Материалы, размещенные в телекоммуникационной библиотеке и представленные в виде цитат,
допускается использовать исключительно в образовательных целях.

Запрещается тиражирование информационных ресурсов **с целью извлечения коммерческой выгоды, а также иное их использование в нарушение соответствующих положений действующего законодательства по защите авторских прав.**



В. Г. Олифер, Н. А. Олифер

КОМПЬЮТЕРНЫЕ СЕТИ
ПРИНЦИПЫ, ТЕХНОЛОГИИ, ПРОТОКОЛЫ

Рекомендовано Министерством образования Российской Федерации

в качестве учебного пособия для студентов высших учебных заведений,
обучающихся направлению «Информатика и вычислительная техника»
и по специальностям «Вычислительные машины, комплексы, системы и сети»,
«Автоматизированные машины, комплексы, системы и сети»,
«Программное обеспечение вычислительной техники
и автоматизированных систем»

Санкт-Петербург
Москва • Харьков • Минск
2001

ПИТЕР

ББК 32.973.202я7
УДК 681.326(075)
О54

Рецензенты:
Кафедра «Вычислительная техника» факультета «Вычислительные машины и системы»
Московского государственного института радиотехники, электроники и автоматики
(Технического университета)
Ю. А. Григорьев, доктор технических наук, профессор,
кафедра «Системы обработки информации и управления»
Московский государственный технический университет им. Н. Э. Баумана
Б. А. Прижуков, кандидат технических наук
Заместитель начальника ИВЦ
ОАО Московский междугородный и международный телефон

О54 Компьютерные сети. Принципы, технологии, протоколы/ В. Г. Олифер, Н. А. Олифер. – СПб.: Питер, 2001. – 672 с.: ил.
ISBN 5-8046-0133-4

Книга представляет собой фундаментальный учебник, в котором последовательно рассматриваются все основные аспекты архитектуры и технологии современных компьютерных сетей. Она предназначена для студентов, аспирантов и технических специалистов, которые хотят получить базовые знания о принципах построения компьютерных сетей, понять особенности традиционных и перспективных технологий локальных и глобальных сетей, изучить способы создания крупных составных сетей и управления такими сетями.

Рекомендуется в качестве учебного пособия для студентов, обучающихся по направлению «Информатика и вычислительная техника».

ББК 32.973.202я7
УДК 681.326(075)
ISBN 5-8046-0133-4
©В. Г. Олифер, Н. А. Олифер, 1999
© Издательский дом «Питер», 2001

ОТ АВТОРОВ

Эта книга является результатом опыта пятилетнего преподавания авторами курсов сетевой тематики в Центре информационных технологий в стенах Московского государственного университета. Основу книги составили материалы курсов «Проблемы построения корпоративных сетей», «Основы сетевых технологий», «Организация удаленного доступа», «Сети TCP/IP», «Стратегическое планирование сетей масштаба предприятия» и ряда других. Эти материалы прошли успешную проверку в бескомпромиссной и сложной аудитории, состоящей из слушателей с существенно разным уровнем подготовки и кругом профессиональных интересов. Среди них были студенты и аспиранты вузов, сетевые администраторы и интеграторы, начальники отделов автоматизации и преподаватели. Учитывая специфику аудитории, курсы лекций были построены так, чтобы начинающий получил основу для дальнейшего изучения, а специалист систематизировал и актуализировал свои знания. В

соответствии с такими же принципами написана и эта книга – она является фундаментальным курсом по компьютерным сетям, который сочетает широту охвата основных областей, проблем и технологий этой быстро развивающейся области знаний с основательным рассмотрением деталей каждой технологии и особенностей оборудования, в котором эта технология реализуется.

Для кого эта книга

Книга предназначена для студентов, аспирантов и технических специалистов, которые хотят получить базовые знания о принципах построения компьютерных сетей, понять особенности традиционных и перспективных технологий локальных и глобальных сетей, изучить способы создания крупных составных сетей и управления такими сетями.

Книга будет полезна начинающим специалистам в области сетевых технологий, которые имеют только общие представления о работе сетей из опыта общения с персональными компьютерами и, возможно, с сетью Internet, но хотели бы получить базовые знания, позволяющие продолжать изучение сетей самостоятельно.

Сложившимся сетевым специалистам книга может оказать помощь при знакомстве с теми технологиями, с которыми им не пришлось столкнуться в практической работе, а также быть справочным средством, в котором можно найти описание конкретного протокола, формата кадра и т. п.

Студенты всех специальностей могут использовать книгу в качестве учебного пособия для общего курса по компьютерным сетям, а для студентов группы специальностей «220000. Информатика и вычислительная техника» книга может служить учебным пособием по специальным дисциплинам.

Структура книги

Книга построена по принципу «спирали познания». Авторы не раз возвращаются к обсуждению отдельных проблем, методов и технологий, но каждый раз на новом уровне.

В *главе* 1 рассматриваются самые общие вопросы построения вычислительных сетей. Как появились первые сети? Что они могут дать предприятию? Какие сетевые задачи возникают уже при связи двух компьютеров? Какие новые проблемы появляются при увеличений числа связываемых машин? Почему такое важное значение в мире вычислительных сетей придается модели ISO/OSI? И так ли уж принципиально отличие локальных и глобальных сетей? Глава заканчивается обсуждением требований, которые предъявляются к современным вычислительным сетям. И здесь главной задачей является пояснение смысла основных характеристик сети. В каких единицах измеряется производительность сети? В чем отличие масштабируемости от расширяемости? Почему для современной сети так желательно совмещение голосового и компьютерного трафика?

Глава 2 носит вспомогательный характер и представляет собой своего рода учебник в учебнике. В ней детально излагаются современные методы передачи дискретной информации в сетях. Хотя этот материал относится не только к компьютерным сетям, но и к любым системам связи, его наличие дает возможность читателю, не тратя время на просмотр большого количества литературы, получить тот необходимый минимум знаний в таких областях, как теория информации, спектральный анализ, физическое и логическое кодирование данных, обнаружение и коррекция ошибок, без которого трудно глубоко вникнуть в методы построения современных протоколов и коммуникационного оборудования.

В *главе* 3 даны подробные описания практически всех основных технологий локальных сетей, как традиционных, таких как Ethernet, Token Ring и FDDI, так и новых высокоскоростных, таких как Fast Ethernet, lOOVG-AnyLAN, Gigabit Ethernet. Кроме того, рассматриваются некоторые факты истории становления технологий, их стандартизации и развития, помогающие понять место каждой технологии в ряду ей подобных, а также ее перспективы. Для технологий семейства Ethernet приводятся рекомендованные IEEE методики расчета корректности конфигурации сети.

Глава 4 посвящена оборудованию локальных сетей, работающему на физическом и канальном уровнях,- структурированным кабельным системам, сетевым адаптерам, повторителям и концентраторам различных технологий, а также мостам и наиболее динамично развивающемуся классу устройств локальных сетей – коммутаторам. Наряду с потребительскими характеристиками коммутаторов приведены типовые схемы их внутренней организации, рассмотрена поддержка ими резервных связей на основе протокола Spanning Tree, а также одна из наиболее перспективных технологий, реализованная в коммутаторах, – технология виртуальных сетей VLAN. Большое внимание

в главе уделяется методам логической структуризации сетей, рассматриваются причины повышения производительности, надежности и управляемости сетей, разделенных на логические сегменты.

В *главе* 5 рассмотрены вопросы построения составных сетей, а также наиболее сложные и интересные сетевые устройства – маршрутизаторы. Центральное место в главе занимает реализация общих принципов и идей межсетевого взаимодействия (internetworking) в IP-сетях. Описываются различные аспекты адресации: способы отображения локальных, сетевых и символьных адресов, использование масок и современных методов агрегирования IP-адресов в Internet, а также способы автоматического конфигурирования узлов. Детально рассмотрена работа протокола IP по продвижению и фрагментации пакетов, изучается общий формат таблицы маршрутизации и примеры ее частных реализаций в программных и аппаратных маршрутизаторах различных типов. Приводится подробное описание протоколов маршрутизации RIP и OSPF, анализируются области применимости этих протоколов и возможности их комбинирования. Всестороннее рассмотрение в главе 5 протоколов стека TCP/IP придает ей самостоятельное значение введения в IP-сети. Главу завершает описание типов и основных характеристик маршрутизаторов, вариантов их внутренней организации, а также методов комбинирования функций коммутации и маршрутизации в одном устройстве – коммутаторе 3-го уровня.

Глава 6 целиком посвящена глобальным сетям. Так как технологии, оборудование и даже терминология этих сетей являются весьма специфическими, то читатель возвращается в некоторой степени к сетевым «азам», несмотря на то, что он уже изучил достаточно много сложных вопросов и поднялся сравнительно высоко по уровням модели OSL Подробно изучается основа любой территориальной сети – выделенные аналоговые и цифровые каналы, технологии первичных сетей FDM, PDH и SONET/SDH, позволяющие создавать такие каналы, соответствующее оборудование доступа – модемы различных типов и цифровые устройства DSU/CSU, а также – протоколы канального уровня HDLC, PPP и SLIP. Глава содержит описание широкого спектра технологий глобальных сетей – как классических X.25, так и более современных ISDN, frame relay и ATM. Наряду с детальным рассмотрением архитектуры, протоколов и услуг сетей каждого типа изучается техника передачи данных на основе виртуальных каналов, отличающая эти технологии от технологий локальных сетей. Глава завершается рассмотрением схем и средств удаленного доступа, позволяющих домашним и мобильным пользователям разделять ресурсы корпоративной сети.

В *главе* 7, завершающей книгу, рассматриваются системы управления сетями. Эти сложные, громоздкие и дорогостоящие системы всегда вызывают много вопросов: В каких случаях их применение оправдано? Что делает платформа управления? Возможно ли создание единой системы управления для разнородной сети, построенной на оборудовании разных производителей? Приведенный в главе материал знакомит читателя с набором функций, который должен поддерживаться системой управления в соответствии с международными стандартами, архитектурой систем управления разнородными сетями, назначением платформ управления. Детально описываются протоколы SNMP, CMIP и структура стандартных управляющих баз MIB-I, MIB-II и RMON MIB,встраиваемых в оборудование. Глава завершается рассмотрением портативных приборов, выполняющих функции мониторинга сети.

Авторы стремились сделать работу читателя с книгой максимально эффективной. Подробный индексный указатель позволяет быстро найти интересующий материал по одному из многочисленных терминов, используемых в сетевой индустрии. Каждый раздел завершается выводами, которые призваны сконцентрировать внимание читателя на главных идеях, темах и результатах раздела, помогая ему не упустить из виду главное за обилием, хотя и полезных, но частных фактов и деталей. В конце каждой главы помещены вопросы и упражнения для проверки степени усвоения основных концепций, а в отдельных случаях и для углубления понимания некоторых идей. Ответы на некоторые из этих вопросов вы найдете в приложении к книге.

Авторы осознают, что ряд важных вопросов, относящихся к компьютерным сетям, не был рассмотрен в данной книге. Прежде всего, это сетевые операционные системы и их ключевые службы, такие как справочная служба и служба безопасности, а также методы и средства согласования неоднородных сетей. Объединение этих тем с рассмотрением транспортных аспектов компьютерных сетей, составивших ядро этой книги, неизбежно привело бы к чрезмерному росту ее объема. В настоящее время авторы работают над новой книгой, которая является логическим продолжением этого учебника и посвящена сетевым операционным системам и сетевым службам.

От издательства

Ваши замечания, предложения, вопросы отправляйте по адресу электронной почты comp@piter-press.ru (издательство «Питер», компьютерная редакция).

Мы будем рады узнать ваше мнение!

Подробную информацию о наших книгах вы найдете на Web-сайте издательства http://www.piter-press.ru.

БЛАГОДАРНОСТИ

Прежде всего мы хотим поблагодарить директоров Центра информационных технологий Алексея и Елену Сальниковых, так как эта книга вряд ли была бы написана, если бы они не организовали в свое время Центр и не пригласили нас возглавить и вести в нем сетевое направление. Кроме того, мы признательны им за постоянную готовность удовлетворять все наши информационные потребности–у нас всегда была возможность получать самые свежие зарубежные журналы, книги известных мировых издательств, технические обзоры, материалы конференций, а также и за ту напряженную рабочую атмосферу, которая требовала от нас постоянного расширения круга интересов и тематики курсов. Выражаем признательность сотрудникам Центра Жене Григорьевой, Кире Кузнецовой и Наташе Новиковой за то, что они максимально облегчали нам работу по подготовке конспектов лекций, послуживших основой этой книги.

Мы благодарим компании IBS, UNI, АйТи, ADV Consulting и лично сотрудников этих компаний Валерия Гущина, Сергея Малышева, Игоря Сунчелея, Алексея Фролова и Петра Митрофанова за высокопрофессиональные консультации, которые мы всегда от них получали, совместную работу по проведению курсов, предоставление свежих технических материалов и готовность поделиться практическим опытом построения сетей.

Мы благодарны также рецензентам д. т. н., профессору МГТУ им. Н. Э. Баумана Григорьеву Ю. А. и к. т. н., заместителю начальника ИВЦ ОАО Московский ММТ Прижукову Б. Ф. за ценные советы и замечания, позволившие улучшить качество этой книги.

Мы признательны сотрудникам издательства «Питер», особенно заведующей редакцией компьютерной литературы и одновременно нашему главному редактору Екатерине Строгановой, а также литературному редактору Алексею Жданову, которые помогли нам в очень сжатые сроки превратить конспект лекций в книгу, которая найдет, надеемся, своего массового читателя.

Виктор и Наталья Олифер

ГЛАВА 1. ОБЩИЕ ПРИНЦИПЫ ПОСТРОЕНИЯ ВЫЧИСЛИТЕЛЬНЫХ СЕТЕЙ

1.1. От централизованных систем - к вычислительным сетям

1.1.1. Эволюция вычислительных систем

Концепция вычислительных сетей является логическим результатом эволюции компьютерной технологии. Первые компьютеры 50-х годов – большие, громоздкие и дорогие – предназначались для очень небольшого числа избранных пользователей. Часто эти монстры занимали целые здания. Такие компьютеры не были предназначены для интерактивной работы пользователя, а использовались в режиме пакетной обработки.

Системы пакетной обработки

Системы пакетной обработки, как правило, строились на базе мэйнфрейма – мощного и надежного компьютера универсального назначения. Пользователи подготавливали перфокарты, содержащие данные и команды программ, и передавали их в вычислительный центр. Операторы вводили эти карты в компьютер, а распечатанные результаты пользователи получали обычно только на следующий день (рис. 1.1). Таким образом, одна неверно набитая карта означала как минимум суточную задержку.

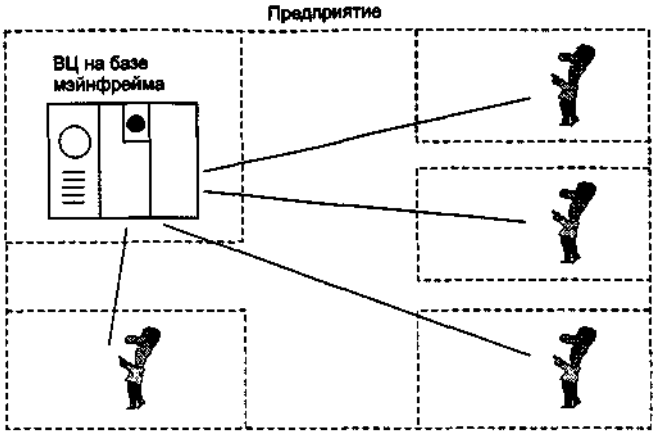


Рис. 1.1. Централизованная система на базе мэйнфрейма

Конечно, для пользователей интерактивный режим работы, при котором можно с терминала оперативно руководить процессом обработки своих данных, был бы гораздо удобней. Но интересами пользователей на первых этапах развития вычислительных систем в значительной степени пренебрегали, поскольку пакетный режим – это самый эффективный режим использования вычислительной мощности, так как он позволяет выполнить в единицу времени больше пользовательских задач, чем любые другие режимы. Во главу угла ставилась эффективность работы самого дорогого устройства вычислительной машины – процессора, в ущерб эффективности работы использующих его специалистов.

Многотерминальные системы – прообраз сети

По мере удешевления процессоров в начале 60-х годов появились новые способы организации вычислительного процесса, которые позволили учесть интересы пользователей. Начали развиваться интерактивные многотерминальные системы разделения времени (рис. 1.2). В таких системах компьютер отдавался в распоряжение сразу нескольким пользователям. Каждый пользователь получал в свое распоряжение терминал, с помощью которого он мог вести диалог с компьютером. Причем время реакции вычислительной системы было достаточно мало для того, чтобы пользователю была не слишком заметна параллельная работа с компьютером и других пользователей. Разделяя таким образом компьютер, пользователи получили возможность за сравнительно небольшую плату пользоваться преимуществами компьютеризации.

Терминалы, выйдя за пределы вычислительного центра, рассредоточились по всему предприятию. И хотя вычислительная мощность оставалась полностью централизованной, некоторые функции – такие как ввод и вывод данных – стали распределенными. Такие многотерминальные централизованные системы внешне уже были очень похожи на локальные вычислительные сети. Действительно, рядовой пользователь работу за терминалом мэйнфрейма воспринимал примерно так же, как сейчас он воспринимает работу за подключенным к сети персональным компьютером. Пользователь мог получить доступ к общим файлам и периферийным устройствам, при этом у него поддерживалась полная иллюзия единоличного владения компьютером, так как он мог запустить нужную ему программу в любой момент и почти сразу же получить результат. (Некоторые, далекие от вычислительной техники пользователи даже были уверены, что все вычисления выполняются внутри их дисплея.)

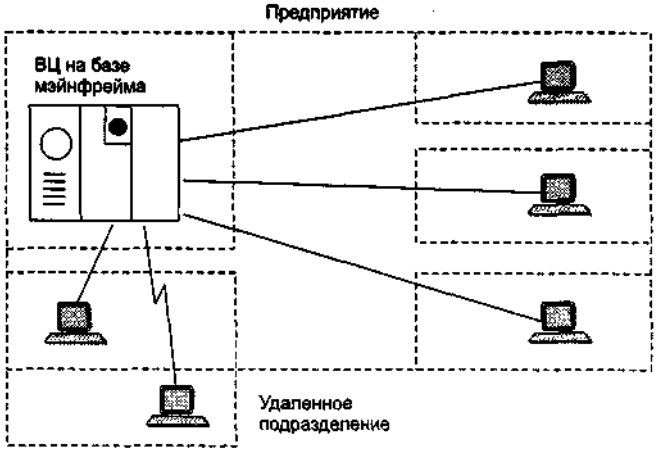


Рис. 1.2. Многотерминальная система – прообраз вычислительной сети

Таким образом, многотерминальные системы, работающие в режиме разделения времени, стали первым шагом на пути создания локальных вычислительных сетей. Но до появления локальных сетей нужно было пройти еще большой путь, так как многотерминальные системы, хотя и имели внешние черты распределенных систем, все еще сохраняли централизованный характер обработки данных. С другой стороны, и потребность предприятий в создании локальных сетей в это время еще не созрела – в одном здании просто нечего было объединять в сеть, так как из-за высокой стоимости вычислительной техники предприятия не могли себе позволить роскошь приобретения нескольких компьютеров. В этот период был справедлив так называемый «закон Гроша», который эмпирически отражал уровень технологии того времени. В соответствии с этим законом производительность компьютера была пропорциональна квадрату его стоимости, отсюда следовало, что за одну и ту же сумму было выгоднее купить одну мощную машину, чем две менее мощных – их суммарная мощность оказывалась намного ниже мощности дорогой машины.

Появление глобальных сетей

Тем не менее потребность в соединении компьютеров, находящихся на большом расстоянии друг от друга, к этому времени вполне назрела. Началось все с решения более простой задачи – доступа к компьютеру с терминалов, удаленных от него на многие сотни, а то и тысячи километров. Терминалы соединялись с компьютерами через телефонные сети с помощью модемов. Такие сети позволяли многочисленным пользователям получать удаленный доступ к разделяемым ресурсам нескольких мощных компьютеров класса суперЭВМ. Затем появились системы, в которых наряду с удаленными соединениями типа терминал–компьютер были реализованы и удаленные связи типа компьютер–компьютер. Компьютеры получили возможность обмениваться данными в автоматическом режиме, что, собственно, и является базовым механизмом любой вычислительной сети. Используя этот механизм, в первых сетях были реализованы службы обмена файлами, синхронизации баз данных, электронной почты и другие, ставшие теперь традиционными сетевые службы.

Таким образом, хронологически первыми появились глобальные вычислительные сети. Именно при построении глобальных сетей были впервые предложены и отработаны многие основные идеи и концепции современных вычислительных сетей. Такие, например, как многоуровневое построение коммуникационных протоколов, технология коммутации пакетов, маршрутизация пакетов в составных сетях.

Первые локальные сети

В начале 70-х годов произошел технологический прорыв в области производства компьютерных компонентов – появились большие интегральные схемы. Их сравнительно невысокая стоимость и высокие функциональные возможности привели к созданию мини-компьютеров, которые стали реальными конкурентами мэйнфреймов. Закон Гроша перестал соответствовать действительности, так как десяток мини-компьютеров выполнял некоторые задачи (как правило, хорошо распараллеливаемые) быстрее одного мэйнфрейма, а стоимость такой мини-компьютерной системы была меньше.

Даже небольшие подразделения предприятий получили возможность покупать для себя компьютеры.

Мини-компьютеры выполняли задачи управления технологическим оборудованием, складом и другие задачи уровня подразделения предприятия. Таким образом, появилась концепция распределения компьютерных ресурсов по всему предприятию. Однако при этом все компьютеры одной организации по-прежнему продолжали работать автономно (рис. 1.3).

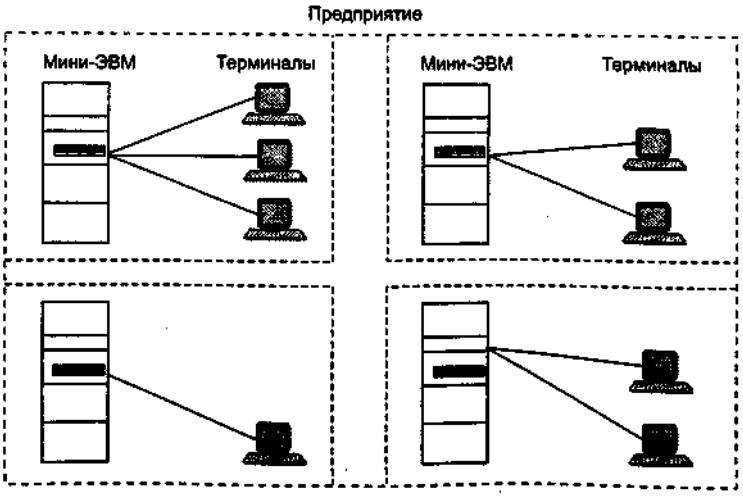


Рис. 1.3. Автономное использование нескольких мини-компьютеров на одном предприятии

Но шло время, потребности пользователей вычислительной техники росли, им стало недостаточно собственных компьютеров, им уже хотелось получить возможность обмена данными с другими близко расположенными компьютерами. В ответ на эту потребность предприятия и организации стали соединять свои мини-компьютеры вместе и разрабатывать программное обеспечение, необходимое для их взаимодействия. В результате появились первые локальные вычислительные сети (рис. 1.4). Они еще во многом отличались от современных локальных сетей, в первую очередь – своими устройствами сопряжения. На первых порах для соединения компьютеров друг с другом использовались самые разнообразные нестандартные устройства со своим способом представления данных на линиях связи, своими типами кабелей и т. п. Эти устройства могли соединять только те типы компьютеров, для которых были разработаны, – например, мини-компьютеры PDP-11 с мэйнфреймом IBM 360 или компьютеры «Наири» с компьютерами «Днепр». Такая ситуация создала большой простор для творчества студентов – названия многих курсовых и дипломных проектов начинались тогда со слов «Устройство сопряжения...».

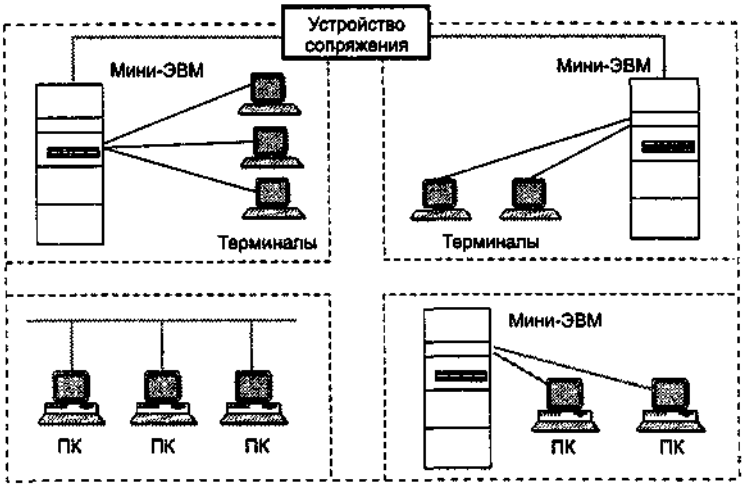


Рис. 1.4. Различные типы связей в первых локальных сетях

Создание стандартных технологий локальных сетей

В середине 80-х годов положение дел в локальных сетях стало кардинально меняться. Утвердились

стандартные технологии объединения компьютеров в сеть – Ethernet, Arcnet, Token Ring. Мощным стимулом для их развития послужили персональные компьютеры. Эти массовые продукты явились идеальными элементами для построения сетей – с одной стороны, они были достаточно мощными для работы сетевого программного обеспечения, а с другой – явно нуждались в объединении своей вычислительной мощности для решения сложных задач, а также разделения дорогих периферийных устройств и дисковых массивов. Поэтому персональные компьютеры стали преобладать в локальных сетях, причем не только в качестве клиентских компьютеров, но и в качестве центров хранения и обработки данных, то есть сетевых серверов, потеснив с этих привычных ролей мини-компьютеры и мэйнфреймы.

Стандартные сетевые технологии превратили процесс построения локальной сети из искусства в рутинную работу. Для создания сети достаточно было приобрести сетевые адаптеры соответствующего стандарта, например Ethernet, стандартный кабель, присоединить адаптеры к кабелю стандартными разъемами и установить на компьютер одну из популярных сетевых операционных систем, например, NetWare. После этого сеть начинала работать и присоединение каждого нового компьютера не вызывало никаких проблем – естественно, если на нем был установлен сетевой адаптер той же технологии.

Локальные сети в сравнении с глобальными сетями внесли много нового в способы организации работы пользователей. Доступ к разделяемым ресурсам стал гораздо удобнее – пользователь мог просто просматривать списки имеющихся ресурсов, а не запоминать их идентификаторы или имена. После соединения с удаленным ресурсом можно было работать с ним с помощью уже знакомых пользователю по работе с локальными ресурсами команд. Последствием и одновременно движущей силой такого прогресса стало появление огромного числа непрофессиональных пользователей, которым совершенно не нужно было изучать специальные (и достаточно сложные) команды для сетевой работы. А возможность реализовать все эти удобства разработчики локальных сетей получили в результате появления качественных кабельных линий связи, на которых даже сетевые адаптеры первого поколения обеспечивали скорость передачи данных до 10 Мбит/с.

Конечно, о таких скоростях разработчики глобальных сетей не могли даже мечтать – им приходилось пользоваться теми каналами связи, которые были в наличии, так как прокладка новых кабельных систем для вычислительных сетей протяженностью в тысячи километров потребовала бы колоссальных капитальных вложений. А «под рукой» были только телефонные каналы связи, плохо приспособленные для высокоскоростной передачи дискретных данных – скорость в 1200 бит/с была для них хорошим достижением. Поэтому экономное расходование пропускной способности каналов связи часто являлось основным критерием эффективности методов передачи данных в глобальных сетях. В этих условиях различные процедуры прозрачного доступа к удаленным ресурсам, стандартные для локальных сетей, для глобальных сетей долго оставались непозволительной роскошью.

Современные тенденции

Сегодня вычислительные сети продолжают развиваться, причем достаточно быстро. Разрыв между локальными и глобальными сетями постоянно сокращается во многом из-за появления высокоскоростных территориальных каналов связи, не уступающих по качеству кабельным системам локальных сетей. В глобальных сетях появляются службы доступа к ресурсам, такие же удобные и прозрачные, как и службы локальных сетей. Подобные примеры в большом количестве демонстрирует самая популярная глобальная сеть – Internet.

Изменяются и локальные сети. Вместо соединяющего компьютеры пассивного кабеля в них в большом количестве появилось разнообразное коммуникационное оборудование – коммутаторы, маршрутизаторы, шлюзы. Благодаря такому оборудованию появилась возможность построения больших корпоративных сетей, насчитывающих тысячи компьютеров и имеющих сложную структуру. Возродился интерес к крупным компьютерам – в основном из-за того, что после спада эйфории по поводу легкости работы с персональными компьютерами выяснилось, что системы, состоящие из сотен серверов, обслуживать сложнее, чем несколько больших компьютеров. Поэтому на новом витке эволюционной спирали мэйнфреймы стали возвращаться в корпоративные вычислительные системы, но уже как полноправные сетевые узлы, поддерживающие Ethernet или Token Ring, а также стек протоколов TCP/IP, ставший благодаря Internet сетевым стандартом де-факто.

Проявилась еще одна очень важная тенденция, затрагивающая в равной степени как локальные, так и

глобальные сети. В них стала обрабатываться несвойственная ранее вычислительным сетям информация – голос, видеоизображения, рисунки. Это потребовало внесения изменений в работу протоколов, сетевых операционных систем и коммуникационного оборудования. Сложность передачи такой мультимедийной информации по сети связана с ее чувствительностью к задержкам при передаче пакетов данных – задержки обычно приводят к искажению такой информации в конечных узлах сети. Так как традиционные службы вычислительных сетей – такие как передача файлов или электронная почта – создают малочувствительный к задержкам трафик и все элементы сетей разрабатывались в расчете на него, то появление трафика реального времени привело к большим проблемам.

Сегодня эти проблемы решаются различными способами, в том числе и с помощью специально рассчитанной на передачу различных типов трафика технологии ATM. Однако, несмотря на значительные усилия, предпринимаемые в этом направлении, до приемлемого решения проблемы пока далеко, и в этой области предстоит еще много сделать, чтобы достичь заветной цели – слияния технологий не только локальных и глобальных сетей, но и технологий любых информационных сетей – вычислительных, телефонных, телевизионных и т. п. Хотя сегодня эта идея многим кажется утопией, серьезные специалисты считают, что предпосылки для такого синтеза уже существуют, и их мнения расходятся только в оценке примерных сроков такого объединения – называются сроки от 10 до 25 лет. Причем считается, что основой для объединения послужит технология коммутации пакетов, применяемая сегодня в вычислительных сетях, а не технология коммутации каналов, используемая в телефонии, что, наверно, должно повысить интерес к сетям этого типа, которым и посвящена данная книга.

1.1.2. Вычислительные сети – частный случай распределенных систем

Компьютерные сети относятся к распределенным (или децентрализованным) вычислительным системам. Поскольку основным признаком распределенной вычислительной системы является наличие нескольких центров обработки данных, то наряду с компьютерными сетями к распределенным системам относят также мультипроцессорные компьютеры и многомашинные вычислительные комплексы.

Мультипроцессорные компьютеры

В мультипроцессорных компьютерах имеется несколько процессоров, каждый из которых может относительно независимо от остальных выполнять свою программу. В мультипроцессоре существует общая для всех процессоров операционная система, которая оперативно распределяет вычислительную нагрузку между процессорами. Взаимодействие между отдельными процессорами организуется наиболее простым способом – через общую оперативную память.

Сам по себе процессорный блок не является законченным компьютером и поэтому не может выполнять программы без остальных блоков мультипроцессорного компьютера – памяти и периферийных устройств. Все периферийные устройства являются для всех процессоров мультипроцессорной системы общими. Территориальную распределенность мультипроцессор не поддерживает – все его блоки располагаются в одном или нескольких близко расположенных конструктивах, как и у обычного компьютера.

Основное достоинство мультипроцессора – его высокая производительность, которая достигается за счет параллельной работы нескольких процессоров. Так как при наличии общей памяти взаимодействие процессоров происходит очень быстро, мультипроцессоры могут эффективно выполнять даже приложения с высокой степенью связи по данным.

Еще одним важным свойством мультипроцессорных систем является отказоустойчивость, то есть способность к продолжению работы при отказах некоторых элементов, например процессоров или блоков памяти. При этом производительность, естественно, снижается, но не до нуля, как в обычных системах, в которых отсутствует избыточность.

Многомашинные системы

Многомашинная система – это вычислительный комплекс, включающий в себя несколько компьютеров (каждый из которых работает под управлением собственной операционной системы), а

также программные и аппаратные средства связи компьютеров, которые обеспечивают работу всех компьютеров комплекса как единого целого.

Работа любой многомашинной системы определяется двумя главными компонентами: высокоскоростным механизмом связи процессоров и системным программным обеспечением, которое предоставляет пользователям и приложениям прозрачный доступ к ресурсам всех компьютеров, входящих в комплекс. В состав средств связи входят программные модули, которые занимаются распределением вычислительной нагрузки, синхронизацией вычислений и реконфигурацией системы. Если происходит отказ одного из компьютеров комплекса, его задачи могут быть автоматически переназначены и выполнены на другом компьютере. Если в состав многомашинной системы входят несколько контроллеров внешних устройств, то в случае отказа одного из них, другие контроллеры автоматически подхватывают его работу. Таким образом, достигается высокая отказоустойчивость комплекса в целом.

Помимо повышения отказоустойчивости, многомашинные системы позволяют достичь высокой производительности за счет организации параллельных вычислений. По сравнению с мультипроцессорными системами возможности параллельной обработки в многомашинных системах ограничены: эффективность распараллеливания резко снижается, если параллельно выполняемые задачи тесно связаны между собой по данным. Это объясняется тем, что связь между компьютерами многомашинной системы менее тесная, чем между процессорами в мультипроцессорной системе, так как основной обмен данными осуществляется через общие многовходовые периферийные устройства. Говорят, что в отличие от мультипроцессоров, где используются сильные программные и аппаратные связи, в многомашинных системах аппаратные и программные связи между обрабатывающими устройствами являются более слабыми. Территориальная распределенность в многомашинных комплексах не обеспечивается, так как расстояния между компьютерами определяются длиной связи между процессорным блоком и дисковой подсистемой.

Вычислительные сети

В *вычислительных сетях* программные и аппаратные связи являются еще более слабыми, а автономность обрабатывающих блоков проявляется в наибольшей степени – основными элементами сети являются стандартные компьютеры, не имеющие ни общих блоков памяти, ни общих периферийных устройств. Связь между компьютерами осуществляется с помощью специальных периферийных устройств – сетевых адаптеров, соединенных относительно протяженными каналами связи. Каждый компьютер работает под управлением собственной операционной системы, а какая-либо «общая» операционная система, распределяющая работу между компьютерами сети, отсутствует. Взаимодействие между компьютерами сети происходит за счет передачи сообщений через сетевые адаптеры и каналы связи. С помощью этих сообщений один компьютер обычно запрашивает доступ к локальным ресурсам другого компьютера. Такими ресурсами могут быть как данные, хранящиеся на диске, так и разнообразные периферийные устройства – принтеры, модемы, факс-аппараты и т. д. Разделение локальных ресурсов каждого компьютера между всеми пользователями сети – основная цель создания вычислительной сети.

Каким же образом сказывается на пользователе тот факт, что его компьютер подключен к сети? Прежде всего, он может пользоваться не только файлами, дисками, принтерами и другими ресурсами своего компьютера, но и аналогичными ресурсами других компьютеров, подключенных к той же сети. Правда, для этого недостаточно снабдить компьютеры сетевыми адаптерами и соединить их кабельной системой. Необходимы еще некоторые добавления к операционным системам этих компьютеров. На тех компьютерах, ресурсы которых должны быть доступны всем пользователям сети, необходимо добавить модули, которые постоянно будут находиться в режиме ожидания запросов, поступающих по сети от других компьютеров. Обычно такие модули называются программными *серверами* (*server*), так как их главная задача – обслуживать (serve) запросы на доступ к ресурсам своего компьютера. На компьютерах, пользователи которых хотят получать доступ к ресурсам других компьютеров, также нужно добавить к операционной системе некоторые специальные программные модули, которые должны вырабатывать запросы на доступ к удаленным ресурсам и передавать их по сети на нужный компьютер. Такие модули обычно называют программными *клиентами* (*client*). Собственно же сетевые адаптеры и каналы связи решают в сети достаточно простую задачу – они передают сообщения с запросами и ответами от одного компьютера к другому, а основную работу по организации совместного

использования ресурсов выполняют клиентские и серверные части операционных систем.

Пара модулей «клиент - сервер» обеспечивает совместный доступ пользователей к определенному типу ресурсов, например к файлам. В этом случае говорят, что пользователь имеет дело с файловой *службой* (*service*). Обычно сетевая операционная система поддерживает несколько видов сетевых служб для своих пользователей – файловую службу, службу печати, службу электронной почты, службу удаленного доступа и т. п.

ПРИМЕЧАНИЕ В технической литературе англоязычный термин «service» обычно переводится как «служба», «сервис» или «услуга». Часто эти термины используются как синонимы. В то же время некоторые специалисты различают термин «служба», с одной стороны, и термины «сервис» и «услуга», с другой. Под «службой» понимается сетевой компонент, который реализует некоторый набор услуг, а «сервисом» называют описание того набора услуг, который предоставляется данной службой. Таким образом, сервис - это интерфейс между потребителем услуг и поставщиком услуг (службой). Далее будет использоваться термин «служба» во всех случаях, когда различие в значении этих терминов не носит принципиального характера.

Термины «клиент» и «сервер» используются не только для обозначения программных модулей, но и компьютеров, подключенных к сети. Если компьютер предоставляет свои ресурсы другим компьютерам сети, то он называется сервером, а если он их потребляет – клиентом. Иногда один и тот же компьютер может одновременно играть роли и сервера, и клиента.

Распределенные программы

Сетевые службы всегда представляют собой распределенные программы. *Распределенная программа* – это программа, которая состоит из нескольких взаимодействующих частей (в приведенном на рис. 1.5 примере – из двух), причем каждая часть, как правило, выполняется на отдельном компьютере сети.

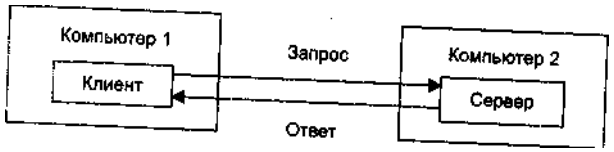


Рис. 1.5. Взаимодействие частей распределенного приложения

До сих пор речь шла о системных распределенных программах. Однако в сети могут выполняться и растределенные пользовательские программы – приложения. Распределенное приложение также состоит из нескольких частей, каждая из которых выполняет какую-то определенную законченную работу по решению прикладной задачи. Например, одна часть приложения, выполняющаяся на компьютере пользователя, может поддерживать специализированный графический интерфейс, вторая – работать на мощном выделенном компьютере и заниматся статистической обработкой введенных пользователем данных, а третья – заносить полученные результаты в базу данных на компьютере с установленной стандартной СУБД. Распределенные приложения в полной мере используют потенциальные возможности распределенной обработки, предоставляемые вычислительной сетью, и поэтому часто называются *сетевыми приложениями*.

Следует подчеркнуть, что не всякое приложение, выполняемое в сети, является сетевым. Существует большое количество популярных приложений, которые не являются распределенными и целиком выполняются на одном компьютере сети. Тем не менее и такие приложения могут использовать преимущества сети за счет встроенных в операционную систему сетевых служб. Значительная часть истории локальных сетей связана как раз с использованием таких нераспределенных приложений. Рассмотрим, например, как проходила работа пользователя с известной в свое время СУБД dBase. Обычно файлы базы данных, с которыми работали все пользователи сети, располагалось на файловом сервере. Сама же СУБД хранилась на каждом клиентском компьютере в виде единого программного модуля. Программа dBase была рассчитана на обработку только локальных данных, то есть данных, расположенных на том же компьютере, что и сама программа. Пользователь запускал dBase на своем компьютере, и она искала данные на локальном диске, совершенно не принимая во внимание

существование сети. Чтобы обрабатывать с помощью dBase данные на удаленном компьютере, пользователь обращался к услугам файловой службы, которая доставляла данные с сервера на клиентский компьютер и создавала для СУБД эффект их локального хранения.

Большинство приложений, используемых в локальных сетях в середине 80-х годов, являлись обычными, нераспределенными приложениями. И это понятно – они были написаны для автономных компьютеров, а потом просто были перенесены в сетевую среду. Создание же распределенных приложений, хотя и сулило много преимуществ (уменьшение сетевого трафика, специализация компьютеров), оказалось делом совсем не простым. Нужно было решать множество дополнительных проблем – на сколько частей разбить приложение, какие функции возложить на каждую часть, как организовать взаимодействие этих частей, чтобы в случае сбоев и отказов оставшиеся части корректно завершали работу, и т. д., и т. п. Поэтому до сих пор только небольшая часть приложений является распределенными, хотя очевидно, что именно за этим классом приложений будущее, так как они в полной мере могут использовать потенциальные возможности сетей по распараллеливанию вычислений.

1.1.3. Основные программные и аппаратные компоненты сети

Даже в результате достаточно поверхностного рассмотрения работы в сети становится ясно, что вычислительная сеть – это сложный комплекс взаимосвязанных и согласованно функционирующих программных и аппаратных компонентов. Изучение сети в целом предполагает знание принципов работы ее отдельных элементов:

- компьютеров;
- коммуникационного оборудования;
- операционных систем;
- сетевых приложений.

Весь комплекс программно-аппаратных средств сети может быть описан многослойной моделью. В основе любой сети лежит аппаратный слой стандартизованных компьютерных платформ. В настоящее время в сетях широко и успешно применяются компьютеры различных классов – от персональных компьютеров до мэйнфреймов и суперЭВМ. Набор компьютеров в сети должен соответствовать набору разнообразных задач, решаемых сетью.

Второй слой – это коммуникационное оборудование. Хотя компьютеры и являются центральными элементами обработки данных в сетях, в последнее время не менее важную роль стали играть коммуникационные устройства. Кабельные системы, повторители, мосты, коммутаторы, маршрутизаторы и модульные концентраторы из вспомогательных компонентов сети превратились в основные наряду с компьютерами и системным программным обеспечением как по влиянию на характеристики сети, так и по стоимости. Сегодня коммуникационное устройство может представлять собой сложный специализированный мультипроцессор, который нужно конфигурировать, оптимизировать и администрировать. Изучение принципов работы коммуникационного оборудования требует знакомства с большим количеством протоколов, используемых как в локальных, так и глобальных сетях.

Третьим слоем, образующим программную платформу сети, являются операционные системы (ОС). От того, какие концепции управления локальными и распределенными ресурсами положены в основу сетевой ОС, зависит эффективность работы всей сети. При проектировании сети важно учитывать, насколько просто данная операционная система может взаимодействовать с другими ОС сети, насколько она обеспечивает безопасность и защищенность данных, до какой степени она позволяет наращивать число пользователей, можно ли перенести ее на компьютер другого типа и многие другие соображения.

Самым верхним слоем сетевых средств являются различные сетевые приложения, такие как сетевые базы данных, почтовые системы, средства архивирования данных, системы автоматизации коллективной работы и др. Очень важно представлять диапазон возможностей, предоставляемых приложениями для различных областей применения, а также знать, насколько они совместимы с другими сетевыми приложениями и операционными системами.

1.1.4. Что дает предприятию использование сетей

Этот вопрос можно уточнить следующим образом: в каких случаях развертывание на предприятии вычислительных сетей предпочтительнее использования автономных компьютеров или многомашинных систем? Какие новые возможности появляются на предприятии с появлением там вычислительной сети? И наконец, всегда ли предприятию нужна сеть?

Если не вдаваться в частности, то конечной целью использования вычислительных сетей на предприятии является повышение эффективности его работы, которое может выражаться, например, в увеличении прибыли предприятия. Действительно, если благодаря компьютеризации снизились затраты на производство уже существующего продукта, сократились сроки разработки новой модели или ускорилось обслуживание заказов потребителей – это означает, что данному предприятию действительно нужна была сеть.

Более обстоятельно отвечая на вопрос, зачем предприятию сеть, начнем с рассмотрения тех принципиальных преимуществ сетей, которые вытекают из их принадлежности к распределенным системам.

Концептуальным преимуществом распределенных систем (а значит, и сетей) перед централизованными системами является их *способность выполнять параллельные вычисления*. За счет этого в системе с несколькими обрабатывающими узлами в принципе может быть достигнута производительность, превышающая максимально возможную на данный момент производительность любого отдельного, сколь угодно мощного процессора. Распределенные системы потенциально имеют *лучшее соотношение производительность-стоимость*, чем централизованные системы.

Еще одно очевидное и важное достоинство распределенных систем – это их принципиально *более высокая отказоустойчивость*. Под отказоустойчивостью понимается способность системы выполнять свои функции (может быть, не в полном объеме) при отказах отдельных элементов аппаратуры и неполной доступности данных. Основой повышенной отказоустойчивости распределенных систем является избыточность. Избыточность обрабатывающих узлов (процессоров в многопроцессорных системах или компьютеров в сетях) позволяет при отказе одного узла переназначать приписанные ему задачи на другие узлы. С этой целью в распределенной системе могут быть предусмотрены процедуры динамической или статической реконфигурации. В вычислительных сетях некоторые наборы данных могут дублироваться на внешних запоминающих устройствах нескольких компьютеров сети, так что при отказе одного их них данные остаются доступными.

Использование территориально распределенных вычислительных систем больше соответствует *распределенному характеру прикладных задач* в некоторых предметных областях, таких как автоматизация технологических процессов, банковская деятельность и т.п. Во всех этих случаях имеются рассредоточенные по некоторой территории отдельные потребители информации – сотрудники, организации или технологические установки. Эти потребители достаточно автономно решают свои задачи, поэтому рациональнее предоставлять им собственные вычислительные средства, но в то же время, поскольку решаемые ими задачи тесно взаимосвязаны, их вычислительные средства должны быть объединены в единую систему. Адекватным решением в такой ситуации является использование вычислительной сети.

Для пользователя, кроме выше названных, распределенные системы дают еще и такие преимущества, как *возможность совместного использования данных и устройств*, а также возможность гибкого распределения работ по всей системе. Такое разделение дорогостоящих периферийных устройств – таких как дисковые массивы большой емкости, цветные принтеры, графопостроители, модемы, оптические диски – во многих случаях является основной причиной развертывания сети на предприятии. Пользователь современной вычислительной сети работает за своим компьютером, часто не отдавая себе отчета в том, что при этом он пользуется данными другого мощного компьютера, находящегося за сотни километров от него. Он отправляет электронную почту через модем, подключенный к коммуникационному серверу, общему для нескольких отделов его предприятия. У пользователя создается иллюзия, что эти ресурсы подключены непосредственно к его компьютеру или же «почти» подключены, так как для их использования нужны незначительные дополнительные действия по сравнению с использованием действительно собственных ресурсов. Такое свойство называется *прозрачностью* сети.

В последнее время стал преобладать другой побудительный мотив развертывания сетей, гораздо более важный в современных условиях, чем экономия средств за счет разделения между сотрудниками корпорации дорогой аппаратуры или программ. Этим мотивом стало стремление обеспечить сотрудникам *оперативный доступ к обширной корпоративной информации*. В условиях жесткой

конкурентной борьбы в любом секторе рынка выигрывает, в конечном счете, та фирма, сотрудники которой могут быстро и правильно ответить на любой вопрос клиента – о возможностях их продукции, об условиях ее применения, о решении любых возможных проблем и т. п. В большой корпорации вряд ли даже хороший менеджер может знать все тонкости каждого из выпускаемых фирмой продуктов, тем более что их номенклатура обновляется сейчас каждый квартал, если не месяц. Поэтому очень важно, чтобы менеджер имел возможность со своего компьютера, подключенного к корпоративной сети, скажем в Магадане, передать вопрос клиента на сервер, расположенный в центральном отделении предприятия в Новосибирске, и оперативно получить качественный ответ, удовлетворяющий клиента. В этом случае клиент не обратится к другой фирме, а будет пользоваться услугами данного менеджера и впредь.

Чтобы такая работа была возможна, необходимо не только наличие быстрых и надежных связей в корпоративной сети, но и наличие структурированной информации на серверах предприятия, а также возможность эффективного поиска нужных данных. Этот аспект сетевой работы всегда был узким местом в организации доставки информации сотрудникам – даже при существовании мощных СУБД информация в них попадала не самая «свежая» и не в том объеме, который был нужен. В последнее время в этой области наметился некоторый прогресс, связанный с использованием гипертекстовой информационной службы WWW – так называемой технологии *intranet*. Эта технология поддерживает достаточно простой способ представления текстовой и графической информации в виде гипертекстовых страниц, что позволяет быстро поместить самую свежую информацию на WWW-серверы корпорации. Кроме того, она унифицирует .просмотр информации с помощью стандартных программ – Web-броузеров, работа с которыми несложна даже для неспециалиста. Сейчас многие крупные корпорации уже перенесли огромные кипы своих документов на страницы WWW-серверов, и сотрудники этих фирм, разбросанные по всему миру, используют информацию этих серверов через Internet или intranet. Получая легкий и более полный доступ к информации, сотрудники принимают решение быстрее, и качество этого решения, как правило, выше.

Использование сети приводит к *совершенствованию коммуникаций*, то есть к улучшению процесса обмена информацией и взаимодействия между сотрудниками предприятия, а также его клиентами и поставщиками. Сети снижают потребность предприятий в других формах передачи информации, таких как телефон или обычная почта. Зачастую именно возможность организации электронной почты является основной причиной и экономическим обоснованием развертывания на предприятии вычислительной сети. Все большее распространение получают новые технологии, которые позволяют передавать по сетевым каналам связи не только компьютерные данные, но голосовую и видеоинформацию. Корпоративная сеть, которая интегрирует данные и мультимедийную информацию, может использоваться для организации аудио- и видеоконференций, кроме того, на ее основе может быть создана собственная внутренняя телефонная сеть.

Конечно, вычислительные сети имеют и свои проблемы. Эти проблемы в основном связаны с организацией эффективного взаимодействия отдельных частей распределенной системы.

Во-первых, это сложности, связанные с программным обеспечением – операционными системами и приложениями. Программирование для распределенных систем принципиально отличается от программирования для централизованных систем. Так, сетевая операционная система, выполняя в общем случае все функции по управлению локальными ресурсами компьютера, сверх того решает многочисленные задачи по предоставлению сетевых служб. Разработка сетевых приложений осложняется из-за необходимости организовать совместную работу их частей, выполняющихся на разных машинах. Много забот доставляет обеспечение совместимости программного обеспечения.

Во-вторых, много проблем связано с транспортировкой сообщений по каналам связи между компьютерами. Основные задачи здесь – обеспечение надежности (чтобы передаваемые данные не терялись и не искажались) и производительности (чтобы обмен данными происходил с приемлемыми задержками). В структуре общих затрат на вычислительную сеть расходы на решение «транспортных вопросов» составляют существенную часть, в то время как в централизованных системах эти проблемы полностью отсутствуют.

В-третьих, это вопросы, связанные с обеспечением безопасности, которые гораздо сложнее решаются в вычислительной сети, чем в централизованной системе. В некоторых случаях, когда безопасность особенно важна, от использования сети лучше вообще отказаться.

Можно приводить еще много «за» и «против» использования сетей, но главным доказательством эффективности является бесспорный факт их повсеместного распространения. Трудно найти сколь-

нибудь крупное предприятие, на котором не было хотя бы односегментной сети персональных компьютеров; все больше и больше появляется крупных сетей с сотнями рабочих станций и десятками серверов, некоторые большие организации и предприятия обзаводятся частными глобальными сетями, объединяющими их филиалы, удаленные на тысячи километров. В каждом конкретном случае для создания сети были свои резоны, но верно и общее утверждение: что-то в этих сетях все-таки есть.

Выводы

- Вычислительные сети явились результатом эволюции компьютерных технологий.
- Вычислительная сеть – это совокупность компьютеров, соединенных линиями связи. Линии связи образованы кабелями, сетевыми адаптерами и другими коммуникационными устройствами. Все сетевое оборудование работает под управлением системного и прикладного программного обеспечения.
- Основная цель сети – обеспечить пользователям сети потенциальную возможность совместного использования ресурсов всех компьютеров.
- Вычислительная сеть – это одна из разновидностей распределенных систем, достоинством которых является возможность распараллеливания вычислений, за счет чего может быть достигнуто повышение производительности и отказоустойчивости системы.
- Важнейший этап в развитии сетей – появление стандартных сетевых технологий типа Ethernet, позволяющих быстро и эффективно объединять компьютеры различных типов.
- Использование вычислительных сетей дает предприятию следующие возможности:
    - разделение дорогостоящих ресурсов;
    - совершенствование коммуникаций;
    - улучшение доступа к информации;
    - быстрое и качественное принятие решений;
    - свобода в территориальном размещении компьютеров.

1.2. Основные проблемы построения сетей

При создании вычислительных сетей их разработчикам пришлось решить много проблем. В этом разделе мы рассмотрим только наиболее важные из них, причем в той последовательности, в которой они естественно возникали в процессе развития и совершенствования сетевых технологий.

Механизмы взаимодействия компьютеров в сети многое позаимствовали у схемы взаимодействия компьютера с периферийными устройствами, поэтому начнем рассмотрение принципов работы сети с этого «досетевого» случая.

1.2.1. Связь компьютера с периферийными устройствами

Для обмена данными между компьютером и периферийным устройством (ПУ) в компьютере предусмотрен внешний *интерфейс* (рис. 1.6), то есть набор проводов, соединяющих компьютер и периферийное устройство, а также набор правил обмена информацией по этим проводам (иногда вместо термина *интерфейс* употребляется термин *протокол* – подробней об этих важных терминах мы еще поговорим). Примерами интерфейсов, используемых в компьютерах, являются параллельный интерфейс Centronics, предназначенный, как правило, для подключения принтеров, и последовательный интерфейс RS-232C, через который подключаются мышь, модем и много других устройств. Интерфейс реализуется со стороны компьютера совокупностью аппаратных и программных средств: контроллером ПУ и специальной программой, управляющей этим контроллером, которую часто называют *драйвером* соответствующего периферийного устройства.

Со стороны ПУ интерфейс чаще всего реализуется аппаратным устройством управления, хотя встречаются и программно-управляемые периферийные устройства.

Программа, выполняемая процессором, может обмениваться данными с помощью команд ввода/вывода с любыми модулями, подключенными к внутренней шине компьютера, в том числе и с контроллерами ПУ.

Периферийные устройства могут принимать от компьютера как данные, например байты информации, которую нужно распечатать на бумаге, так и команды управления, в ответ на которые ПУ

может выполнить специальные действия, например перевести головку диска на требуемую дорожку или же вытолкнуть лист бумаги из принтера. Периферийное устройство использует внешний интерфейс компьютера не только для приема информации, но и для передачи информации в компьютер, то есть обмен данными по внешнему интерфейсу, как правило, является двунаправленным. Так, например, даже принтер, который по своей природе является устройством вывода информации, возвращает в компьютер данные о своем состоянии.

Контроллеры ПУ принимают команды и данные от процессора в свой внутренний буфер, который часто называется регистром или портом, затем выполняют необходимые преобразования этих данных и команд в соответствии с форматами, понятными ПУ, и выдают их на внешний интерфейс.

Распределение обязанностей между контроллером и драйвером ПУ может быть разным, но обычно контроллер выполняет набор простых команд по управлению ПУ, а драйвер использует эти команды, чтобы заставить устройство совершать более сложные действия по некоторому алгоритму. Например, контроллер принтера может поддерживать такие элементарные команды, как «Печать символа», «Перевод строки», «Возврат каретки» и т. п. Драйвер же принтера с помощью этих команд организует печать строк символов, разделение документа на страницы и другие более высокоуровневые операции. Для одного и того же контроллера можно разработать различные драйверы, которые будут управлять данным ПУ поразному – одни лучше, а другие хуже – в зависимости от опыта и способностей программистов, их разработавших.

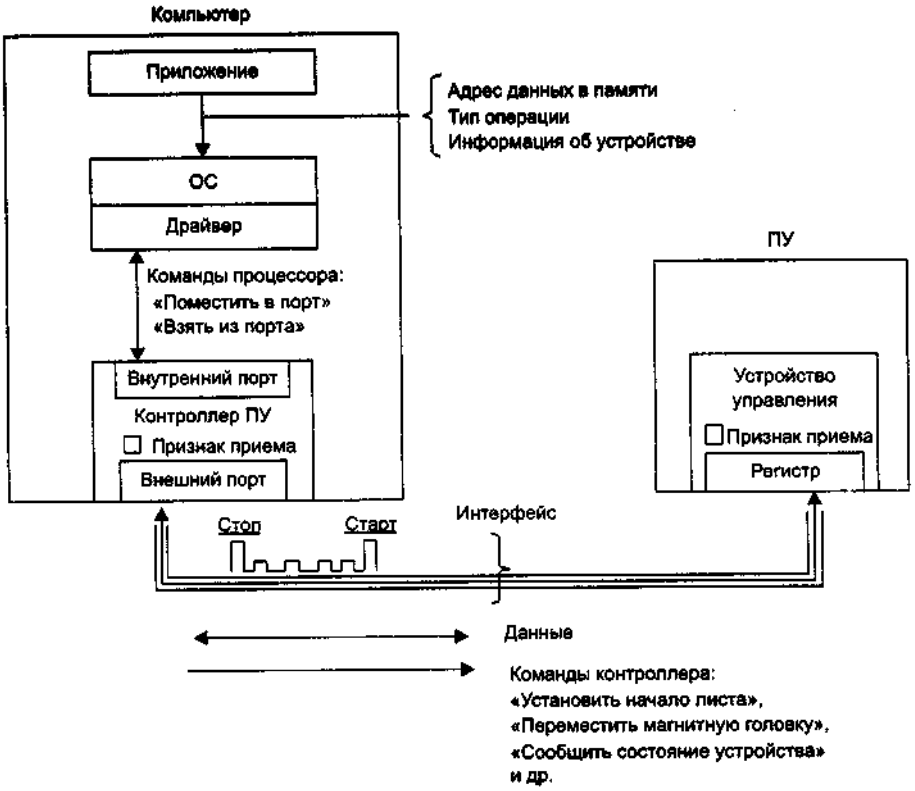


Рис. 1.6. Связь компьютера с периферийным устройством

Рассмотрим схему передачи одного байта информации от прикладной программы на периферийное устройство. Программа, которой потребовалось выполнить обмен данными с ПУ, обращается к драйверу этого устройства, сообщая ему в качестве параметра адрес байта памяти, который нужно передать. Драйвер загружает значение этого байта в буфер контроллера ПУ, который начинает последовательно передавать биты в линию связи, представляя каждый бит соответствующим электрическим сигналом. Чтобы устройству управления ПУ стало понятно, что начинается передача байта, перед передачей первого бита информации контроллер ПУ формирует стартовый сигнал специфической формы, а после передачи последнего информационного бита – стоповый сигнал. Эти сигналы *синхронизируют* передачу байта.

Кроме информационных бит, контроллер может передавать бит контроля четности для повышения достоверности обмена. Устройство управления, обнаружив на соответствующей линии стартовый бит, выполняет подготовительные действия и начинает принимать информационные биты, формируя из них

байт в своем приемном буфере. Если передача сопровождается битом четности, то выполняется проверка правильности передачи: при правильно выполненной передаче в соответствующем регистре устройства управления устанавливается признак завершения приема информации.

Обычно на драйвер возлагаются наиболее сложные функции протокола (например, подсчет контрольной суммы последовательности передаваемых байтов, анализ состояния периферийного устройства, проверка правильности выполнения команды). Но даже самый примитивный драйвер контроллера должен поддерживать как минимум две операции: «Взять данные из контроллера в оперативную память» и «Передать данные из оперативной памяти в контроллер».

Существуют как весьма специализированные интерфейсы, пригодные для подключения узкого класса устройств (например, графических мониторов высокого разрешения фирмы Vista), так и интерфейсы общего назначения, являющиеся стандартными и позволяющие подключать различные периферийные устройства. Примером такого интерфейса является интерфейс RS-232C, который поддерживается многими терминалами, принтерами, графопостроителями, манипуляторами типа «мышь» и многими другими устройствами.

1.2.2. Простейший случай взаимодействия двух компьютеров

В самом простом случае взаимодействие компьютеров может быть реализовано с помощью тех же самых средств, которые используются для взаимодействия компьютера с периферией, например, через последовательный интерфейс RS-232C. В отличие от взаимодействия компьютера с периферийным устройством, когда программа работает, как правило, только с одной стороны – со стороны компьютера, в этом случае происходит взаимодействие двух программ, работающих на каждом из компьютеров.

Программа, работающая на одном компьютере, не может получить непосредственный доступ к ресурсам другого компьютера – его дискам, файлам, принтеру. Она может только «попросить» об этом программу, работающую на том компьютере, которому принадлежат эти ресурсы. Эти «просьбы» выражаются в виде *сообщений*, передаваемых по каналам связи между компьютерами. Сообщения могут содержать не только команды на выполнение некоторых действий, но и собственно информационные данные (например, содержимое некоторого файла).

Рассмотрим случай, когда пользователю, работающему с текстовым редактором на персональном компьютере А, нужно прочитать часть некоторого файла, расположенного на диске персонального компьютера В (рис. 1.7), Предположим, что мы связали эти компьютеры по кабелю связи через COM-порты, которые, как известно, реализуют интерфейс RS-232C (такое соединение часто называют нуль-модемным). Пусть для определенности компьютеры работают под управлением MS-DOS, хотя принципиального значения в данном случае это не .имеет.

Драйвер COM-порта вместе с контроллером COM-порта работают примерно так же, как и в описанном выше случае взаимодействия ПУ с компьютером. Однако при этом роль устройства управления ПУ выполняет контроллер и драйвер COM-порта другого компьютера. Вместе они обеспечивают передачу по кабелю между компьютерами одного байта информации. (В «настоящих» локальных сетях подобные функции передачи данных в линию связи выполняются сетевыми адаптерами и их драйверами.)

Драйвер компьютера В периодически опрашивает признак завершения приема, устанавливаемый контроллером при правильно выполненной передаче данных, и при его появлении считывает принятый байт из буфера контроллера в оперативную память, делая его тем самым доступным для программ компьютера В. В некоторых случаях драйвер вызывается асинхронно, по прерываниям от контроллера.

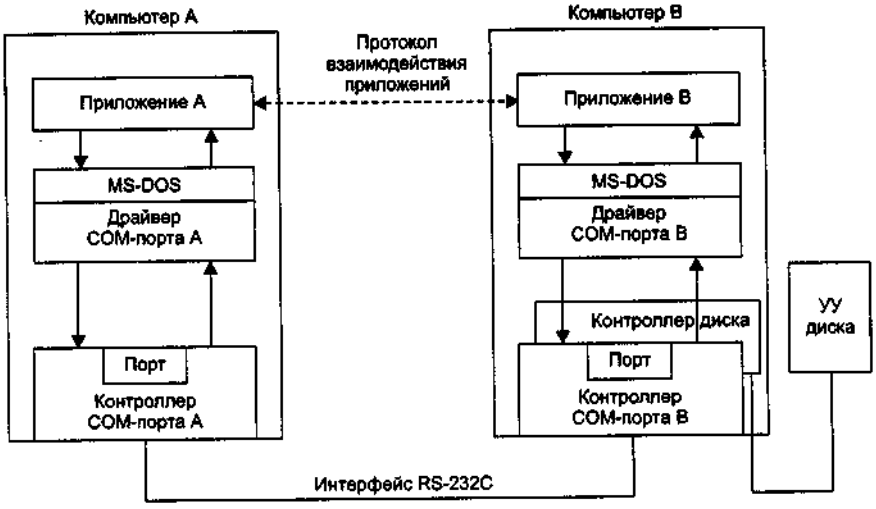


Рис. 1.7. Взаимодействие двух компьютеров

Таким образом, в распоряжении программ компьютеров А и В имеется средство для передачи одного байта информации. Но рассматриваемая в нашем примере задача значительно сложнее, так как нужно передать не один байт, а определенную часть заданного файла. Все связанные с этим дополнительные проблемы должны решить программы более высокого уровня, чем драйверы COM-портов. Для определенности назовем такие программы компьютеров А и В приложением А и приложением В соответственно. Итак, приложение А должно сформировать сообщение-запрос для приложения В. В запросе необходимо указать имя файла, тип операции (в данном случае – чтение), смещение и размер области файла, содержащей нужные данные.

Чтобы передать это сообщение компьютеру В, приложение А обращается к драйверу COM-порта, сообщая ему адрес в оперативной памяти, по которому драйвер находит сообщение и затем передает его байт за байтом приложению В. Приложение В, приняв запрос, выполняет его, то есть считывает требуемую область файла с диска с помощью средств локальной ОС в буферную область своей оперативной памяти, а далее с помощью драйвера COM-порта передает считанные данные по каналу связи в компьютер А, где они и попадают к приложению А.

Описанные функции приложения А могла бы выполнить сама программа текстового редактора, но включать эти функции в состав каждого приложения – текстовых редакторов, графических редакторов, систем управления базами данных и других приложений, которым нужен доступ к файлам, – не очень рационально (хотя существует большое количество программ, которые действительно самостоятельно решают все задачи по межмашинному обмену данными, например Kermit – программа обмена файлами через COM-порты, реализованная для различных ОС, Norton Commander 3.0 с его функцией Link). Гораздо выгоднее создать специальный программный модуль, который будет выполнять функции формирования сообщений-запросов и приема результатов для всех приложений компьютера. Как уже было ранее сказано, такой служебный модуль называется клиентом. На стороне же компьютера В должен работать другой модуль – сервер, постоянно ожидающий прихода запросов на удаленный доступ к файлам, расположенным на диске этого компьютера. Сервер, приняв запрос из сети, обращается к локальному файлу и выполняет с ним заданные действия, возможно, с участием локальной ОС.

Программные клиент и сервер выполняют системные функции по обслуживанию запросов приложений компьютера А на удаленный доступ к файлам компьютера В. Чтобы приложения компьютера В могли пользоваться файлами компьютера А, описанную схему нужно симметрично дополнить клиентом для компьютера В и сервером для компьютера А.

Схема взаимодействия клиента и сервера с приложениями и операционной системой приведена на рис. 1.8. Несмотря на то что мы рассмотрели очень простую схему аппаратной связи компьютеров, функции программ, обеспечивающих доступ к удаленным файлам, очень похожи на функции модулей сетевой операционной системы, работающей в сети с более сложными аппаратными связями компьютеров.

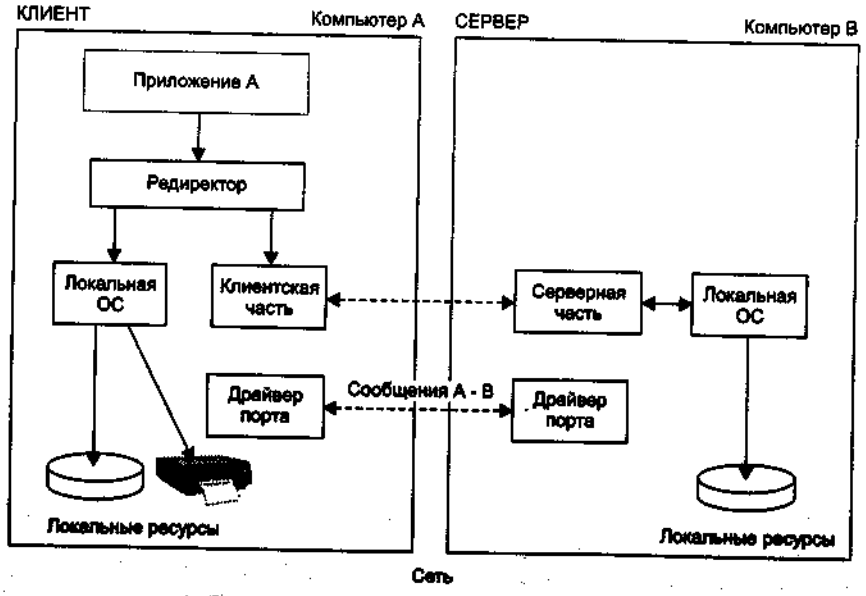


Рис. 1.8. Взаимодействие программных компонентов при связи двух компьютеров

Очень удобной и полезной функцией клиентской программы является способность отличить запрос к удаленному файлу от запроса к локальному файлу. Если клиентская программа умеет это делать, то приложения не должны заботиться о том, с каким файлом они работают (локальным или удаленным), клиентская программа сама распознает и *перенаправляет* (*redirect*) запрос к удаленной машине. Отсюда и название, часто используемое для клиентской части сетевой ОС, – *редиректор*. Иногда функции распознавания выделяются в отдельный программный модуль, в этом случае редиректором называют не всю клиентскую часть, а только этот модуль.

1.2.3. Проблемы физической передачи данных по линиям связи

Даже при рассмотрении простейшей сети, состоящей всего из двух машин, можно увидеть многие проблемы, присущие любой вычислительной сети, в том числе проблемы, связанные с физической передачей сигналов по линиям связи, без решения которой невозможен любой вид связи.

В вычислительной технике для представления данных используется двоичный код. Внутри компьютера единицам и нулям данных соответствуют дискретные электрические сигналы. Представление данных в виде электрических или оптических сигналов называется *кодированием*. Существуют различные способы кодирования двоичных цифр 1 и 0, например, потенциальный способ, при котором единице соответствует один уровень напряжения, а нулю – другой, или импульсный способ, когда для представления цифр используются импульсы различной или одной полярности.

Аналогичные подходы могут быть использованы для кодирования данных и при передаче их между двумя компьютерами по линиям связи. Однако эти линии связи отличаются по своим электрическим характеристикам от тех, которые существуют внутри компьютера. Главное отличие внешних линий связи от внутренних состоит в их гораздо большей протяженности, а также в том, что они проходят вне экранированного корпуса по пространствам, зачастую подверженным воздействию сильных электромагнитных помех. Все это приводит к значительно большим искажениям прямоугольных импульсов (например, «заваливанию» фронтов), чем внутри компьютера. Поэтому для надежного распознавания импульсов на приемном конце линии связи при передаче данных внутри и вне компьютера не всегда можно использовать одни и те же скорости и способы кодирования. Например, медленное нарастание фронта импульса из-за высокой емкостной нагрузки линии требует передачи импульсов с меньшей скоростью (чтобы передний и задний фронты соседних импульсов не перекрывались и импульс успел дорасти до требуемого уровня).

В вычислительных сетях применяют как потенциальное, так и импульсное кодирование дискретных данных, а также специфический способ представления данных, который никогда не используется внутри компьютера, – *модуляцию* (рис. 1.9). При модуляции дискретная информация представляется синусоидальным сигналом той частоты, которую хорошо передает имеющаяся линия связи.

Потенциальное или импульсное кодирование применяется, на каналах высокого качества, а модуляция на основе синусоидальных сигналов предпочтительнее в том случае, когда канал вносит сильные искажения в передаваемые сигналы. Обычно модуляция используется в глобальных сетях при передаче данных через аналоговые телефонные каналы связи, которые были разработаны для передачи голоса в аналоговой форме и поэтому плохо подходят для непосредственной передачи импульсов.

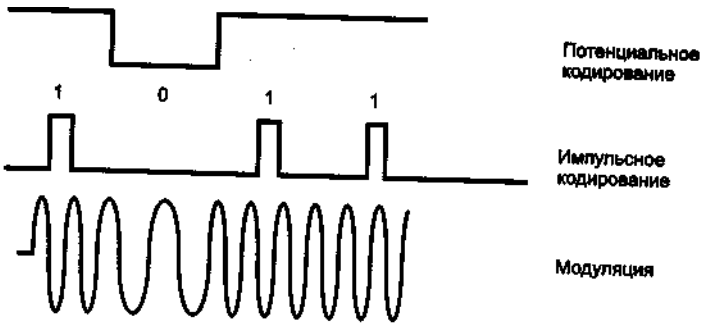


Рис. 1.9. Примеры представления дискретной информации

На способ передачи сигналов влияет и количество проводов в линиях связи между компьютерами. Для сокращения стоимости линий связи в сетях обычно стремятся к сокращению количества проводов и из-за этого используют не параллельную передачу всех бит одного байта или даже нескольких байт, как это делается внутри компьютера, а последовательную, побитную передачу, требующую всего одной пары проводов.

Еще одной проблемой, которую нужно решать при передаче сигналов, является проблема взаимной *синхронизации* передатчика одного компьютера с приемником другого. При организации взаимодействия модулей внутри компьютера эта проблема решается очень просто, так как в этом случае все модули синхронизируются от общего тактового генератора. Проблема синхронизации при связи компьютеров может решаться разными способами, как с помощью обмена специальными тактовыми синхроимпульсами по отдельной линии, так и с помощью периодической синхронизации заранее обусловленными кодами или импульсами характерной формы, отличающейся от формы импульсов данных.

Несмотря на предпринимаемые меры – выбор соответствующей скорости обмена данными, линий связи с определенными характеристиками, способа синхронизации приемника и передатчика, – существует вероятность искажения некоторых бит передаваемых данных. Для повышения надежности передачи данных между компьютерами часто используется стандартный прием – подсчет *контрольной суммы* и передача ее по линиям связи после каждого байта или после некоторого блока байтов. Часто в протокол обмена данными включается как обязательный элемент сигнал-квитанция, который подтверждает правильность приема данных и посылается от получателя отправителю.

Задачи надежного обмена двоичными сигналами, представленными соответствующими электромагнитными сигналами, в вычислительных сетях решает определенный класс оборудования. В локальных сетях это *сетевые адаптеры*, а в глобальных сетях – аппаратура передачи данных, к которой относятся, например, устройства, выполняющие модуляцию и демодуляцию дискретных сигналов, - *модемы*. Это оборудование кодирует и декорирует каждый информационный бит, синхронизирует передачу электромагнитных сигналов по линиям связи, проверяет правильность передачи по контрольной сумме и может выполнять некоторые другие операции. Сетевые адаптеры рассчитаны, как правило, на работу с определенной *передающей средой* – коаксиальным кабелем, витой парой, оптоволокном и т. п. Каждый тип передающей среды обладает определенными электрическими характеристиками, влияющими на способ использования данной среды, и определяет скорость передачи сигналов, способ их кодирования и некоторые другие параметры.

1.2.4. Проблемы объединения нескольких компьютеров

До сих пор мы рассматривали вырожденную сеть, состоящую всего из двух машин. При объединении в сеть большего числа компьютеров возникает целый комплекс новых проблем.

Топология физических связей

В первую очередь необходимо выбрать способ организации физических связей, то есть *топологию*. Под топологией вычислительной сети понимается конфигурация графа, вершинам которого соответствуют компьютеры сети (иногда и другое оборудование, например концентраторы), а ребрам – физические связи между ними. Компьютеры, подключенные к сети, часто называют *станциями* или *узлами сети.*

Заметим, что конфигурация *физических связей* определяется электрическими соединениями компьютеров между собой и может отличаться от конфигурации *логических связей* между узлами сети. Логические связи представляют собой маршруты передачи данных между узлами сети и образуются путем соответствующей настройки коммуникационного оборудования.

Выбор топологии электрических связей существенно влияет на многие характеристики сети. Например, наличие резервных связей повышает надежность сети и делает возможным балансирование загрузки отдельных каналов. Простота присоединения новых узлов, свойственная некоторым топологиям, делает сеть легко расширяемой. Экономические соображения часто приводят к выбору топологий, для которых характерна минимальная суммарная длина линий связи.

Рассмотрим некоторые, наиболее часто встречающиеся топологии.

Полносвязная топология (рис. 1.10, *а*) соответствует сети, в которой каждый компьютер сети связан со всеми остальными. Несмотря на логическую простоту, этот вариант оказывается громоздким и неэффективным. Действительно, каждый компьютер в сети должен иметь большое количество коммуникационных портов, достаточное для связи с каждым из остальных компьютеров сети. Для каждой пары компьютеров должна быть выделена отдельная электрическая линия связи. Полносвязные топологии применяются редко, так как не удовлетворяют ни одному из приведенных выше требований. Чаще этот вид топологии используется в многомашинных комплексах или глобальных сетях при небольшом количестве компьютеров.

Все другие варианты основаны на неполносвязных топологиях, когда для обмена данными между двумя компьютерами может потребоваться промежуточная передача данных через другие узлы сети.

Ячеистая топология (*mesh*) получается из полносвязной путем удаления некоторых возможных связей (рис. 1.10,*б*). В сети с ячеистой топологией непосредственно связываются только те компьютеры, между которыми происходит интенсивный обмен данными, а для обмена данными между компьютерами, не соединенными прямыми связями, используются транзитные передачи через промежуточные узлы. Ячеистая топология допускает соединение большого количества компьютеров и характерна, как правило, для глобальных сетей.

Общая шина (рис. 1.10, *в*) является очень распространенной (а до недавнего времени самой распространенной) топологией для локальных сетей. В этом случае компьютеры подключаются к одному коаксиальному кабелю по схеме «монтажного ИЛИ». Передаваемая информация может распространяться в обе стороны. Применение общей шины снижает стоимость проводки, унифицирует подключение различных модулей, обеспечивает возможность почти мгновенного широковещательного обращения ко всем станциям сети. Таким образом, основными преимуществами такой схемы являются дешевизна и простота разводки кабеля по помещениям. Самый серьезный недостаток общей шины заключается в ее низкой надежности: любой дефект кабеля или какого-нибудь из многочисленных разъемов полностью парализует всю сеть. К сожалению, дефект коаксиального разъема редкостью не является. Другим недостатком общей шины является ее невысокая производительность, так как при таком способе подключения в каждый момент времени только один компьютер может передавать данные в сеть. Поэтому пропускная способность канала связи всегда делится здесь между всеми узлами сети.

Топология *звезда* (рис. 1.10, *г*). В этом случае каждый компьютер подключается отдельным кабелем к общему устройству, называемому *концентратором*, который находится в центре сети. В функции концентратора входит направление передаваемой компьютером информации одному или всем остальным компьютерам сети. Главное преимущество этой топологии перед общей шиной – существенно большая надежность. Любые неприятности с кабелем касаются лишь того компьютера, к которому этот кабель присоединен, и только неисправность концентратора может вывести из строя всю сеть. Кроме того, концентратор может играть роль интеллектуального фильтра информации, поступающей от узлов в сеть, и при необходимости блокировать запрещенные администратором передачи.

К недостаткам топологии типа звезда относится более высокая стоимость сетевого оборудования из-за необходимости приобретения концентратора. Кроме того, возможности по наращиванию количества узлов в сети ограничиваются количеством портов концентратора. Иногда имеет смысл строить сеть с использованием нескольких концентраторов, иерархически соединенных между собой связями типа звезда (рис. 1.10, *д*). В настоящее время иерархическая звезда является самым распространенным типом топологии связей как в локальных, так и глобальных сетях.

В сетях с *кольцевой* конфигурацией (рис. 1.10, *е*) данные передаются по кольцу от одного компьютера к другому, как правило, в одном направлении. Если компьютер распознает данные как «свои», то он копирует их себе во внутренний буфер. В сети с кольцевой топологией необходимо принимать специальные меры, чтобы в случае выхода из строя или отключения какой-либо станции не прервался канал связи между остальными станциями. Кольцо представляет собой очень удобную конфигурацию для организации обратной связи – данные, сделав полный оборот, возвращаются к узлу-источнику. Поэтому этот узел может контролировать процесс доставки данных адресату. Часто это свойство кольца используется для тестирования связности сети и поиска узла, работающего некорректно. Для этого в сеть посылаются специальные тестовые сообщения.

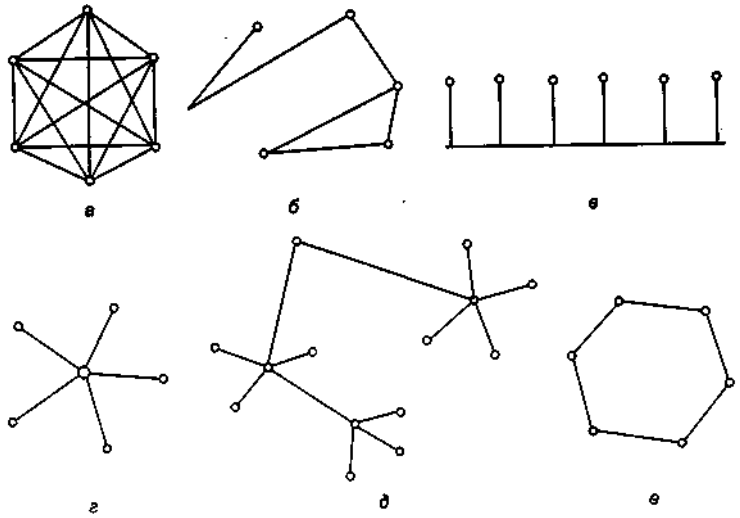


Рис. 1.10. Типовые топологии сетей

В то время как небольшие сети, как правило, имеют типовую топологию – звезда, кольцо или общая шина, для крупных сетей характерно наличие произвольных связей между компьютерами. В таких сетях можно выделить отдельные произвольно связанные фрагменты (подсети), имеющие типовую топологию, поэтому их называют сетями со *смешанной топологией* (рис. 1.11).

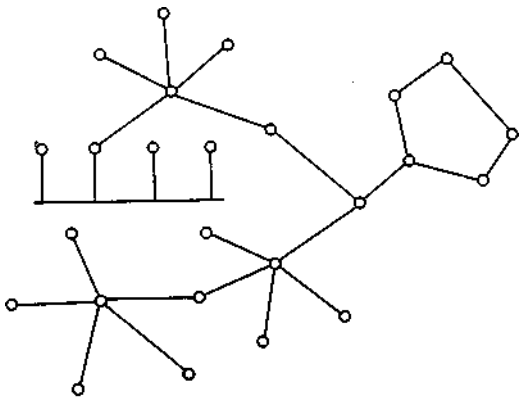


Рис. 1.11. Смешанная топология

Организация совместного использования линий связи

Только в сети с полносвязной топологией для соединения каждой пары компьютеров имеется отдельная линия связи. Во всех остальных случаях неизбежно возникает вопрос о том, как организовать

совместное использование линий связи несколькими компьютерами сети. Как и всегда при разделении ресурсов, главной целью здесь является удешевление сети.

В вычислительных сетях используют как индивидуальные линии связи между компьютерами, так и *разделяемые* (*shared*), когда одна линия связи попеременно используется несколькими компьютерами. В случае применения разделяемых линий связи (часто используется также термин разделяемая среда передачи данных – shared media) возникает комплекс проблем, связанных с их совместным использованием, который включает как чисто электрические проблемы обеспечения нужного качества сигналов при подключении к одному и тому же проводу нескольких приемников и передатчиков, так и логические проблемы разделения во времени доступа к этим линиям.

Классическим примером сети с разделяемыми линиями связи являются сети с топологией «общая шина», в которых один кабель совместно используется всеми компьютерами сети. Ни один из компьютеров сети в принципе не может индивидуально, независимо от всех других компьютеров сети, использовать кабель, так как при одновременной передаче данных сразу несколькими узлами сигналы смешиваются и искажаются. В топологиях «кольцо» или «звезда» индивидуальное использование линий связи, соединяющих компьютеры, принципиально возможно, но эти кабели часто также рассматривают как разделяемые для всех компьютеров сети, так что, например, только один компьютер кольца имеет право в данный момент времени отправлять по кольцу пакеты другим компьютерам.

Существуют различные способы решения задачи организации совместного доступа к разделяемым линиям связи. Внутри компьютера проблемы разделения линий связи между различными модулями также существуют – примером является доступ к системной шине, которым управляет либо процессор, либо специальный арбитр шины. В сетях организация совместного доступа к линиям связи имеет свою специфику из-за существенно большего времени распространения сигналов по длинным проводам, к тому же это время для различных пар компьютеров может быть различным. Из-за этого процедуры согласования доступа к линии связи могут занимать слишком большой промежуток времени и приводить к значительным потерям производительности сети.

Несмотря на все эти сложности, в локальных сетях разделяемые линии связи используются очень часто. Этот подход, в частности, реализован в широко распространенных классических технологиях Ethernet и Token Ring. Однако в последние годы наметилась тенденция отказа от разделяемых сред передачи данных и в локальных сетях. Это связано с тем, что за достигаемое таким образом удешевление сети приходится расплачиваться производительностью.

Сеть с разделяемой средой при большом количестве узлов будет работать всегда медленнее, чем аналогичная сеть с индивидуальными линиями связи, так как пропускная способность индивидуальной линии связи достается одному компьютеру, а при ее совместном использовании делится на все компьютеры сети. Часто с такой потерей производительности мирятся ради увеличения экономической эффективности сети. Не только в классических, но и в совеем новых технологиях, разработанных для локальных сетей, сохраняется режим разделяемых линий связи. Например, разработчики технологии Gigabit Ethernet, принятой в 1998 году в качестве нового стандарта, включили режим разделения передающей среды в свои спецификации наряду с режимом работы по индивидуальным линиям связи.

При использовании индивидуальных линий связи в полносвязных топологиях конечные узлы должны иметь по одному порту на каждую линию связи. В звездообразных топологиях конечные узлы могут подключаться индивидуальными линиями связи к специальному устройству – коммутатору. В глобальных сетях коммутаторы использовались уже на начальном этапе, а в локальных сетях – с начала 90-х годов. Коммутаторы приводят к существенному удорожанию локальной сети, поэтому пока их применение ограничено, но по мере снижения стоимости коммутации этот подход, возможно, вытеснит применение разделяемых линий связи. Необходимо подчеркнуть, что индивидуальными в таких сетях являются только линии связи между конечными узлами и коммутаторами сети, а связи между коммутаторами остаются разделяемыми, так как по ним передаются сообщения разных конечных узлов (рис. 1.12)

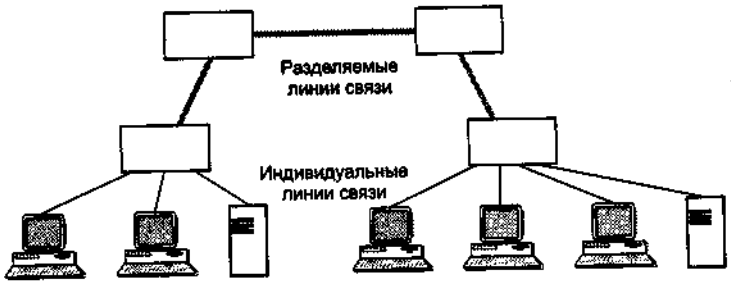


Рис. 1.12. Индивидуальные и разделяемые линии связи в сетях на основе коммутаторов

В глобальных сетях отказ от разделяемых линий связи объясняется техническими причинами. Здесь большие временные задержки распространения сигналов принципиально ограничивают применимость техники разделения линии связи. Компьютеры могут затратить больше времени на переговоры о том, кому сейчас можно использовать линию связи, чем непосредственно на передачу данных по этой линии связи. Однако это не относится к линиям связи типа «коммутатор - коммутатор». В этом случае только два коммутатора борются за доступ к линии связи, и это существенно упрощает задачу организации совместного использования линии.

Адресация компьютеров

Еще одной новой проблемой, которую нужно учитывать при объединении трех и более компьютеров, является проблема их адресации. К адресу узла сети и схеме его назначения можно предъявить несколько требований.

- Адрес должен уникально идентифицировать компьютер в сети любого масштаба.
- Схема назначения адресов должна сводить к минимуму ручной труд администратора и вероятность дублирования адресов.
- Адрес должен иметь иерархическую структуру, удобную для построения больших сетей. Эту проблему хорошо иллюстрируют международные почтовые адреса, которые позволяют почтовой службе, организующей доставку писем между странами, пользоваться только названием страны адресата и не учитывать название его города, а тем более улицы. В больших сетях, состоящих из многих тысяч узлов, отсутствие иерархии адреса может привести к большим издержкам – конечным узлам и коммуникационному оборудованию придется оперировать с таблицами адресов, состоящими из тысяч записей.
- Адрес должен быть удобен для пользователей сети, а это значит, что он должен иметь символьное представление например, Server3 или www.cisco.com.
- Адрес должен иметь по возможности компактное представление, чтобы не перегружать память коммуникационной аппаратуры – сетевых адаптеров, маршрутизаторов и т. п.

Нетрудно заметить, что эти требования противоречивы – например, адрес, имеющий иерархическую структуру, скорее всего будет менее компактным, чем неиерархический (такой адрес часто называют «плоским», то есть не имеющим структуры). Символьный же адрес скорее всего потребует больше памяти, чем адрес-число.

Так как все перечисленные требования трудно совместить в рамках какой-либо одной схемы адресации, то на практике обычно используется сразу несколько схем, так что компьютер одновременно имеет несколько адресов-имен. Каждый адрес используется в той ситуации, когда соответствующий вид адресации наиболее удобен. А чтобы не возникало путаницы и компьютер всегда однозначно определялся своим адресом, используются специальные вспомогательные протоколы, которые по адресу одного типа могут определить адреса других типов.

Наибольшее распространение получили три схемы адресации узлов.

- *Аппаратные* (*hardware*) адреса. Эти адреса предназначены для сети небольшого или среднего размера, поэтому они не имеют иерархической структуры. Типичным представителем адреса такого типа является адрес сетевого адаптера локальной сети. Такой адрес обычно используется только аппаратурой, поэтому его стараются сделать по возможности компактным и записывают в виде двоичного или шестнадцатеричного значения, например 0081005e24a8. При задании аппаратных адресов обычно не требуется выполнение ручной

работы, так как они либо встраиваются в аппаратуру компанией-изготовителем, либо генерируются автоматически при каждом новом запуске оборудования, причем уникальность адреса в пределах сети обеспечивает оборудование. Помимо отсутствия иерархии, использование аппаратных адресов связано еще с одним недостатком – при замене аппаратуры, например, сетевого адаптера, изменяется и адрес компьютера. Более того, при установке нескольких сетевых адаптеров у компьютера появляется несколько адресов, что не очень удобно для пользователей сети.

- *Символьные адреса* или *имена*. Эти адреса предназначены для запоминания людьми и поэтому обычно несут смысловую нагрузку. Символьные адреса легко использовать как в небольших, так и крупных сетях. Для работы в больших сетях символьное имя может иметь сложную иерархическую структуру, например ftp-arch1.ucl.ac.uk. Этот адрес говорит о том, что данный компьютер поддерживает ftp-архив в сети одного из колледжей Лондонского университета (University College London – ucl) и эта сеть относится к академической ветви (ac) Internet Великобритании (United Kingdom – uk). При работе в пределах сети Лондонского университета такое длинное символьное имя явно избыточно и вместо него удобно пользоваться кратким символьным именем, на роль которого хорошо подходит самая младшая составляющего полного имени, то есть имя ftp-arch1.
- *Числовые составные адреса*. Символьные имена удобны для людей, но из-за переменного формата и потенциально большой длины их передача по сети не очень экономична. Поэтому во многих случаях для работы в больших сетях в качестве адресов узлов используют числовые составные адреса фиксированного и компактного форматов. Типичным представителями адресов этого типа являются IP- и IPX-адреса. В них поддерживается двухуровневая иерархия, адрес делится на старшую часть – номер сети и младшую – номер узла. Такое деление позволяет передавать сообщения между сетями только на основании номера сети, а номер узла используется только после доставки сообщения в нужную сеть; точно так же, как название улицы используется почтальоном только после того, как письмо доставлено в нужный город. В последнее время, чтобы сделать маршрутизацию в крупных сетях более эффективной, предлагаются более сложные варианты числовой адресации, в соответствии с которыми адрес имеет три и более составляющих. Такой подход, в частности, реализован в новой версии протокола IPv6, предназначенного для работы в сети Internet.

В современных сетях для адресации узлов применяются, как правило, одновременно все три приведенные выше схемы. Пользователи адресуют компьютеры символьными именами, которые автоматически заменяются в сообщениях, передаваемых по сети, на числовые номера. С помощью этих числовых номеров сообщения передаются из одной сети в другую, а после доставки сообщения в сеть назначения вместо числового номера используется аппаратный адрес компьютера. Сегодня такая схема характерна даже для небольших автономных сетей, где, казалось бы, она явно избыточна – это делается для того, чтобы при включении этой сети в большую сеть не нужно было менять состав операционной системы.

Проблема установления соответствия между адресами различных типов, которой занимается *служба разрешения имен*, может решаться как полностью централизованными, так и распределенными средствами. В случае централизованного подхода в сети выделяется один компьютер (сервер имен), в котором хранится таблица соответствия друг другу имен различных типов, например символьных имен и числовых номеров. Все остальные компьютеры обращаются к серверу имен, чтобы по символьному имени найти числовой номер компьютера, с которым необходимо обменяться данными.

При другом, распределенном подходе, каждый компьютер сам решает задачу установления соответствия между именами. Например, если пользователь указал для узла назначения числовой номер, то перед началом передачи данных компьютер-отправитель посылает всем компьютерам сети сообщение (такое сообщение называется широковещательным) с просьбой опознать это числовое имя. Все компьютеры, получив это сообщение, сравнивают заданный номер со своим собственным. Тот компьютер, у которого обнаружилось совпадение, посылает ответ, содержащий его аппаратный адрес, после чего становится возможным отправка сообщений по локальной сети.

Распределенный подход хорош тем, что не предполагает выделения специального компьютера, который к тому же часто требует ручного задания таблицы соответствия имен. Недостатком распределенного подхода является необходимость широковещательных сообщений – такие сообщения перегружают сеть, так как они требуют обязательной обработки всеми узлами, а не только узлом

назначения. Поэтому распределенный подход используется только в небольших локальных сетях. В крупных сетях распространение широковещательных сообщений по всем ее сегментам становится практически нереальным, поэтому для них характерен централизованный подход. Наиболее известной службой централизованного разрешения имен является служба Domain Name System (DNS) сети Internet.

1.2.5. Ethernet–пример стандартного решения сетевых проблем

Рассмотрим, каким образом описанные выше общие подходы к решению наиболее важных проблем построения сетей воплощены в наиболее популярной сетевой технологии – *Ethernet*.

Сетевая технология – это согласованный набор стандартных протоколов и реализующих их программно-аппаратных средств (например, сетевых адаптеров, драйверов, кабелей и разъемов), достаточный для построения вычислительной сети. Эпитет «достаточный» подчеркивает то обстоятельство, что этот набор представляет собой минимальный набор средств, с помощью которых можно построить работоспособную сеть. Возможно, эту сеть можно улучшить, например, за счет выделения в ней подсетей, что сразу потребует кроме протоколов стандарта Ethernet применения протокола IP, а также специальных коммуникационных устройств – маршрутизаторов. Улучшенная сеть будет, скорее всего, более надежной и быстродействующей, но за счет надстроек над средствами технологии Ethernet, которая составила базис сети.

Термин «сетевая технология» чаще всего используется в описанном выше узком смысле, но иногда применяется и его расширенное толкование как любого набора средств и правил для построения сети, например, «технология сквозной маршрутизации», «технология создания защищенного канала», «технология IP-сетей».

Протоколы, на основе которых строится сеть определенной технологии (в узком смысле), специально разрабатывались для совместной работы, поэтому от разработчика сети не требуется дополнительных усилий по организации их взаимодействия. Иногда сетевые технологии называют *базовыми технологиями*, имея в виду то, что на их основе строится базис любой сети. Примерами базовых сетевых технологий могут служить наряду с Ethernet такие известные технологии локальных сетей как, Token Ring и FDDI, или же технологии территориальных сетей X.25 и frame relay. Для получения работоспособной сети в этом случае достаточно приобрести программные и аппаратные средства, относящиеся к одной базовой технологии – сетевые адаптеры с драйверами, концентраторы, коммутаторы, кабельную систему и т. п., – и соединить их в соответствии с требованиями стандарта на данную технологию.

Стандарт Ethernet был принят в 1980 году. Число сетей, построенных на основе этой технологии, к настоящему моменту оценивается в 5 миллионов, а количество компьютеров, работающих в таких сетях, – в 50 миллионов.

Основной принцип, положенный в основу Ethernet, – *случайный метод доступа* к разделяемой среде передачи данных. В качестве такой среды может использоваться толстый или тонкий коаксиальный кабель, витая пара, оптоволокно или радиоволны (кстати, первой сетью, построенной на принципе случайного доступа к разделяемой среде, была радиосеть Aloha Гавайского университета).

В стандарте Ethernet строго зафиксирована топология электрических связей. Компьютеры подключаются к разделяемой среде в соответствии с типовой структурой «общая шина» (рис. 1.13). С помощью разделяемой во времени шины любые два компьютера могут обмениваться данными. Управление доступом к линии связи осуществляется специальными контроллерами – сетевыми адаптерами Ethernet. Каждый компьютер, а более точно, каждый сетевой адаптер, имеет уникальный адрес. Передача данных происходит со скоростью 10 Мбит/с. Эта величина является пропускной способностью сети Ethernet.

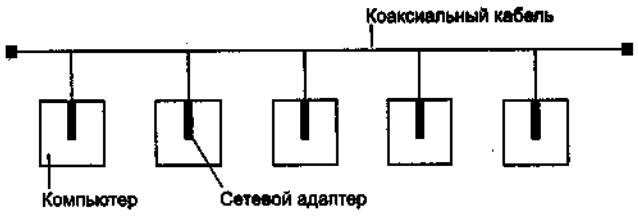


Рис. 1.13. Сеть Ethernet

Суть случайного метода доступа состоит в следующем. Компьютер в сети Ethernet может передавать данные по сети, только если сеть свободна, то есть если никакой другой компьютер в данный момент не занимается обменом. Поэтому важной частью технологии Ethernet является процедура определения доступности среды.

После того как компьютер убедился, что сеть свободна, он начинает передачу, при этом «захватывает» среду. Время монопольного использования разделяемой среды одним узлом ограничивается временем передачи одного кадра. *Кадр* – это единица данных, которыми обмениваются компьютеры в сети Ethernet. Кадр имеет фиксированный формат и наряду с полем данных содержит различную служебную информацию, например адрес получателя и адрес отправителя.

Сеть Ethernet устроена так, что при попадании кадра в разделяемую среду передачи данных все сетевые адаптеры одновременно начинают принимать этот кадр. Все они анализируют адрес назначения, располагающийся в одном из начальных полей кадра, и, если этот адрес совпадает с их собственным адресом, кадр помещается во внутренний буфер сетевого адаптера. Таким образом компьютер-адресат получает предназначенные ему данные.

Иногда может возникать ситуация, когда одновременно два или более компьютера решают, что сеть свободна, и начинают передавать информацию. Такая ситуация, называемая *коллизией*, препятствует правильной передаче данных по сети. В стандарте Ethernet предусмотрен алгоритм обнаружения и корректной обработки коллизий. Вероятность возникновения коллизии зависит от интенсивности сетевого трафика.

После обнаружения коллизии сетевые адаптеры, которые пытались передать свои кадры, прекращают передачу и после паузы случайной длительности пытаются снова получить доступ к среде и передать тот кадр, который вызвал коллизию.

Главным достоинством сетей Ethernet, благодаря которому они стали такими популярными, является их экономичность. Для построения сети достаточно иметь по одному сетевому адаптеру для каждого компьютера плюс один физический сегмент коаксиального кабеля нужной длины. Другие базовые технологии, например Token Ring, для создания даже небольшой сети требуют наличия дополнительного устройства – концентратора.

Кроме того, в сетях Ethernet реализованы достаточно простые алгоритмы доступа к среде, адресации и передачи данных. Простота логики работы сети ведет к упрощению и, соответственно, удешевлению сетевых адаптеров и их драйверов. По той же причине адаптеры сети Ethernet обладают высокой надежностью.

И наконец, еще одним замечательным свойством сетей Ethernet является их хорошая расширяемость, то есть легкость подключения новых узлов.

Другие базовые сетевые технологии – Token Ring, FDDI, 100VGAny-LAN, хотя и обладают многими индивидуальными чертами, в то же время имеют много общих свойств с Ethernet. В первую очередь – это применение регулярных фиксированных топологий (иерархическая звезда и кольцо), а также разделяемых сред передачи данных. Существенные отличия одной технологии от другой связаны с особенностями используемого метода доступа к разделяемой среде. Так, отличия технологии Ethernet от технологии Token Ring во многом определяются спецификой заложенных в них методов разделения среды – случайного алгоритма доступа в Ethernet и метода доступа путем передачи маркера в Token Ring.

1.2.6. Структуризация как средство построения больших сетей

В сетях с небольшим (10-30) количеством компьютеров чаще всего используется одна из типовых топологий – общая шина, кольцо, звезда или полносвязная сеть. Все перечисленные топологии обладают свойством однородности, то есть все компьютеры в такой сети имеют одинаковые права в отношении доступа к другим компьютерам (за исключением центрального компьютера при соединении звезда). Такая однородность структуры делает простой процедуру наращивания числа компьютеров, облегчает обслуживание и эксплуатацию сети.

Однако при построении больших сетей однородная структура связей превращается из преимущества в недостаток. В таких сетях использование типовых структур порождает различные ограничения, важнейшими из которых являются:

- ограничения на длину связи между узлами;

- ограничения на количество узлов в сети;
- ограничения на интенсивность трафика, порождаемого узлами сети.

Например, технология Ethernet на тонком коаксиальном кабеле позволяет использовать кабель длиной не более 185 метров, к которому можно подключить не более 30 компьютеров. Однако, если компьютеры интенсивно обмениваются информацией между собой, иногда приходится снижать число подключенных к кабелю компьютеров до 20, а то и до 10, чтобы каждому компьютеру доставалась приемлемая доля общей пропускной способности сети.

Для снятия этих ограничений используются специальные методы структуризации сети и специальное структурообразующее оборудование – повторители, концентраторы, мосты, коммутаторы, маршрутизаторы. Оборудование такого рода также называют коммуникационным, имея в виду, что с помощью него отдельные сегменты сети взаимодействуют между собой.

Физическая структуризация сети

Простейшее из коммуникационных устройств – *повторитель* (*repeater*) – используется для физического соединения различных сегментов кабеля локальной сети с целью увеличения общей длины сети. Повторитель передает сигналы, приходящие из одного сегмента сети, в другие ее сегменты (рис. 1.14). Повторитель позволяет преодолеть ограничения на длину линий связи за счет улучшения качества передаваемого сигнала – восстановления его мощности и амплитуды, улучшения фронтов и т. п.

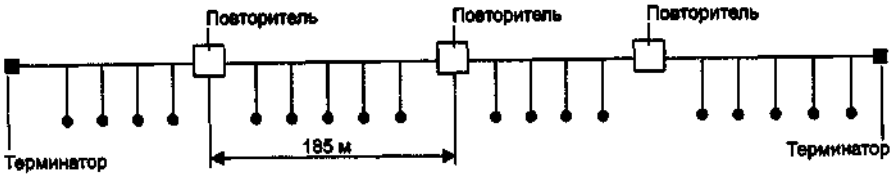


Рис. 1.14. Повторитель позволяет увеличить длину сети Ethernet

Повторитель, который имеет несколько портов и соединяет несколько физических сегментов, часто называют *концентратором* (*concentrator*) или *хабом* (*hub*). Эти названия (hub – основа, центр деятельности) отражают тот факт, что в данном устройстве сосредоточиваются все связи между сегментами сети.

Концентраторы характерны практически для всех базовых технологий локальных сетей – Ethernet, ArcNet, Token Ring, FDDI, Fast Ethernet, Gigabit Ethernet, lOOVG-AnyLAN.

Нужно подчеркнуть, что в работе концентраторов любых технологий много общего – они повторяют сигналы, пришедшие с одного из своих портов, на других своих портах. Разница состоит в том, на каких именно портах повторяются входные сигналы. Так, концентратор Ethernet повторяет входные сигналы на всех своих портах, кроме того, с которого сигналы поступают (рис, 1.15, *а*). А концентратор Token Ring (рис. 1.15, *б*) повторяет входные сигналы, поступающие с некоторого порта, только на одном порту – на том, к которому подключен следующий в кольце компьютер.

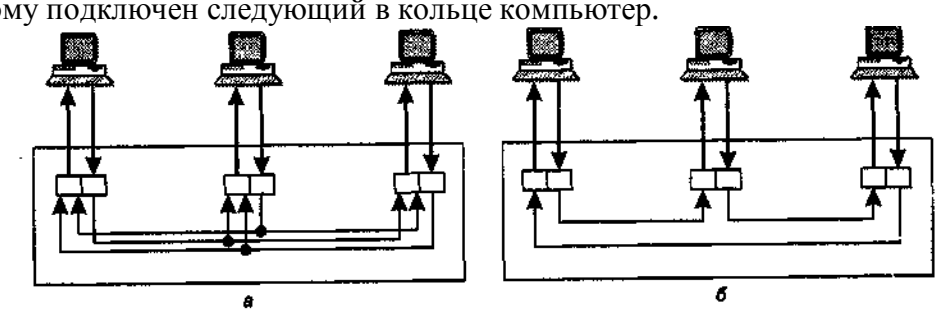


Рис. 1.15. Концентраторы различных технологий

ВНИМАНИЕ Концентратор всегда изменяет физическую топологию сети, но при этом оставляет без изменения ее логическую топологию.

Напомним, что под физической топологией понимается конфигурация связей, образованных

отдельными частями кабеля, а под логической – конфигурация информационных потоков между компьютерами сети. Во многих случаях физическая и логическая топологии сети совпадают. Например, сеть, представленная на рис. 1.16, *а*, имеет физическую топологию кольцо. Компьютеры этой сети получают доступ к кабелям кольца за счет передачи друг другу специального кадра – маркера, причем этот маркер также передается последовательно от компьютера к компьютеру в том же порядке, в котором компьютеры образуют физическое кольцо, то есть компьютер А передает маркер компьютеру В, компьютер В – компьютеру С и т.д.

Сеть, показанная на рис. 1.16, *б*, демонстрирует пример несовпадения физической и логической топологии. Физически компьютеры соединены по топологии общая шина. Доступ же к шине происходит не по алгоритму случайного доступа, применяемому в технологии Ethernet, а путем передачи маркера в кольцевом порядке: от компьютера А – компьютеру В, от компьютера В – компьютеру С и т.д. Здесь порядок передачи маркера уже не повторяет физические связи, а определяется логическим конфигурированием драйверов сетевых адаптеров. Ничто не мешает настроить сетевые адаптеры и их драйверы так, чтобы компьютеры образовали кольцо в другом порядке, например: В, А, С... При этом физическая структура сети никак не изменяется.

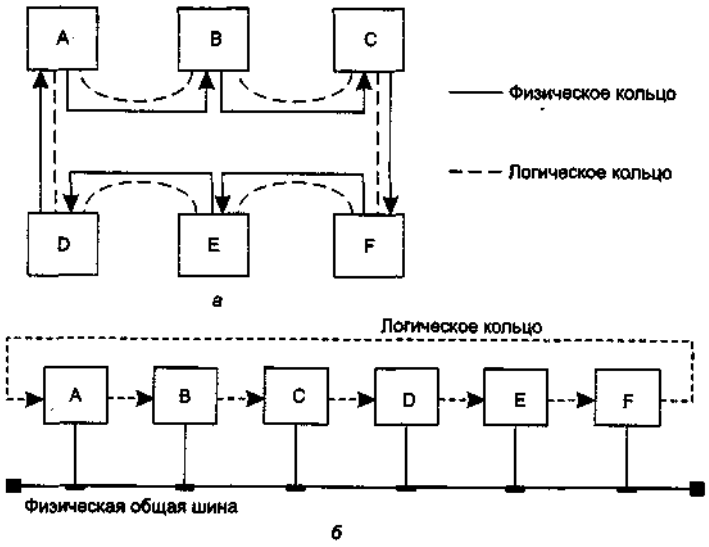


Рис. 1.16. Логическая и физическая топологии сети

Другим примером несовпадения физической и логической топологий сети является уже рассмотренная сеть на рис. 1.15, *а*. Концентратор Ethernet поддерживает в сети физическую топологию звезда. Однако логическая топология сети осталась без изменений – это общая шина. Так как концентратор повторяет данные, пришедшие с любого порта, на всех остальных портах, то они появляются одновременно на всех физических сегментах сети, как и в сети с физической общей шиной. Логика доступа к сети совершенно не меняется: все компоненты алгоритма случайного доступа – определение незанятости среды, захват среды, распознавание и отработка коллизий – остаются в силе.

Физическая структуризация сети с помощью концентраторов полезна не только для увеличения расстояния между узлами сети, но и для повышения ее надежности. Например, если какой-либо компьютер сети Ethernet с физической общей шиной из-за сбоя начинает непрерывно передавать данные до общему кабелю, то вся сеть выходит из строя, и для решения этой проблемы остается только один выход – вручную отсоединить сетевой адаптер этого компьютера от кабеля. В сети Ethernet, построенной с использованием концентратора, эта проблема может быть решена автоматически – концентратор отключает свой порт, если обнаруживает, что присоединенный к нему узел слишком долго монопольно занимает сеть. Концентратор может блокировать некорректно работающий узел и в других случаях, выполняя роль некоторого управляющего узла.

Логическая структуризация сети

Физическая структуризация сети полезна во многих отношениях, однако в ряде случаев, обычно относящихся к сетям большого и среднего размера, невозможно обойтись без логической структуризации сети. Наиболее важной проблемой, не решаемой путем физической структуризации,

остается проблема перераспределения передаваемого трафика между различными физическими сегментами сети.

В большой сети естественным образом возникает неоднородность информационных потоков: сеть состоит из множества подсетей рабочих групп, отделов, филиалов предприятия и других административных образований. Очень часто наиболее интенсивный обмен данными наблюдается между компьютерами, принадлежащими к одной подсети, и только небольшая часть обращений происходит к ресурсам компьютеров, находящихся вне локальных рабочих групп. (До недавнего времени такое соотношение трафиков не подвергалось сомнению, и был даже сформулирован эмпирический закон «80/20», в соответствии с которым в каждой подсети 80 % трафика является внутренним и только 20 % – внешним.) Сейчас характер нагрузки сетей во многом изменился, широко внедряется технология intranet, на многих предприятиях имеются централизованные хранилища корпоративных данных, активно используемые всеми сотрудниками предприятия. Все это не могло не повлиять на распределение информационных потоков. И теперь не редки ситуации, когда интенсивность внешних обращений выше интенсивности обмена между «соседними» машинами. Но независимо от того, в какой пропорции распределяются внешний и внутренний трафик, для повышения эффективности работы сети неоднородность информационных потоков необходимо учитывать.

Сеть с типовой топологией (шина, кольцо, звезда), в которой все физические сегменты рассматриваются в качестве одной разделяемой среды, оказывается неадекватной структуре информационных потоков в большой сети. Например, в сети с общей шиной взаимодействие любой пары компьютеров занимает ее на все время обмена, поэтому при увеличении числа компьютеров в сети шина становится узким местом. Компьютеры одного отдела вынуждены ждать, когда окончит обмен пара компьютеров другого отдела, и это при том, что необходимость в связи между компьютерами двух разных отделов возникает гораздо реже и требует совсем небольшой пропускной способности.

Этот случай иллюстрирует рис. 1.17, *а*. Здесь показана сеть, построенная с использованием концентраторов. Пусть компьютер А, находящийся в одной подсети с компьютером В, посылает ему данные. Несмотря на разветвленную физическую структуру сета, концентраторы распространяют любой кадр по всем ее сегментам. Поэтому кадр, посылаемый компьютером А компьютеру В, хотя и не нужен компьютерам отделов 2 и 3, в соответствии с логикой работы концентраторов поступает на эти сегменты тоже. И до тех пор, пока компьютер В не получит адресованный ему кадр, ни один из компьютеров этой сети не сможет передавать данные.

Такая ситуация возникает из-за того, что логическая структура данной сети осталась однородной – она никак не учитывает увеличение интенсивности трафика внутри отдела и предоставляет всем парам компьютеров равные возможности по обмену информацией (рис. 1.17, *б*).

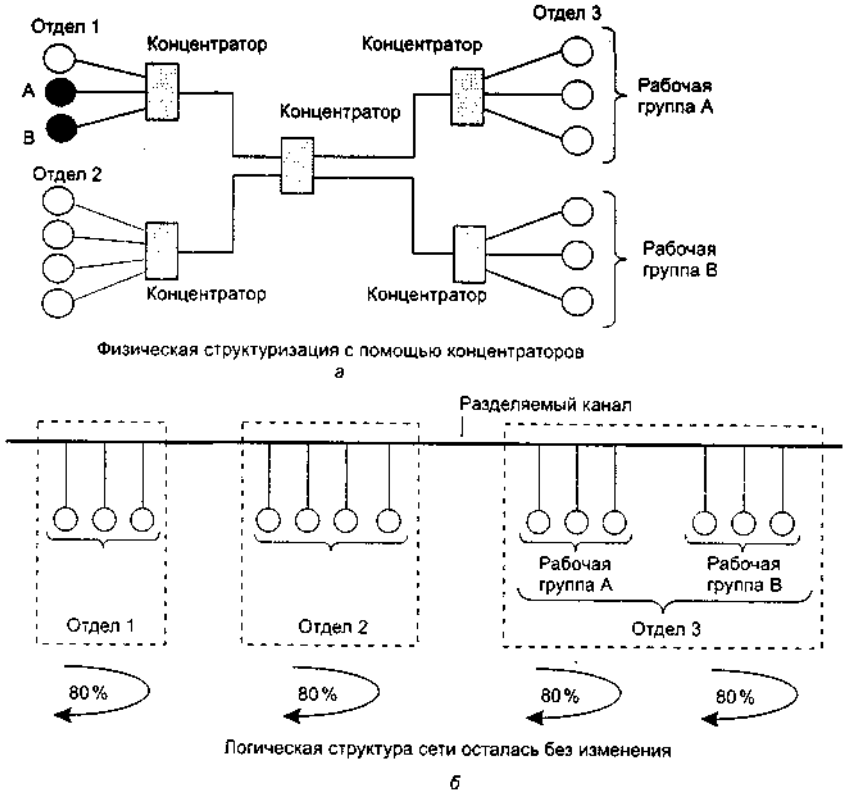


Рис. 1.17. Противоречие между логической структурой сети и структурой информационных потоков

Решение проблемы состоит в отказе от идеи единой однородной разделяемой среды. Например, в рассмотренном выше примере желательно было бы сделать так, чтобы кадры, которые передают компьютеры отдела 1, выходили бы за пределы этой части сети в том и только в том случае, если эти кадры направлены какому-либо компьютеру из других отделов. С другой стороны, в сеть каждого из отделов должны попадать те и только те кадры, которые адресованы узлам этой сети. При такой организации работы сети ее производительность существенно повысится, так как компьютеры одного отдела не будут простаивать в то время, когда обмениваются данными компьютеры других отделов.

Нетрудно заметить, что в предложенном решении мы отказались от идеи общей разделяемой среды в пределах всей сети, хотя и оставили ее в пределах каждого отдела. Пропускная способность линий связи между отделами не должна совпадать с пропускной способностью среды внутри отделов. Если трафик между отделами составляет только 20 % трафика внутри отдела (как уже отмечалось, эта величина может быть другой), то и пропускная способность линий связи и коммуникационного оборудования, соединяющего отделы, может быть значительно ниже внутреннего трафика сети отдела.

ВНИМАНИЕ Распространение трафика, предназначенного для компьютеров некоторого сегмента сети, только в пределах этого сегмента, называется локализацией трафика. Логическая структуризация сети - это процесс разбиения сети на сегменты с локализованным трафиком.

Для логической структуризации сети используются такие коммуникационные устройства, как мосты, коммутаторы, маршрутизаторы и шлюзы.

Мост (*bridge*) делит разделяемую среду передачи сети на части (часто называемые логическими сегментами), передавая информацию из одного сегмента в другой только в том случае, если такая передача действительно необходима, то есть если адрес компьютера назначения принадлежит другой подсети. Тем самым мост изолирует трафик одной подсети от трафика другой, повышая общую производительность передачи данных в сети. Локализация трафика не только экономит пропускную способность, но и уменьшает возможность несанкционированного доступа к данным, так как кадры не выходят за пределы своего сегмента и их сложнее перехватить злоумышленнику.

На рис. 1.18 показана сеть, которая была получена из сети с центральным концентратором (см. рис. 1.17) путем его замены на мост. Сети 1-го и 2-го отделов состоят из отдельных логических сегментов, а

сеть отдела 3 – из двух логических сегментов. Каждый логический сегмент построен на базе концентратора и имеет простейшую физическую структуру, образованную отрезками кабеля, связывающими компьютеры с портами концентратора.

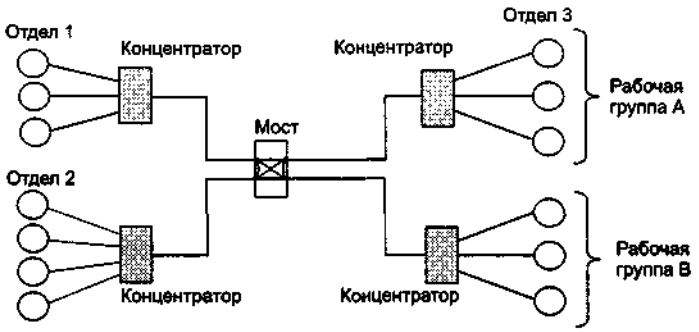


Рис. 1.18. Логическая структуризации сети с помощью моста

Мосты используют для локализации трафика аппаратные адреса компьютеров. Это затрудняет распознавание принадлежности того или иного компьютера к определенному логическому сегменту – сам адрес не содержит никакой информации по этому поводу. Поэтому мост достаточно упрощенно представляет деление сети на сегменты – он запоминает, через какой порт на него поступил кадр данных от каждого компьютера сети, и в дальнейшем передает кадры, предназначенные для этого компьютера, на этот порт. Точной топологии связей между логическими сегментами мост не знает. Из-за этого применение мостов приводит к значительным ограничениям на конфигурацию связей сети – сегменты должны быть соединены таким образом, чтобы в сети не образовывались замкнутые контуры,

Коммутатор (*switch, switching hub*) по принципу обработки кадров ничем не отличается от моста. Основное его отличие от моста состоит в том, что он является своего рода коммуникационным мультипроцессором, так как каждый его порт оснащен специализированным процессором, который обрабатывает кадры по алгоритму моста независимо от процессоров других портов. За счет этого общая производительность коммутатора обычно намного выше производительности традиционного моста, имеющего один процессорный блок. Можно сказать, что коммутаторы – это мосты нового поколения, которые обрабатывают кадры в параллельном режиме.

Ограничения, связанные с применением мостов и коммутаторов – по топологии связей, а также ряд других, – привели к тому, что в ряду коммуникационных устройств появился еще один тип оборудования – *маршрутизатор* (*router*). Маршрутизаторы более надежно и более эффективно, чем мосты, изолируют трафик отдельных частей сети друг от друга. Маршрутизаторы образуют логические сегменты посредством явной адресации, поскольку используют не плоские аппаратные, а составные числовые адреса. В этих адресах имеется поле номера сети, так что все компьютеры, у которых значение этого поля одинаково, принадлежат к одному сегменту, называемому в данном случае *подсетью* (*subnet*).

Кроме локализации трафика маршрутизаторы выполняют еще много других полезных функций. Так, маршрутизаторы могут работать в сети с замкнутыми контурами, при этом они осуществляют выбор наиболее рационального маршрута из нескольких возможных. Сеть, представленная на рис. 1.19, отличается от своей предшественницы (см. рис. 1.18) тем, что между подсетями отделов 1 и 2 проложена дополнительная связь, которая может использоваться как для повышения производительности сети, так и для повышения ее надежности.

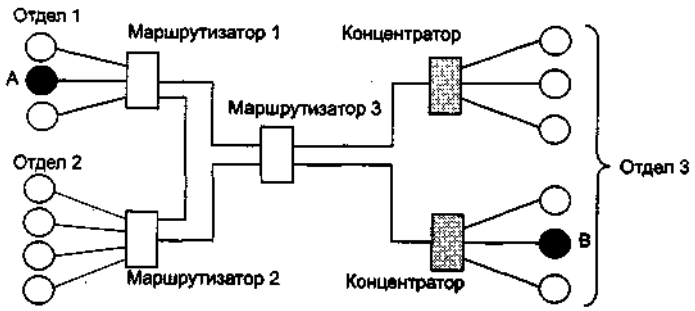


Рис. 1.19. Логическая структуризация сети с помощью маршрутизаторов

Другой очень важной функцией маршрутизаторов является их способность связывать в единую сеть подсети, построенные с использованием разных сетевых технологий, например Ethernet и X.25.

Кроме перечисленных устройств отдельные части сети может соединять *шлюз* (*gateway*). Обычно основной причиной, по которой в сети используют шлюз, является необходимость объединить сети с разными типами системного и прикладного программного обеспечения, а не желание локализовать трафик. Тем не менее шлюз обеспечивает и локализацию трафика в качестве некоторого побочного эффекта.

Крупные сети практически никогда не строятся без логической структуризации. Для отдельных сегментов и подсетей характерны типовые однородные топологии базовых технологий, и для их объединения всегда используется оборудование, обеспечивающее локализацию трафика, – мосты, коммутаторы, маршрутизаторы и шлюзы.

1.2.7. Сетевые службы

Для конечного пользователя сеть – это не компьютеры, кабели и концентраторы и даже не информационные потоки, для него сеть – это, прежде всего, тот набор сетевых служб, с помощью которых он получает возможность просмотреть список имеющихся в сети компьютеров, прочитать удаленный файл, распечатать документ на «чужом» принтере или послать почтовое сообщение. Именно совокупность предоставляемых возможностей – насколько широк их выбор, насколько они удобны, надежны и безопасны – определяет для пользователя облик той или иной сети.

Кроме собственно обмена данными, сетевые службы должны решать и другие, более специфические задачи, например, задачи, порождаемые распределенной обработкой данных. К таким задачам относится обеспечение непротиворечивости нескольких копий данных, размещенных на разных машинах (служба репликации), или организация выполнения одной задачи параллельно на нескольких машинах сети (служба вызова удаленных процедур). Среди сетевых служб можно выделить административные, то есть такие, которые в основном ориентированы не на простого пользователя, а на администратора и служат для организации правильной работы сети в целом. Служба администрирования учетных записей о пользователях, которая позволяет администратору вести общую базу данных о пользователях сети, система мониторинга сети, позволяющая захватывать и анализировать сетевой трафик, служба безопасности, в функции которой может входить среди прочего выполнение процедуры логического входа с последующей проверкой пароля, – все это примеры административных служб.

Реализация сетевых служб осуществляется программными средствами. Основные службы – файловая служба и служба печати – обычно предоставляются сетевой операционной системой, а вспомогательные, например служба баз данных, факса или передачи голоса, – системными сетевыми приложениями или утилитами, работающими в тесном контакте с сетевой ОС. Вообще говоря, распределение служб между ОС и утилитами достаточно условно и меняется в конкретных реализациях ОС.

При разработке сетевых служб приходится решать проблемы, которые свойственны любым распределенным приложениям: определение протокола взаимодействия между клиентской и серверной частями, распределение функций между ними, выбор схемы адресации приложений и др.

Одним из главных показателей качества сетевой службы является ее удобство. Для одного и того же ресурса может быть разработано несколько служб, по-разному решающих в общем-то одну и ту же задачу. Отличия могут заключаться в производительности или в уровне удобства предоставляемых

услуг. Например, файловая служба может быть основана на использовании команды передачи файла из одного компьютера в другой по имени файла, а это требует от пользователя знания имени нужного файла. Та же файловая служба может быть реализован и так, что пользователь монтирует удаленную файловую систему к локальному каталогу, а далее обращается к удаленным файлам как к своим собственным, что гораздо более удобно. Качество сетевой службы зависит и от качества пользовательского интерфейса – интуитивной понятности, наглядности, рациональности.

При определении степени удобства разделяемого ресурса часто употребляют термин «прозрачность». *Прозрачный доступ* – это такой доступ, при котором пользователь не замечает, где расположен нужный ему ресурс - на его компьютере или на удаленном. После того как он смонтировал удаленную файловую систему в свое дерево каталогов, доступ к удаленным файлам становится для него совершенно прозрачным. Сама операция монтирования также может иметь разную степень прозрачности – в сетях с меньшей прозрачностью пользователь должен знать и задавать в команде имя компьютера, на котором хранится удаленная файловая система, в сетях с большей степенью прозрачности соответствующий программный компонент сети производит поиск разделяемых томов файлов безотносительно мест их хранения, а затем предоставляет их пользователю в удобном для него виде, например в виде списка или набора пиктограмм.

Для обеспечения прозрачности важен способ адресации (именования) разделяемых сетевых ресурсов. Имена разделяемых сетевых ресурсов не должны зависеть от их физического расположения на том или ином компьютере. В идеале пользователь не должен ничего менять в своей работе, если администратор сети переместил том или каталог с одного компьютера на другой. Сам администратор и сетевая операционная система имеют информацию о расположении файловых систем, но от пользователя она скрыта. Такая степень прозрачности пока редко встречается в сетях, – обычно для получения доступа к ресурсам определенного компьютера сначала приходится устанавливать с ним логическое соединение. Такой подход применяется, например, в сетях Windows NT.

Выводы

- Задачи надежного обмена двоичными сигналами по линиям связи в локальных сетях решают сетевые адаптеры, а в глобальных сетях – аппаратура передачи данных. Это оборудование кодирует и декодирует информацию, синхронизирует передачу электромагнитных сигналов по линиям связи и проверяет правильность передачи.
- Программные средства, реализующие простейшую схему удаленного доступа к файлам, включают классические элементы сетевой операционной системы: сервер, клиент и средства транспортировки сообщений по линии связи.
- Важной характеристикой сети является топология – тип графа, вершинам которого соответствуют компьютеры сети (иногда и другое оборудование, например концентраторы), а ребрам – физические связи между ними. Конфигурация физических связей определяется электрическими соединениями компьютеров между собой и может отличаться от конфигурации логических связей между узлами сети. Логические связи представляют собой маршруты передачи данных между узлами сети.
- Типовыми топологиями физических связей являются: полносвязная, ячеистая, общая шина, кольцевая топология и топология типа звезда.
- Для вычислительных сетей характерны как индивидуальные линии связи между компьютерами, так и разделяемые, когда одна линия связи попеременно используется несколькими компьютерами. В последнем случае возникают как чисто электрические проблемы обеспечения нужного качества сигналов при подключении к одному и тому же проводу нескольких приемников и передатчиков, так и логические проблемы разделения времени доступа к этим линиям.
- Для адресации узлов сети используются три типа адресов: аппаратные адреса, символьные имена, числовые составные адреса. В современных сетях, как правило, одновременно применяются все эти три схемы адресации. Важной сетевой проблемой является задача установления соответствия между адресами различных типов. Эта проблема может решаться как полностью централизованными, так и распределенными средствами.
- Для снятия ограничений на длину сети и количество ее узлов используется физическая структуризация сети с помощью повторителей и концентраторов.

- Для повышения производительности и безопасности сети используется логическая структуризация сети, состоящая в разбиении сети на сегменты таким образом, что основная часть трафика компьютеров каждого сегмента не выходит за пределы этого сегмента. Средствами логической структуризации служат мосты, коммутаторы, маршрутизаторы и шлюзы.

1.3. Понятие «открытая система» и проблемы стандартизации

Универсальный тезис о пользе стандартизации, справедливый для всех отраслей, в компьютерных сетях приобретает особое значение. Суть сети – это соединение разного оборудования, а значит, проблема совместимости является одной из наиболее острых. Без принятия всеми производителями общепринятых правил построения оборудования прогресс в деле «строительства» сетей был бы невозможен. Поэтому все развитие компьютерной отрасли в конечном счете отражено в стандартах – любая новая технология только тогда приобретает «законный» статус, когда ее содержание закрепляется в соответствующем стандарте.

В компьютерных сетях идеологической основой стандартизации является многоуровневый подход к разработке средств сетевого взаимодействия. Именно на основе этого подхода была разработана стандартная семиуровневая модель взаимодействия открытых систем, ставшая своего рода универсальным языком сетевых специалистов.

1.3.1. Многоуровневый подход. Протокол. Интерфейс. Стек протоколов

Организация взаимодействия между устройствами в сети является сложной задачей. Как известно, для решения сложных задач используется универсальный прием – декомпозиция, то есть разбиение одной сложной задачи на несколько более простых задач-модулей (рис. 1.20). Процедура декомпозиции включает в себя четкое определение функций каждого модуля, решающего отдельную задачу, и интерфейсов между ними. В результате достигается логическое упрощение задачи, а кроме того, появляется возможность модификации отдельных модулей без изменения остальной части системы.

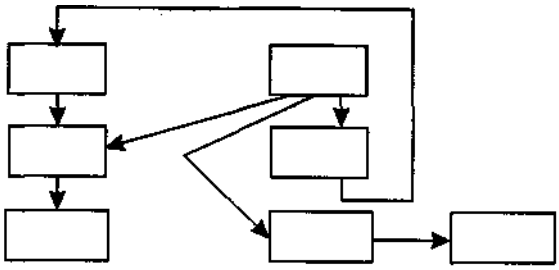


Рис. 1.20. Пример декомпозиции задачи

При декомпозиции часто используют многоуровневый подход. Он заключается в следующем. Все множество модулей разбивают на уровни. Уровни образуют иерархию, то есть имеются вышележащие и нижележащие уровни (рис. 1.21). Множество модулей, составляющих каждый уровень, сформировано таким образом, что для выполнения своих задач они обращаются с запросами только к модулям непосредственно примыкающего нижележащего уровня. С другой стороны, результаты работы всех модулей, принадлежащих некоторому уровню, могут быть переданы только модулям соседнего вышележащего уровня. Такая иерархическая декомпозиция задачи предполагает четкое определение функции каждого уровня и интерфейсов между уровнями. Интерфейс определяет набор функций, которые нижележащий уровень предоставляет вышележащему. В результате иерархической декомпозиции достигается относительная независимость уровней, а значит, и возможность их легкой замены.

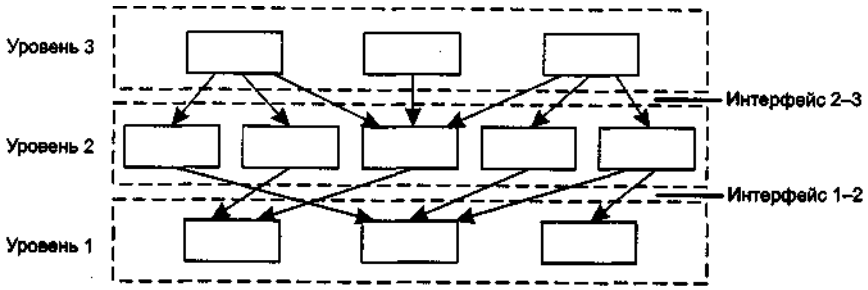


Рис. 1.21. Многоуровневый подход – создание иерархии задач

Средства сетевого взаимодействия, конечно, тоже могут быть представлены в виде иерархически организованного множества модулей. При этом модули нижнего уровня могут, например, решать все вопросы, связанные с надежной передачей электрических сигналов между двумя соседними узлами. Модули более высокого уровня организуют транспортировку сообщений в пределах всей сети, пользуясь для этого средствами упомянутого нижележащего уровня. А на верхнем уровне работают модули, предоставляющие пользователям доступ к различным службам – файловой, печати и т. п. Конечно, это только один из множества возможных вариантов деления общей задачи организации сетевого взаимодействия на частные подзадачи.

Многоуровневый подход к описанию и реализации функций системы применяется не только в отношении сетевых средств. Такая модель функционирования используется, например, в локальных файловых системах, когда поступивший запрос на доступ к файлу последовательно обрабатывается несколькими программными уровнями (рис. 1.22). Запрос вначале анализируется верхним уровнем, на котором осуществляется последовательный разбор составного символьного имени файла и определение уникального идентификатора файла. Следующий уровень находит по уникальному имени все основные характеристики файла: адрес, атрибуты доступа и т. п. Затем на более низком уровне осуществляется проверка прав доступа к этому файлу, а далее, после расчета координат области файла, содержащей требуемые данные, выполняется физический обмен с внешним устройством с помощью драйвера диска.

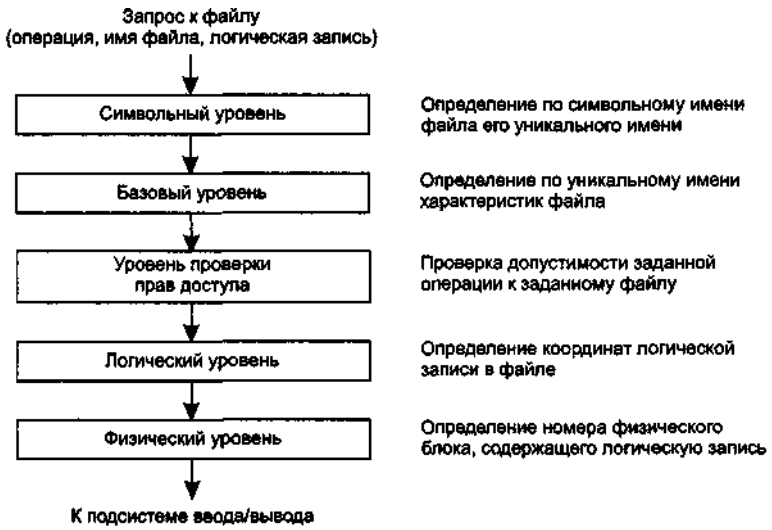


Рис. 1.22. Многоуровневая модель файловой системы

Многоуровневое представление средств сетевого взаимодействия имеет свою специфику, связанную с тем, что в процессе обмена сообщениями участвуют *две* машины, то есть в данном случае необходимо организовать согласованную работу двух «иерархий». При передаче сообщений оба участника сетевого обмена должны принять множество соглашений. Например, они должны согласовать уровни и форму электрических сигналов, способ определения длины сообщений, договориться о методах контроля достоверности и т. п. Другими словами, соглашения должны быть приняты для всех уровней, начиная от самого низкого – уровня передачи битов – до самого высокого, реализующего сервис для пользователей сети.

На рис. 1.23 показана модель взаимодействия двух узлов. С каждой стороны средства взаимодействия представлены четырьмя уровнями. Процедура взаимодействия этих двух узлов может

быть описана в виде набора правил взаимодействия каждой пары соответствующих уровней обеих участвующих сторон. Формализованные правила, определяющие последовательность и формат сообщений, которыми обмениваются сетевые компоненты, лежащие на одном уровне, но в разных узлах, называются *протоколом.*

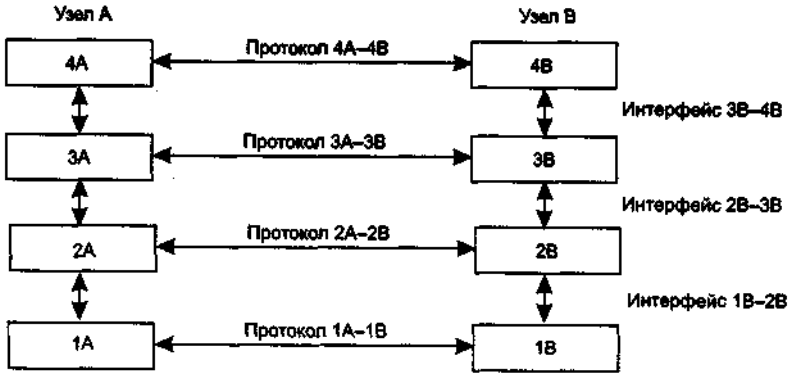


Рис. 1.23. Взаимодействие двух узлов

Модули, реализующие протоколы соседних уровней и находящиеся в одном узле, также взаимодействуют друг с другом в соответствии с четко определенными правилами и с помощью стандартизованных форматов сообщений. Эти правила принято называть *интерфейсом.* Интерфейс определяет набор сервисов, предоставляемый данным уровнем соседнему уровню. В сущности, протокол и интерфейс выражают одно и то же понятие, но традиционно в сетях за ними закрепили разные области действия: протоколы определяют правила взаимодействия модулей одного уровня в разных узлах, а интерфейсы – модулей соседних уровней в одном узле.

Средства каждого уровня должны отрабатывать, во-первых, свой собственный протокол, а во-вторых, интерфейсы с соседними уровнями.

Иерархически организованный набор протоколов, достаточный для организации взаимодействия узлов в сети, называется *стеком коммуникационных протоколов*.

Коммуникационные протоколы могут быть реализованы как программно, так и аппаратно. Протоколы нижних уровней часто реализуются комбинацией программных и аппаратных средств, а протоколы верхних уровней – как правило, чисто программными средствами.

Программный модуль, реализующий некоторый протокол, часто для краткости также называют «протоколом». При этом соотношение между протоколом – формально определенной процедурой и протоколом – программным модулем, реализующим эту процедуру, аналогично соотношению между алгоритмом решения некоторой задачи и программой, решающей эту задачу.

Понятно, что один и тот же алгоритм может быть запрограммирован с разной степенью эффективности. Точно так же и протокол может иметь несколько программных реализаций. Именно поэтому при сравнении протоколов следует учитывать не только логику их работы, но и качество программных решений. Более того, на эффективность взаимодействия устройств в сети влияет качество всей совокупности протоколов, составляющих стек, в частности, насколько рационально распределены функции между протоколами разных уровней и насколько хорошо определены интерфейсы между ними.

Протоколы реализуются не только компьютерами, но и другими сетевыми устройствами – концентраторами, мостами, коммутаторами, маршрутизаторами и т. д. Действительно, в общем случае связь компьютеров в сети осуществляется не напрямую, а через различные коммуникационные устройства. В зависимости от типа устройства в нем должны быть встроенные средства, реализующие тот или иной набор протоколов.

Чтобы еще раз пояснить понятия «протокол» и «интерфейс», рассмотрим пример, не имеющий отношения к вычислительным сетям, а именно обсудим взаимодействие двух предприятий А и В, связанных между собой деловым сотрудничеством. Между предприятиями существуют многочисленные договоренности и соглашения, такие, например, как регулярные поставки продукции одного предприятия другому. В соответствии с этой договоренностью начальник отдела продаж предприятия А регулярно в начале каждого месяца посылает официальное сообщение начальнику отдела закупок предприятия В о том, сколько и какого товара может быть поставлено в этом месяце. В

ответ на это сообщение начальник отдела закупок предприятия В посылает в ответ заявку установленного образца на требуемое количество продукции. Возможно, процедура взаимодействия этих начальников включает дополнительные согласования, в любом случае существует установленный порядок взаимодействия, который можно считать «протоколом уровня начальников». Начальники посылают свои сообщения и заявки через своих секретарей. Порядок взаимодействия начальника и секретаря соответствует понятию межуровневого интерфейса «начальник - секретарь». На предприятии А обмен документами между начальником и секретарем идет через специальную папку, а на предприятии В начальник общается с секретарем по факсу. Таким образом, интерфейсы «начальник - секретарь» на этих двух предприятиях отличаются.

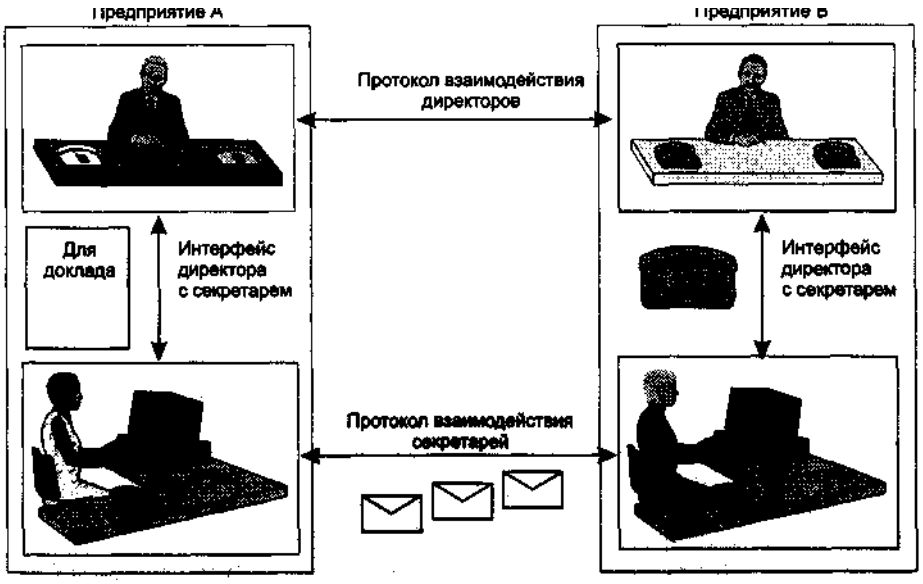


Рис. 1.24. Пример многоуровневого взаимодействия предприятий

После того как сообщения переданы секретарям, начальников не волнует, каким образом эти сообщения будут перемещаться дальше – обычной или электронной почтой, факсом или нарочным. Выбор способа передачи – это уровень компетенции секретарей, они могут решать этот вопрос, не уведомляя об этом своих начальников, так как их протокол взаимодействия связан только с передачей сообщений, поступающих сверху, и не касается содержания этих сообщений. На рис. 1.24 показано, что в качестве протокола взаимодействия «секретарь–секретарь» используется обмен письмами. При решении других вопросов начальники могут взаимодействовать по другим правилам-протоколам, но это не повлияет на работу секретарей, для которых не важно, какие сообщения отправлять, а важно, чтобы они дошли до адресата. Итак, в данном случае мы имеем дело с двумя уровнями – уровнем начальников и уровнем секретарей, и каждый из них имеет собственный протокол, который может быть изменен независимо от протокола другого уровня. Эта независимость протоколов друг от друга и делает привлекательным многоуровневый подход.

1.3.2. Модель OSI

Из того, что протокол является соглашением, принятым двумя взаимодействующими объектами, в данном случае двумя работающими в сети компьютерами, совсем не следует, что он обязательно является стандартным. Но на практике при реализации сетей стремятся использовать стандартные протоколы. Это могут быть фирменные, национальные или международные стандарты.

В начале 80-х годов ряд международных организаций по стандартизации – ISO, ITU-T и некоторые другие – разработали модель, которая сыграла значительную роль в развитии сетей. Эта модель называется *моделью взаимодействия открытых систем* (*Open System Interconnection, OSI*) или моделью OSI. Модель OSI определяет различные уровни взаимодействия систем, дает им стандартные имена и указывает, какие функции должен выполнять каждый уровень. Модель OSI была разработана на основании большого опыта, полученного при создании компьютерных сетей, в основном глобальных, в 70-е годы. Полное описание этой модели занимает более 1000 страниц текста.

В модели OSI (рис. 1.25) средства взаимодействия делятся на семь уровней: прикладной,

представительный, сеансовый, транспортный, сетевой, канальный и физический. Каждый уровень имеет дело с одним определенным аспектом взаимодействия сетевых устройств.

Модель OSI описывает только системные средства взаимодействия, реализуемые операционной системой, системными утилитами, системными аппаратными средствами. Модель не включает средства взаимодействия приложений конечных пользователей. Свои собственные протоколы взаимодействия приложения реализуют, обращаясь к системным средствам. Поэтому необходимо различать уровень взаимодействия приложений и прикладной уровень.

Следует также иметь в виду, что приложение может взять на себя функции некоторых верхних уровней модели OSI. Например, некоторые СУБД имеют встроенные средства удаленного доступа к файлам. В этом случае приложение, выполняя доступ к удаленным ресурсам, не использует системную файловую службу; оно обходит верхние уровни модели OSI и обращается напрямую к системным средствам, ответственным за транспортировку сообщений по сети, которые располагаются на нижних уровнях модели OSI.

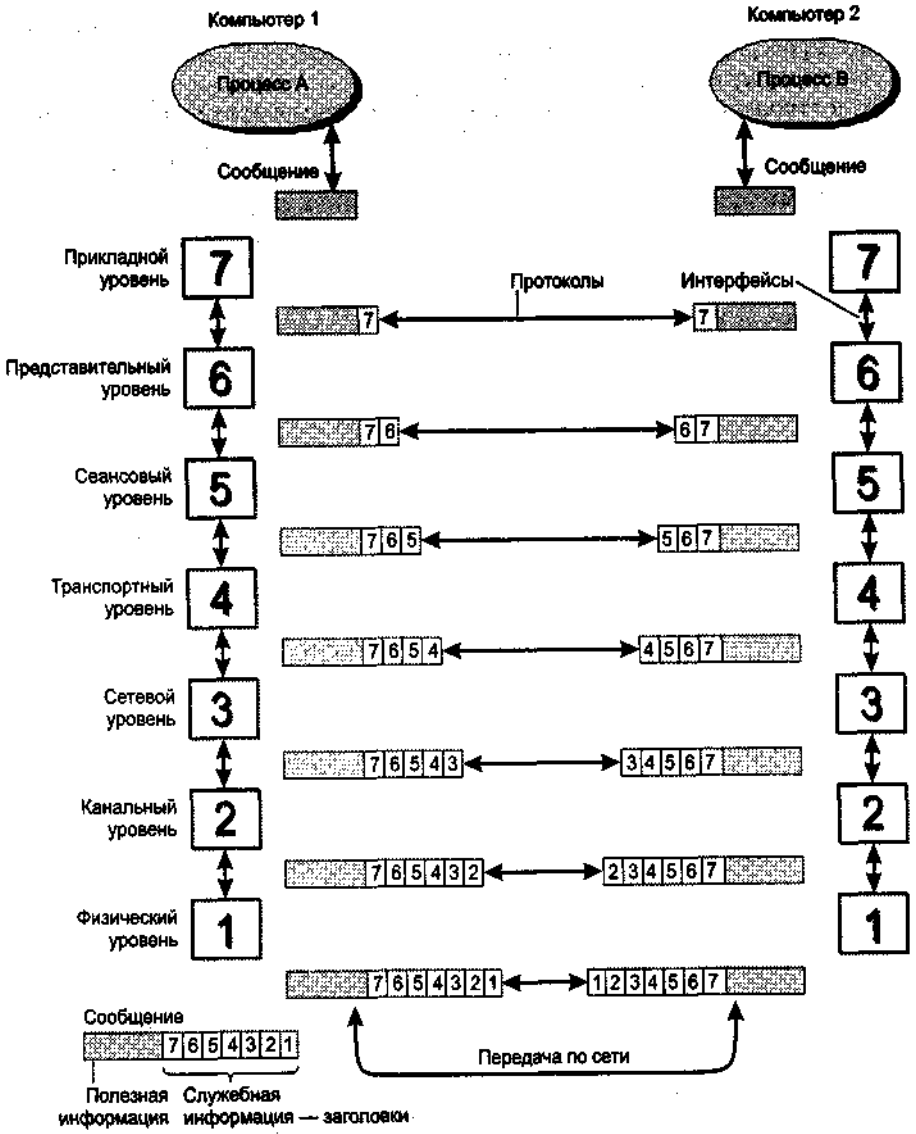


Рис. 1.25. Модель взаимодействия открытых систем ISO/OSI

Итак, пусть приложение обращается с запросом к прикладному уровню, например к файловой службе. На основании этого запроса программное обеспечение прикладного уровня формирует сообщение стандартного формата. Обычное сообщение состоит из заголовка и поля данных. Заголовок содержит служебную информацию, которую необходимо передать через сеть прикладному уровню машины – адресата, чтобы сообщить ему, какую работу надо выполнить. В нашем случае заголовок, очевидно, должен содержать информацию о месте нахождения файла и о типе операции, которую необходимо над ним выполнить. Поле данных сообщения может быть пустым или содержать какие –

либо данные, например те, которые необходимо записать в удаленный файл. Но для того чтобы доставить эту информацию по назначению, предстоит решить еще много задач, ответственность за которые несут нижележащие уровни.

После формирования сообщения прикладной уровень направляет его вниз по стеку представительному уровню. Протокол представительного уровня на основании информации, полученной из заголовка прикладного уровня, выполняет требуемые действия и добавляет к сообщению собственную служебную информацию – заголовок представительного уровня, в котором содержатся указания для протокола представительного уровня машины-адресата. Полученное в результате сообщение передается вниз сеансовому уровню, который в свою очередь добавляет свой заголовок, и т. д. (Некоторые реализации протоколов помещают служебную информацию не только в начале сообщения в виде заголовка, но и в конце, в виде так называемого «концевика».) Наконец, сообщение достигает нижнего, физического уровня, который собственно и передает его по линиям связи машине-адресату. К этому моменту сообщение «обрастает» заголовками всех уровней (рис. 1.26).

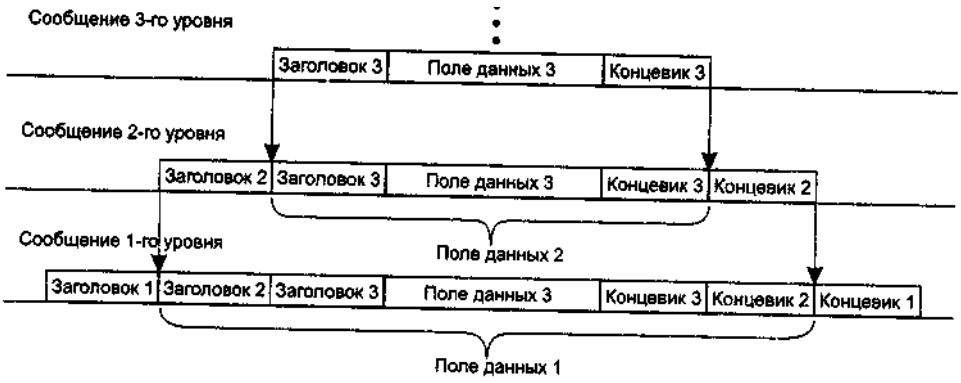


Рис. 1.26. Вложенность сообщений различных уровней

Когда сообщение по сети поступает на машину-адресат, оно принимается ее физическим уровнем и последовательно перемещается вверх с уровня на уровень. Каждый уровень анализирует и обрабатывает заголовок своего уровня, выполняя соответствующие данному уровню функции, а затем удаляет этот заголовок и передает сообщение вышележащему уровню.

Наряду с термином *сообщение* (*message*) существуют и другие термины, применяемые сетевыми специалистами для обозначения единиц данных в процедурах обмена. В стандартах ISO для обозначения единиц данных, с которыми имеют дело протоколы разных уровней, используется общее название *протокольный блок данных* (*Protocol Data Unit, PDU*). Для обозначения блоков данных определенных уровней часто используются специальные названия: кадр (frame), пакет (packet), дейтаграмма (datagram), сегмент (segment).

В модели OSI различаются два основных типа протоколов. В протоколах с *установлением соединения* (*connection-oriented*) перед обменом данными отправитель и получатель должны сначала установить соединение и, возможно, выбрать некоторые параметры протокола, которые они будут использовать при обмене данными. После завершения диалога они должны разорвать это соединение. Телефон – это пример взаимодействия, основанного на установлении соединения.

Вторая группа протоколов – протоколы *без предварительного установления соединения* (*connectionless*). Такие протоколы называются также *дейтаграммными* протоколами. Отправитель просто передает сообщение, когда оно готово. Опускание письма в почтовый ящик – это пример связи без предварительного установления соединения. При взаимодействии компьютеров используются протоколы обоих типов.

1.3.3. Уровни модели OSI

Физический уровень

Физический уровень (Physical layer) имеет дело с передачей битов по физическим каналам связи, таким, например, как коаксиальный кабель, витая пара, оптоволоконный кабель или цифровой территориальный канал. К этому уровню имеют отношение характеристики физических сред передачи данных, такие как полоса пропускания, помехозащищенность, волновое сопротивление и другие. На

этом же уровне определяются характеристики электрических сигналов, передающих дискретную информацию, например, крутизна фронтов импульсов, уровни напряжения или тока передаваемого сигнала, тип кодирования, скорость передачи сигналов. Кроме этого, здесь стандартизуются типы разъемов и назначение каждого контакта.

Функции физического уровня реализуются во всех устройствах, подключенных к сети. Со стороны компьютера функции физического уровня выполняются сетевым адаптером или последовательным портом.

Примером протокола физического уровня может служить спецификация 10Base-T технологии Ethernet, которая определяет в качестве используемого кабеля неэкранированную витую пару категории 3 с волновым сопротивлением 100 Ом, разъем RJ-45, максимальную длину физического сегмента 100 метров, манчестерский код для представления данных в кабеле, а также некоторые другие характеристики среды и электрических сигналов.

Канальный уровень

На физическом уровне просто пересылаются биты. При этом не учитывается, что в некоторых сетях, в которых линии связи используются (разделяются) попеременно несколькими парами взаимодействующих компьютеров, физическая среда передачи может быть занята. Поэтому одной из задач канального уровня (Data Link layer) является проверка доступности среды передачи. Другой задачей канального уровня является реализация механизмов обнаружения и коррекции ошибок. Для этого на канальном уровне биты группируются в наборы, называемые *кадрами* (*frames*). Канальный уровень обеспечивает корректность передачи каждого кадра, помещая специальную последовательность бит в начало и конец каждого кадра, для его выделения, а также вычисляет контрольную сумму, обрабатывая все байты кадра определенным способом и добавляя контрольную сумму к кадру. Когда кадр приходит по сети, получатель снова вычисляет контрольную сумму полученных данных и сравнивает результат с контрольной суммой из кадра. Если они совпадают, кадр считается правильным и принимается. Если же контрольные суммы не совпадают, то фиксируется ошибка. Канальный уровень может не только обнаруживать ошибки, но и исправлять их за счет повторной передачи поврежденных кадров. Необходимо отметить, что функция исправления ошибок не является обязательной для канального уровня, поэтому в некоторых протоколах этого уровня она отсутствует, например, в Ethernet и frame relay.

В протоколах канального уровня, используемых в локальных сетях, заложена определенная структура связей между компьютерами и способы их адресации. Хотя канальный уровень и обеспечивает доставку кадра между любыми двумя узлами локальной сети, он это делает только в сети с совершенно определенной топологией связей, именно той топологией, для которой он был разработан. К таким типовым топологиям, поддерживаемым протоколами канального уровня локальных сетей, относятся общая шина, кольцо и звезда, а также структуры, полученные из них с помощью мостов и коммутаторов. Примерами протоколов канального уровня являются протоколы Ethernet, Token Ring, FDDI, lOOVG-AnyLAN.

В локальных сетях протоколы канального уровня используются компьютерами, мостами, коммутаторами и маршрутизаторами. В компьютерах функции канального уровня реализуются совместными усилиями сетевых адаптеров и их драйверов.

В глобальных сетях, которые редко обладают регулярной топологией, канальный уровень часто обеспечивает обмен сообщениями только между двумя соседними компьютерами, соединенными индивидуальной линией связи. Примерами протоколов «точка-точка» (как часто называют такие протоколы) могут служить широко распространенные протоколы PPP и LAP-B. В таких случаях для доставки сообщений между конечными узлами через всю сеть используются средства сетевого уровня. Именно так организованы сети X.25. Иногда в глобальных сетях функции канального уровня в чистом виде выделить трудно, так как в одном и том же протоколе они объединяются с функциями сетевого уровня. Примерами такого подхода могут служить протоколы технологий ATM и frame relay.

В целом канальный уровень представляет собой весьма мощный и законченный набор функций по пересылке сообщений между узлами сети. В некоторых случаях протоколы канального уровня оказываются самодостаточными транспортными средствами и могут допускать работу поверх них непосредственно протоколов прикладного уровня или приложений, без привлечения средств сетевого и транспортного уровней. Например, существует реализация протокола управления сетью SNMP

непосредственно поверх Ethernet, хотя стандартно этот протокол работает поверх сетевого протокола IP и транспортного протокола UDP. Естественно, что применение такой реализации будет ограниченным – она не подходит для составных сетей разных технологий, например Ethernet и X.25, и даже для такой сети, в которой во всех сегментах применяется Ethernet, но между сегментами существуют петлевидные связи. А вот в двухсегментной сети Ethernet, объединенной мостом, реализация SNMP над канальным уровнем будет вполне работоспособна.

Тем не менее для обеспечения качественной транспортировки сообщений в сетях любых топологий и технологий функций канального уровня оказывается недостаточно, поэтому в модели OSI решение этой задачи возлагается на два следующих уровня – сетевой я транспортный.

Сетевой уровень

Сетевой уровень (Network layer) служит для образования единой транспортной системы, объединяющей несколько сетей, причем эти сети могут использовать совершенно различные принципы передачи сообщений между конечными узлами и обладать произвольной структурой связей. Функции сетевого уровня достаточно разнообразны. Начнем их рассмотрение на примере объединения локальных сетей.

Протоколы канального уровня локальных сетей обеспечивают доставку данных между любыми узлами только в сети с соответствующей типовой топологией, например топологией иерархической звезды. Это очень жесткое ограничение, которое не позволяет строить сети с развитой структурой, например, сети, объединяющие несколько сетей предприятия в единую сеть, или высоконадежные сети, в которых существуют избыточные связи между узлами. Можно было бы усложнять протоколы канального уровня для поддержания петлевидных избыточных связей, но принцип разделения обязанностей между уровнями приводит к другому решению. Чтобы с одной стороны сохранить простоту процедур передачи данных для типовых топологий, а с другой допустить использование произвольных топологий, вводится дополнительный сетевой уровень.

На сетевом уровне сам термин *сеть* наделяют специфическим значением. В данном случае под сетью понимается совокупность компьютеров, соединенных между собой в соответствии с одной из стандартных типовых топологий и использующих для передачи данных один из протоколов канального уровня, определенный для этой топологии.

Внутри сети доставка данных обеспечивается соответствующим канальным уровнем, а вот доставкой данных между сетями занимается сетевой уровень, который и поддерживает возможность правильного выбора маршрута передачи сообщения даже в том случае, когда структура связей между составляющими сетями имеет характер, отличный от принятого в протоколах канального уровня.

Сети соединяются между собой специальными устройствами, называемыми маршрутизаторами. *Маршрутизатор* – это устройство, которое собирает информацию о топологии межсетевых соединений и на ее основании пересылает пакеты сетевого уровня в сеть назначения. Чтобы передать сообщение от отправителя, находящегося в одной сети, получателю, находящемуся в другой сети, нужно совершить некоторое количество *транзитных передач между сетями*, или *хопов* (от *hop* – прыжок), каждый раз выбирая подходящий маршрут. Таким образом, маршрут представляет собой последовательность маршрутизаторов, через которые проходит пакет.

На рис. 1.27 показаны четыре сети, связанные тремя маршрутизаторами. Между узлами A и B данной сети пролегают два маршрута: первый через маршрутизаторы 1 и 3, а второй через маршрутизаторы 1, 2 и 3.

Проблема выбора наилучшего пути называется *маршрутизацией*, и ее решение является одной из главных задач сетевого уровня. Эта проблема осложняется тем, что самый короткий путь не всегда самый лучший. Часто критерием при выборе маршрута является время передачи данных по этому маршруту; оно зависит от пропускной способности каналов связи и интенсивности трафика, которая может изменяться с течением времени. Некоторые алгоритмы маршрутизации пытаются приспособиться к изменению нагрузки, в то время как другие принимают решения на основе средних показателей за длительное время. Выбор маршрута может осуществляться и по другим критериям, например надежности передачи.

В общем случае функции сетевого уровня шире, чем функций передачи сообщений по связям с нестандартной структурой, которые мы сейчас рассмотрели на примере объединения нескольких локальных сетей, Сетевой уровень решает также задачи согласования разных технологий, упрощения

адресации в крупных сетях и создания надежных и гибких барьеров на пути нежелательного трафика между сетями.

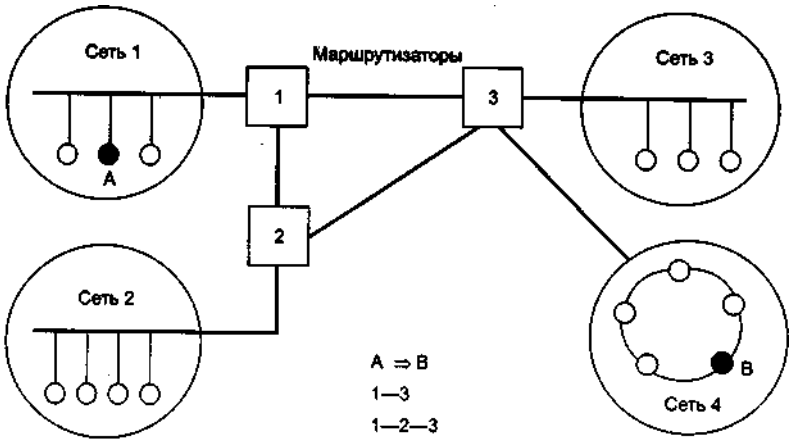


Рис. 1.27. Пример составной сети

Сообщения сетевого уровня принято называть *пакетами* (*packets*). При организации доставки пакетов на сетевом уровне используется понятие «номер сети». В этом случае адрес получателя состоит из старшей части – номера сети и младшей – номера узла в этой сети. Все узлы одной сети должны иметь одну и ту же старшую часть адреса, поэтому термину «сеть» на сетевом уровне можно дать и другое, более формальное определение: сеть – это совокупность узлов, сетевой адрес которых содержит один и тот же номер сети.

На сетевом уровне определяются два вида протоколов. Первый вид – *сетевые протоколы* (*routed protocols*) – реализуют продвижение пакетов через сеть. Именно эти протоколы обычно имеют в виду, когда говорят о протоколах сетевого уровня. Однако часто к сетевому уровню относят и другой вид протоколов, называемых протоколами обмена маршрутной информацией или просто *протоколами маршрутизации* (*routing protocols*). С помощью этих протоколов маршрутизаторы собирают информацию о топологии межсетевых соединений. Протоколы сетевого уровня реализуются программными модулями операционной системы, а также программными и аппаратными средствами маршрутизаторов.

На сетевом уровне работают протоколы еще одного типа, которые отвечают за отображение адреса узла, используемого на сетевом уровне, в локальный адрес сети. Такие протоколы часто называют *протоколами разрешения адресов* – Address Resolution Protocol, ARP. Иногда их относят не к сетевому уровню, а к канальному, хотя тонкости классификации не изменяют их сути.

Примерами протоколов сетевого уровня являются протокол межсетевого взаимодействия IP стека TCP/IP и протокол межсетевого обмена пакетами IPX стека Novell,

Транспортный уровень

На пути от отправителя к получателю пакеты могут быть искажены или утеряны. Хотя некоторые приложения имеют собственные средства обработки ошибок, существуют и такие, которые предпочитают сразу иметь дело с надежным соединением. Транспортный уровень (Transport layer) обеспечивает приложениям или верхним уровням стека – прикладному и сеансовому – передачу данных с той степенью надежности, которая им требуется. Модель OSI определяет пять классов сервиса, предоставляемых транспортным уровнем. Эти виды сервиса отличаются качеством предоставляемых услуг: срочностью, возможностью восстановления прерванной связи, наличием средств мультиплексирования нескольких соединений между различными прикладными протоколами через общий транспортный протокол, а главное – способностью к обнаружению и исправлению ошибок передачи, таких как искажение, потеря и дублирование пакетов.

Выбор класса сервиса транспортного уровня определяется, с одной стороны, тем, в какой степени задача обеспечения надежности решается самими приложениями и протоколами более высоких, чем транспортный, уровней, а с другой стороны, этот выбор зависит от того, насколько надежной является система транспортировки данных в сети, обеспечиваемая уровнями, расположенными ниже транспортного – сетевым, канальным и физическим. Так, например, если качество каналов передачи

связи очень высокое и вероятность возникновения ошибок, не обнаруженных протоколами более низких уровней, невелика, то разумно воспользоваться одним из облегченных сервисов транспортного уровня, не обремененных многочисленными проверками, квитированием и другими приемами повышения надежности. Если же транспортные средства нижних уровней изначально очень ненадежны, то целесообразно обратиться к наиболее развитому сервису транспортного уровня, который работает, используя максимум средств для обнаружения и устранения ошибок, – с помощью предварительного установления логического соединения, контроля доставки сообщений по контрольным суммам и циклической нумерации пакетов, установления тайм-аутов доставки и т. п.

Как правило, все протоколы, начиная с транспортного уровня и выше, реализуются программными средствами конечных узлов сети – компонентами их сетевых операционных систем. В качестве примера транспортных протоколов можно привести протоколы TCP и UDP стека TCP/IP и протокол SPX стека Novell.

Протоколы нижних четырех уровней обобщенно называют сетевым транспортом или транспортной подсистемой, так как они полностью решают задачу транспортировки сообщений с заданным уровнем качества в составных сетях с произвольной топологией и различными технологиями. Остальные три верхних уровня решают задачи предоставления прикладных сервисов на основании имеющейся транспортной подсистемы.

Сеансовый уровень

Сеансовый уровень (Session layer) обеспечивает управление диалогом: фиксирует, какая из сторон является активной в настоящий момент, предоставляет средства синхронизации. Последние позволяют вставлять контрольные точки в длинные передачи, чтобы в случае отказа можно было вернуться назад к последней контрольной точке, а не начинать все с начала. На практике немногие приложения используют сеансовый уровень, и он редко реализуется в виде отдельных протоколов, хотя функции этого уровня часто объединяют с функциями прикладного уровня и реализуют в одном протоколе.

Представительный уровень

Представительный уровень (Presentation layer) имеет дело с формой представления передаваемой по сети информации, не меняя при этом ее содержания. За счет уровня представления информация, передаваемая прикладным уровнем одной системы, всегда понятна прикладному уровню другой системы. С помощью средств данного уровня протоколы прикладных уровней могут преодолеть синтаксические различия в представлении данных или же различия в кодах символов, например кодов ASCII и EBCDIC. На этом уровне может выполняться шифрование и дешифрование данных, благодаря которому секретность обмена данными обеспечивается сразу для всех прикладных служб. Примером такого протокола является протокол Secure Socket Layer (SSL), который обеспечивает секретный обмен сообщениями для протоколов прикладного уровня стека TCP/IP.

Прикладной уровень

Прикладной уровень (Application layer) – это в действительности просто набор разнообразных протоколов, с помощью которых пользователи сети получают доступ к разделяемым ресурсам, таким как файлы, принтеры или гипертекстовые Web-страницы, а также организуют свою совместную работу, например, с помощью протокола электронной почты. Единица данных, которой оперирует прикладной уровень, обычно называется *сообщением* (*message*).

Существует очень большое разнообразие служб прикладного уровня. Приведем в качестве примера хотя бы несколько наиболее распространенных реализаций файловых служб: NCP в операционной системе Novell NetWare, SMB в Microsoft Windows NT, NFS, FTP и TFTP, входящие в стек TCP/IP.

Сетезависимые и сетенезависимые уровни

Функции всех уровней модели OSI могут быть отнесены к одной из двух групп: либо к функциям, зависящим от конкретной технической реализации сети, либо к функциям, ориентированным на работу с приложениями.

Три нижних уровня – физический, канальный и сетевой – являются сетезависимыми, то есть протоколы этих уровней тесно связаны с технической реализацией сети и используемым коммуникационным оборудованием. Например, переход на оборудование FDDI означает полную смену протоколов физического и канального уровней во всех узлах сети.

Три верхних уровня – прикладной, представительный и сеансовый – ориентированы на приложения и мало зависят от технических особенностей построения сети. На протоколы этих уровней не влияют какие бы то ни было изменения в топологии сети, замена оборудования или переход на другую сетевую технологию. Так, переход от Ethernet на высокоскоростную технологию lOOVG-AnyLAN не потребует никаких изменений в программных средствах, реализующих функции прикладного, представительного и сеансового уровней.

Транспортный уровень является промежуточным, он скрывает все детали функционирования нижних уровней от верхних. Это позволяет разрабатывать приложения, не зависящие от технических средств непосредственной транспортировки сообщений.

На рис. 1.28 показаны уровни модели OSI, на которых работают различные элементы сети. Компьютер с установленной на нем сетевой ОС взаимодействует с другим компьютером с помощью протоколов всех семи уровней. Это взаимодействие компьютеры осуществляют опосредовано через различные коммуникационные устройства: концентраторы, модемы, мосты, коммутаторы, маршрутизаторы, мультиплексоры. В зависимости от типа коммуникационное устройство может работать либо только на физическом уровне (повторитель), либо на физическом и канальном (мост), либо на физическом, канальном и сетевом, иногда захватывая и транспортный уровень (маршрутизатор). На рис. 1.29 показано соответствие функций различных коммуникационных устройств уровням модели OSI.

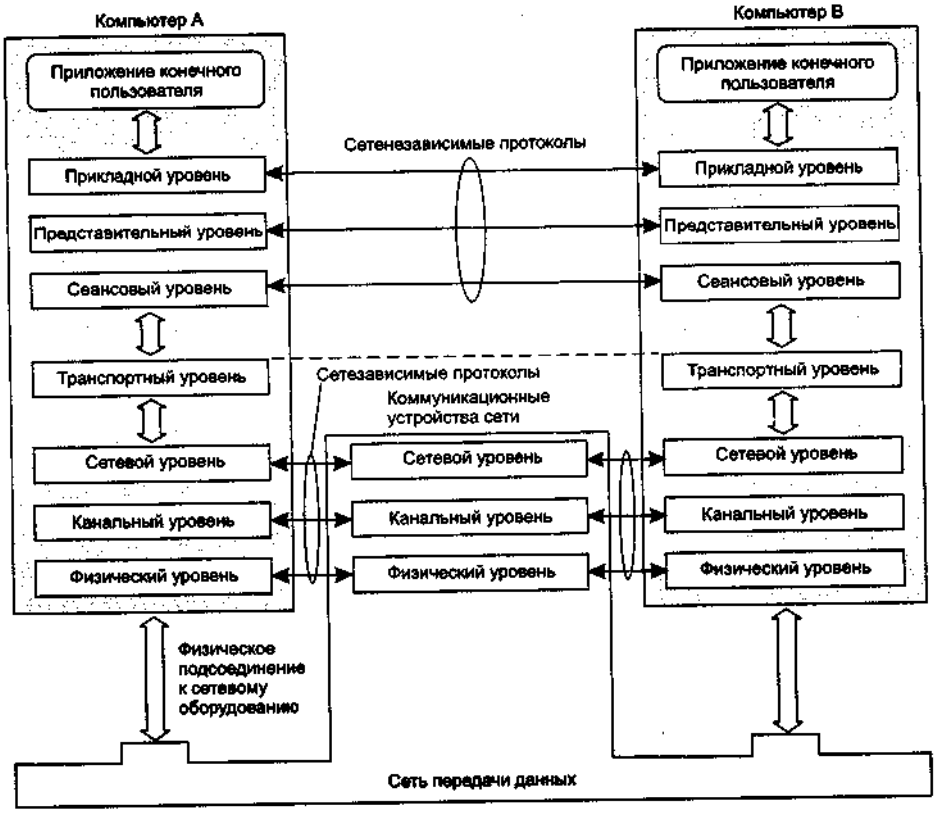


Рис. 1.28. Сетезависимые и сетенезависимые уровни модели OSI

Модель OSI представляет хотя и очень важную, но только одну из многих моделей коммуникаций. Эти модели и связанные с ними стеки протоколов могут отличаться количеством уровней, их функциями, форматами сообщений, службами, поддерживаемыми на верхних уровнях, и прочими параметрами.

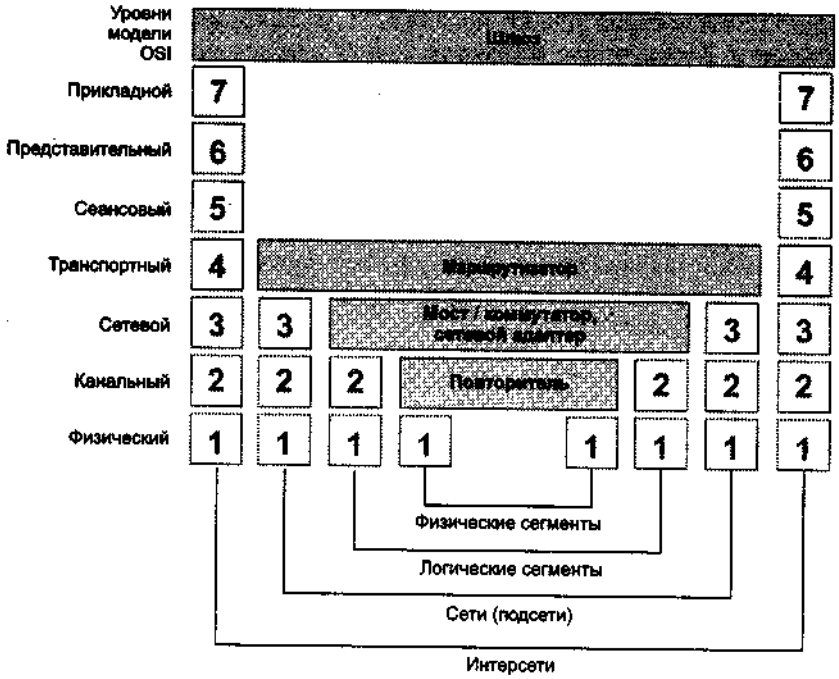


Рис. 1.29. Соответствие функций различных устройств сети уровням модели OSI

1.3.4. Понятие «открытая система»

Модель OSI, как это следует из ее названия (Open System Interconnection), описывает взаимосвязи открытых систем. Что же такое открытая система?

В широком смысле *открытой системой* может быть названа любая система (компьютер, вычислительная сеть, ОС, программный пакет, другие аппаратные и программные продукты), которая построена в соответствии с открытыми спецификациями.

Напомним, что под термином «спецификация» (в вычислительной технике) понимают формализованное описание аппаратных или программных компонентов, способов их функционирования, взаимодействия с другими компонентами, условий эксплуатации, ограничений и особых характеристик. Понятно, что не всякая спецификация является стандартом. В свою очередь, под открытыми спецификациями понимаются опубликованные, общедоступные спецификации, соответствующие стандартам и принятые в результате достижения согласия после всестороннего обсуждения всеми заинтересованными сторонами.

Использование при разработке систем открытых спецификаций позволяет третьим сторонам разрабатывать для этих систем различные аппаратные или программные средства расширения и модификации, а также создавать программно-аппаратные комплексы из продуктов разных производителей.

Для реальных систем полная открытость является недостижимым идеалом. Как правило, даже в системах, называемых открытыми, этому определению соответствуют лишь некоторые части, поддерживающие внешние интерфейсы. Например, открытость семейства операционных систем Unix заключается, кроме всего прочего, в наличии стандартизованного программного интерфейса между ядром и приложениями, что позволяет легко переносить приложения из среды одной версии Unix в среду другой версии. Еще одним примером частичной открытости является применение в достаточно закрытой операционной системе Novell NetWare открытого интерфейса Open Driver Interface (ODI) для включения в систему драйверов сетевых адаптеров независимых производителей. Чем больше открытых спецификаций использовано при разработке системы, тем более открытой она является.

Модель OSI касается только одного аспекта открытости, а именно открытости средств взаимодействия устройств, связанных в вычислительную сеть. Здесь под открытой системой понимается сетевое устройство, готовое взаимодействовать с другими сетевыми устройствами с использованием стандартных правил, определяющих формат, содержание и значение принимаемых и отправляемых сообщений.

Если две сети построены с соблюдением принципов открытости, то это дает следующие

преимущества:

- возможность построения сети из аппаратных и программных средств различных производителей, придерживающихся одного и того же стандарта;
- возможность безболезненной замены отдельных компонентов сети другими, более совершенными, что позволяет сети развиваться с минимальными затратами;
- возможность легкого сопряжения одной сети с другой;
- простота освоения и обслуживания сети.

Ярким примером открытой системы является международная сеть Internet. Эта сеть развивалась в полном соответствии с требованиями, предъявляемыми к открытым системам. В разработке ее стандартов принимали участие тысячи специалистов-пользователей этой сети из различных университетов, научных организаций и фирм-производителей вычислительной аппаратуры и программного обеспечения, работающих в разных странах. Само название стандартов, определяющих работу сети Internet – Request For Comments (RFC), что можно перевести как «запрос на комментарии», – показывает гласный и открытый характер принимаемых стандартов. В результате сеть Internet сумела объединить в себе самое разнообразное оборудование и программное обеспечение огромного числа сетей, разбросанных по всему миру.

1.3.5. Модульность и стандартизация

Модульность – это одно из неотъемлемых и естественных свойств вычислительных сетей. Модульность проявляется не только в многоуровневом представлении коммуникационных протоколов в конечных узлах сети, хотя это, безусловно, важная и принципиальная особенность сетевой архитектуры. Сеть состоит из огромного числа различных модулей – компьютеров, сетевых адаптеров, мостов, маршрутизаторов, модемов, операционных систем и модулей приложений. Разнообразные требования, предъявляемые предприятиями к компьютерным сетям, привели к такому же разнообразию выпускаемых для построения сети устройств и программ. Эти продукты отличаются не только основными функциями (имеются в виду функции, выполняемые, например, повторителями, мостами или программными редиректорами), но и многочисленными вспомогательными функциями, предоставляющими пользователям или администраторам дополнительные удобства, такие как автоматизированное конфигурирование параметров устройства, автоматическое обнаружение и устранение некоторых неисправностей, возможность программного изменения связей в сети и т. п. Разнообразие увеличивается также потому, что, многие устройства и программы отличаются сочетаниями тех или иных основных и дополнительных функций – существуют, например, устройства, сочетающие основные возможности коммутаторов и маршрутизаторов, к которым добавляется еще и набор некоторых дополнительных функций, характерный только для данного продукта.

В результате не существует компании, которая смогла бы обеспечить производство полного набора всех типов и подтипов оборудования и программного обеспечения, требуемого для построения сети. Но, так как все компоненты сети должны работать согласованно, совершенно необходимым оказалось принятие многочисленных стандартов, которые, если не во всех, то хотя бы в большинстве случаев, гарантировали бы совместимость оборудования и программ различных фирм-изготовителей. Таким образом, понятия модульности и стандартизации в сетях неразрывно связаны, и модульный подход только тогда дает преимущества, когда он сопровождается следованием стандартам.

В результате открытый характер стандартов и спецификаций важен не только для коммуникационных протоколов, но и для всех многочисленных функций разнообразных устройств и программ, выпускаемых для построения сети. Нужно отметить, что большинство стандартов, принимаемых сегодня, носят открытый характер. Время закрытых систем, точные спецификации на которые были известны только фирме-производителю, ушло. Все осознали, что возможность легкого взаимодействия с продуктами конкурентов не снижает, а наоборот, повышает ценность изделия, так как его можно применить в большем количестве работающих сетей, построенных на продуктах разных производителей. Поэтому даже фирмы, ранее выпускавшие весьма закрытые системы – такие как IBM, Novell или Microsoft, – сегодня активно участвуют в разработке открытых стандартов и применяют их в своих продуктах.

Сегодня в секторе сетевого оборудования и программ с совместимостью продуктов разных производителей сложилась следующая ситуация. Практически все продукты, как программные, так и аппаратные, совместимы по функциям и свойствам, которые были внедрены в практику уже достаточно

давно и стандарты на которые уже разработаны и приняты по крайней мере 3-4 года назад. В то же время очень часто принципиально новые устройства, протоколы и свойства оказываются несовместимыми даже у ведущих производителей. Такая ситуация наблюдается не только для тех устройств или функций, стандарты на которые еще не успели принять (это естественно), но и для устройств, стандарты на которые существуют уже несколько лет. Совместимость достигается только после того, как все производители реализуют этот стандарт в своих изделиях, причем одинаковым образом.

1.3,6. Источники стандартов

Работы по стандартизации вычислительных сетей ведутся большим количеством организаций, В зависимости от статуса организаций различают следующие виды стандартов:

- *стандарты отдельных фирм* (например, стек протоколов DECnet фирмы Digital Equipment или графический интерфейс OPEN LOOK для Unix-систем фирмы Sun);
- *стандарты специальных комитетов и объединений*, создаваемых несколькими фирмами, например стандарты технологии ATM, разрабатываемые специально созданным объединением ATM Forum, насчитывающем около 100 коллективных участников, или стандарты союза Fast Ethernet Alliance по разработке стандартов 100 Мбит Ethernet;
- *национальные стандарты*, например, стандарт FDDI, представляющий один из многочисленных стандартов, разработанных Американским национальным институтом стандартов (ANSI), или стандарты безопасности для операционных систем, разработанные Национальным центром компьютерной безопасности (NCSC) Министерства обороны США;
- *международные стандарты*, например, модель и стек коммуникационных протоколов Международной организации по стандартам (ISO), многочисленные стандарты Международного союза электросвязи (ITU), в том числе стандарты на сети с коммутацией пакетов X.25, сети frame relay, ISDN, модемы и многие другие.

Некоторые стандарты, непрерывно развиваясь, могут переходить из одной категории в другую. В частности, фирменные стандарты на продукцию, получившую широкое распространение, обычно становятся международными стандартами де-факто, так как вынуждают производителей из разных стран следовать фирменным стандартам, чтобы обеспечить совместимость своих изделий с этими популярными продуктами. Например, из-за феноменального успеха персонального компьютера компании IBM фирменный стандарт на архитектуру IBM PC стал международным стандартом де-факто.

Более того, ввиду широкого распространения некоторые фирменные стандарты становятся основой для национальных и международных стандартов де-юре. Например, стандарт Ethernet, первоначально разработанный компаниями Digital Equipment, Intel и Xerox, через некоторое время и в несколько измененном виде был принят как национальный стандарт IEEE 802.3, а затем организация ISO утвердила его в качестве международного стандарта ISO 8802.3.

Далее приводятся краткие сведения об организациях, наиболее активно и успешно занимающихся разработкой стандартов в области вычислительных сетей.

- *Международная организация по стандартизации* (*International Organization for Standardization, ISO*, часто называемая также *International Standards Organization*) представляет собой ассоциацию ведущих национальных организаций по стандартизации разных стран. Главным достижением ISO явилась модель взаимодействия открытых систем OSI, которая в настоящее время является концептуальной основой стандартизации в области вычислительных сетей. В соответствии с моделью OSI этой организацией был разработан стандартный стек коммуникационных протоколов OSI.
- *Международный союз электросвязи* (*International Telecommunications Union, ITU*) – организация, являющаяся в настоящее время специализированным органом Организации Объединенных Наций. Наиболее значительную роль в стандартизации вычислительных сетей играет постоянно действующий в рамках этой организации Международный консультативный комитет по телефонии и телеграфии (МККТТ) (Consultative Committee on International Telegraphy and Telephony, CCITT). В результате проведенной в 1993 году реорганизации ITU CCITT несколько изменил направление своей деятельности и сменил название – теперь он называется сектором телекоммуникационной стандартизации ITU (ITU Telecommunication Standardization Sector, ITU-T). Основу деятельности ITU-T составляет разработка

международных стандартов в области телефонии, телематических служб (электронной почты, факсимильной связи, телетекста, телекса и т. д.), передачи данных, аудио- и видеосигналов. За годы своей деятельности ITU-T выпустил огромное число рекомендаций-стандартов. Свою работу ITU-T строит на изучении опыта сторонних организаций, а также на результатах собственных исследований. Раз в четыре года издаются труды ITU-T в виде так называемой «Книги», которая на самом деле представляет собой целый набор обычных книг, сгруппированных в выпуски, которые, в свою очередь, объединяются в тома. Каждый том и выпуск содержат логически взаимосвязанные рекомендации. Например, том III Синей Книги содержит рекомендации для цифровых сетей с интеграцией услуг (ISDN), а весь том VIII (за исключением выпуска VIII. 1, который содержит рекомендации серии V для передачи данных по телефонной сети) посвящен рекомендациям серии X: X.25 для сетей с коммутацией пакетов, X.400 для систем электронной почты, X.500 для глобальной справочной службы и многим другим.

- *Институт инженеров по электротехнике и радиоэлектронике – Institute of Electrical and Electronics Engineers, IEEE*) – национальная организация США, определяющая сетевые стандарты. В 1981 году рабочая группа 802 этого института сформулировала основные требования, которым должны удовлетворять локальные вычислительные сети. Группа 802 определила множество стандартов, из них самыми известными являются стандарты 802.1, 802.2, 802.3 и 802.5, которые описывают общие понятия, используемые в области локальных сетей, а также стандарты на два нижних уровня сетей Ethernet и Token Ring.
- *Европейская ассоциация производителей компьютеров* (*European Computer Manufacturers Association, ECMA*) – некоммерческая организация, активно сотрудничающая с ITU-T и ISO, занимается разработкой стандартов и технических обзоров, относящихся к компьютерной и коммуникационной технологиям. Известна своим стандартом ECMA-101, используемым при передаче отформатированного текста и графических изображений с сохранением оригинального формата.
- *Ассоциация производителей компьютеров и оргтехники* (*Computer and Business Equipment Manufacturers Association, CBEMA*) – организация американских фирм-производителей аппаратного обеспечения; аналогична европейской ассоциации ЕКМА; участвует в разработке стандартов на обработку информации и соответствующее оборудование.
- *Ассоциация электронной промышленности* (*Eelktronic Industries Association, EIA*) – промышленно-торговая группа производителей электронного и сетевого оборудования; является национальной коммерческой ассоциацией США; проявляет значительную активность в разработке стандартов для проводов, коннекторов и других сетевых компонентов. Ее наиболее известный стандарт – RS-232C.
- *Министерство обороны США* (*Department of Defense, DoD*) имеет многочисленные подразделения, занимающиеся созданием стандартов для компьютерных систем. Одной из самых известных разработок DoD является стек транспортных протоколов TCP/IP.
- *Американский национальный институт стандартов* (*American National Standards Institute, ANSI*) – эта организация представляет США в Международной организации по стандартизации ISO. Комитеты ANSI ведут работу по разработке стандартов в различных областях вычислительной техники. Так, комитет ANSI X3T9.5 совместно с фирмой IBM занимается стандартизацией локальных сетей крупных ЭВМ (архитектура сетей SNA). Известный стандарт FDDI также является результатом деятельности этого комитета ANSI. В области микрокомпьютеров ANSI разрабатывает стандарты на языки программирования, интерфейс SCSI. ANSI разработал рекомендации по переносимости для языков C. FORTRAN, COBOL.

Особую роль в выработке международных открытых стандартов играют стандарты Internet. Ввиду большой и постоянной растущей популярности Internet, эти стандарты становятся международными стандартами «де-факто», многие из которых затем приобретают статус официальных международных стандартов за счет их утверждения одной из вышеперечисленных организаций, в том числе ISO и ITU-T. Существует несколько организационных подразделений, отвечающие за развитие Internet и, в частности, за стандартизацию средств Internet.

Основным из них является Internet Society (ISOC) – профессиональное сообщество, которое занимается общими вопросами эволюции и роста Internet как глобальной коммуникационной

инфраструктуры. Под управлением ISOC работает Internet Architecture Board (IAB) – организация, в ведении которой находится технический контроль и координация работ для Internet. IAB координирует направление исследований и новых разработок для стека TCP/IP и является конечной инстанцией при определении новых стандартов Internet.

BIAB входят две основные группы: Internet Engineering Task Force (IETF) и Internet Research Task Force (IRTF). IETF – это инженерная группа, которая занимается решением ближайших технических проблем Internet. Именно IETF определяет спецификации, которые затем становятся стандартами Internet. В свою очередь, IRTF координирует долгосрочные исследовательские проекты по протоколам TCP/IP.

В любой организации, занимающейся стандартизацией, процесс выработки и принятия стандарта состоит из ряда обязательных этапов, которые, собственно, и составляют процедуру стандартизации. Рассмотрим эту процедуру на примере разработки стандартов Internet.

- Сначала в IETF представляется так называемый *рабочий проект* (*draft*) в виде, доступном для комментариев. Он публикуется в Internet, после чего широкий круг заинтересованных лиц включается в обсуждение этого документа, в него вносятся исправления, и наконец наступает момент, когда можно зафиксировать содержание документа. На этом этапе проекту присваивается номер RFC (возможен и другой вариант развития событий – после обсуждения рабочий проект отвергается и удаляется из Internet).
- После присвоения номера проект приобретает статус *предлагаемого стандарта*. В течение 6 месяцев этот предлагаемый стандарт проходит проверку практикой, в результате в него вносятся изменения.
- Если результаты практических исследований показывают эффективность предлагаемого стандарта, то ему, со всеми внесенными изменениями, присваивается статус *проекта стандарта*. Затем в течение не менее 4-х месяцев проходят его дальнейшие испытания «на прочность», в число которых входит создание по крайней мере двух программных реализаций.
- Если во время пребывания в ранге проекта стандарта в документ не было внесено никаких исправлений, то ему может быть присвоен статус *официального стандарта* Internet. Список утвержденных официальных стандартов Internet публикуется в виде документа RFC и доступен в Internet.

Следует заметить, что все стандарты Internet носят название RFC с соответствующим порядковым номером, но далеко не все RFC являются стандартами Internet – часто эти документы представляют собой комментарии к какому-либо стандарту или просто описания некоторой проблемы Internet.

1.3.7. Стандартные стеки коммуникационных протоколов

Важнейшим направлением стандартизации в области вычислительных сетей является стандартизация коммуникационных протоколов. В настоящее время в сетях используется большое количество стеков коммуникационных протоколов. Наиболее популярными являются стеки: TCP/IP, IPX/SPX, NetBIOS/SMB, DECnet, SNA и OSI. Все эти стеки, кроме SNA на нижних уровнях – физическом и канальном, – используют одни и те же хорошо стандартизованные протоколы Ethernet, Token Ring, FDDI и некоторые другие, которые позволяют использовать во всех сетях одну и ту же аппаратуру. Зато на верхних уровнях все стеки работают по своим собственным протоколам. Эти протоколы часто не соответствуют рекомендуемому моделью OSI разбиению на уровни. В частности, функции сеансового и представительного уровня, как правило, объединены с прикладным уровнем. Такое несоответствие связано с тем, что модель OSI появилась как результат обобщения уже существующих и реально используемых стеков, а не наоборот.

Стек OSI

Следует четко различать модель OSI и стек OSI. В то время как модель OSI является концептуальной схемой взаимодействия открытых систем, стек OSI представляет собой набор вполне конкретных спецификаций протоколов. В отличие от других стеков протоколов стек OSI полностью соответствует модели OSI, он включает спецификации протоколов для всех семи уровней взаимодействия, определенных в этой модели. На нижних уровнях стек OSI поддерживает Ethernet, Token Ring, FDDI, протоколы глобальных сетей, X.25 и ISDN, – то есть использует разработанные вне стека протоколы

нижних уровней, как и все другие стеки. Протоколы сетевого, транспортного и сеансового уровней стека OSI специфицированы и реализованы различными производителями, но распространены пока мало. Наиболее популярными протоколами стека OSI являются прикладные протоколы. К ним относятся: протокол передачи файлов FTAM, протокол эмуляции терминала VTP, протоколы справочной службы X.500, электронной почты X.400 и ряд других.

Протоколы стека OSI отличает большая сложность и неоднозначность спецификаций. Эти свойства явились результатом общей политики разработчиков стека, стремившихся учесть в своих протоколах все случаи жизни и все существующие и появляющиеся технологии. К этому нужно еще добавить и последствия большого количества политических компромиссов, неизбежных при принятии международных стандартов по такому злободневному вопросу, как построение открытых вычислительных сетей.

Из-за своей сложности протоколы OSI требуют больших затрат вычислительной мощности центрального процессора, что делает их наиболее подходящими для мощных машин, а не для сетей персональных компьютеров.

Стек OSI – международный, независимый от производителей стандарт. Его поддерживает правительство США в своей программе GOSIP, в соответствии с которой все компьютерные сети, устанавливаемые в правительственных учреждениях США после 1990 года, должны или непосредственно поддерживать стек OSI или обеспечивать средства для перехода на этот стек в будущем. Тем не менее стек OSI более популярен в Европе, чем в США, так как в Европе осталось меньше старых сетей, работающих по своим собственным протоколам. Большинство организаций пока только планируют переход к стеку OSI, и очень немногие приступили к созданию пилотных проектов. Из тех, кто работает в этом направлении, можно назвать Военно-морское ведомство США и сеть NFSNET. Одним из крупнейших производителей, поддерживающих OSI, является компания AT&T, ее сеть Stargroup полностью базируется на этом стеке.

Стек TCP/IP

Стек TCP/IP был разработан по инициативе Министерства обороны США более 20 лет назад для связи экспериментальной сети ARPAnet с другими сетями как набор общих протоколов для разнородной вычислительной среды. Большой вклад в развитие стека TCP/IP, который получил свое название по популярным протоколам IP и TCP, внес университет Беркли, реализовав протоколы стека в своей версии ОС UNIX. Популярность этой операционной системы привела к широкому распространению протоколов TCP, IP и других протоколов стека. Сегодня этот стек используется для связи компьютеров всемирной информационной сети Internet, а также в огромном числе корпоративных сетей.

Стек TCP/IP на нижнем уровне поддерживает все популярные стандарты физического и канального уровней: для локальных сетей – это Ethernet, Token Ring, FDDI, для глобальных – протоколы работы на аналоговых коммутируемых и выделенных линиях SLIP, PPP, протоколы территориальных сетей X.25 и ISDN.

Основными протоколами стека, давшими ему название, являются протоколы IP и TCP. Эти протоколы в терминологии модели OSI относятся к сетевому и транспортному уровням соответственно. IP обеспечивает продвижение пакета по составной сети, а TCP гарантирует надежность его доставки.

За долгие годы использования в сетях различных стран и организаций стек TCP/IP вобрал в себя большое количество протоколов прикладного уровня. К ним относятся такие популярные протоколы, как протокол пересылки файлов FTP, протокол эмуляции терминала telnet, почтовый протокол SMTP, используемый в электронной почте сети Internet, гипертекстовые сервисы службы WWW и многие другие.

Сегодня стек TCP/IP представляет собой один из самых распространенных стеков транспортных протоколов вычислительных сетей. Действительно, только в сети Internet объединено около 10 миллионов компьютеров по всему миру, которые взаимодействуют друг с другом с помощью стека протоколов TCP/IP.

Стремительный рост популярности Internet привел и к изменениям в расстановке сил в мире коммуникационных протоколов – протоколы TCP/IP, на которых построен Internet, стали быстро теснить бесспорного лидера прошлых лет – стек IPX/SPX компании Novell. Сегодня в мире общее количество компьютеров, на которых установлен стек TCP/IP, сравнялось с общим количеством

компьютеров, на которых работает стек IPX/SPX, и это говорит о резком переломе в отношении администраторов локальных сетей к протоколам, используемым на настольных компьютерах, так как именно они составляют подавляющее число мирового компьютерного парка и именно на них раньше почти везде работали протоколы компании Novell, необходимые для доступа к файловым серверам NetWare. Процесс становления стека TCP/IP в качестве стека номер один в любых типах сетей продолжается, и сейчас любая промышленная операционная система обязательно включает программную реализацию этого стека в своем комплекте поставки.

Хотя протоколы TCP/IP неразрывно связаны с Internet и каждый из многомиллионной армады компьютеров Internet работает на основе этого стека, существует большое количество локальных, корпоративных и территориальных сетей, непосредственно не являющихся частями Internet, в которых также используют протоколы TCP/IP. Чтобы отличать их от Internet, эти сети называют сетями TCP/IP или просто IP-сетями.

Поскольку стек TCP/IP изначально создавался для глобальной сети Internet, он имеет много особенностей, дающих ему преимущество перед другими протоколами, когда речь заходит о построении сетей, включающих глобальные связи. В частности, очень полезным свойством, делающим возможным применение этого протокола в больших сетях, является его способность фрагментировать пакеты. Действительно, большая составная сеть часто состоит из сетей, построенных на совершенно разных принципах. В каждой из этих сетей может быть установлена собственная величина максимальной длины единицы передаваемых данных (кадра). В таком случае при переходе из одной сети, имеющей большую максимальную длину, в сеть с меньшей максимальной длиной может возникнуть необходимость деления передаваемого кадра на несколько частей. Протокол IP стека TCP/IP эффективно решает эту задачу.

Другой особенностью технологии TCP/IP является гибкая система адресации, позволяющая более просто по сравнению с другими протоколами аналогичного назначения включать в интерсеть сети других технологий. Это свойство также способствует применению стека TCP/IP для построения больших гетерогенных сетей.

В стеке TCP/IP очень экономно используются возможности широковещательных рассылок. Это свойство совершенно необходимо при работе на медленных каналах связи, характерных для территориальных сетей.

Однако, как и всегда, за получаемые преимущества надо платить, и платой здесь оказываются высокие требования к ресурсам и сложность администрирования IP-сетей. Мощные функциональные возможности протоколов стека TCP/IP требуют для своей реализации высоких вычислительных затрат. Гибкая система адресации и отказ от широковещательных рассылок приводят к наличию в IP-сети различных централизованных служб типа DNS, DHCP и т. п. Каждая из этих служб направлена на облегчение администрирования сети, в том числе и на облегчение конфигурирования оборудования, но в то же время сама требует пристального внимания со стороны администраторов.

Можно приводить и другие доводы за и против стека протоколов Internet, однако факт остается фактом – сегодня это самый популярный стек протоколов, широко используемый как в глобальных, так и локальных сетях.

Стек IPX/SPX

Этот стек является оригинальным стеком протоколов фирмы Novell, разработанным для сетевой операционной системы NetWare еще в начале 80-х годов. Протоколы сетевого и сеансового уровней Internetwork Packet Exchange (IPX) и Sequenced Packet Exchange (SPX), которые дали название стеку, являются прямой адаптацией протоколов XNS фирмы Xerox, распространенных в гораздо меньшей степени, чем стек IPX/SPX. Популярность стека IPX/SPX непосредственно связана с операционной системой Novell NetWare, которая еще сохраняет мировое лидерство по числу установленных систем, хотя в последнее время ее популярность несколько снизилась и по темпам роста она отстает от Microsoft Windows NT.

Многие особенности стека IPX/SPX обусловлены ориентацией ранних версий ОС NetWare (до версии 4.0) на работу в локальных сетях небольших размеров, состоящих из персональных компьютеров со скромными ресурсами. Понятно, что для таких компьютеров компании Novell нужны были протоколы, на реализацию которых требовалось бы минимальное количество оперативной памяти (ограниченной в IBM-совместимых компьютерах под управлением MS-DOS объемом 640 Кбайт) и которые бы быстро

работали на процессорах небольшой вычислительной мощности. В результате протоколы стека IPX/SPX до недавнего времени хорошо работали в локальных сетях и не очень – в больших корпоративных сетях, так как они слишком перегружали медленные глобальные связи широковещательными пакетами, которые интенсивно используются несколькими протоколами этого стека (например, для установления связи между клиентами и серверами). Это обстоятельство, а также тот факт, что стек IPX/SPX является собственностью фирмы Novell и на его реализацию нужно получать лицензию (то есть открытые спецификации не поддерживались), долгое время ограничивали распространенность его только сетями NetWare. Однако с момента выпуска версии NetWare 4.0 Novell внесла и продолжает вносить в свои протоколы серьезные изменения, направленные на их адаптацию для работы в корпоративных сетях. Сейчас стек IPX/ SPX реализован не только в NetWare, но и в нескольких других популярных сетевых ОС, например SCO UNIX, Sun Solaris, Microsoft Windows NT.

Стек NetBIOS/SMB

Этот стек широко используется в продуктах компаний IBM и Microsoft. На физическом и канальном уровнях этого стека используются все наиболее распространенные протоколы Ethernet, Token Ring, FDDI и другие. На верхних уровнях работают протоколы NetBEUI и SMB.

Протокол NetBIOS (Network Basic Input/Output System) появился в 1984 году как сетевое расширение стандартных функций базовой системы ввода/вывода (BIOS) IBM PC для сетевой программы PC Network фирмы IBM. В дальнейшем этот протокол был заменен так называемым протоколом расширенного пользовательского интерфейса NetBEUI – NetBIOS Extended User Interface. Для обеспечения совместимости приложений в качестве интерфейса к протоколу NetBEUI был сохранен интерфейс NetBIOS. Протокол NetBEUI разрабатывался как эффективный протокол, потребляющий немного ресурсов и предназначенный для сетей, насчитывающих не более 200 рабочих станций. Этот протокол содержит много полезных сетевых функций, которые можно отнести к сетевому, транспортному и сеансовому уровням модели OSI, однако с его помощью невозможна маршрутизация пакетов. Это ограничивает применение протокола NetBEUI локальными сетями, не разделенными на подсети, и делает невозможным его использование в составных сетях. Некоторые ограничения NetBEUI снимаются реализацией этого протокола NBF (NetBEUI Frame), которая включена в операционную систему Microsoft Windows NT.

Протокол SMB (Server Message Block) выполняет функции сеансового, представительного и прикладного уровней. На основе SMB реализуется файловая служба, а также службы печати и передачи сообщений между приложениями.

Стеки протоколов SNA фирмы IBM, DECnet корпорации Digital Equipment и AppleTalk/AFP фирмы Apple применяются в основном в операционных системах и сетевом оборудовании этих фирм.

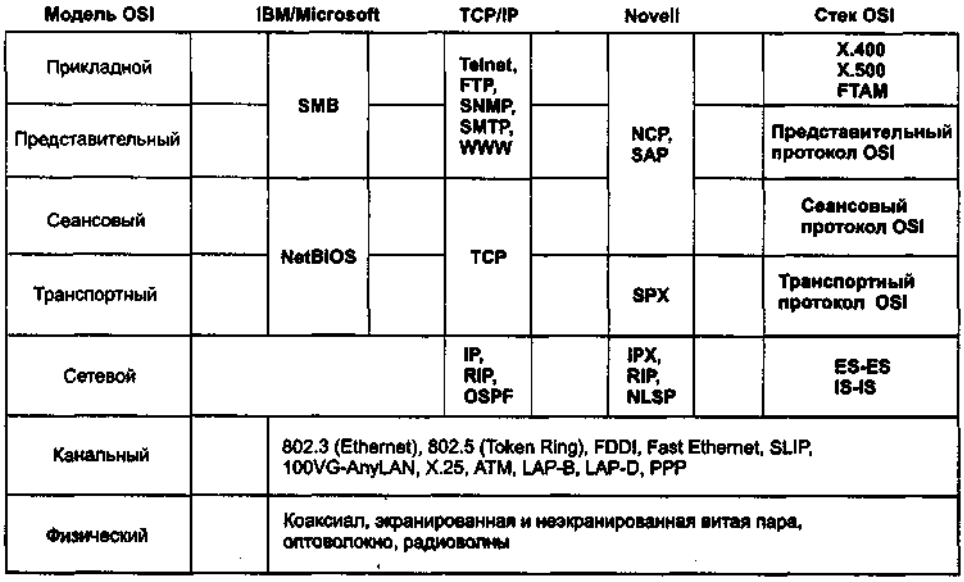


Рис. 1.30. Соответствие популярных стеков протоколов модели OSI

На рис. 1.30 показано соответствие некоторых, наиболее популярных протоколов уровням модели

OSI. Часто это соответствие весьма условно, так как модель OSI – это только руководство к действию, причем достаточно общее, а конкретные протоколы разрабатывались для решения специфических задач, причем многие из них появились до разработки модели OSI. В большинстве случаев разработчики стеков отдавали предпочтение скорости работы сети в ущерб модульности – ни один стек, кроме стека OSI, не разбит на семь уровней. Чаще всего в стеке явно выделяются 3-4 уровня: уровень сетевых адаптеров, в котором реализуются протоколы физического и канального уровней, сетевой уровень, транспортный уровень и уровень служб, вбирающий в себя функции сеансового, представительного и прикладного уровней.

Выводы

- В компьютерных сетях идеологической основой стандартизации является многоуровневый подход к разработке средств сетевого взаимодействия.
- Формализованные правила, определяющие последовательность и формат сообщений, которыми обмениваются сетевые компоненты, лежащие на одном уровне, но в разных узлах, называются протоколом.
- Формализованные правила, определяющие взаимодействие сетевых компонентов соседних уровней одного узла, называются интерфейсом. Интерфейс определяет набор сервисов, предоставляемый данным уровнем соседнему уровню.
- Иерархически организованный набор протоколов, достаточный для организации взаимодействия узлов в сети, называется стеком коммуникационных протоколов.
- Открытой системой может быть названа любая система, которая построена в соответствии с общедоступными спецификациями, соответствующими стандартам и принятыми в результате публичного обсуждения всеми заинтересованными сторонами.
- Модель OSI стандартизует взаимодействие открытых систем. Она определяет 7 уровней взаимодействия: прикладной, представительный, сеансовый, транспортный, сетевой, канальный и физический.
- Важнейшим направлением стандартизации в области вычислительных сетей является стандартизация коммуникационных протоколов. Наиболее популярными являются стеки: TCP/IP, IPX/SPX, NetBIOS/SMB, DECnet, SNA и OSI.

1.4. Локальные и глобальные сети

Для классификации компьютерных сетей используются различные признаки, но чаще всего сети делят на типы по территориальному признаку, то есть по величине территории, которую покрывает сеть. И для этого есть веские причины, так как отличия технологий локальных и глобальных сетей очень значительны, несмотря на их постоянное сближение.

1.4.1. Особенности локальных, глобальных и городских сетей

К *локальным сетям – Local Area Afetowfe* (*MN*) – относят сети компьютеров, сосредоточенные на небольшой территории (обычно в радиусе не более 1-2 км). В общем случае локальная сеть представляет собой коммуникационную систему, принадлежащую одной организации. Из-за коротких расстояний в локальных сетях имеется возможность использования относительно дорогих высококачественных линий связи, которые позволяют, применяя простые методы передачи данных, достигать высоких скоростей обмена данными порядка 100 Мбит/с. В связи с этим услуги, предоставляемые локальными сетями, отличаются широким разнообразием и обычно предусматривают реализацию в режиме on-line.

Глобальные сети – Wide Area Networks (*WAN*) - объединяют территориально рассредоточенные компьютеры, которые могут находиться в различных городах и странах. Так как прокладка высококачественных линий связи на большие расстояния обходится очень дорого, в глобальных сетях часто используются уже существующие линии связи, изначально предназначенные совсем для других целей. Например, многие глобальные сети строятся на основе телефонных и телеграфных каналов общего назначения. Из-за низких скоростей таких линий связи в глобальных сетях (десятки килобит в

секунду) набор предоставляемых услуг обычно ограничивается передачей файлов, преимущественно не в оперативном, а в фоновом режиме, с использованием электронной почты. Для устойчивой передачи дискретных данных по некачественным линиям связи применяются методы и оборудование, существенно отличающиеся от методов и оборудования, характерных для локальных сетей. Как правило, здесь применяются сложные процедуры контроля и восстановления данных, так как наиболее типичный режим передачи данных по территориальному каналу связи связан со значительными искажениями сигналов.

Городские сети (*или сети мегаполисов*) ~ *Metropolitan Area Networks* (*MAN*) -являются менее распространенным типом сетей. Эти сети появились сравнительно недавно. Они предназначены для обслуживания территории крупного города – мегаполиса. В то время как локальные сети наилучшим образом подходят для разделения ресурсов на коротких расстояниях и широковещательных передач, а глобальные сети обеспечивают работу на больших расстояниях, но с ограниченной скоростью и небогатым набором услуг, сети мегаполисов занимают некоторое промежуточное положение. Они используют цифровые магистральные линии связи, часто оптоволоконные, со скоростями от 45 Мбит/с, и предназначены для связи локальных сетей в масштабах города и соединения локальных сетей с глобальными. Эти сети первоначально были разработаны для передачи данных, но сейчас они поддерживают и такие услуги, как видеоконференции и интегральную передачу голоса и текста. Развитие технологии сетей мегаполисов осуществлялось местными телефонными компаниями. Исторически сложилось так, что местные телефонные компании всегда обладали слабыми техническими возможностями и из-за этого не могли привлечь крупных клиентов. Чтобы преодолеть свою отсталость и занять достойное место в мире локальных и глобальных сетей, местные предприятия связи занялись разработкой сетей на основе самых современных технологий, например технологии коммутации ячеек SMDS или ATM. Сети мегаполисов являются общественными сетями, и поэтому их услуги обходятся дешевле, чем построение собственной (частной) сети в пределах города.

1.4.2. Отличия локальных сетей от глобальных

Рассмотрим основные отличия локальных сетей от глобальных более детально. Так как в последнее время эти отличия становятся все менее заметными, то будем считать, что в данном разделе мы рассматриваем сети конца 80-х годов, когда эти отличия проявлялись весьма отчетливо, а современные тенденции сближения технологий локальных и глобальных сетей будут рассмотрены в следующем разделе.

- *Протяженность, качество и способ прокладки линий связи*. Класс локальных вычислительных сетей по определению отличается от класса глобальных сетей небольшим расстоянием между узлами сети. Это в принципе делает возможным использование в локальных сетях качественных линий связи: коаксиального кабеля, витой пары, оптоволоконного кабеля, которые не всегда доступны (из-за экономических ограничений) на больших расстояниях, свойственных глобальным сетям. В глобальных сетях часто применяются уже существующие линии связи (телеграфные или телефонные), а в локальных сетях они прокладываются заново.
- *Сложность методов передачи и оборудования*. В условиях низкой надежности физических каналов в глобальных сетях требуются более сложные, чем в локальных сетях, методы передачи данных и соответствующее оборудование. Так, в глобальных сетях широко применяются модуляция, асинхронные методы, сложные методы контрольного суммирования, квитирование и повторные передачи искаженных кадров. С другой стороны, качественные линии связи в локальных сетях позволили упростить процедуры передачи данных за счет применения немодулированных сигналов и отказа от обязательного подтверждения получения пакета.
- *Скорость обмена данными*. Одним из главных отличий локальных сетей от глобальных является наличие высокоскоростных каналов обмена данными между компьютерами, скорость которых (10, 16 и 100 Мбит/с) сравнима со скоростями работы устройств и узлов компьютера – дисков, внутренних шин обмена данными и т. п. За счет этого у пользователя локальной сети, подключенного к удаленному разделяемому ресурсу (например, диску сервера), складывается впечатление, что он пользуется этим диском, как «своим». Для глобальных сетей типичны гораздо более низкие скорости передачи данных – 2400, 9600, 28800,33600

бит/с, 56 и 64 Кбит/с и только на магистральных каналах – до 2 Мбит/с.

- *Разнообразие услуг*. Локальные сети предоставляют, как правило, широкий набор услуг – это различные виды услуг файловой службы, услуги печати, услуги службы передачи факсимильных сообщений, услуги баз данных, электронная почта и другие, в то время как глобальные сети в основном предоставляют почтовые услуги и иногда файловые услуги с ограниченными возможностями – передачу файлов из публичных архивов удаленных серверов без предварительного просмотра их содержания.
- *Оперативность выполнения запросов*. Время прохождения пакета через локальную сеть обычно составляет несколько миллисекунд, время же его передачи через глобальную сеть может достигать нескольких секунд. Низкая скорость передачи данных в глобальных сетях затрудняет реализацию служб для режима on-line, который является обычным для локальных сетей.
- *Разделение каналов*. В локальных сетях каналы связи используются, как правило, совместно сразу несколькими узлами сети, а в глобальных сетях – индивидуально.
- *Использование метода коммутации пакетов*. Важной особенностью локальных сетей является неравномерное распределение нагрузки. Отношение пиковой нагрузки к средней может составлять 100:1 и даже выше. Такой трафик обычно называют *пульсирующим*. Из-за этой особенности трафика в локальных сетях для связи узлов применяется метод коммутации пакетов, который для пульсирующего трафика оказывается гораздо более эффективным, чем традиционный для глобальных сетей метод коммутации каналов. Эффективность метода коммутации пакетов состоит в том, что сеть в целом передает в единицу времени больше данных своих абонентов. В глобальных сетях метод коммутации пакетов также используется, но наряду с ним часто применяется и метод коммутации каналов, а также некоммутируемые каналы – как унаследованные технологии некомпьютерных сетей.
- *Масштабируемость*. «Классические» локальные сети обладают плохой масштабируемостью из-за жесткости базовых топологий, определяющих способ подключения станций и длину линии. При использовании многих базовых топологий характеристики сети резко ухудшаются при достижении определенного предела по количеству узлов или протяженности линий связи. Глобальным же сетям присуща хорошая масштабируемость, так как они изначально разрабатывались в расчете на работу с произвольными топологиями.

1.4.3. Тенденция к сближению локальных и глобальных сетей

Если принять во внимание все перечисленные выше различия локальных и глобальных сетей, то становится понятным, почему так долго могли существовать раздельно два сообщества специалистов, занимающиеся этими двумя видами сетей. Но за последние годы ситуация резко изменилась.

Специалисты по локальным сетям, перед которыми встали задачи объединения нескольких локальных сетей, расположенных в разных, географически удаленных друг от друга пунктах, были вынуждены начать освоение чуждого для них мира глобальных сетей и телекоммуникаций. Тесная интеграция удаленных локальных сетей не позволяет рассматривать глобальные сети в виде «черного ящика», представляющего собой только инструмент транспортировки сообщений на большие расстояния. Поэтому все, что связано с глобальными связями и удаленным доступом, стало предметом повседневного интереса многих специалистов по локальным сетям.

С другой стороны, стремление повысить пропускную способность, скорость передачи данных, расширить набор и оперативность служб, другими словами, стремление улучшить качество предоставляемых услуг – все это заставило специалистов по глобальным сетям обратить пристальное внимание на технологии, используемые в локальных сетях.

Таким образом, в мире локальных и глобальных сетей явно наметилось движение навстречу друг другу, которое уже сегодня привело к значительному взаимопроникновению технологий локальных и глобальных сетей.

Одним из проявлений этого сближения является появление сетей масштаба большого города (MAN), занимающих промежуточное положение между локальными и глобальными сетями. При достаточно больших расстояниях между узлами они обладают качественными линиями связи и высокими скоростями обмена, даже более высокими, чем в классических локальных сетях. Как и в случае локальных сетей, при построении MAN уже существующие линии связи не используются, а

прокладываются заново.

Сближение в методах передачи данных происходит на платформе оптической цифровой (немодулированной) передачи данных по оптоволоконным линиям связи. Из-за резкого улучшения качества каналов связи в глобальных сетях начали отказываться от сложных и избыточных процедур обеспечения корректности передачи данных. Примером могут служить сети frame relay. В этих сетях предполагается, что искажение бит происходит настолько редко, что ошибочный пакет просто уничтожается, а все проблемы, связанные с его потерей, решаются программами прикладного уровня, которые непосредственно не входят в состав сети frame relay.

За счет новых сетевых технологий и, соответственно, нового оборудования, рассчитанного на более качественные линии связи, скорости передачи данных в уже существующих коммерческих глобальных сетях нового поколения приближаются к традиционным скоростям локальных сетей (в сетях frame relay сейчас доступны скорости 2 Мбит/с), а в глобальных сетях ATM и превосходят их, достигая 622 Мбит/с.

В результате службы для режима on-line становятся обычными и в глобальных сетях. Наиболее яркий пример – гипертекстовая информационная служба World Wide Web, ставшая основным поставщиком информации в сети Internet. Ее интерактивные возможности превзошли возможности многих аналогичных служб локальных сетей, так что разработчикам локальных сетей пришлось просто позаимствовать эту службу у глобальных сетей. Процесс переноса служб и технологий из глобальных сетей в локальные приобрел такой массовый характер, что появился даже специальный термин – intranet-технологии (intra – внутренний), обозначающий применение служб внешних (глобальных) сетей во внутренних – локальных.

Локальные сети перенимают у глобальных сетей и транспортные технологии. Все новые скоростные технологии (Fast Ethernet, Gigabit Ethernet, lOOVG-AnyLAN) поддерживают работу по индивидуальным линиям связи наряду с традиционными для локальных сетей разделяемыми линиями. Для организации индивидуальных линий связи используется специальный тип коммуникационного оборудования – коммутаторы. Коммутаторы локальных сетей соединяются между собой по иерархической схеме, подобно тому, как это делается в телефонных сетях: имеются коммутаторы нижнего уровня, к которым непосредственно подключаются компьютеры сети, коммутаторы следующего уровня соединяют между собой коммутаторы нижнего уровня и т. д. Коммутаторы более высоких уровней обладают, как правило, большей производительностью и работают с более скоростными каналами, уплотняя данные нижних уровней. Коммутаторы поддерживают не только новые протоколы локальных сетей, но и традиционные – Ethernet и Token Ring.

В локальных сетях в последнее время уделяется такое же большое внимание методам обеспечения защиты информации от несанкционированного доступа, как и в глобальных сетях. Такое внимание обусловлено тем, что локальные сети перестали быть изолированными, чаще всего они имеют выход в «большой мир» через глобальные связи. При этом часто используются те же методы – шифрование данных, аутентификация пользователей, возведение защитных барьеров, предохраняющих от проникновения в сеть извне.

И наконец, появляются новые технологии, изначально предназначенные для обоих видов сетей. Наиболее ярким представителем нового поколения технологий является технология ATM, которая может служить основой не только локальных и глобальных компьютерных сетей, но и телефонных сетей, а также широковещательных видеосетей, объединяя все существующие типы трафика в одной транспортной сети.

Выводы

- Классифицируя сети по территориальному признаку, различают локальные (LAN), глобальные (WAN) и городские (MAN) сети.
- LAN – сосредоточены на территории не более 1-2 км; построены с использованием дорогих высококачественных линий связи, которые позволяют, применяя простые методы передачи данных, достигать высоких скоростей обмена данными порядка 100 Мбит/с. Предоставляемые услуги отличаются широким разнообразием и обычно предусматривают реализацию в режиме on-line.
- WAN – объединяют компьютеры, рассредоточенные на расстоянии сотен и тысяч километров. Часто используются уже существующие не очень качественные линии связи. Более низкие, чем в локальных сетях, скорости передачи данных (десятки килобит в секунду) ограничивают

набор предоставляемых услуг передачей файлов, преимущественно не в оперативном, а в фоновом режиме, с использованием электронной почты. Для устойчивой передачи дискретных данных применяются более сложные методы и оборудование, чем в локальных сетях.

- MAN – занимают промежуточное положение между локальными и глобальными сетями. При достаточно больших расстояниях между узлами (десятки километров) они обладают качественными линиями связи и высокими скоростями обмена, иногда даже более высокими, чем в классических локальных сетях. Как и в случае локальных сетей, при построении MAN уже существующие линии связи не используются, а прокладываются заново.

1.5. Сети отделов, кампусов и корпораций

Еще одним популярным способом классификации сетей является их классификация по масштабу производственного подразделения, в пределах которого действует сеть. Различают сети отделов, сети кампусов и корпоративные сети.

1.5.1. Сети отделов

Сети отделов – это сети, которые используются сравнительно небольшой группой сотрудников, работающих в одном отделе предприятия. Эти сотрудники решают некоторые общие задачи, например ведут бухгалтерский учет или занимаются маркетингом. Считается, что отдел может насчитывать до 100-150 сотрудников.

Главной целью сети отдела является разделение локальных ресурсов, таких как приложения, данные, лазерные принтеры и модемы. Обычно сети отделов имеют один или два файловых сервера и не более тридцати пользователей (рис. 1.31). Сети отделов обычно не разделяются на подсети. В этих сетях локализуется большая часть трафика предприятия. Сети отделов обычно создаются на основе какой-либо одной сетевой технологии – Ethernet, Token Ring. Для такой сети характерен один или, максимум, два типа операционных систем. Чаще всего – это сеть с выделенным сервером, например NetWare, хотя небольшое количество пользователей делает возможным использование одноранговых сетевых ОС, таких, например, как Windows 95.

Задачи управления сетью на уровне отдела относительно просты: добавление новых пользователей, устранение простых отказов, инсталляция новых узлов и установка новых версий программного обеспечения. Такой сетью может управлять сотрудник, посвящающий обязанностям администратора только часть своего времени. Чаще всего администратор сети отдела не имеет специальной подготовки, но является тем человеком в отделе, который лучше всех разбирается в компьютерах, и само собой получается так, что он занимается администрированием сети.

Существует и другой тип сетей, близкий к сетям отделов, – *сети рабочих групп*. К таким сетям относят совсем небольшие сети, включающие до 10-20 компьютеров. Характеристики сетей рабочих групп практически не отличаются от описанных выше характеристик сетей отделов. Такие свойства, как простота сети и однородность, здесь проявляются в наибольшей степени, в то время как сети отделов могут приближаться в некоторых случаях к следующему по масштабу типу сетей – сетям кампусов.

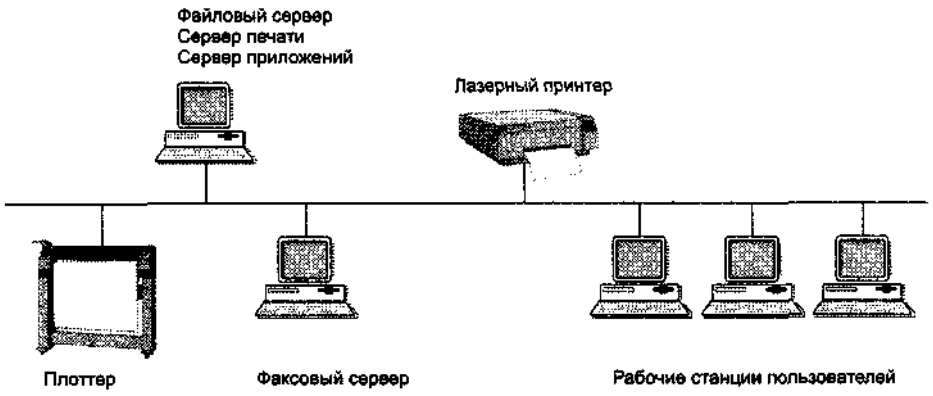


Рис. 1.31. Пример сети масштаба отдела

1.5.2. Сети кампусов

Сети кампусов получили свое название от английского слова campus – студенческий городок. Именно на территории университетских городков часто возникала необходимость объединения нескольких мелких сетей в одну большую сеть. Сейчас это название не связывают со студенческими городками, а используют для обозначения сетей любых предприятий и организаций.

Главными особенностями сетей кампусов являются следующие (рис. 1.32). Сети этого типа объединяют множество сетей различных отделов одного предприятия в пределах отдельного здания или в пределах одной территории, покрывающей площадь в несколько квадратных километров. При этом глобальные соединения в сетях кампусов не используются. Службы такой сети включают взаимодействие между сетями отделов, доступ к общим базам данных предприятия, доступ к общим факс-серверам, высокоскоростным модемам и высокоскоростным принтерам. В результате сотрудники каждого отдела предприятия получают доступ к некоторым файлам и ресурсам сетей других отделов. Важной службой, предоставляемой сетями кампусов, стал доступ к корпоративным базам данных независимо от того, на каких типах компьютеров они располагаются.

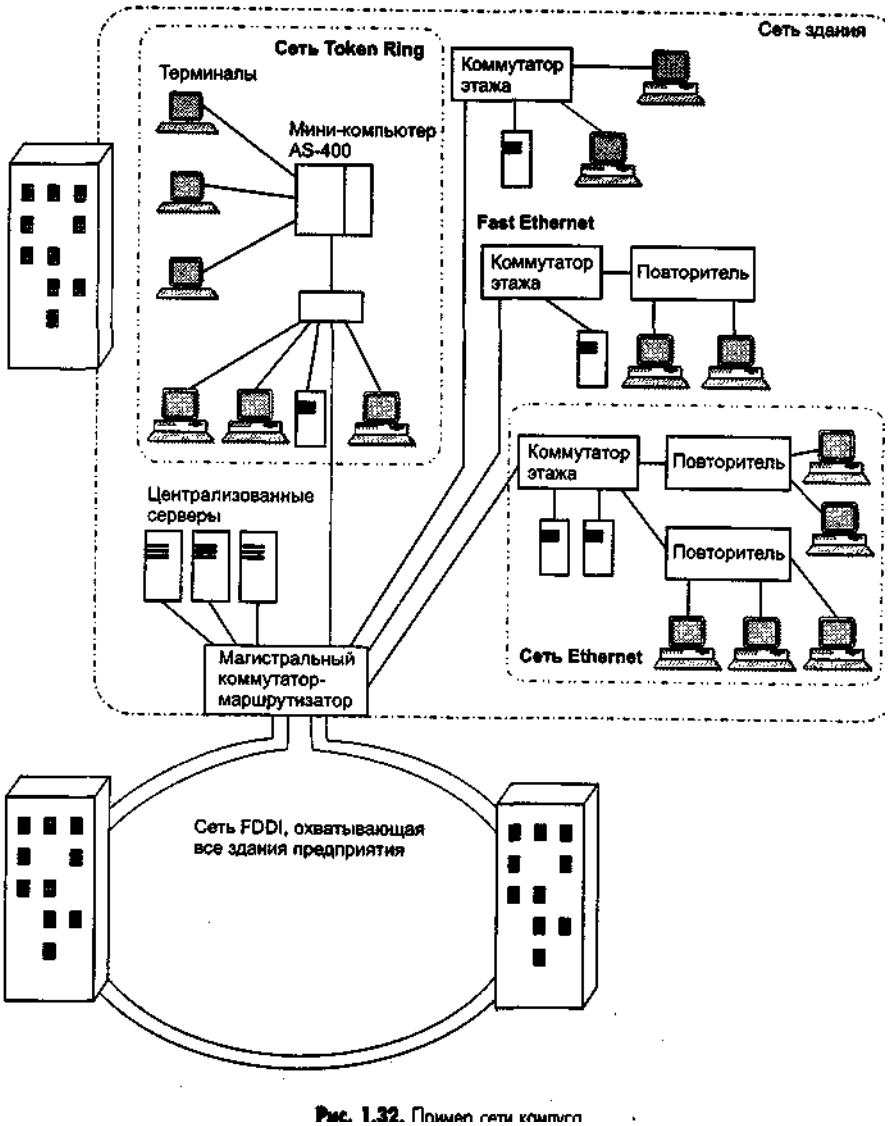


Рис. 1.32. Пример сети кампуса

Имение на уровне сети кампуса возникают проблемы интеграции неоднородного аппаратного и программного обеспечения. Типы компьютеров, сетевых операционных систем, сетевого аппаратного обеспечения могут отличаться в каждом отделе. Отсюда вытекают сложности управления сетями кампусов. Администраторы должны быть в этом случае более квалифицированными, а средства оперативного управления сетью – более совершенными.

1.5.3. Корпоративные сети

Корпоративные сети называют также сетями масштаба предприятия, что соответствует дословному переводу термина «enterprise-wide networks», используемого в англоязычной литературе для обозначения этого типа сетей. Сети масштаба предприятия (корпоративные сети) объединяют большое количество компьютеров на всех территориях отдельного предприятия. Они могут быть сложно связаны и покрывать город, регион или даже континент. Число пользователей и компьютеров может измеряться тысячами, а число серверов – сотнями, расстояния между сетями отдельных территорий могут оказаться такими, что становится необходимым использование глобальных связей (рис. 1.33). Для соединения удаленных локальных сетей и отдельных компьютеров в корпоративной сети применяются разнообразные телекоммуникационные средства, в том числе телефонные каналы, радиоканалы, спутниковая связь. Корпоративную сеть можно представить в виде «островков локальных сетей», плавающих в телекоммуникационной среде.

Непременным атрибутом такой сложной и крупномасштабной сети является высокая степень гетерогенности – нельзя удовлетворить потребности тысяч пользователей с помощью однотипных программных и аппаратных средств. В корпоративной сети обязательно будут использоваться различные типы компьютеров – от мэйнфреймов до персоналок, несколько типов операционных систем и множество различных приложений. Неоднородные части корпоративной сети должны работать как единое целое, предоставляя пользователям по возможности прозрачный доступ ко всем необходимым ресурсам.

Появление корпоративных сетей – это хорошая иллюстрация известного философского постулата о переходе количества в качество. При объединении отдельных сетей крупного предприятия, имеющего филиалы в разных городах и даже странах, в единую сеть многие количественные характеристики объединенной сети превосходят некоторый критический порог, за которым начинается новое качество. В этих условиях существующие методы и подходы к решению традиционных задач сетей меньших масштабов для корпоративных сетей оказались непригодными. На первый план вышли такие задачи и проблемы, которые в сетях рабочих групп, отделов и даже кампусов либо имели второстепенное значение, либо вообще не проявлялись. Примером может служить простейшая (для небольших сетей) задача – ведение учетных данных о пользователях сети.

Наиболее простой способ ее решения – помещение учетных данных каждого пользователя в локальную базу учетных данных каждого компьютера, к ресурсам которого пользователь должен иметь доступ. При попытке доступа эти данные извлекаются из локальной учетной базы и на их основе доступ предоставляется или не предоставляется. Для небольшой сети, состоящей из 5-10 компьютеров и примерно такого же количества пользователей такой способ работает очень хорошо. Но если в сети насчитывается несколько тысяч пользователей, каждому из которых нужен доступ к нескольким десяткам серверов, то, очевидно, это решение становится крайне неэффективным. Администратор должен повторить несколько десятков раз операцию занесения учетных данных пользователя. Сам пользователь также вынужден повторять процедуру логического входа каждый раз, когда ему нужен доступ к ресурсам нового сервера. Хорошее решение этой проблемы для крупной сети – использование централизованной справочной службы, в базе данной которой хранятся учетные записи всех пользователей сети. Администратор один раз выполняет операцию занесения данных пользователя в эту базу, а пользователь один раз выполняет процедуру логического входа, причем не в отдельный сервер, а в сеть целиком.

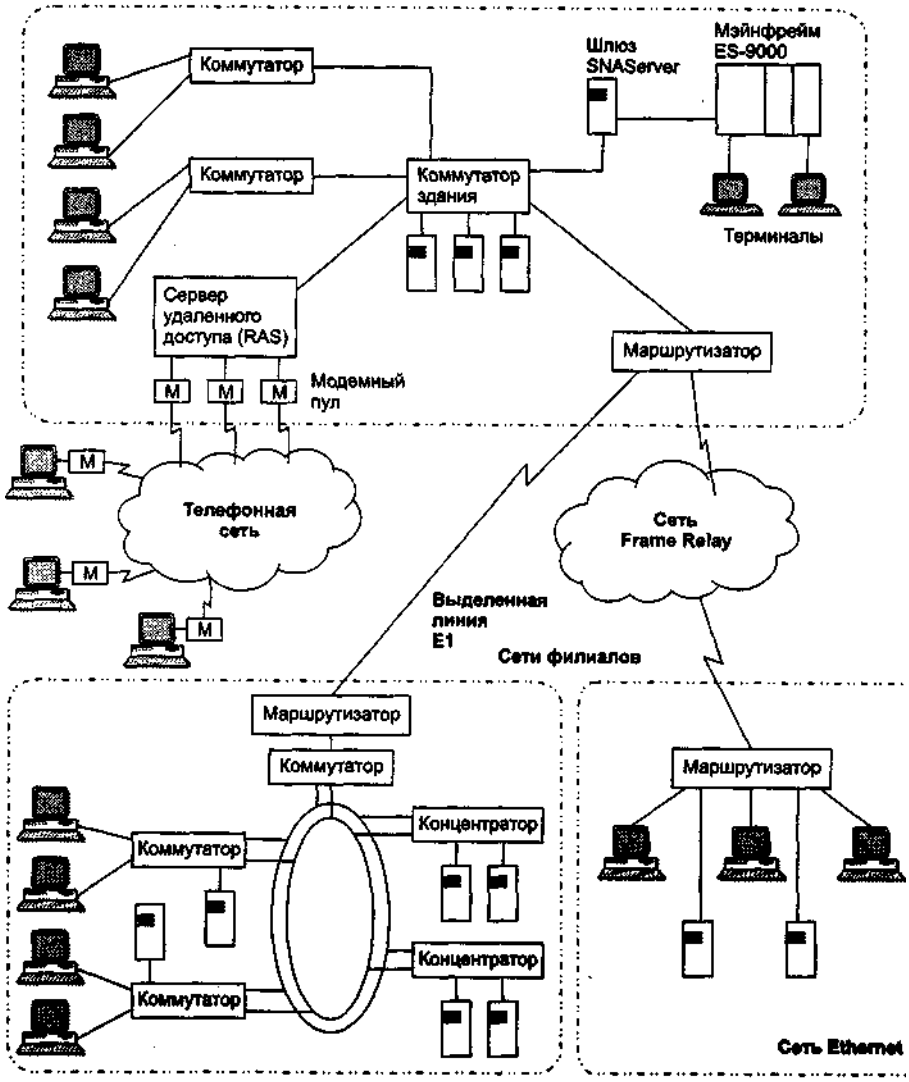


Рис. 1.33. Пример корпоративной сети

При переходе от более простого типа сетей к более сложному – от сетей отдела к корпоративной сети – сеть должна быть все более надежной и отказоустойчивой, при этом требования к ее производительности также существенно возрастают. По мере увеличения масштабов сети увеличиваются и ее функциональные возможности. По сети циркулирует все возрастающее количество данных, и сеть должна обеспечивать их безопасность и защищенность наряду с доступностью. Соединения, обеспечивающие взаимодействие, должны быть более прозрачными. При каждом переходе на следующий уровень сложности компьютерное оборудование сети становится все более разнообразным, а географические расстояния увеличиваются, делая достижение целей более сложным; более проблемным и дорогостоящим становится управление такими соединениями.

Выводы

- В зависимости от масштаба производственного подразделения, в пределах которого действует сеть, различают сети отделов, сети кампусов и корпоративные сети.
- *Сети отделов* используются небольшой группой сотрудников в основном с целью разделения дорогостоящих периферийных устройств, приложений и данных; имеют один-два файловых сервера и не более тридцати пользователей; обычно не разделяются на подсети; создаются на основе какой-либо одной сетевой технологии; могут работать на базе одноранговых сетевых ОС.
- *Сети кампусов* объединяют сети отделов в пределах отдельного здания или одной территории площадью в несколько квадратных километров, при этом глобальные соединения не используются. На уровне сети кампуса возникают проблемы интеграции и управления неоднородным аппаратным и программным обеспечением.

- *Корпоративные сети* объединяют большое количество компьютеров на всех территориях отдельного предприятия. Для корпоративной сети характерны:
- масштабность – тысячи пользовательских компьютеров, сотни серверов, огромные объемы хранимых и передаваемых по линиям связи данных, множество разнообразных приложений;
- высокая степень гетерогенности – типы компьютеров, коммуникационного оборудования, операционных систем и приложений различны;
- использование глобальных связей – сети филиалов соединяются с помощью телекоммуникационных средств, в том числе телефонных каналов, радиоканалов, спутниковой связи.

1.6. Требования, предъявляемые к современным вычислительным сетям

Главным требованием, предъявляемым к сетям, является выполнение сетью ее основной функции - обеспечение пользователям потенциальной возможности доступа к разделяемым ресурсам всех компьютеров, объединенных в сеть. Все остальные требования – производительность, надежность, совместимость, управляемость, защищенность, расширяемость и масштабируемость – связаны с качеством выполнения этой основной задачи.

Хотя все эти требования весьма важны, часто понятие «качество обслуживания» (Quality of Service, QoS) компьютерной сети трактуется более узко – в него включаются только две самые важные характеристики сети – производительность и надежность.

Независимо от выбранного показателя качества обслуживания сети существуют два подхода к его обеспечению. Первый подход, очевидно, покажется наиболее естественным с точки зрения пользователя сети. Он состоит в том, что сеть (точнее, обслуживающий ее персонал) гарантирует пользователю соблюдение некоторой числовой величины показателя качества обслуживания. Например, сеть может гарантировать пользователю А, что любой из его пакетов, посланных пользователю В, будет задержан сетью не более, чем на 150 мс. Или, что средняя пропускная способность канала между пользователями А и В не будет ниже 5 Мбит/с, при этом канал будет разрешать пульсации трафика в 10 Мбит на интервалах времени не более 2 секунд. Технологии frame relay и ATM позволяют строить сети, гарантирующие качество обслуживания по производительности.

Второй подход состоит в том, что сеть обслуживает пользователей в соответствии с их приоритетами. То есть качество обслуживания зависит от степени привилегированности пользователя или группы пользователей, к которой он принадлежит. Качество обслуживания в этом случае не гарантируется, а гарантируется только уровень привилегий пользователя. Такое обслуживание называется обслуживанием *best effort* – с наибольшим старанием. Сеть старается по возможности более качественно обслужить пользователя, но ничего при этом не гарантирует. По такому принципу работают, например, локальные сети, построенные на коммутаторах с приоритезацией кадров.

1.6.1. Производительность

Потенциально высокая производительность – это одно из основных свойств распределенных систем, к которым относятся компьютерные сети. Это свойство обеспечивается возможностью распараллеливания работ между несколькими компьютерами сети. К сожалению, эту возможность не всегда удается реализовать.

Существует несколько основных характеристик производительности сети:

- время реакции;
- пропускная способность;
- задержка передачи и вариация задержки передачи.

Время реакции сети является интегральной характеристикой производительности сети с точки зрения пользователя. Именно эту характеристику имеет в виду пользователь, когда говорит: «Сегодня сеть работает медленно».

В общем случае время реакции определяется как интервал времени между возникновением запроса пользователя к какой-либо сетевой службе и получением ответа на этот запрос.

Очевидно, что значение этого показателя зависит от типа службы, к которой обращается пользователь, от того, какой пользователь и к какому серверу обращается, а также от текущего состояния элементов сети – загруженности сегментов, коммутаторов и маршрутизаторов, через которые

проходит запрос, загруженности сервера и т. п.

Поэтому имеет смысл использовать также и средневзвешенную оценку времени реакции сети, усредняя этот показатель по пользователям, серверам и времени дня (от которого в значительной степени зависит загрузка сети).

Время реакции сети обычно складывается из нескольких составляющих. В общем случае в него входит время подготовки запросов на клиентском компьютере, время передачи запросов между клиентом и сервером через сегменты сети и промежуточное коммуникационное оборудование, время обработки запросов на сервере, время передачи ответов от сервера клиенту и время обработки получаемых от сервера ответов на клиентском компьютере.

Ясно, что пользователя разложение времени реакции на составляющие не интересует – ему важен конечный результат, однако для сетевого специалиста очень важно выделить из общего времени реакции составляющие, соответствующие этапам собственно сетевой обработки данных, – передачу данных от клиента к серверу через сегменты сети и коммуникационное оборудование.

Знание сетевых составляющих времени реакции дает возможность оценить производительность отдельных элементов сети, выявить узкие места и в случае необходимости выполнить модернизацию сети для повышения ее общей производительности.

Пропускная способность отражает объем данных, переданных сетью или ее частью в единицу времени. Пропускная способность уже не является пользовательской характеристикой, так как она говорит о скорости выполнения внутренних операций сети – передачи пакетов данных между узлами сети через различные коммуникационные устройства. Зато она непосредственно характеризует качество выполнения основной функции сети – транспортировки сообщений – и поэтому чаще используется при анализе производительности сети, чем время реакции.

Пропускная способность измеряется либо в битах в секунду, либо в пакетах в секунду. Пропускная способность может быть мгновенной, максимальной и средней.

Средняя пропускная способность вычисляется путем деления общего объема переданных данных на время их передачи, причем выбирается достаточно длительный промежуток времени – час, день или неделя.

Мгновенная пропускная способность отличается от средней тем, что для усреднения выбирается очень маленький промежуток времени – например, 10 мс или 1 с.

Максимальная пропускная способность – это наибольшая мгновенная пропускная способность, зафиксированная в течение периода наблюдения.

Чаще всего при проектировании, настройке и оптимизации сети используются такие показатели, как средняя и максимальная пропускные способности. Средняя пропускная способность отдельного элемента или всей сети позволяет оценить работу сети на большом промежутке времени, в течение которого в силу закона больших чисел пики и спады интенсивности трафика компенсируют друг друга. Максимальная пропускная способность позволяет оценить возможности сети справляться с пиковыми нагрузками, характерными для особых периодов работы сети, например утренних часов, когда сотрудники предприятия почти одновременно регистрируются в сети и обращаются к разделяемым файлам и базам данных.

Пропускную способность можно измерять между любыми двумя узлами или точками сети, например между клиентским компьютером и сервером, между входным и выходным портами маршрутизатора. Для анализа и настройки сети очень полезно знать данные о пропускной способности отдельных элементов сети.

Важно отметить, что из-за последовательного характера передачи пакетов различными элементами сети общая пропускная способность сети любого составного пути в сети будет равна *минимальной* из пропускных способностей составляющих элементов маршрута. Для повышения пропускной способности составного пути необходимо в первую очередь обратить внимание на самые медленные элементы – в данном случае таким элементом, скорее всего, будет маршрутизатор. Следует подчеркнуть, что если передаваемый по составному пути трафик будет иметь среднюю интенсивность, превосходящую среднюю пропускную способность самого медленного элемента пути, то очередь пакетов к этому элементу будет расти теоретически до бесконечности, а практически – до тех пор, пока не заполниться его буферная память, а затем пакеты просто начнут отбрасываться и теряться.

Иногда полезно оперировать с *общей пропускной способностью сети*, которая определяется как среднее количество информации, переданной между всеми узлами сети в единицу времени. Этот показатель характеризует качество сети в целом, не дифференцируя его по отдельным сегментам или

устройствам.

Обычно при определении пропускной способности сегмента или устройства в передаваемых данных не выделяются пакеты какого-то определенного пользователя, приложения или компьютера – подсчитывается общий объем передаваемой информации. Тем не менее для более точной оценки качества обслуживания такая детализации желательна, и в последнее время системы управления сетями все чаще позволяют ее выполнять.

Задержка передачи определяется как задержка между моментом поступления пакета на вход какого-либо сетевого устройства или части сети и моментом появления его на выходе этого устройства. Этот параметр производительности по смыслу близок ко времени реакции сети, но отличается тем, что всегда характеризует только сетевые этапы обработки данных, без задержек обработки компьютерами сети. Обычно качество сети характеризуют величинами *максимальной задержки передачи* и *вариацией задержки*. Не все типы трафика чувствительны к задержкам передачи, во всяком случае, к тем величинам задержек, которые характерны для компьютерных сетей, – обычно задержки не превышают сотен миллисекунд, реже – нескольких секунд. Такого порядка задержки пакетов, порождаемых файловой службой, службой электронной почты или службой печати, мало влияют на качество этих служб с точки зрения пользователя сети. С другой стороны, такие же задержки пакетов, переносящих голосовые данные или видеоизображение, могут приводить к значительному снижению качества предоставляемой пользователю информации – возникновению эффекта «эха», невозможности разобрать некоторые слова, дрожание изображения и т. п.

Пропускная способность и задержки передачи являются независимыми параметрами, так что сеть может обладать, например, высокой пропускной способностью, но вносить значительные задержки при передаче каждого пакета Пример такой ситуации дает канал связи, образованный геостационарным спутником. Пропускная способность этого, канала может быть весьма высокой, например 2 Мбит/с, в то время как задержка передачи всегда составляет не менее 0,24 с, что определяется скоростью распространения сигнала (около 300 000 км/с) и длиной канала (72 000 км).

1.6.2. Надежность и безопасность

Одной из первоначальных целей создания распределенных систем, к которым относятся и вычислительные сети, являлось достижение большей надежности по сравнению с отдельными вычислительными машинами.

Важно различать несколько аспектов надежности. Для технических устройств используются такие показатели надежности, как среднее время наработки на отказ, вероятность отказа, интенсивность отказов. Однако эти показатели пригодны для оценки надежности простых элементов и устройств, которые могут находиться только в двух состояниях – работоспособном или неработоспособном. Сложные системы, состоящие из многих элементов, кроме состояний работоспособности и неработоспособности, могут иметь и другие промежуточные состояния, которые эти характеристики не учитывают. В связи с этим для оценки надежности сложных систем применяется другой набор характеристик.

Готовность или коэффициент готовности (*availability*) означает долю времени, в течение которого система может быть использована. Готовность может быть улучшена путем введения избыточности в структуру системы: ключевые элементы системы должны существовать в нескольких экземплярах, чтобы при отказе одного из них функционирование системы обеспечивали другие.

Чтобы систему можно было отнести к высоконадежным, она должна как минимум обладать высокой готовностью, но этого недостаточно. Необходимо обеспечить *сохранность данных* и защиту их от искажений. Кроме этого, должна поддерживаться *согласованность* (непротиворечивость) данных, например, если для повышения надежности на нескольких файловых серверах хранится несколько копий данных, то нужно постоянно обеспечивать их идентичность.

Так как сеть работает на основе механизма передачи пакетов между конечными узлами, то одной из характерных характеристик надежности является *вероятность доставки пакета* узлу назначения без искажений. Наряду с этой характеристикой могут использоваться и другие показатели: вероятность потери пакета (по любой из причин – из-за переполнения буфера маршрутизатора, из-за несовпадения контрольной суммы, из-за отсутствия работоспособного пути к узлу назначения и т. д.), вероятность искажения отдельного бита передаваемых данных, отношение потерянных пакетов к доставленным.

Другим аспектом общей надежности является *безопасность* (*security*), то есть способность системы

защитить данные от несанкционированного доступа. В распределенной системе это сделать гораздо сложнее, чем в централизованной. В сетях сообщения передаются по линиям связи, часто проходящим через общедоступные помещения, в которых могут быть установлены средства прослушивания линий. Другим уязвимым местом могут быть оставленные без присмотра персональные компьютеры. Кроме того, всегда имеется потенциальная угроза взлома защиты сети от неавторизованных пользователей, если сеть имеет выходы в глобальные сети общего пользования.

Еще одной характеристикой надежности является *отказоустойчивость* (*fault tolerance*). В сетях под отказоустойчивостью понимается способность системы скрыть от пользователя отказ отдельных ее элементов. Например, если копии таблицы базы данных хранятся одновременно на нескольких файловых серверах, то пользователи могут просто не заметить отказ одного из них. В отказоустойчивой системе отказ одного из ее элементов приводит к некоторому снижению качества ее работы (деградации), а не к полному останову. Так, при отказе одного из файловых серверов в предыдущем примере увеличивается только время доступа к базе данных из-за уменьшения степени распараллеливания запросов, но в целом система будет продолжать выполнять свои функции.

1.6.3. Расширяемость и масштабируемость

Термины расширяемость и масштабируемость иногда используют как синонимы, но это неверно – каждый из них имеет четко определенное самостоятельное значение.

Расширяемость (*extensibility*) означает возможность сравнительно легкого добавления отдельных элементов сети (пользователей, компьютеров, приложений, служб), наращивания длины сегментов сети и замены существующей аппаратуры более мощной. При этом принципиально важно, что легкость расширения системы иногда может обеспечиваться в некоторых весьма ограниченных пределах. Например, локальная сеть Ethernet, построенная на основе одного сегмента толстого коаксиального кабеля, обладает хорошей расширяемостью, в том смысле, что позволяет легко подключать новые станции. Однако такая сеть имеет ограничение на число станций – их число не должно превышать 30-40. Хотя сеть допускает физическое подключение к сегменту и большего числа станций (до 100), но при этом чаще всего резко снижается производительность сети. Наличие такого ограничения и является признаком плохой масштабируемости системы при хорошей расширяемости.

Масштабируемость (*scalability*) означает, что сеть позволяет наращивать количество узлов и протяженность связей в очень широких пределах, при этом производительность сети не ухудшается. Для обеспечения масштабируемости сети приходится применять дополнительное коммуникационное оборудование и специальным образом структурировать сеть. Например, хорошей масштабируемостью обладает многосегментная сеть, построенная с использованием коммутаторов и маршрутизаторов и имеющая иерархическую структуру связей. Такая сеть может включать несколько тысяч компьютеров и при этом обеспечивать каждому пользователю сети нужное качество обслуживания.

1.6.4. Прозрачность

Прозрачность (*transparency*) сети достигается в том случае, когда сеть представляется пользователям не как множество отдельных компьютеров, связанных между собой сложной системой кабелей, а как единая традиционная вычислительная машина с системой разделения времени. Известный лозунг компании Sun Microsystems: «Сеть – это компьютер» – говорит именно о такой прозрачной сети.

Прозрачность может быть достигнута на двух различных уровнях – на уровне пользователя и на уровне программиста. На уровне пользователя прозрачность означает, что для работы с удаленными ресурсами он использует те же команды и привычные ему процедуры, что и для работы с локальными ресурсами. На программном уровне прозрачность заключается в том, что приложению для доступа к удаленным ресурсам требуются те же вызовы, что и для доступа к локальным ресурсам. Прозрачность на уровне пользователя достигается проще, так как все особенности процедур, связанные с распределенным характером системы, маскируются от пользователя программистом, который создает приложение. Прозрачность на уровне приложения требует сокрытия всех деталей распределенности средствами сетевой операционной системы.

Сеть должна скрывать все особенности операционных систем и различия в типах компьютеров. Пользователь компьютера Macintosh должен иметь возможность обращаться к ресурсам, поддерживаемым UNIX-системой, а пользователь UNIX должен иметь возможность разделять

информацию с пользователями Windows 95. Подавляющее число пользователей ничего не хочет знать о внутренних форматах файлов или о синтаксисе команд UNIX. Пользователь терминала IBM 3270 должен иметь возможность обмениваться сообщениями с пользователями сети персональных компьютеров без необходимости вникать в секреты трудно запоминаемых адресов.

Концепция прозрачности может быть применена к различным аспектам сети. Например, прозрачность расположения означает, что от пользователя не требуется знаний о месте расположения программных и аппаратных ресурсов, таких как процессоры, принтеры, файлы и базы данных. Имя ресурса не должно включать информацию о месте его расположения, поэтому имена типа mashinel:prog.c или \\ftp_serv\pub прозрачными не являются. Аналогично, прозрачность перемещения означает, что ресурсы должны свободно перемещаться из одного компьютера в другой без изменения своих имен. Еще одним из возможных аспектов прозрачности является прозрачность параллелизма, заключающаяся в том, что процесс распараллеливания вычислений происходит автоматически, без участия программиста, при этом система сама распределяет параллельные ветви приложения по процессорам и компьютерам сети. В настоящее время нельзя сказать, что свойство прозрачности в полной мере присуще многим вычислительным сетям, это скорее цель, к которой стремятся разработчики современных сетей.

1.6.5. Поддержка разных видов трафика

Компьютерные сети изначально предназначены для совместного доступа пользователя к ресурсам компьютеров: файлам, принтерам и т. п. Трафик, создаваемый этими традиционными службами компьютерных сетей, имеет свои особенности и существенно отличается от трафика сообщений в телефонных сетях или, например, в сетях кабельного телевидения. Однако 90-е годы стали годами проникновения в компьютерные сети трафика мультимедийных данных, представляющих в цифровой форме речь и видеоизображение. Компьютерные сети стали использоваться для организации видеоконференций, обучения и развлечения на основе видеофильмов и т. п. Естественно, что для динамической передачи мультимедийного трафика требуются иные алгоритмы и протоколы и, соответственно, другое оборудование. Хотя доля мультимедийного трафика пока невелика, он уже начал свое проникновение как в глобальные, так и локальные сети, и этот процесс, очевидно, будет продолжаться с возрастающей скоростью.

Главной особенностью трафика, образующегося при динамической передаче голоса или изображения, является наличие жестких требований к синхронности передаваемых сообщений. Для качественного воспроизведения непрерывных процессов, которыми являются звуковые колебания или изменения интенсивности света в видеоизображении, необходимо получение измеренных и закодированных амплитуд сигналов с той же частотой, с которой они были измерены на передающей стороне. При запаздывании сообщений будут наблюдаться искажения.

В то же время трафик компьютерных данных характеризуется крайне неравномерной интенсивностью поступления сообщений в сеть при отсутствии жестких требований к синхронности доставки этих сообщений. Например, доступ пользователя, работающего с текстом на удаленном диске, порождает случайный поток сообщений между удаленным и локальным компьютерами, зависящий от действий пользователя по редактированию текста, причем задержки при доставке в определенных (и достаточно широких с компьютерной точки зрения) пределах мало влияют на качество обслуживания пользователя сети. Все алгоритмы компьютерной связи, соответствующие протоколы и коммуникационное оборудование были рассчитаны именно на такой «пульсирующий» характер трафика, поэтому необходимость передавать мультимедийный трафик требует внесения принципиальных изменений как в протоколы, так и оборудование. Сегодня практически все новые протоколы в той или иной степени предоставляют поддержку мультимедийного трафика.

Особую сложность представляет *совмещение* в одной сети традиционного *компьютерного* и *мультимедийного трафика*. Передача исключительно мультимедийного трафика компьютерной сетью хотя и связана с определенными сложностями, но вызывает меньшие трудности. А вот случай сосуществования двух типов трафика с противоположными требованиями к качеству обслуживания является намного более сложной задачей. Обычно протоколы и оборудование компьютерных сетей относят мультимедийный трафик к факультативному, поэтому качество его обслуживания оставляет желать лучшего. Сегодня затрачиваются большие усилия по созданию сетей, которые не ущемляют интересы одного из типов трафика. Наиболее близки к этой цели сети на основе технологии ATM,

разработчики которой изначально учитывали случай сосуществования разных типов трафика в одной сети.

1.6.6. Управляемость

Управляемость сети подразумевает возможность централизованно контролировать состояние основных элементов сети, выявлять и разрешать проблемы, возникающие при работе сети, выполнять анализ производительности и планировать развитие сети. В идеале средства управления сетями представляют собой систему, осуществляющую наблюдение, контроль и управление каждым элементом сети – от простейших до самых сложных устройств, при этом такая система рассматривает сеть как единое целое, а не как разрозненный набор отдельных устройств.

Хорошая система управления наблюдает за сетью и, обнаружив проблему, активизирует определенное действие, исправляет ситуацию и уведомляет администратора о том, что произошло и какие шаги предприняты. Одновременно с этим система управления должна накапливать данные, на основании которых можно планировать развитие сети. Наконец, система управления должна быть независима от производителя и обладать удобным интерфейсом, позволяющим выполнять все действия с одной консоли.

Решая тактические задачи, администраторы и технический персонал сталкиваются с ежедневными проблемами обеспечения работоспособности сети. Эти задачи требуют быстрого решения, обслуживающий сеть персонал должен оперативно реагировать на сообщения о неисправностях, поступающих от пользователей или автоматических средств управления сетью. Постепенно становятся заметны более общие проблемы производительности, конфигурирования сети, обработки сбоев и безопасности данных, требующие стратегического подхода, то есть *планирования* сети. Планирование, кроме этого, включает прогноз изменений требований пользователей к сети, вопросы применения новых приложений, новых сетевых технологий и т. п.

Полезность системы управления особенно ярко проявляется в больших сетях: корпоративных или публичных глобальных. Без системы управления в таких сетях нужно присутствие квалифицированных специалистов по эксплуатации в каждом здании каждого города, где установлено оборудование сети, что в итоге приводит к необходимости содержания огромного штата обслуживающего персонала.

В настоящее время в области систем управления сетями много нерешенных проблем. Явно недостаточно действительно удобных, компактных и многопротокольных средств управления сетью. Большинство существующих средств вовсе не управляют сетью, а всего лишь осуществляют *наблюдение* за ее работой. Они следят за сетью, но не выполняют активных действий, если с сетью что-то произошло или может произойти. Мало масштабируемых систем, способных обслуживать как сети масштаба отдела, так и сети масштаба предприятия, – очень многие системы управляют только отдельными элементами сети и не анализируют способность сети выполнять качественную передачу данных между конечными пользователями сети.

1.6.7. Совместимость

Совместимость или *интегрируемость* означает, что сеть способна включать в себя самое разнообразное программное и аппаратное обеспечение, то есть в ней могут сосуществовать различные операционные системы, поддерживающие разные стеки коммуникационных протоколов, и работать аппаратные средства и приложения от разных производителей. Сеть, состоящая из разнотипных элементов, называется неоднородной или гетерогенной, а если гетерогенная сеть работает без проблем, то она является интегрированной. Основной путь построения интегрированных сетей – использование модулей, выполненных в соответствии с открытыми стандартами и спецификациями.

Выводы

- Качество работы сети характеризуют следующие свойства: производительность, надежность, совместимость, управляемость, защищенность, расширяемость и масштабируемость.
- Существуют два основных подхода к обеспечению качества работы сети. Первый – состоит в том, что сеть гарантирует пользователю соблюдение некоторой числовой величины показателя качества обслуживания. Например, сети frame relay и ATM могут гарантировать

пользователю заданный уровень пропускной способности. При втором подходе (best effort) сеть старается по возможности более качественно обслужить пользователя, но ничего при этом не гарантирует.

- К основным характеристикам производительности сети относятся: *время реакции*, которое определяется как время между возникновением запроса к какому-либо сетевому сервису и получением ответа на него; *пропускная способность*, которая отражает объем данных, переданных сетью в единицу времени, и *задержка передачи*, которая равна интервалу между моментом поступления пакета на вход какого-либо сетевого устройства и моментом его появления на выходе этого устройства.
- Для оценки надежности сетей используются различные характеристики, в том числе: *коэффициент готовности*, означающий долю времени, в течение которого система может быть использована; *безопасность*, то есть способность системы защитить данные от несанкционированного доступа; *отказоустойчивость* – способность системы работать в условиях отказа некоторых ее элементов.
- *Расширяемость* означает возможность сравнительно легкого добавления отдельных элементов сети (пользователей, компьютеров, приложений, сервисов), наращивания длины сегментов сети и замены существующей аппаратуры более мощной.
- *Масштабируемость* означает, что сеть позволяет наращивать количество узлов и протяженность связей в очень широких пределах, при этом производительность сети не ухудшается.
- *Прозрачность* – свойство сети скрывать от пользователя детали своего внутреннего устройства, упрощая тем самым его работу в сети.
- *Управляемость* сети подразумевает возможность централизованно контролировать состояние основных элементов сети, выявлять и разрешать проблемы, возникающие при работе сети, выполнять анализ производительности и планировать развитие сети.
- *Совместимость* означает, что сеть способна включать в себя самое разнообразное программное и аппаратное обеспечение.

ВОПРОСЫ И УПРАЖНЕНИЯ

1. Чем можно объяснить тот факт, что глобальные сети появились раньше, чем локальные?
2. Поясните использование термина «сеть» в следующих предложениях:
- *сеть* нашего предприятия включает *сеть* Ethernet и *сеть* Token Ring;
- маршрутизатор – это устройство, которое соединяет *сети*;
- чтобы получить выход в Internet, необходимо получить у поставщика услуг Internet номер сети;
- в последнее время IP-*сети* становятся все более распространенными;
- гетерогенность корпоративной *сети* приводит к тому, что на первый план часто выходит проблема согласования *сетей*.
3. Всякое ли приложение, выполняемое в сети, можно назвать сетевым?
4. Что общего и в чем отличие между взаимодействием компьютеров в сети и взаимодействием компьютера с периферийным устройством?
5. Как распределяются функции между сетевым адаптером и его драйвером?
6. Поясните значения терминов «клиент», «сервер», «редиректор».
7. Назовите главные недостатки полносвязной топологии, а также топологий типа общая шина, звезда, кольцо.
8. Какую топологию имеет односегментная сеть Ethernet, построенная на основе концентратора: общая шина или звезда?
9. Какие из следующих утверждений верны:
- (A) разделение линий связи приводит к повышению пропускной способности канала;
- (B) конфигурация физических связей может совпадать с конфигурацией логических связей;
- (C) главной задачей службы разрешения имен является проверка сетевых имен и адресов на допустимость;
- (D) протоколы без установления соединений называются также дейтаграммными протоколами.

10. Определите функциональное назначение основных типов коммуникационного оборудования - повторителей, концентраторов, мостов, коммутаторов, маршрутизаторов.
11. В чем отличие логической структуризации сети от физической?
12. Если все коммуникационные устройства в приведенном ниже фрагменте сети (рис. 1.34) являются концентраторами, то на каких портах появится кадр, если его отправил компьютер А компьютеру В? Компьютеру С? Компьютеру D?

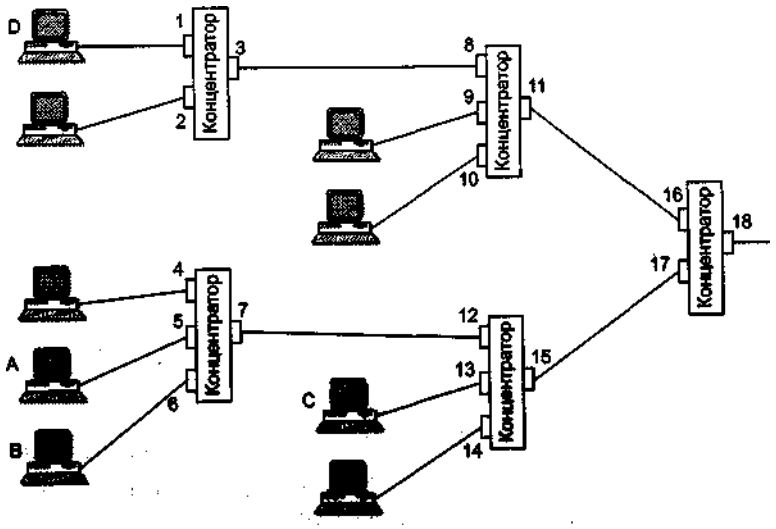


Рис. 1.34. Фрагмент сети

13. Если в предыдущем упражнении изменить условия и считать, что все коммуникационные устройства являются коммутаторами, то на каких портах появится кадр, посланный компьютером А компьютеру В? Компьютеру С? Компьютеру D?
14. Что такое «открытая система»? Приведите примеры закрытых систем.
15. Поясните разницу в употреблении терминов «протокол» и «интерфейс» применительно к многоуровневой модели взаимодействия устройств в сети.
16. Что стандартизует модель OSI?
17. Что стандартизует стек OSI?
18. Почему в модели OSI семь уровней?
19. Дайте краткое описание функций каждого уровня и приведите примеры стандартных протоколов для каждого уровня модели OSI.
20. Являются ли термины «спецификация» и «стандарт» синонимами?
21. Какая организация разработала основные стандарты сетей Ethernet и Token Ring?
22. Из приведенной ниже последовательности названий стандартных стеков коммуникационных протоколов выделите названия, которые относятся к одному и тому же стеку:
    TCP/IP, Microsoft, IPX/SPX, Novell, Internet, DoD, NetBIOS/SMB, DECnet.
23. В чем состоит отличие локальных сетей от глобальных на уровне служб? На уровне транспортной системы?
24. Назовите наиболее часто используемые характеристики производительности сети?
25. Что важнее для передачи мультимедийного трафика: надежность или синхронность?
26. Поясните значение некоторых сетевых характеристик, названия которых помещены в англоязычном написании:

- availability;
- fault tolerance;
- security;
- extensibility;
- scalability;
- transparency.

ГЛАВА 2. ОСНОВЫ ПЕРЕДАЧИ ДИСКРЕТНЫХ ДАННЫХ

Любая сетевая технология должна обеспечить надежную и быструю передачу дискретных данных по линиям связи. И хотя между технологиями имеются большие различия, они базируются на общих принципах передачи дискретных данных, которые рассматриваются в этой главе. Эти принципы находят свое воплощение в методах представления двоичных единиц и нулей с помощью импульсных или синусоидальных сигналов в линиях связи различной физической природы, методах обнаружения и коррекции ошибок, методах компрессии и методах коммутации.

2.1. Линии связи

2.1.1. Типы линий связи

Линия связи (рис. 2.1) состоит в общем случае из физической среды, по которой передаются электрические информационные сигналы, аппаратуры передачи данных и промежуточной аппаратуры. Синонимом термина *линия связи* (*line*) является термин *канал связи* (*channel*).



Рис. 2.1. Состав линии связи

Физическая среда передачи данных (*medium*) может представлять собой кабель, то есть набор проводов, изоляционных и защитных оболочек и соединительных разъемов, а также земную атмосферу или космическое пространство, через которые распространяются электромагнитные волны.

В зависимости от среды передачи данных линии связи разделяются на следующие (рис. 2.2.):

- проводные (воздушные);
- кабельные (медные и волоконно-оптические);
- радиоканалы наземной и спутниковой связи.

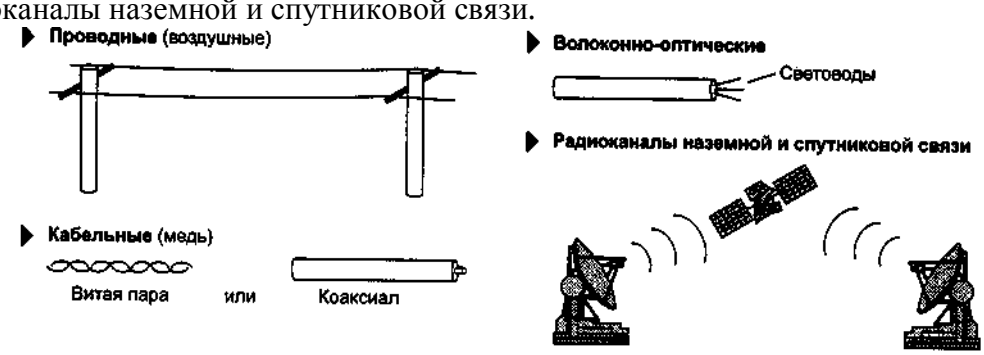


Рис. 2.2. Типы линий связи

Проводные (*воздушные*) *линии связи* представляют собой провода без каких-либо изолирующих или экранирующих оплеток, проложенные между столбами и висящие в воздухе. По таким линиям связи традиционно передаются телефонные или телеграфные сигналы, но при отсутствии других возможностей эти линии используются и для передачи компьютерных данных. Скоростные качества и помехозащищенность этих линий оставляют желать много лучшего. Сегодня проводные линии связи быстро вытесняются кабельными.

Кабельные линии представляют собой достаточно сложную конструкцию. Кабель состоит из проводников, заключенных в несколько слоев изоляции: электрической, электромагнитной, механической, а также, возможно, климатической. Кроме того, кабель может быть оснащен разъемами, позволяющими быстро выполнять присоединение к нему различного оборудования. В компьютерных сетях применяются три основных типа кабеля: кабели на основе скрученных пар медных проводов,

коаксиальные кабели с медной жилой, а также волоконно-оптические кабели.

Скрученная пара проводов называется *витой парой* (*twisted pair*). Витая пара существует в экранированном варианте (*Shielded Twisted Pair, STP*), когда пара медных проводов обертывается в изоляционный экран, и неэкранированном (*Unshielded Twisted Pair, UTP*), когда изоляционная обертка отсутствует. Скручивание проводов снижает влияние внешних помех на полезные сигналы, передаваемые по кабелю. *Коаксиальный кабель* (*coaxial*) имеет несимметричную конструкцию и состоит из внутренней медной жилы и оплетки, отделенной от жилы слоем изоляции. Существует несколько типов коаксиального кабеля, отличающихся характеристиками и областями применения – для локальных сетей, для глобальных сетей, для кабельного телевидения и т. п. *Волоконно-оптический кабель* (*optical fiber*) состоит из тонких (5-60 микрон) волокон, по которым распространяются световые сигналы. Это наиболее качественный тип кабеля – он обеспечивает передачу данных с очень высокой скоростью (до 10 Гбит/с и выше) и к тому же лучше других типов передающей среды обеспечивает защиту данных от внешних помех.

Радиоканалы наземной и спутниковой связи образуются с помощью передатчика и приемника радиоволн. Существует большое количество различных типов радиоканалов, отличающихся как используемым частотным диапазоном, так и дальностью канала. Диапазоны коротких, средних и длинных волн (КВ, СВ и ДВ), называемые также диапазонами амплитудной модуляции (Amplitude Modulation, AM) по типу используемого в них метода модуляции сигнала, обеспечивают дальнюю связь, но при невысокой скорости передачи данных. Более скоростными являются каналы, работающие на диапазонах ультракоротких волн (УКВ), для которых характерна частотная модуляция (Frequency Modulation, FM), а также диапазонах сверхвысоких частот (СВЧ или microwaves). В диапазоне СВЧ (свыше 4 ГГц) сигналы уже не отражаются ионосферой Земли и для устойчивой связи требуется наличие прямой видимости между передатчиком и приемником. Поэтому такие частоты используют либо спутниковые каналы, либо радиорелейные каналы, где это условие выполняется.

В компьютерных сетях сегодня применяются практически все описанные типы физических сред передачи данных, но наиболее перспективными являются волоконно-оптические. На них сегодня строятся как магистрали крупных территориальных сетей, так и высокоскоростные линии связи локальных сетей. Популярной средой является также витая пара, которая характеризуется отличным соотношением качества к стоимости, а также простотой монтажа. С помощью витой пары обычно подключают конечных абонентов сетей на расстояниях до 100 метров от концентратора. Спутниковые каналы и радиосвязь используются чаще всего в тех случаях, когда кабельные связи применить нельзя – например, при прохождении канала через малонаселенную местность или же для связи с мобильным пользователем сети, таким как шофер грузовика, врач, совершающий обход, и т. п.

2.1.2. Аппаратура линий связи

Аппаратура передачи данных (*АПД* или *DCE – Data Circuit terminating Equipment*) непосредственно связывает компьютеры или локальные сети пользователя с линией связи и является, таким образом, пограничным оборудованием. Традиционно аппаратуру передачи данных включают в состав линии связи. Примерами DCE являются модемы, терминальные адаптеры сетей ISDN, оптические модемы, устройства подключения к цифровым каналам. Обычно DCE работает на физическом уровне, отвечая за передачу и прием сигнала нужной формы и мощности в физическую среду.

Аппаратура пользователя линии связи, вырабатывающая данные для передачи по линии связи и подключаемая непосредственно к аппаратуре передачи данных, обобщенно носит название *оконечное оборудование данных* (*ООД* или *DTE – Data Terminal Equipment*). Примером DTE могут служить компьютеры или маршрутизаторы локальных сетей. Эту аппаратуру не включают в состав линии связи.

Разделение оборудования на классы DCE и DTE в локальных сетях является достаточно условным. Например, адаптер локальной сети можно считать как принадлежностью компьютера, то есть DTE, так и составной частью канала связи, то есть DCE.

Промежуточная аппаратура обычно используется на линиях связи большой протяженности. Промежуточная аппаратура решает две основные задачи:

- улучшение качества сигнала;
- создание постоянного составного канала связи между двумя абонентами сети.

В локальных сетях промежуточная аппаратура может совсем не использоваться, если протяженность физической среды – кабелей или радиоэфира – позволяет одному сетевому адаптеру принимать сигналы

непосредственно от другого сетевого адаптера, без промежуточного усиления. В противном случае применяются устройства типа повторителей и концентраторов.

В глобальных сетях необходимо обеспечить качественную передачу сигналов на расстояния в сотни и тысячи километров. Поэтому без усилителей сигналов, установленных через определенные расстояния, построить территориальную линию связи невозможно. В глобальной сети необходима также и промежуточная аппаратура другого рода – мультиплексоры, демультиплексоры и коммутаторы. Эта аппаратура решает вторую указанную задачу, то есть создает между двумя абонентами сети составной канал из некоммутируемых отрезков физической среды – кабелей с усилителями. Важно отметить, что приведенные на рис. 2.1 мультиплексоры, демультиплексоры и коммутаторы образуют составной канал на *долговременной* основе, например на месяц или год, причем абонент не может влиять на процесс коммутации этого канала – эти устройства управляются по отдельным входам, абоненту недоступным (на рисунке не показаны). Наличие промежуточной коммутационной аппаратуры избавляет создателей глобальной сети от необходимости прокладывать отдельную кабельную линию для каждой пары соединяемых узлов сети. Вместо этого между мультиплексорами и коммутаторами используется высокоскоростная физическая среда, например волоконно-оптический или коаксиальный кабель, по которому передаются одновременно данные от большого числа сравнительно низкоскоростных абонентских линий. А когда нужно образовать постоянное соединение между какими-либо двумя конечными узлами сети, находящимися, например, в разных городах, то мультиплексоры, коммутаторы и демультиплексоры настраиваются оператором канала соответствующим образом. Высокоскоростной канал обычно называют уплотненным каналом.

Промежуточная аппаратура канала связи прозрачна для пользователя, он ее не замечает и не учитывает в своей работе. Для него важны только качество полученного канала, влияющее на скорость передачи дискретных данных. В действительности же промежуточная аппаратура образует сложную сеть, которую называют *первичной сетью*, так как сама по себе она никаких высокоуровневых служб (например, файловой или передачи голоса) не поддерживает, а только служит основой для построения компьютерных, телефонных или иных сетей.

В зависимости от типа промежуточной аппаратуры все линии связи делятся на аналоговые и цифровые. В *аналоговых линиях* промежуточная аппаратура предназначена для усиления аналоговых сигналов, то есть сигналов, которые имеют непрерывный диапазон значений. Такие линии связи традиционно применялись в телефонных сетях для связи АТС между собой. Для создания высокоскоростных каналов, которые мультиплексируют несколько низкоскоростных аналоговых абонентских каналов, при аналоговом подходе обычно используется техника частотного мультиплексирования (Frequency Division Multiplexing FDM).

В *цифровых линиях* связи передаваемые сигналы имеют конечное число состояний. Как правило, элементарный сигнал, то есть сигнал, передаваемый за один такт работы передающей аппаратуры, имеет 2 или 3 состояния, которые передаются в линиях связи импульсами прямоугольной формы. С помощью таких сигналов передаются как компьютерные данные, так и оцифрованные речь и изображение. В цифровых каналах связи используется промежуточная аппаратура, которая улучшает форму импульсов и обеспечивает их ресинхронизацию, то есть восстанавливает период их следования. Промежуточная аппаратура образования высокоскоростных цифровых каналов (мультиплексоры, демультиплексоры, коммутаторы) работает по принципу временного мультиплексирования каналов (Time Division Multiplexing, TDM), когда каждому низкоскоростному каналу выделяется определенная доля времени (тайм-слот или квант) высокоскоростного канала.

Аппаратура передачи дискретных компьютерных данных по аналоговым и цифровым линиям связи существенно отличается, так как в первом случае линия связи предназначена для передачи сигналов произвольной формы и не предъявляет никаких требований к способу представления единиц и нулей аппаратурой передачи данных, а во втором – все параметры передаваемых линией импульсов стандартизованы. Другими словами, на цифровых линиях связи протокол физического уровня определен, а на аналоговых линиях – нет.

2.1.3. Характеристики линий связи

Типы характеристик и способы их определения

К основным характеристикам линий связи относятся:

- амплитудно-частотная характеристика;
- полоса пропускания;
- затухание;
- помехоустойчивость;
- перекрестные наводки на ближнем конце линии;
- пропускная способность;
- достоверность передачи данных;
- удельная стоимость.

В первую очередь разработчика вычислительной сети интересуют пропускная способность и достоверность передачи данных, поскольку эти характеристики прямо влияют на производительность и надежность создаваемой сети. Пропускная способность и достоверность – это характеристики как линии связи, так и способа передачи данных. Поэтому если способ передачи (протокол) уже определен, то известны и эти характеристики. Например, пропускная способность цифровой линии всегда известна, так как на ней определен протокол физического уровня, который задает битовую скорость передачи данных – 64 Кбит/с, 2 Мбит/с и т. п.

Однако нельзя говорить о пропускной способности линии связи, до того как для нее определен протокол физического уровня. Именно в таких случаях, когда только предстоит определить, какой из множества существующих протоколов можно использовать на данной линии, очень важными являются остальные характеристики линии, такие как полоса пропускания, перекрестные наводки, помехоустойчивость и другие характеристики.

Для определения характеристик линии связи часто используют анализ ее реакций на некоторые эталонные воздействия. Такой подход позволяет достаточно просто и однотипно определять характеристики линий связи любой природы, не прибегая к сложным теоретическим исследованиям. Чаше всего в качестве эталонных сигналов для исследования реакций линий связи используются синусоидальные сигналы различных частот. Это связано с тем, что сигналы этого типа часто встречаются в технике и с их помощью можно представить любую функцию времени – как непрерывный процесс колебаний звука, так и прямоугольные импульсы, генерируемые компьютером.

Спектральный анализ сигналов на линиях связи

Из теории гармонического анализа известно, что любой периодический процесс можно представить в виде суммы синусоидальных колебаний различных частот и различных амплитуд (рис. 2.3). Каждая составляющая синусоида называется также гармоникой, а набор всех гармоник называют спектральным разложением исходного сигнала. Непериодические сигналы можно представить в виде интеграла синусоидальных сигналов с непрерывным спектром частот. Например, спектральное разложение идеального импульса (единичной мощности и нулевой длительности) имеет составляющие всего спектра частот, от -∞ до +∞ (рис. 2.4).

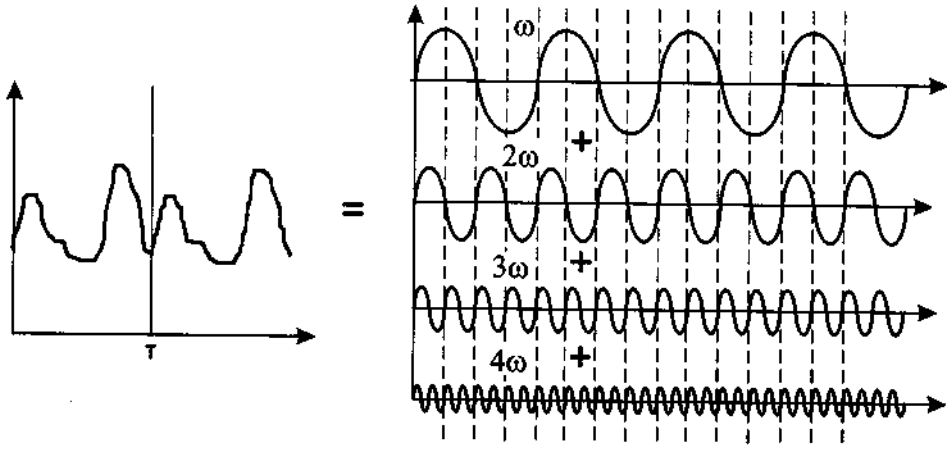


Рис.2.3. Представление периодического сигнала суммой синусоид

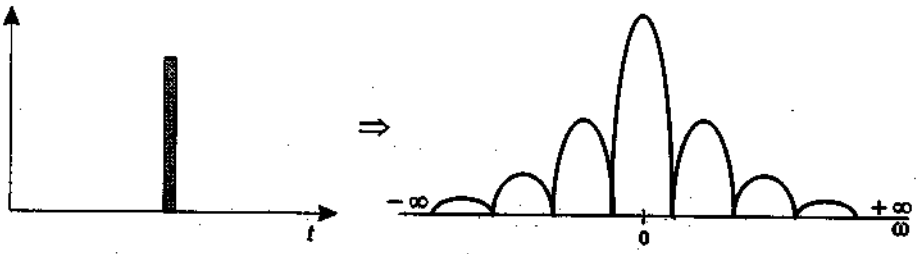


Рис.2.4. Спектральное разложение идеального импульса

Техника нахождения спектра любого исходного сигнала хорошо известна. Для некоторых сигналов, которые хорошо описываются аналитически (например, для последовательности прямоугольных импульсов одинаковой длительности и амплитуды), спектр легко вычисляется на основании формул Фурье. Для сигналов произвольной формы, встречающихся на практике, спектр можно найти с помощью специальных приборов – спектральных анализаторов, которые измеряют спектр реального сигнала и отображают амплитуды составляющих гармоник на экране или распечатывают их на принтере.

Искажение передающим каналом синусоиды какой-либо частоты приводит в конечном счете к искажению передаваемого сигнала любой формы, особенно если синусоиды различных частот искажаются неодинаково. Если это аналоговый сигнал, передающий речь, то изменяется тембр голоса за счет искажения обертонов – боковых частот. При передаче импульсных сигналов, характерных для компьютерных сетей, искажаются низкочастотные и высокочастотные гармоники, в результате фронты импульсов теряют свою прямоугольную форму (рис. 2.5). Вследствие этого на приемном конце линии сигналы могут плохо распознаваться.

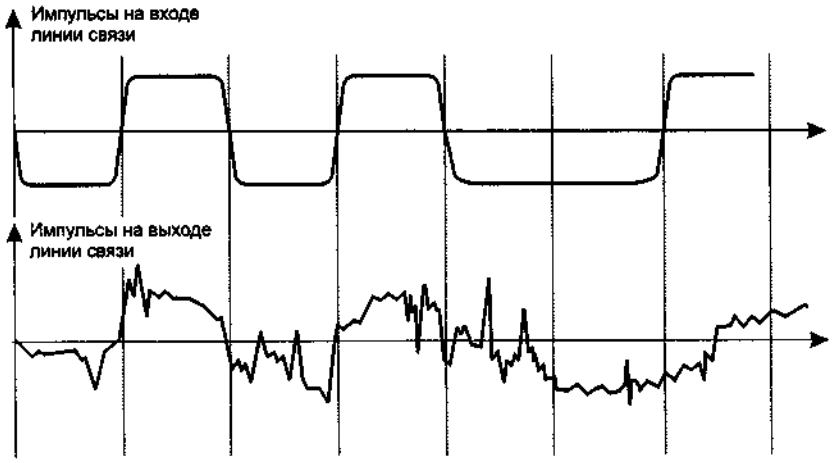


Рис.2.5. Искажение импульсов в линии связи

Линия связи искажает передаваемые сигналы из-за того, что ее физические параметры отличаются от идеальных. Так, например, медные провода всегда представляют собой некоторую распределенную по

длине комбинацию активного сопротивления, емкостной и индуктивной нагрузки (рис. 2.6). В результате для синусоид различных частот линия будет обладать различным полным сопротивлением, а значит, и передаваться они будут по-разному. Волоконно-оптический кабель также имеет отклонения, мешающие идеальному распространению света. Если линия связи включает промежуточную аппаратуру, то она также может вносить дополнительные искажения, так как невозможно создать устройства, которые бы одинаково хорошо передавали весь спектр синусоид, от нуля до бесконечности.

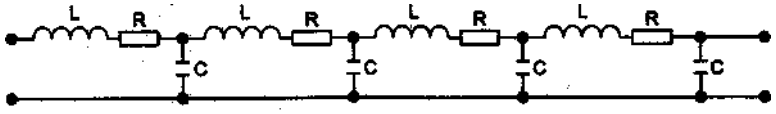


Рис. 2.6. Представление линии как распределенной индуктивно-емкостной нагрузки

Кроме искажений сигналов, вносимых внутренними физическими параметрами линии связи, существуют и внешние помехи, которые вносят свой вклад в искажение формы сигналов на выходе линии. Эти помехи создают различные электрические двигатели, электронные устройства, атмосферные явления и т. д. Несмотря на защитные меры, предпринимаемые разработчиками кабелей и усилительно-коммутирующей аппаратуры, полностью компенсировать влияние внешних помех не удается. Поэтому сигналы на выходе линии связи обычно имеют сложную форму (как это и показано на рис. 2.5), по которой иногда трудно понять, какая дискретная информация была подана на вход линии.

Амплитудно-частотная характеристика, полоса пропускания и затухание

Степень искажения синусоидальных сигналов линиями связи оценивается с помощью таких характеристик, как амплитудно-частотная характеристика, полоса пропускания и затухание на определенной частоте.

Амплитудно-частотная характеристика (рис. 2.7) показывает, как затухает амплитуда синусоиды на выходе линии связи по сравнению с амплитудой на ее входе для всех возможных частот передаваемого сигнала. Вместо амплитуды в этой характеристике часто используют также такой параметр сигнала, как его мощность.

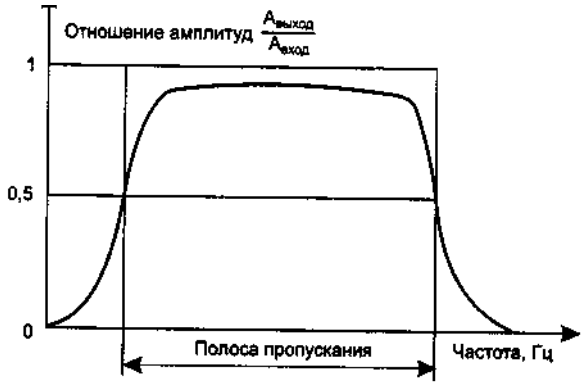


Рис. 2.7. Амплитудно-частотная характеристика

Знание амплитудно-частотной характеристики реальной линии позволяет определить форму выходного сигнала практически для любого входного сигнала. Для этого необходимо найти спектр входного сигнала, преобразовать амплитуду составляющих его гармоник в соответствии с амплитудно-частотной характеристикой, а затем найти форму выходного сигнала, сложив преобразованные гармоники.

Несмотря на полноту информации, предоставляемой амплитудно-частотной характеристикой о линии связи, ее использование осложняется тем обстоятельством, что получить ее весьма трудно. Ведь для этого нужно провести тестирование линии эталонными синусоидами по всему диапазону частот от нуля до некоторого максимального значения, которое может встретиться во входных сигналах. Причем менять частоту входных синусоид нужно с небольшим шагом, а значит, количество экспериментов должно быть очень большим. Поэтому на практике вместо амплитудно-частотной характеристики

применяются другие, упрощенные характеристики – полоса пропускания и затухание.

Полоса пропускания (*bandwidth*) – это непрерывный диапазон частот, для которого отношение амплитуды выходного сигнала ко входному превышает некоторый заранее заданный предел, обычно 0,5. То есть полоса пропускания определяет диапазон частот синусоидального сигнала, при которых этот сигнал передается по линии связи без значительных искажений. Знание полосы пропускания позволяет получить с некоторой степенью приближения тот же результат, что и знание амплитудно-частотной характеристики. Как мы увидим ниже, *ширина* полосы пропускания в наибольшей степени влияет на максимально возможную скорость передачи информации по линии связи. Именно этот факт нашел отражение в английском эквиваленте рассматриваемого термина (width – ширина).

Затухание (*attenuation*) определяется как относительное уменьшение амплитуды или мощности сигнала при передаче по линии сигнала определенной частоты. Таким образом, затухание представляет собой одну точку из амплитудно-частотной характеристики линии. Часто при эксплуатации линии заранее известна основная частота передаваемого сигнала, то есть та частота, гармоника которой имеет наибольшую амплитуду и мощность. Поэтому достаточно знать затухание на этой частоте, чтобы приблизительно оценить искажения передаваемых по линии сигналов. Более точные оценки возможны при знании затухания на нескольких частотах, соответствующих нескольким основным гармоникам передаваемого сигнала.

Затухание A обычно измеряется в децибелах (дБ, decibel – dB) и вычисляется по следующей формуле:

$A = 10 \log_{10} P_{вых}/P_{вх}$,

где $P_{вых}$ – мощность сигнала на выходе линии, $P_{вх}$ – мощность сигнала на входе линии.

Так как мощность выходного сигнала кабеля без промежуточных усилителей всегда меньше, чем мощность входного сигнала, затухание кабеля всегда является отрицательной величиной.

Например, кабель на витой паре категории 5 характеризуется затуханием не ниже -23,6 дБ для частоты 100 МГц при длине кабеля 100 м. Частота 100 МГц выбрана потому, что кабель этой категории предназначен для высокоскоростной передачи данных, сигналы которых имеют значимые гармоники с частотой примерно 100 МГц. Кабель категории 3 предназначен для низкоскоростной передачи данных, поэтому для него определяется затухание на частоте 10 МГц (не ниже -11,5 дБ). Часто оперируют с абсолютными значениями затухания, без указания знака.

Абсолютный *уровень мощности*, например уровень мощности передатчика, также измеряется в децибелах. При этом в качестве базового значения мощности сигнала, относительно которого измеряется текущая мощность, принимается значение в 1 мВт. Таким образом, уровень мощности p вычисляется по следующей формуле:

$p = 10\log_{10} P / 1мВт[дБм]$,

где P – мощность сигнала в милливаттах, а дБм (dBm) – это единица измерения уровня мощности (децибел на 1 мВт).

Таким образом, амплитудно-частотная характеристика, полоса пропускания и затухание являются универсальными характеристиками, и их знание позволяет сделать вывод о том, как через линию связи будут передаваться сигналы любой формы.

Полоса пропускания зависит от типа линии и ее протяженности. На рис. 2.8 показаны полосы пропускания линий связи различных типов, а также наиболее часто используемые в технике связи частотные диапазоны.

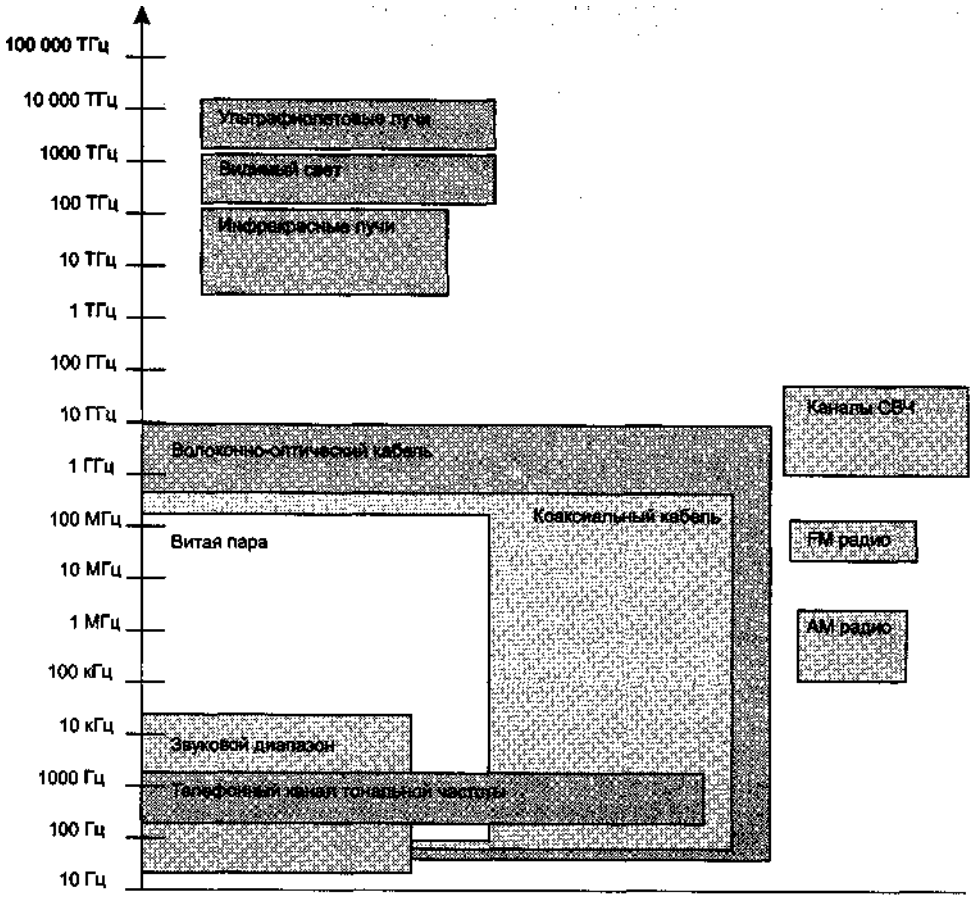


Рис. 2.8. Полосы пропускания линий связи и популярные частотные диапозоны

Пропускная способность линии

Пропускная способность (*throughput*) линии характеризует максимально возможную скорость передачи данных по линии связи. Пропускная способность измеряется в битах в секунду – бит/с, а также в производных единицах, таких как килобит в секунду (Кбит/с), мегабит в секунду (Мбит/с), гигабит в секунду (Гбит/с) и т. д.

ПРИМЕЧАНИЕ Пропускная способность линий связи и коммуникационного сетевого оборудования традиционно измеряется в битах в секунду, а не в байтах в секунду. Это связано с тем, что данные в сетях передаются последовательно, то есть побитно, а не параллельно, байтами, как это происходит между устройствами внутри компьютера. Такие единицы измерения, как килобит, мегабит или гигабит, в сетевых технологиях строго соответствуют степеням 10 (то есть килобит - это 1000 бит, а мегабит – это1 000 000 бит), как это принято во всех отраслях науки и техники, а не близким к этим числам степеням 2, как это принято в программировании, где приставка «кило» равна 2^{10}=1024, а «мега» - 2^{20}=1048576.

Пропускная способность линии связи зависит не только от ее характеристик, таких как амплитудно-частотная характеристика, но и от спектра передаваемых сигналов. Если значимые гармоники сигнала (то есть те гармоники, амплитуды которых вносят основной вклад в результирующий сигнал) попадают в полосу пропускания линии, то такой сигнал будет хорошо передаваться данной линией связи и приемник сможет правильно распознать информацию, отправленную по линии передатчиком (рис. 2.9, *а*). Если же значимые гармоники выходят за границы полосы пропускания линии связи, то сигнал будет значительно искажаться, приемник будет ошибаться при распознавании информации, а значит, информация не сможет передаваться с заданной пропускной способностью (рис. 2.9, *б*).

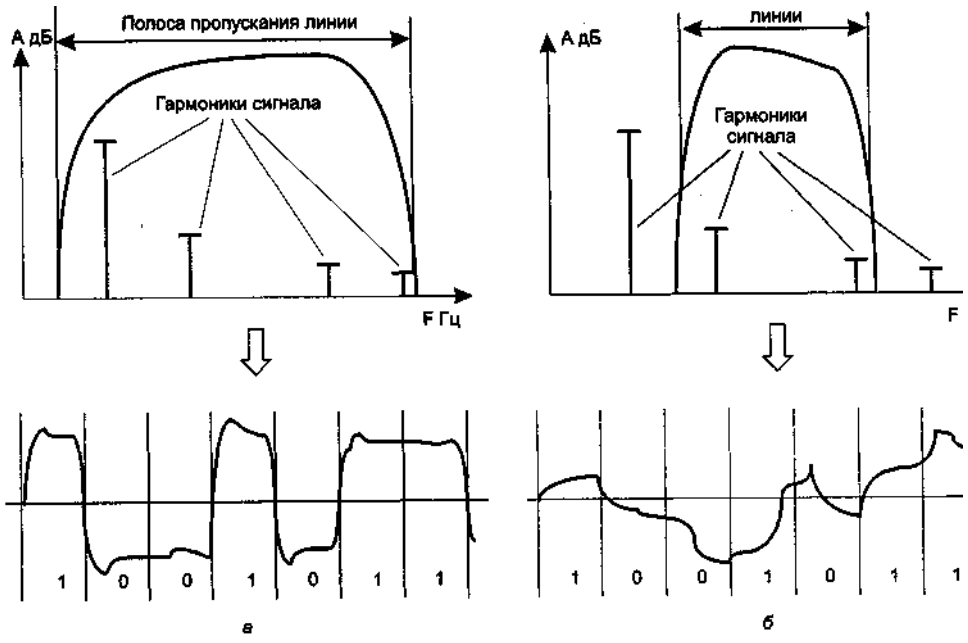


Рис. 2.9. Соответствие между полосой пропускания линии связи и спектром сигнала

Выбор способа представления дискретной информации в виде сигналов, подаваемых на линию связи, называется *физическим* или *линейным кодированием*. От выбранного способа кодирования зависит спектр сигналов и, соответственно, пропускная способность линии. Таким образом, для одного способа кодирования линия может обладать одной пропускной способностью, а для другого – другой. Например, витая пара категории 3 может передавать данные с пропускной способностью 10 Мбит/с при способе кодирования стандарта физического уровня 10Base-T и 33 Мбит/с при способе кодирования стандарта 100Base-T4. В примере, приведенном на рис. 2.9, принят следующий способ кодирования – логическая 1 представлена на линии положительным потенциалом, а логический 0 – отрицательным.

Теория информации говорит, что любое различимое и непредсказуемое изменение принимаемого сигнала несет в себе информацию. В соответствии с этим прием синусоиды, у которой амплитуда, фаза и частота остаются неизменными, информации не несет, так как изменение сигнала хотя и происходит, но является хорошо предсказуемым. Аналогично, не несут в себе информации импульсы на тактовой шине компьютера, так как их изменения также постоянны во времени. А вот импульсы на шине данных предсказать заранее нельзя, поэтому они переносят информацию между отдельными блоками или устройствами.

Большинство способов кодирования используют изменение какого-либо параметра периодического сигнала – частоты, амплитуды и фазы синусоиды или же знак потенциала последовательности импульсов. Периодический сигнал, параметры которого изменяются, называют *несущим сигналом* или *несущей частотой*, если в качестве такого сигнала используется синусоида.

Если сигнал изменяется так, что можно различить только два его состояния, то любое его изменение будет соответствовать наименьшей единице информации – биту. Если же сигнал может иметь более двух различимых состояний, то любое его изменение будет нести несколько бит информации.

Количество изменений информационного параметра несущего периодического сигнала в секунду измеряется в *бодах* (*baud*). Период времени между соседними изменениями информационного сигнала называется тактом работы передатчика.

Пропускная способность линии в битах в секунду в общем случае не совпадает с числом бод. Она может быть как выше, так и ниже числа бод, и это соотношение зависит от способа кодирования.

Если сигнал имеет более двух различимых состояний, то пропускная способность в битах в секунду будет выше, чем число бод. Например, если информационными параметрами являются фаза и амплитуда синусоиды, причем различаются 4 состояния фазы в 0, 90, 180 и 270 градусов и два значения амплитуды сигнала, то информационный сигнал может иметь 8 различимых состояний. В этом случае модем, работающий со скоростью 2400 бод (с тактовой частотой 2400 Гц) передает информацию со скоростью 7200 бит/с, так как при одном изменении сигнала передается 3 бита информации.

При использовании сигналов с двумя различимыми состояниями может наблюдаться обратная картина. Это часто происходит потому, что для надежного распознавания приемником пользовательской информации каждый бит в последовательности кодируется с помощью нескольких

изменений информационного параметра несущего сигнала. Например, при кодировании единичного значения бита импульсом положительной полярности, а нулевого значения бита – импульсом отрицательной полярности физический сигнал дважды изменяет свое состояние при передаче каждого бита. При таком кодировании пропускная способность линии в два раза ниже, чем число бод, передаваемое по линии.

На пропускную способность линии оказывает влияние не только физическое, но и логическое кодирование. *Логическое кодирование* выполняется до физического кодирования и подразумевает замену бит исходной информации новой последовательностью бит, несущей ту же информацию, но обладающей, кроме этого, дополнительными свойствами, например возможностью для приемной стороны обнаруживать ошибки в принятых данных. Сопровождение каждого байта исходной информации одним битом четности – это пример очень часто применяемого способа логического кодирования при передаче данных с помощью модемов. Другим примером логического кодирования может служить шифрация данных, обеспечивающая их конфиденциальность при передаче через общественные каналы связи. При логическом кодировании чаще всего исходная последовательность бит заменяется более длинной последовательностью, поэтому пропускная способность канала по отношению к полезной информации при этом уменьшается.

Связь между пропускной способностью линии и ее полосой пропускания

Чем выше частота несущего периодического сигнала, тем больше информации в единицу времени передается по линии и тем выше пропускная способность линии при фиксированном способе физического кодирования. Однако, с другой стороны, с увеличением частоты периодического несущего сигнала увеличивается и ширина спектра этого сигнала, то есть разность между максимальной и минимальной частотами того набора синусоид, которые в сумме дадут выбранную для физического кодирования последовательность сигналов. Линия передает этот спектр синусоид с теми искажениями, которые определяются ее полосой пропускания. Чем больше несоответствие между полосой пропускания линии и шириной спектра передаваемых информационных сигналов, тем больше сигналы искажаются и тем вероятнее ошибки в распознавании информации принимающей стороной, а значит, скорость передачи информации на самом деле оказывается меньше, чем можно было предположить.

Связь между полосой пропускания линии и ее *максимально возможной пропускной способностью*, вне зависимости от принятого способа физического кодирования, установил Клод Шеннон:

$C = F \log2(1 + P_c/P_ш)$,

где C – максимальная пропускная способность линии в битах в секунду, F – ширина полосы пропускания линии в герцах, Рс – мощность сигнала, Рш – мощность шума.

Из этого соотношения видно, что хотя теоретического предела пропускной способности линии с фиксированной полосой пропускания не существует, на практике такой предел имеется. Действительно, повысить пропускную способность линии можно за счет увеличения мощности передатчика или же уменьшения мощности шума (помех) на линии связи. Обе эти составляющие поддаются изменению с большим трудом. Повышение мощности передатчика ведет к значительному увеличению его габаритов и стоимости. Снижение уровня шума требует применения специальных кабелей с хорошими защитными экранами, что весьма дорого, а также снижения шума в передатчике и промежуточной аппаратуре, чего достичь весьма не просто. К тому же влияние мощностей полезного сигнала и шума на пропускную способность ограничено логарифмической зависимостью, которая растет далеко не так быстро, как прямо-пропорциональная. Так, при достаточно типичном исходном отношении мощности сигнала к мощности шума в 100 раз повышение мощности передатчика в 2 раза даст только 15 % увеличения пропускной способности линии.

Близким по сути к формуле Шеннона является следующее соотношение, полученное Найквистом, которое также определяет максимально возможную пропускную способность линии связи, но без учета шума на линии:

$C = 2F \log_2 M$,

где M – количество различимых состояний информационного параметра.

Если сигнал имеет 2 различимых состояния, то пропускная способность равна удвоенному значению ширины полосы пропускания линии связи (рис. 2.10, *а*). Если же передатчик использует более чем 2 устойчивых состояния сигнала для кодирования данных, то пропускная способность линии повышается, так как за один такт работы передатчик передает несколько бит исходных данных, например 2 бита при

наличии четырех различимых состояний сигнала (рис. 2.10,*б*).

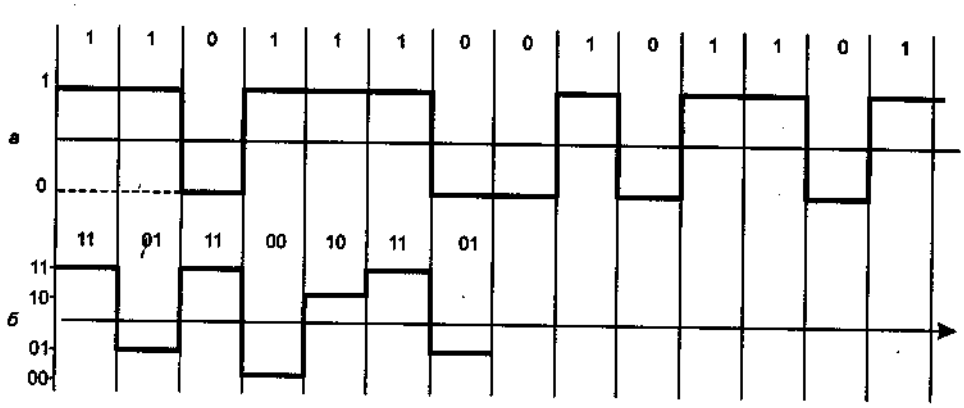


Рис. 2.10. Повышение скорости передачи за счет дополнительных состояний сигнала

Хотя формула Найквиста явно не учитывает наличие шума, косвенно его влияние отражается в выборе количества состояний информационного сигнала. Для повышения пропускной способности канала хотелось бы увеличить это количество до значительных величин, но на практике мы не можем этого сделать из-за шума на линии. Например, для примера, приведенного на рис. 2.10, можно увеличить пропускную способность линии еще в два раза, использовав для кодирования данных не 4, а 16 уровней. Однако если амплитуда шума часто превышает разницу между соседними 16-ю уровнями, то приемник не сможет устойчиво распознавать передаваемые данные. Поэтому количество возможных состояний сигнала фактически ограничивается соотношением мощности сигнала и шума, а формула Найквиста определяет предельную скорость передачи данных в том случае, когда количество состояний уже выбрано с учетом возможностей устойчивого распознавания приемником.

Приведенные соотношения дают предельное значение пропускной способности линии, а степень приближения к этому пределу зависит от конкретных методов физического кодирования, рассматриваемых ниже.

Помехоустойчивость и достоверность

Помехоустойчивость линии определяет ее способность уменьшать уровень помех, создаваемых во внешней среде, на внутренних проводниках. Помехоустойчивость линии зависит от типа используемой физической среды, а также от экранирующих и подавляющих помехи средств самой линии. Наименее помехоустойчивыми являются радиолинии, хорошей устойчивостью обладают кабельные линии и отличной – волоконно-оптические линии, малочувствительные ко внешнему электромагнитному излучению. Обычно для уменьшения помех, появляющихся из-за внешних электромагнитных полей, проводники экранируют и/или скручивают.

Перекрестные наводки на ближнем конце (*Near End Cross Talk – NEXT*) определяют помехоустойчивость кабеля к внутренним источникам помех, когда электромагнитное поле сигнала, передаваемого выходом передатчика по одной паре проводников, наводит на другую пару проводников сигнал помехи. Если ко второй паре будет подключен приемник, то он может принять наведенную внутреннюю помеху за полезный сигнал. Показатель NEXT, выраженный в децибелах, равен 10 log $P_{вых}/P_{нав}$, где $P_{вых}$ – мощность выходного сигнала, $P_{нав}$ – мощность наведенного сигнала.

Чем меньше значение NEXT, тем лучше кабель. Так, для витой пары категории 5 показатель NEXT должен быть меньше -27 дБ на частоте 100 МГц.

Показатель NEXT обычно используется применительно к кабелю, состоящему из нескольких витых пар, так как в этом случае взаимные наводки одной пары на другую могут достигать значительных величин. Для одинарного коаксиального кабеля (то есть состоящего из одной экранированной жилы) этот показатель не имеет смысла, а для двойного коаксиального кабеля он также не применяется вследствие высокой степени защищенности каждой жилы. Оптические волокна также не создают сколь-нибудь заметных помех друг для друга.

В связи с тем, что в некоторых новых технологиях используется передача данных одновременно по нескольким витым парам, в последнее время стал применяться показатель *PowerSUM*, являющийся модификацией показателя NEXT. Этот показатель отражает суммарную мощность перекрестных

наводок от всех передающих пар в кабеле.

Достоверность передачи данных характеризует вероятность искажения для каждого передаваемого бита данных. Иногда этот же показатель называют *интенсивностью битовых ошибок* (*Bit Error Rate, BER*). Величина BER для каналов связи без дополнительных средств защиты от ошибок (например, самокорректирующихся кодов или протоколов с повторной передачей искаженных кадров) составляет, как правило, 10^{-4}-10^{-6}, в оптоволоконных линиях связи – 10^{-9}. Значение достоверности передачи данных, например, в 10^{-4} говорит о том, что в среднем из 10 000 бит искажается значение одного бита.

Искажения бит происходят как из-за наличия помех на линии, так и по причине искажений формы сигнала ограниченной полосой пропускания линии. Поэтому для повышения достоверности передаваемых данных нужно повышать степень помехозащищенности линии, снижать уровень перекрестных наводок в кабеле, а также использовать более широкополосные линии связи.

2.1.4. Стандарты кабелей

Кабель – это достаточно сложное изделие, состоящее из проводников, слоев экрана и изоляции. В некоторых случаях в состав кабеля входят разъемы, с помощью которых кабели присоединяются к оборудованию. Кроме этого, для обеспечения быстрой перекоммутации кабелей и оборудования используются различные электромеханические устройства, называемые кроссовыми секциями, кроссовыми коробками или шкафами.

В компьютерных сетях применяются кабели, удовлетворяющие определенным стандартам, что позволяет строить кабельную систему сети из кабелей и соединительных устройств разных производителей. Сегодня наиболее употребительными стандартами в мировой практике являются следующие.

- Американский стандарт EIA/TIA-568A, который был разработан совместными усилиями нескольких организаций; ANSI, EIA/TIA и лабораторией Underwriters Labs (UL). Стандарт EIA/TIA-568 разработан на основе предыдущей версии стандарта EIA/TIA-568 и дополнений к этому стандарту TSB-36 и TSB 40A).
- Международный стандарт ISO/IEC 11801.
- Европейский стандарт EN50173.

Эти стандарты близки между собой и по многим позициям предъявляют к кабелям идентичные требования. Однако есть и различия между этими стандартами, например, в международный стандарт 11801 и европейский EN50173 вошли некоторые типы кабелей, которые отсутствуют в стандарте EIA/TAI-568A.

До появления стандарта EIA/TIA большую роль играл американский стандарт *системы категорий кабелей* Underwriters Labs, разработанный совместно с компанией Anixter. Позже этот стандарт вошел в стандарт EIA/TIA-568.

Кроме этих открытых стандартов, многие компании в свое время разработали свои фирменные стандарты, из которых до сих пор имеет практическое значение только один – стандарт компании IBM.

При стандартизации кабелей принят протокольно-независимый подход. Это означает, что в стандарте оговариваются электрические, оптические и механические характеристики, которым должен удовлетворять тот или иной тип кабеля или соединительного изделия – разъема, кроссовой коробки и т. п. Однако для какого протокола предназначен данный кабель, стандарт не оговаривает. Поэтому нельзя приобрести кабель для протокола Ethernet или FDDI, нужно просто знать, какие типы стандартных кабелей поддерживают протоколы Ethernet и FDDI.

В ранних версиях стандартов определялись только характеристики кабелей, без соединителей. В последних версиях стандартов появились требования к соединительным элементам (документы TSB-36 и TSB-40A, вошедшие затем в стандарт 568A), а также к *линиям* (*каналам*), представляющим типовую сборку элементов кабельной системы, состоящую из шнура от рабочей станции до розетки, самой розетки, основного кабеля (длиной до 90 м для витой пары), точки перехода (например, еще одной розетки или жесткого кроссового соединения) и шнура до активного оборудования, например концентратора или коммутатора.

Мы остановимся только на основных требованиях к самим кабелям, не рассматривая характеристик соединительных элементов и собранных линий.

В стандартах кабелей оговаривается достаточно много характеристик, из которых наиболее важные перечислены ниже (первые две из них уже были достаточно детально рассмотрены).

- *Затухание* (*Attenuation*). Затухание измеряется в децибелах на метр для определенной частоты или диапазона частот сигнала.
- *Перекрестные наводки на ближнем конце* (*Near End Cross Talk, NEXT*). Измеряются в децибелах для определенной частоты сигнала.
- *Импеданс* (*волновое сопротивление*) – это полное (активное и реактивное) сопротивление в электрической цепи. Импеданс измеряется в Омах и является относительно постоянной величиной для кабельных систем (например, для коаксиальных кабелей, используемых в стандартах Ethernet, импеданс кабеля должен составлять 50 Ом). Для неэкранированной витой пары наиболее часто используемые значения импеданса – 100 и 120 Ом. В области высоких частот (100-200 МГц) импеданс зависит от частоты.
- *Активное сопротивление* – это сопротивление постоянному току в электрической цепи. В отличие от импеданса активное сопротивление не зависит от частоты и возрастает с увеличением длины кабеля.
- *Емкость* – это свойство металлических проводников накапливать энергию. Два электрических проводника в кабеле, разделенные диэлектриком, представляют собой конденсатор, способный накапливать заряд. Емкость является нежелательной величиной, поэтому следует стремиться к тому, чтобы она была как можно меньше (иногда применяют термин «паразитная емкость»). Высокое значение емкости в кабеле приводит к искажению сигнала и ограничивает полосу пропускания линии.
- *Уровень внешнего электромагнитного излучения или электрический шум*. Электрический шум – это нежелательное переменное напряжение в проводнике. Электрический шум бывает двух типов: фоновый и импульсный. Электрический шум можно также разделить на низко-, средне- и высокочастотный. Источниками фонового электрического шума в диапазоне до 150 кГц являются линии электропередачи, телефоны и лампы дневного света; в диапазоне от 150 кГц до 20 МГц – компьютеры, принтеры, ксероксы; в диапазоне от 20 МГц до 1 ГГц – телевизионные и радиопередатчики, микроволновые печи. Основными источниками импульсного электрического шума являются моторы, переключатели и сварочные агрегаты. Электрический шум измеряется в милливольтах.
- *Диаметр или площадь сечения проводника*. Для медных проводников достаточно употребительной является американская система AWG (American Wire Gauge), которая вводит некоторые условные типы проводников, например 22 AWG, 24 AWG, 26 AWG. Чем больше номер типа проводника, тем меньше его диаметр. В вычислительных сетях наиболее употребительными являются типы проводников, приведенные выше в качестве примеров. В европейских и международных стандартах диаметр проводника указывается в миллиметрах.

Естественно, приведенный перечень характеристик далеко не полон, причем в нем представлены только электромагнитные характеристики и его нужно дополнить механическими и конструктивными характеристиками, определяющими тип изоляции, конструкцию разъема и т. п. Помимо универсальных характеристик, таких, например, как затухание, которые применимы для всех типов кабелей, существуют характеристики, которые применимы только к определенному типу кабеля. Например, параметр *шаг скрутки проводов* используется только для характеристики витой пары, а параметр NEXT применим только к многопарным кабелям на основе витой пары.

Основное внимание в современных стандартах уделяется кабелям на основе витой пары и волоконно-оптическим кабелям.

Кабели на основе неэкранированной витой пары

Медный неэкранированный кабель UTP в зависимости от электрических и механических характеристик разделяется на 5 категорий (Category 1 – Category 5). Кабели категорий 1 и 2 были определены в стандарте EIA/TIA-568, но в стандарт 568A уже не вошли, как устаревшие.

Кабели *категории* 1 применяются там, где требования к скорости передачи минимальны. Обычно это кабель для цифровой и аналоговой передачи голоса и низкоскоростной (до 20 Кбит/с) передачи данных. До 1983 года это был основной тип кабеля для телефонной разводки.

Кабели *категории* 2 были впервые применены фирмой IBM при построении собственной кабельной системы. Главное требование к кабелям этой категории – способность передавать сигналы со спектром до 1 МГц.

Кабели *категории* 3 были стандартизованы в 1991 году, когда был разработан *Стандарт телекоммуникационных кабельных систем для коммерческих зданий* (EIA-568), на основе которого затем был создан действующий стандарт EIA-568A. Стандарт EIA-568 определил электрические характеристики кабелей категории 3 для частот в диапазоне до 16 МГц, поддерживающих, таким образом, высокоскоростные сетевые приложения. Кабель категории 3 предназначен как для передачи данных, так и для передачи голоса. Шаг скрутки проводов равен примерно 3 витка на 1 фут (30,5 см). Кабели категории 3 сейчас составляют основу многих кабельных систем зданий, в которых они используются для передачи и голоса, и данных.

Кабели *категории* 4 представляют собой несколько улучшенный вариант кабелей категории 3. Кабели категории 4 обязаны выдерживать тесты на частоте передачи сигнала 20 МГц и обеспечивать повышенную помехоустойчивость и низкие потери сигнала. Кабели категории 4 хорошо подходят для применения в системах с увеличенными расстояниями (до 135 метров) и в сетях Token Ring с пропускной способностью 16 Мбит/с. На практике используются редко.

Кабели *категории* 5 были специально разработаны для поддержки высокоскоростных протоколов. Поэтому их характеристики определяются в диапазоне до 100 МГц. Большинство новых высокоскоростных стандартов ориентируются на использование витой пары 5 категории. На этом кабеле работают протоколы со скоростью передачи данных 100 Мбит/с – FDDI (с физическим стандартом TP-PMD), Fast Ethernet, lOOVG-AnyLAN, а также более скоростные протоколы – ATM на скорости 155 Мбит/с, и Gigabit Ethernet на скорости 1000 Мбит/с (вариант Gigabit Ethernet на витой паре категории 5 стал стандартом в июне 1999 г.). Кабель категории 5 пришел на замену кабелю категории 3, и сегодня все новые кабельные системы крупных зданий строятся именно на этом типе кабеля (в сочетании с волоконно-оптическим).

Наиболее важные электромагнитные характеристики кабеля категории 5 имеют следующие значения:

- в полное волновое сопротивление в диапазоне частот до 100 МГц равно 100 Ом (стандарт ISO 11801 допускает также кабель с волновым сопротивлением 120 Ом);
- величина перекрестных наводок NEXT в зависимости от частоты сигнала должна принимать значения не менее 74 дБ на частоте 150 кГц и не менее 32 дБ на частоте 100 МГц;
- затухание имеет предельные значения от 0,8 дБ (на частоте 64 кГц) до 22 дБ (на частоте 100 МГц);
- активное сопротивление не должно превышать 9,4 Ом на 100 м;
- емкость кабеля не должна превышать 5,6 нф на 100 м.

Все кабели UTP независимо от их категории выпускаются в 4-парном исполнении. Каждая из четырех пар кабеля имеет определенный цвет и шаг скрутки. Обычно две пары предназначены для передачи данных, а две – для передачи голоса.

Для соединения кабелей с оборудованием используются вилки и розетки RJ-45, представляющие 8-контактные разъемы, похожие на обычные телефонные разъемы, RJ-ll.

Особое место занимают кабели *категорий 6 и 7*, которые промышленность начала выпускать сравнительно недавно. Для кабеля категории 6 характеристики определяются до частоты 200 МГц, а для кабелей категории 7 до 600 МГц. Кабели категории 7 обязательно экранируются, причем как каждая пара, так и весь кабель в целом. Кабель категории 6 может быть как экранированным, так и неэкранированным. Основное назначение этих кабелей поддержка высокоскоростных протоколов на отрезках кабеля большей длины, чем кабель UTP категории 5. Некоторые специалисты сомневаются в необходимости применения кабелей категории 7, так как стоимость кабельной системы при их использовании получается соизмеримой по стоимости сети с использованием волоконно-оптических кабелей, а характеристики кабелей на основе оптических волокон выше.

Кабели на основе экранированной витой пары

Экранированная витая пара STP хорошо защищает передаваемые сигналы от внешних помех, а также меньше излучает электромагнитных колебаний вовне, что защищает, в свою очередь, пользователей сетей от вредного для здоровья излучения. Наличие заземляемого экрана удорожает кабель и усложняет его прокладку, так как требует выполнения качественного заземления. Экранированный кабель применяется только для передачи данных, а голос по нему не передают.

Основным стандартом, определяющим параметры экранированной витой пары, является фирменный

стандарт IBM. В этом стандарте кабели делятся не на категории, а на типы: Type 1, Type 2,..., Type 9.

Основным типом экранированного кабеля является кабель Type 1 стандарта IBM. Он состоит из 2-х пар скрученных проводов, экранированных проводящей оплеткой, которая заземляется. Электрические параметры кабеля Type 1 примерно соответствуют параметрам кабеля UTP категории 5. Однако волновое сопротивление кабеля Type 1 равно 150 Ом (UTP категории 5 имеет волновое сопротивление 100 Ом), поэтому простое «улучшение» кабельной проводки сети путем замены неэкранированной пары UTP на STP Type 1 невозможно. Трансиверы, рассчитанные на работу с кабелем, имеющим волновое сопротивление 100 Ом, будут плохо работать на волновое сопротивление 150 Ом. Поэтому при использовании STP Type 1 необходимы соответствующие трансиверы. Такие трансиверы имеются в сетевых адаптерах Token Ring, так как эти сети разрабатывались для работы на экранированной витой паре. Некоторые другие стандарты также поддерживают кабель STP Type 1 – например, lOOVG-AnyLAN, а также Fast Ethernet (хотя основным типом кабеля для Fast Ethernet является UTP категории 5). В случае если технология может использовать UTP и STP, нужно убедиться, на какой тип кабеля рассчитаны приобретаемые трансиверы. Сегодня кабель STP Type 1 включен в стандарты EIA/TIA-568A, ISO 11801 и EN50173, то есть приобрел международный статус.

Экранированные витые пары используются также в кабеле IBM Type 2, который представляет кабель Type 1 с добавленными 2 парами неэкранированного провода для передачи голоса.

Для присоединения экранированных кабелей к оборудованию используются разъемы конструкции IBM.

Не все типы кабелей стандарта IBM относятся к экранированным кабелям – некоторые определяют характеристики неэкранированного телефонного кабеля (Type 3) и оптоволоконного кабеля (Type 5).

Коаксиальные кабели

Существует большое количество типов коаксиальных кабелей, используемых в сетях различного типа - телефонных, телевизионных и компьютерных. Ниже приводятся основные типы и характеристики этих кабелей.

- RG-8 и RG-11 – «толстый» коаксиальный кабель, разработанный для сетей Ethernet 10Base-5. Имеет волновое сопротивление 50 Ом и внешний диаметр 0,5 дюйма (около 12 мм). Этот кабель имеет достаточно толстый внутренний проводник диаметром 2,17 мм, который обеспечивает хорошие механические и электрические характеристики (затухание на частоте 10 МГц – не хуже 18 дБ/км). Зато этот кабель сложно монтировать – он плохо гнется.
- RG-58/U, RG-58 A/U и RG-58 C/U – разновидности «тонкого» коаксиального кабеля для сетей Ethernet 10Base-2. Кабель RG-58/U имеет сплошной внутренний проводник, а кабель RG-58 A/U - многожильный. Кабель RG-58 C/U проходит «военную приемку». Все эти разновидности кабеля имеют волновое сопротивление 50 Ом, но обладают худшими механическими и электрическими характеристиками по сравнению с «толстым» коаксиальным кабелем. Тонкий внутренний проводник 0,89 мм не так прочен, зато обладает гораздо большей гибкостью, удобной при монтаже. Затухание в этом типе кабеля выше, чем в «толстом» коаксиальном кабеле, что приводит к необходимости уменьшать длину кабеля для получения одинакового затухания в сегменте. Для соединения кабелей с оборудованием используется разъем типа BNC.
- RG-59 – телевизионный кабель с волновым сопротивлением 75 Ом. Широко применяется в кабельном телевидении.
- RG-62 – кабель с волновым сопротивлением 93 Ома, использовался в сетях ArcNet, оборудование которых сегодня практически не выпускается.

Коаксиальные кабели с волновым сопротивлением 50 Ом (то есть «тонкий» и «толстый») описаны в стандарте EIA/TIA-568. Новый стандарт EIA/TIA-568A коаксиальные кабели не описывает, как морально устаревшие.

Волоконно-оптические кабели

Волоконно-оптические кабели состоят из центрального проводника света (сердцевины) – стеклянного волокна, окруженного другим слоем стекла – оболочкой, обладающей меньшим показателем преломления, чем сердцевина. Распространяясь по сердцевине, лучи света не выходят за ее

пределы, отражаясь от покрывающего слоя оболочки. В зависимости от распределения показателя преломления и от величины диаметра сердечника различают:

- многомодовое волокно со ступенчатым изменением показателя преломления (рис. 2.11, *а*);
- многомодовое волокно с плавным изменением показателя преломления (рис. 2.11,*б*);
- одномодовое волокно (рис. 2.11, *в*).

Понятие «мода» описывает режим распространения световых лучей во внутреннем сердечнике кабеля. В *одномодовом кабеле* (*Single Mode Fiber, SMF*) используется центральный проводник очень малого диаметра, соизмеримого с длиной волны света – от 5 до 10 мкм. При этом практически все лучи света распространяются вдоль оптической оси световода, не отражаясь от внешнего проводника. Полоса пропускания одномодового кабеля очень широкая – до сотен гигагерц на километр. Изготовление тонких качественных волокон для одномодового кабеля представляет сложный технологический процесс, что делает одномодовый кабель достаточно дорогим. Кроме того, в волокно такого маленького диаметра достаточно сложно направить пучок света, не потеряв при этом значительную часть его энергии.

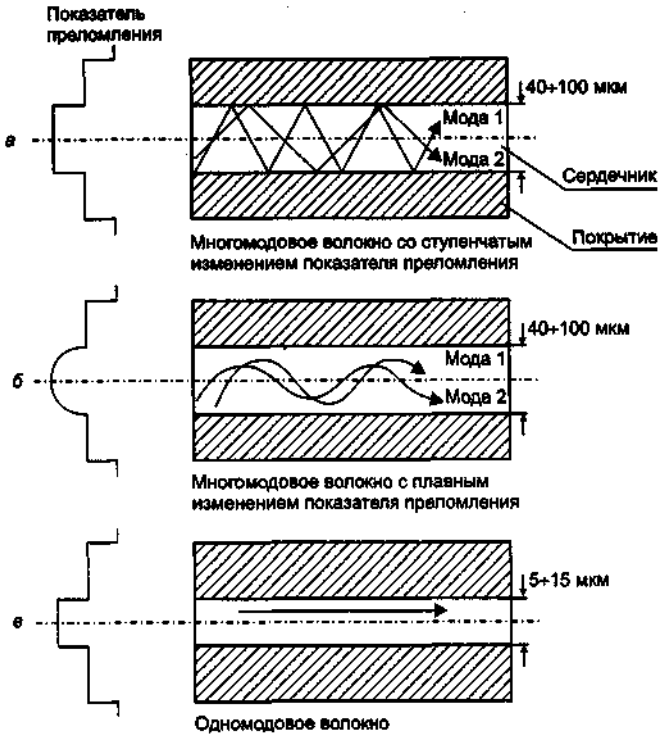


Рис. 2.11. Типы оптического кабеля

В *многомодовых кабелях* (*Multi Mode Fiber, MMF*) используются более широкие внутренние сердечники, которые легче изготовить технологически. В стандартах определены два наиболее употребительных многомодовых кабеля: 62,5/125 мкм и 50/125 мкм, где 62,5 мкм или 50 мкм – это диаметр центрального проводника, а 125 мкм – диаметр внешнего проводника.

В многомодовых кабелях во внутреннем проводнике одновременно существует несколько световых лучей, отражающихся от внешнего проводника под разными углами. Угол отражения луча называется модой луча. В многомодовых кабелях с плавным изменением коэффициента преломления режим распространения каждой моды имеет более сложный характер.

Многомодовые кабели имеют более узкую полосу пропускания – от 500 до 800 МГц/км. Сужение полосы происходит из-за потерь световой энергии при отражениях, а также из-за интерференции лучей разных мод.

В качестве источников излучения света в волоконно-оптических кабелях применяются:

- светодиоды;
- полупроводниковые лазеры.

Для одномодовых кабелей применяются только полупроводниковые лазеры, так как при таком малом диаметре оптического волокна световой поток, создаваемый светодиодом, невозможно без больших потерь направить в волокно. Для многомодовых кабелей используются более дешевые светодиодные

излучатели.

Для передачи информации применяется свет с длиной волны 1550 нм (1,55 мкм), 1300 нм (1,3 мкм) и 850 нм (0,85 мкм). Светодиоды могут излучать свет с длиной волны 850 нм и 1300 нм. Излучатели с длиной волны 850 нм существенно дешевле, чем излучатели с длиной волны 1300 нм, но полоса пропускания кабеля для волн 850 нм уже, например 200 МГц/км вместо 500 МГц/км.

Лазерные излучатели работают на длинах волн 1300 и 1550 нм. Быстродействие современных лазеров позволяет модулировать световой поток с частотами 10 ГГц и выше. Лазерные излучатели создают когерентный поток света, за счет чего потери в оптических волокнах становятся меньше, чем при использовании некогерентного потока светодиодов.

Использование только нескольких длин волн для передачи информации в оптических волокнах связанно с особенностью их амплитудно-частотной характеристики. Именно для этих дискретных длин волн наблюдаются ярко выраженные максимумы передачи мощности сигнала, а для других волн затухание в волокнах существенно выше.

Волоконно-оптические кабели присоединяют к оборудованию разъемами MIC, ST и SC.

Волоконно-оптические кабели обладают отличными характеристиками всех типов: электромагнитными, механическими (хорошо гнутся, а в соответствующей изоляции обладают хорошей механической прочностью). Однако у них есть один серьезный недостаток – сложность соединения волокон с разъемами и между собой при необходимости наращивания длины кабеля.

Сама стоимость волоконно-оптических кабелей ненамного превышает стоимость кабелей на витой паре, однако проведение монтажных работ с оптоволокном обходится намного дороже из-за трудоемкости операций и высокой стоимости применяемого монтажного оборудования. Так, присоединение оптического волокна к разъему требует проведения высокоточной обрезки волокна в плоскости строго перпендикулярной оси волокна, а также выполнения соединения путем сложной операции склеивания, а не обжатия, как это делается для витой пары. Выполнение же некачественных соединений сразу резко сужает полосу пропускания волоконно-оптических кабелей и линий.

Выводы

- При построении сетей применяются линии связи, использующие различную физическую среду: телефонные и телеграфные провода, подвешенные в воздухе, медные коаксиальные кабели, медные витые пары, волоконно-оптические кабели, радиоволны.
- Линии связи могут использовать, кроме кабеля, промежуточную аппаратуру, прозрачную для пользователей. Промежуточная аппаратура выполняет две основные функции: усиливает сигналы и обеспечивает постоянную коммутацию между парой пользователей линии.
- В зависимости от типа промежуточной аппаратуры линии связи делятся на аналоговые и цифровые. В аналоговых линиях связи для уплотнения низкоскоростных каналов абонентов в общий высокоскоростной канал используется метод разделения частот (FDM), а в цифровых – метод разделения во времени (TDM).
- Для характеристики способности линии передавать сигналы произвольной формы без значительных искажений применяется ряд показателей, использующих в качестве тестового сигнала синусоиды различной частоты. К этим показателям относятся: амплитудно-частотная характеристика, полоса пропускания и затухание сигнала на определенной частоте.
- В компьютерных сетях применяются кабели, удовлетворяющие определенным стандартам. Современные стандарты определяет характеристики не отдельного кабеля, а полного набора элементов, необходимого для создания кабельного соединения, например шнура от рабочей станции до розетки, самой розетки, основного кабеля, жесткого кроссового соединения и шнура до концентратора. Сегодня наиболее употребительными стандартами являются: американский стандарт EIA/TIA-568A, международный стандарт ISO/IEC 11801, европейский стандарт EN50173, а также фирменный стандарт компании IBM.
- Стандарты определены для четырех типов кабеля: на основе неэкранированной витой пары, на основе экранированной витой пары, коаксиального и волоконно-оптического кабелей.
- Кабель на основе неэкранированной витой пары в зависимости от электрических и механических характеристик разделяется на 5 категорий. Кабели *категории* 1 применяются там, где требования к скорости передачи минимальны. Главная особенность кабелей *категории* 2 – способность передавать сигналы со спектром до 1 МГц. Кабели *категории* 3

широко распространены и предназначены как для передачи данных, так и для передачи голоса. Кабели *категории* 4 представляют собой несколько улучшенный вариант кабелей категории 3 и на практике используются редко. Кабели *категории* 5 были специально разработаны для поддержки высокоскоростных протоколов FDDI, Fast Ethernet, 100VG-AnyLAN, ATM и Gigabit Ethernet.

- Кабель на основе экранированной витой пары хорошо защищает передаваемые сигналы от внешних помех, а пользователей сетей – от вредного для здоровья излучения. Наличие заземляемого экрана удорожает кабель и усложняет его прокладку. Экранированный кабель применяется только для передачи данных. Основным стандартом, определяющим параметры экранированной витой пары, является фирменный стандарт IBM. В этом стандарте кабели делятся на типы: Type 1, Type 2,..., Type 9, из которых основным является кабель Type 1.
- Коаксиальные кабели существует в большом количестве вариантов: «толстый» коаксиальный кабель, различные разновидности «тонкого» коаксиального кабеля, которые обладают худшими механическими и электрическими характеристиками по сравнению с «толстым» коаксиальным кабелем, зато за счет своей гибкости более удобны при монтаже, сюда же относится телевизионный кабель.
- Волоконно-оптические кабели обладают отличными электромагнитными и механическими характеристиками, недостаток их состоит в сложности и высокой стоимости монтажных работ.

2.2. Методы передачи дискретных данных на физическом уровне

При передаче дискретных данных по каналам связи применяются два основных типа физического кодирования – на основе синусоидального несущего сигнала и на основе последовательности прямоугольных импульсов. Первый способ часто называется также *модуляцией* или *аналоговой модуляцией*, подчеркивая тот факт, что кодирование осуществляется за счет изменения параметров аналогового сигнала. Второй способ обычно называют *цифровым кодированием*. Эти способы отличаются шириной спектра результирующего сигнала и сложностью аппаратуры, необходимой для их реализации.

При использовании прямоугольных импульсов спектр результирующего сигнала получается весьма широким. Это не удивительно, если вспомнить, что спектр идеального импульса имеет бесконечную ширину. Применение синусоиды приводит к спектру гораздо меньшей ширины при той же скорости передачи информации. Однако для реализации синусоидальной модуляции требуется более сложная и дорогая аппаратура, чем для реализации прямоугольных импульсов.

В настоящее время все чаще данные, изначально имеющие аналоговую форму – речь, телевизионное изображение, – передаются по каналам связи в дискретном виде, то есть в виде последовательности единиц и нулей. Процесс представления аналоговой информации в дискретной форме называется *дискретной модуляцией*. Термины «модуляция» и «кодирование» часто используют как синонимы.

2.2.1. Аналоговая модуляция

Аналоговая модуляция применяется для передачи дискретных данных по каналам с узкой полосой частот, типичным представителем которых является *канал тональной частоты*, предоставляемый в распоряжение пользователям общественных телефонных сетей. Типичная амплитудно-частотная характеристика канала тональной частоты представлена на рис. 2.12. Этот канал передает частоты в диапазоне от 300 до 3400 Гц, таким образом, его полоса пропускания равна 3100 Гц. Хотя человеческий голос имеет гораздо более широкий спектр – примерно от 100 Гц до 10 кГц, – для приемлемого качества передачи речи диапазон в 3100 Гц является хорошим решением. Строгое ограничение полосы пропускания тонального канала связано с использованием аппаратуры уплотнения и коммутации каналов в телефонных сетях.

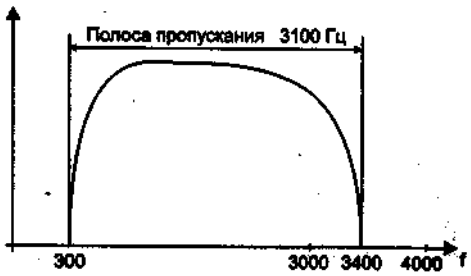


Рис. 2.12. Амплитудно-частотная характеристика канала тональной частоты

Устройство, которое выполняет функции модуляции несущей синусоиды на передающей стороне и демодуляции на приемной стороне, носит название *модем* (*модулятор-демодулятор*).

Методы аналоговой модуляции

Аналоговая модуляция является таким способом физического кодирования, при котором информация кодируется изменением амплитуды, частоты или фазы синусоидального сигнала несущей частоты. Основные способы аналоговой модуляции показаны на рис. 2.13. На диаграмме (рис. 2.13, а) показана последовательность бит исходной информации, представленная потенциалами высокого уровня для логической единицы и потенциалом нулевого уровня для логического нуля. Такой способ кодирования называется потенциальным кодом, который часто используется при передаче данных между блоками компьютера.

При *амплитудной модуляции* (рис. 2.13, *б*) для логической единицы выбирается один уровень амплитуды синусоиды несущей частоты, а для логического нуля – другой. Этот способ редко используется в чистом виде на практике из-за низкой помехоустойчивости, но часто применяется в сочетании с другим видом модуляции – фазовой модуляцией.

При *частотной модуляции* (рис. 2.13, *в*) значения 0 и 1 исходных данных передаются синусоидами с различной частотой – fo и fi. Этот способ модуляции не требует сложных схем в модемах и обычно применяется в низкоскоростных модемах, работающих на скоростях 300 или 1200 бит/с.

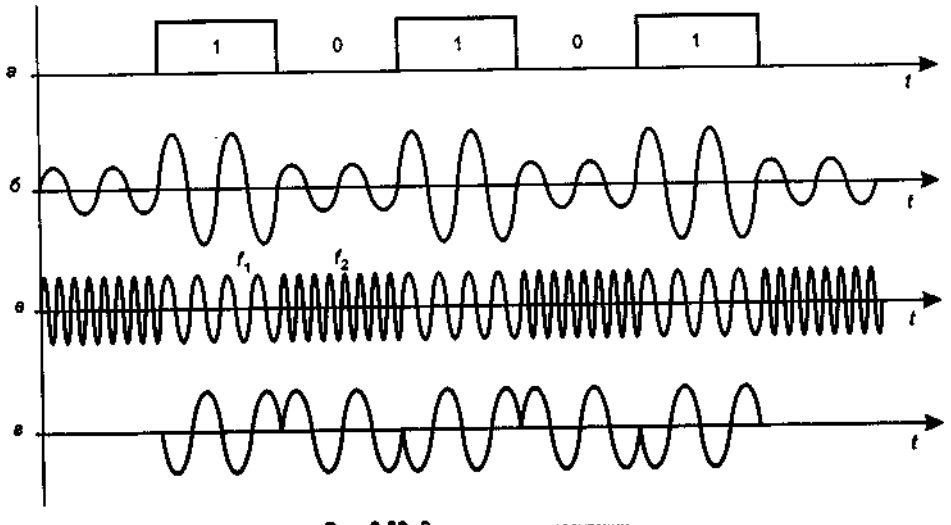


Рис. 2.13. Различные типы модуляции

При *фазовой модуляции* (рис. 2.13, *г*) значениям данных 0 и 1 соответствуют сигналы одинаковой частоты, но с различной фазой, например 0 и 180 градусов или 0,90,180 и 270 градусов.

В скоростных модемах часто используются комбинированные методы модуляции, как правило, амплитудная в сочетании с фазовой.

Спектр модулированного сигнала

Спектр результирующего модулированного сигнала зависит от типа модуляции и скорости

модуляции, то есть желаемой скорости передачи бит исходной информации.

Рассмотрим сначала спектр сигнала при потенциальном кодировании. Пусть логическая единица кодируется положительным потенциалом, а логический ноль – отрицательным потенциалом такой же величины. Для упрощения вычислений предположим, что передается информация, состоящая из бесконечной последовательности чередующихся единиц и нулей, как это и показано на рис. 2.13, *а*. Заметим, что в данном случае величины бод и бит в секунду совпадают.

Для потенциального кодирования спектр непосредственно получается из формул Фурье для периодической функции. Если дискретные данные передаются с битовой скоростью N бит/с, то спектр состоит из постоянной составляющей нулевой частоты и бесконечного ряда гармоник с частотами fo, 3fo, 5fo, 7fo,..., где fo = N/2. Амплитуды этих гармоник убывают достаточно медленно – с коэффициентами 1/3, 1/5, 1/7,... от амплитуды гармоники fo (рис. 2.14, *а*). В результате спектр потенциального кода требует для качественной передачи широкую полосу пропускания. Кроме того, нужно учесть, что реально спектр сигнала постоянно меняется в зависимости от того, какие данные передаются по линии связи. Например, передача длинной последовательности нулей или единиц сдвигает спектр в сторону низких частот, а в крайнем случае, когда передаваемые данные состоят только из единиц (или только из нулей), спектр состоит из гармоники нулевой частоты. При передаче чередующихся единиц и нулей постоянная составляющая отсутствует. Поэтому спектр результирующего сигнала потенциального кода при передаче произвольных данных занимает полосу от некоторой величины, близкой к 0 Гц, до примерно 7fo (гармониками с частотами выше 7fo можно пренебречь из-за их малого вклада в результирующий сигнал). Для канала тональной частоты верхняя граница при потенциальном кодировании достигается для скорости передачи данных в 971 бит/с, а нижняя неприемлема для любых скоростей, так как полоса пропускания канала начинается с 300 Гц. В результате потенциальные коды на каналах тональной частоты никогда не используются.

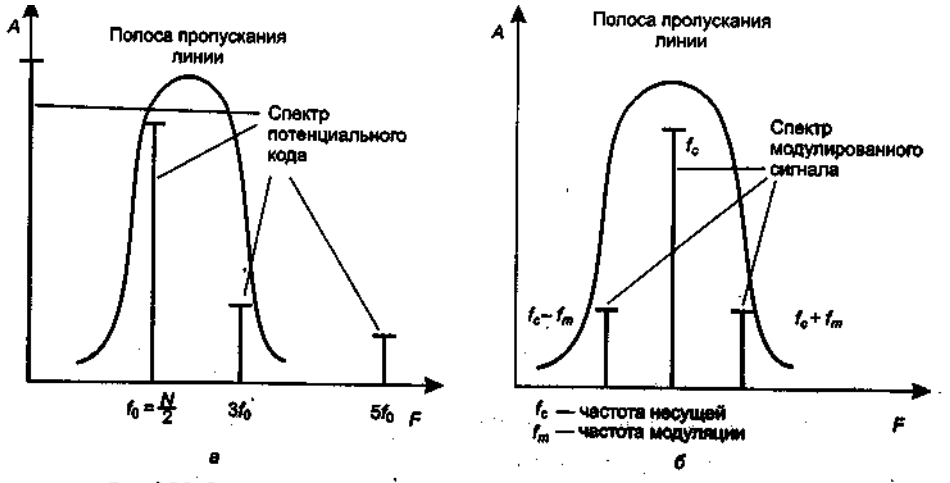


Рис. 2.14. Спектры сигналов при потенциальном кодировании и амплитудной модуляции

При амплитудной модуляции спектр состоит из синусоиды несущей частоты fc и двух боковых гармоник: (fc + fm) и (fc – fm), где fm – частота изменения информационного параметра синусоиды, которая совпадает со скоростью передачи данных при использовании двух уровней амплитуды (рис. 2.14,*б*). Частота fm определяет пропускную способность линии при данном способе кодирования. При небольшой частоте модуляции ширина спектра сигнала будет также небольшой (равной 2fm), поэтому сигналы не будут искажаться линией, если ее полоса пропускания будет больше или равна 2fm. Для канала тональной частоты такой способ модуляции приемлем при скорости передачи данных не больше 3100/2=1550 бит/с. Если же для представления данных используются 4 уровня амплитуды, то пропускная способность канала повышается до 3100 бит/с.

При фазовой и частотной модуляции спектр сигнала получается более сложным, чем при амплитудной модуляции, так как боковых гармоник здесь образуется более двух, но они также симметрично расположены относительно основной несущей частоты, а их амплитуды быстро убывают. Поэтому эти виды модуляции также хорошо подходят для передачи данных по каналу тональной частоты.

Для повышения скорости передачи данных используют комбинированные методы модуляции.

Наиболее распространенными являются методы *квадратурной амплитудной модуляции* (*Quadrature Amplitude Modulation, QAM*). Эти методы основаны на сочетании фазовой модуляции с 8 значениями величин сдвига фазы и амплитудной модуляции с 4 уровнями амплитуды. Однако из возможных 32 комбинаций сигнала используются далеко не все. Например, в кодах *Треллиса* допустимы всего 6, 7 или 8 комбинаций для представления исходных данных, а остальные комбинации являются запрещенными. Такая избыточность кодирования требуется для распознавания модемом ошибочных сигналов, являющихся следствием искажений из-за помех, которые на телефонных каналах, особенно коммутируемых, весьма значительны по амплитуде и продолжительны по времени.

2.2.2. Цифровое кодирование

При цифровом кодировании дискретной информации применяют потенциальные и импульсные коды.

В потенциальных кодах для представления логических единиц и нулей используется только значение потенциала сигнала, а его перепады, формирующие законченные импульсы, во внимание не принимаются. Импульсные коды позволяют представить двоичные данные либо импульсами определенной полярности, либо частью импульса – перепадом потенциала определенного направления.

Требования к методам цифрового кодирования

При использовании прямоугольных импульсов для передачи дискретной информации необходимо выбрать такой способ кодирования, который одновременно достигал бы нескольких целей:

- имел при одной и той же битовой скорости наименьшую ширину спектра результирующего сигнала;
- обеспечивал синхронизацию между передатчиком и приемником;
- обладал способностью распознавать ошибки;
- обладал низкой стоимостью реализации.

Более узкий спектр сигналов позволяет на одной и той же линии (с одной и той же полосой пропускания) добиваться более высокой скорости передачи данных. Кроме того, часто к спектру сигнала предъявляется требование отсутствия постоянной составляющей, то есть наличия постоянного тока между передатчиком и приемником. В частности, применение различных трансформаторных схем *гальванической развязки* препятствует прохождению постоянного тока.

Синхронизация передатчика и приемника нужна для того, чтобы приемник точно знал, в какой момент времени необходимо считывать новую информацию с линии связи. Эта проблема в сетях решается сложнее, чем при обмене данными между близко расположенными устройствами, например между блоками внутри компьютера или же между компьютером и принтером. На небольших расстояниях хорошо работает схема, основанная на отдельной тактирующей линии связи (рис. 2.15), так что информация снимается с линии данных только в момент прихода тактового импульса. В сетях использование этой схемы вызывает трудности из-за неоднородности характеристик проводников в кабелях. На больших расстояниях неравномерность скорости распространения сигнала может привести к тому, что тактовый импульс придет настолько позже или раньше соответствующего сигнала данных, что бит данных будет пропущен или считан повторно. Другой причиной, по которой в сетях отказываются от использования тактирующих импульсов, является экономия проводников в дорогостоящих кабелях.

Поэтому в сетях применяются так называемые *самосинхронизирующиеся коды*, сигналы которых несут для передатчика указания о том, в какой момент времени нужно осуществлять распознавание очередного бита (или нескольких бит, если код ориентирован более чем на два состояния сигнала). Любой резкий перепад сигнала – так называемый фронт – может служить хорошим указанием для синхронизации приемника с передатчиком.

При использовании синусоид в качестве несущего сигнала результирующий код обладает свойством самосинхронизации, так как изменение амплитуды несущей частоты дает возможность приемнику определить момент появления входного кода.

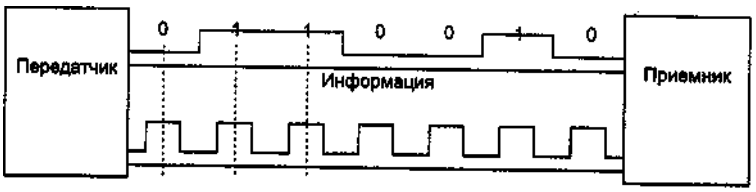


Рис. 2.15. Синхронизация приемника и передатчика на небольших расстояниях

Распознавание и коррекцию искаженных данных сложно осуществить средствами физического уровня, поэтому чаще всего эту работу берут на себя протоколы, лежащие выше: канальный, сетевой, транспортный или прикладной. С другой стороны, распознавание ошибок на физическом уровне экономит время, так как приемник не ждет полного помещения кадра в буфер, а отбраковывает его сразу при распознавании ошибочных бит внутри кадра.

Требования, предъявляемые к методам кодирования, являются взаимно противоречивыми, поэтому каждый из рассматриваемых ниже популярных методов цифрового кодирования обладает своими преимуществами и своими недостатками по сравнению с другими.

Потенциальный код без возвращения к нулю

На рис. 2.16, *а* показан уже упомянутый ранее метод потенциального кодирования, называемый также кодированием *без возвращения к нулю* (*Non Return to Zero, NRZ*). Последнее название отражает то обстоятельство, что при передаче последовательности единиц сигнал не возвращается к нулю в течение такта (как мы увидим ниже, в других методах кодирования возврат к нулю в этом случае происходит). Метод NRZ прост в реализации, обладает хорошей распознаваемостью ошибок (из-за двух резко отличающихся потенциалов), но не обладает свойством самосинхронизации. При передаче длинной последовательности единиц или нулей сигнал на линии не изменяется, поэтому приемник лишен возможности определять по входному сигналу моменты времени, когда нужно в очередной раз считывать данные. Даже при наличии высокоточного тактового генератора приемник может ошибиться с моментом съема данных, так как частоты двух генераторов никогда не бывают полностью идентичными. Поэтому при высоких скоростях обмена данными и длинных последовательностях единиц или нулей небольшое рассогласование тактовых частот может привести к ошибке в целый такт и, соответственно, считыванию некорректного значения бита.

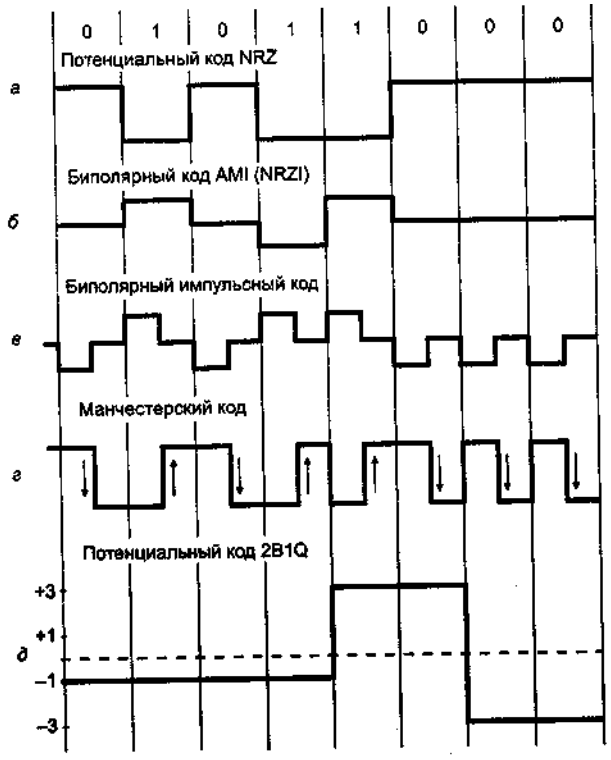


Рис. 2.16. Способы дискретного кодирования данных

Другим серьезным недостатком метода NRZ является наличие низкочастотной составляющей, которая приближается к нулю при передаче длинных последовательностей единиц или нулей. Из-за этого многие каналы связи, не обеспечивающие прямого гальванического соединения между приемником и источником, этот вид кодирования не поддерживают. В результате в чистом виде код NRZ в сетях не используется. Тем не менее используются его различные модификации, в которых устраняют как плохую самосинхронизацию кода NRZ, так и наличие постоянной составляющей. Привлекательность кода NRZ, из-за которой имеет смысл заняться его улучшением, состоит в достаточно низкой частоте основной гармоники fo, которая равна N/2 Гц, как это было показано в предыдущем разделе. У других методов кодирования, например манчестерского, основная гармоника имеет более высокую частоту.

Метод биполярного кодирования с альтернативной инверсией

Одной из модификаций метода NRZ является метод *биполярного кодирования с альтернативной инверсией* (*Bipolar Alternate Mark Inversion, AMI*). В этом методе (рис. 2.16, *б*) используются три уровня потенциала – отрицательный, нулевой и положительный. Для кодирования логического нуля используется нулевой потенциал, а логическая единица кодируется либо положительным потенциалом, либо отрицательным, при этом потенциал каждой новой единицы противоположен потенциалу предыдущей.

Код AMI частично ликвидирует проблемы постоянной составляющей и отсутствия самосинхронизации, присущие коду NRZ. Это происходит при передаче длинных последовательностей единиц. В этих случаях сигнал на линии представляет собой последовательность разнополярных импульсов с тем же спектром, что и у кода NRZ, передающего чередующиеся нули и единицы, то есть без постоянной составляющей и с основной гармоникой N/2 Гц (где N – битовая скорость передачи данных). Длинные же последовательности нулей также опасны для кода AMI, как и для кода NRZ – сигнал вырождается в постоянный потенциал нулевой амплитуды. Поэтому код AMI требует дальнейшего улучшения, хотя задача упрощается – осталось справиться только с последовательностями нулей.

В целом, для различных комбинаций бит на линии использование кода AMI приводит к более узкому спектру сигнала, чем для кода NRZ, а значит, и к более высокой пропускной способности линии. Например, при передаче чередующихся единиц и нулей основная гармоника fo имеет частоту N/4 Гц. Код AMI предоставляет также некоторые возможности по распознаванию ошибочных сигналов. Так, нарушение строгого чередования полярности сигналов говорит о ложном импульсе или исчезновении с линии корректного импульса. Сигнал с некорректной полярностью называется *запрещенным сигналом* (*signal violation*).

В коде AMI используются не два, а три уровня сигнала на линии. Дополнительный уровень требует увеличение мощности передатчика примерно на 3 дБ для обеспечения той же достоверности приема бит на линии, что является общим недостатком кодов с несколькими состояниями сигнала по сравнению с кодами, которые различают только два состояния.

Потенциальный код с инверсией при единице

Существует код, похожий на AMI, но только с двумя уровнями сигнала. При передаче нуля он передает потенциал, который был установлен в предыдущем такте (то есть не меняет его), а при передаче единицы потенциал инвертируется на противоположный. Этот код называется *потенциальным кодом с инверсией при единице* (*Non Return to Zero with ones Inverted, NRZI*). Этот код удобен в тех случаях, когда использование третьего уровня сигнала весьма нежелательно, например в оптических кабелях, где устойчиво распознаются два состояния сигнала – свет и темнота.

Для улучшения потенциальных кодов, подобных AMI и NRZI, используются два метода. Первый метод основан на добавлении в исходный код избыточных бит, содержащих логические единицы. Очевидно, что в этом случае длинные последовательности нулей прерываются и код становится самосинхронизирующимся для любых передаваемых данных. Исчезает также постоянная составляющая, а значит, еще более сужается спектр сигнала. Но этот метод снижает полезную пропускную способность линии, так как избыточные единицы пользовательской информации не несут. Другой метод основан на предварительном «перемешивании» исходной информации таким образом,

чтобы вероятность появления единиц и нулей на линии становилась близкой. Устройства, или блоки, выполняющие такую операцию, называются *скрэмблерами* (scramble – свалка, беспорядочная сборка). При скремблировании используется известный алгоритм, поэтому приемник, получив двоичные данные, передает их на *дескрэмблер*, который восстанавливает исходную последовательность бит. Избыточные биты при этом по линии не передаются. Оба метода относятся к логическому, а не физическому кодированию, так как форму сигналов на линии они не определяют. Более детально они изучаются в следующем разделе.

Биполярный импульсный код

Кроме потенциальных кодов в сетях используются и импульсные коды, когда данные представлены полным импульсом или же его частью – фронтом. Наиболее простым случаем такого подхода является *биполярный импульсный код*, в котором единица представлена импульсом одной полярности, а ноль – другой (рис. 2.16, *в*). Каждый импульс длится половину такта. Такой код обладает отличными самосинхронизирующими свойствами, но постоянная составляющая может присутствовать, например, при передаче длинной последовательности единиц или нулей. Кроме того, спектр у него шире, чем у потенциальных кодов. Так, при передаче всех нулей или единиц частота основной гармоники кода будет равна N Гц, что в два раза выше основной гармоники кода NRZ и в четыре раза выше основной гармоники кода AMI при передаче чередующихся единиц и нулей. Из-за слишком широкого спектра биполярный импульсный код используется редко.

Манчестерский код

В локальных сетях до недавнего времени самым распространенным методом кодирования был так называемый *манчестерский код* (рис. 2.16, *г*). Он применяется в технологиях Ethernet и Token Ring.

В манчестерском коде для кодирования единиц и нулей используется перепад потенциала, то есть фронт импульса. При манчестерском кодировании каждый такт делится на две части. Информация кодируется перепадами потенциала, происходящими в середине каждого такта. Единица кодируется перепадом от низкого уровня сигнала к высокому, а ноль – обратным перепадом. В начале каждого такта может происходить служебный перепад сигнала, если нужно представить несколько единиц или нулей подряд. Так как сигнал изменяется по крайней мере один раз за такт передачи одного бита данных, то манчестерский код обладает хорошими самосинхронизирующими свойствами. Полоса пропускания манчестерского кода уже, чем у биполярного импульсного. У него также нет постоянной составляющей, а основная гармоника в худшем случае (при передаче последовательности единиц или нулей) имеет частоту N Гц, а в лучшем (при передаче чередующихся единиц и нулей) она равна N/2 Гц, как и у кодов AMI или NRZ. В среднем ширина полосы манчестерского кода в полтора раза уже, чем у биполярного импульсного кода, а основная гармоника колеблется вблизи значения 3N/4. Манчестерский код имеет еще одно преимущество перед биполярным импульсным кодом. В последнем для передачи данных используются три уровня сигнала, а в манчестерском – два.

Потенциальный код 2B1Q

На рис. 2.16, *д* показан потенциальный код с четырьмя уровнями сигнала для кодирования данных. Это код *2B1Q*, название которого отражает его суть – каждые два бита (2B) передаются за один такт сигналом, имеющим четыре состояния (1Q). Паре бит 00 соответствует потенциал -2,5 В, паре бит 01 соответствует потенциал -0,833 В, паре 11 – потенциал +0,833 В, а паре 10 – потенциал +2,5 В. При этом способе кодирования требуются дополнительные меры по борьбе с длинными последовательностями одинаковых пар бит, так как при этом сигнал превращается в постоянную составляющую. При случайном чередовании бит спектр сигнала в два раза уже, чем у кода NRZ, так как при той же битовой скорости длительность такта увеличивается в два раза. Таким образом, с помощью кода 2B1Q. можно по одной и той же линии передавать данные в два раза быстрее, чем с помощью кода AMI или NRZI. Однако для его реализации мощность передатчика должна быть выше, чтобы четыре уровня четко различались приемником на фоне помех.

2.2.3. Логическое кодирование

Логическое кодирование используется для улучшения потенциальных кодов типа AMI, NRZI или 2Q1B. Логическое кодирование должно заменять длинные последовательности бит, приводящие к постоянному потенциалу, вкраплениями единиц. Как уже отмечалось выше, для логического кодирования характерны два метода – избыточные коды и скрэмблирование.

Избыточные коды

Избыточные коды основаны на разбиении исходной последовательности бит на порции, которые часто называют символами. Затем каждый исходный символ заменяется на новый, который имеет большее количество бит, чем исходный. Например, логический код 4B/5B, используемый в технологиях FDDI и Fast Ethernet, заменяет исходные символы длиной в 4 бита на символы длиной в 5 бит. Так как результирующие символы содержат избыточные биты, то общее количество битовых комбинаций в них больше, чем в исходных. Так, в коде 4B/5B результирующие символы могут содержать 32 битовых комбинации, в то время как исходные символы – только 16. Поэтому в результирующем коде можно отобрать 16 таких комбинаций, которые не содержат большого количества нулей, а остальные считать *запрещенными кодами* (*code violation*). Кроме устранения постоянной составляющей и придания коду свойства самосинхронизации, избыточные коды позволяют приемнику распознавать искаженные биты. Если приемник принимает запрещенный код, значит, на линии произошло искажение сигнала.

Соответствие исходных и результирующих кодов 4B/5B представлено ниже.

Исходный код	Результирующий код	Исходный код	Результирующий код
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 затем передается по линии с помощью физического кодирования по одному из методов потенциального кодирования, чувствительному только к длинным последовательностям нулей. Символы кода 4B/5B длиной 5 бит гарантируют, что при любом их сочетании на линии не могут встретиться более трех нулей подряд.

Буква B в названии кода означает, что элементарный сигнал имеет 2 состояния – от английского binary – двоичный. Имеются также коды и с тремя состояниями сигнала, например, в коде 8B/6T для кодирования 8 бит исходной информации используется код из 6 сигналов, каждый из которых имеет три состояния. Избыточность кода 8B/6T выше, чем кода 4B/5B, так как на 256 исходных кодов приходится 3^6=729 результирующих символов.

Использование таблицы перекодировки является очень простой операцией, поэтому этот подход не усложняет сетевые адаптеры и интерфейсные блоки коммутаторов и маршрутизаторов.

Для обеспечения заданной пропускной способности линии передатчик, использующий избыточный код, должен работать с повышенной тактовой частотой. Так, для передачи кодов 4B/5B со скоростью 100 Мб/с передатчик должен работать с тактовой частотой 125 МГц. При этом спектр сигнала на линии расширяется по сравнению со случаем, когда по линии передается чистый, не избыточный код. Тем не менее спектр избыточного потенциального кода оказывается уже спектра манчестерского кода, что оправдывает дополнительный этап логического кодирования, а также работу приемника и передатчика на повышенной тактовой частоте.

Скрэмблирование

Перемешивание данных скрэмблером перед передачей их в линию с помощью потенциального кода

является другим способом логического кодирования.

Методы скрэмблирования заключаются в побитном вычислении результирующего кода на основании бит исходного кода и полученных в предыдущих тактах бит результирующего кода. Например, скрэмблер может реализовывать следующее соотношение:

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

где B_i – двоичная цифра результирующего кода, полученная на i-м такте работы скрэмблера, A_i – двоичная цифра исходного кода, поступающая на i-м такте на вход скрэмблера, B_{i-3} и B_{i-5} – двоичные цифры результирующего кода, полученные на предыдущих тактах работы скрэмблера, соответственно на 3 и на 5 тактов ранее текущего такта, + – операция исключающего ИЛИ (сложение по модулю 2).

Например, для исходной последовательности 110110000001 скрэмблер даст следующий результирующий код:

$B_1 = A_1 = 1$ (первые три цифры результирующего кода будут совпадать с исходным, так как еще нет нужных предыдущих цифр)

$B_2 = A_2 = 1$

$B_3 = A_3 = 0$

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

$B_5 = A_5 \oplus B_2 = 1 \oplus 1 = 0$

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

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

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

$B_9 = A_9 \oplus B_6 \oplus B_4 = 0 \oplus 1 \oplus 0 = 1$

$B_{10} = A_{10} \oplus B_7 \oplus B_5 = 0 \oplus 1 \oplus 0 = 1$

$B_{11} = A_{11} \oplus B_8 \oplus B_6 = 0 \oplus 0 \oplus 1 = 1$

$B_{12} = A_{12} \oplus B_9 \oplus B_7 = 1 \oplus 1 \oplus 1 = 1$

Таким образом, на выходе скрэмблера появится последовательность 110001101111, в которой нет последовательности из шести нулей, присутствовавшей в исходном коде.

После получения результирующей последовательности приемник передает ее дескрэмблеру, который восстанавливает исходную последовательность на основании обратного соотношения:

$C_i = B_i \oplus B_{i-3} \oplus Bi\text{-}_5 = (A_i \oplus B_{i-3} \oplus B_{i-5}) \oplus B_{i-3} \oplus B_{i-5} = A_i$.

Различные алгоритмы скрэмблирования отличаются количеством слагаемых, дающих цифру результирующего кода, и сдвигом между слагаемыми. Так, в сетях ISDN при передаче данных от сети к абоненту используется преобразование со сдвигами в 5 и 23 позиции, а при передаче данных от абонента в сеть – со сдвигами 18 и 23 позиции.

Существуют и более простые методы борьбы с последовательностями единиц, также относимые к классу скрэмблирования.

Для улучшения кода Bipolar AMI используются два метода, основанные на искусственном искажении последовательности нулей запрещенными символами.

На рис. 2.17 показано использование метода B8ZS (Bipolar with 8-Zeros Substitution) и метода HDB3 (High-Density Bipolar 3-Zeros) для корректировки кода AMI. Исходный код состоит из двух длинных последовательностей нулей: в первом случае – из 8, а во втором – из 5.

Код B8ZS исправляет только последовательности, состоящие из 8 нулей. Для этого он после первых трех нулей вместо оставшихся пяти нулей вставляет пять цифр: V-1*-0-V-1*. V здесь обозначает сигнал единицы, запрещенной для данного такта полярности, то есть сигнал, не изменяющий полярность предыдущей единицы,1* – сигнал единицы корректной полярности, а знак звездочки отмечает тот факт, что в исходном коде в этом такте была не единица, а ноль. В результате на 8 тактах приемник наблюдает 2 искажения – очень маловероятно, что это случилось из-за шума на линии или других сбоев передачи. Поэтому приемник считает такие нарушения кодировкой 8 последовательных нулей и после приема заменяет их на исходные 8 нулей. Код B8ZS построен так, что его постоянная составляющая равна нулю при любых последовательностях двоичных цифр.

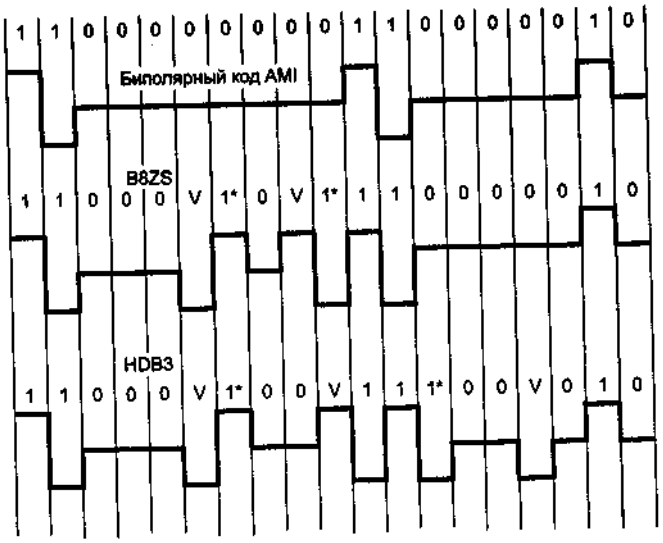


Рис. 2.17. Коды B8ZS и HDB3. V – сигнал единицы запрещенной полярности; 1* – сигнал единицы корректной полярности, но заменившей 0 в исходном коде

Код HDB3 исправляет любые четыре подряд идущих нуля в исходной последовательности. Правила формирования кода HDB3 более сложные, чем кода B8ZS. Каждые четыре нуля заменяются четырьмя сигналами, в которых имеется один сигнал V. Для подавления постоянной составляющей полярность сигнала V чередуется при последовательных заменах. Кроме того, для замены используются два образца четырехтактовых кодов. Если перед заменой исходный код содержал нечетное число единиц, то используется последовательность OOOV, а если число единиц было четным – последовательность 1*OOV.

Улучшенные потенциальные коды обладают достаточно узкой полосой пропускания для любых последовательностей единиц и нулей, которые встречаются в передаваемых данных. На рис. 2.18 приведены спектры сигналов разных кодов, полученные при передаче произвольных данных, в которых различные сочетания нулей и единиц в исходном коде равновероятны. При построении графиков спектр усреднялся по всем возможным наборам исходных последовательностей. Естественно, что результирующие коды могут иметь и другое распределение нулей и единиц. Из рис. 2.18 видно, что потенциальный код NRZ обладает хорошим спектром с одним недостатком – у него имеется постоянная составляющая. Коды, полученные из потенциального путем логического кодирования, обладают более узким спектром, чем манчестерский, даже при повышенной тактовой частоте (на рисунке спектр кода 4B/5B должен был бы примерно совпадать с кодом B8ZS, но он сдвинут в область более высоких частот, так как его тактовая частота повышена на 1/4 по сравнению с другими кодами). Этим объясняется применение потенциальных избыточных и скрэмблированных кодов в современных технологиях, подобных FDDI, Fast Ethernet, Gigabit Ethernet, ISDN и т. п. вместо манчестерского и биполярного импульсного кодирования.

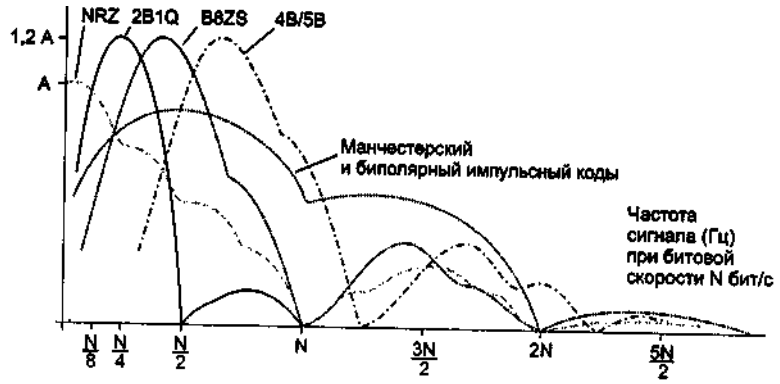


Рис. 2.18. Спектры потенциальных и импульсных кодов

2.2.4. Дискретная модуляция аналоговых сигналов

Одной из основных тенденций развития сетевых технологий является передача в одной сети как дискретных, так и аналоговых по своей природе данных. Источниками дискретных данных являются компьютеры и другие вычислительные устройства, а источниками аналоговых данных являются такие устройства, как телефоны, видеокамеры, звуко- и видеовоспроизводящая аппаратура. На ранних этапах решения этой проблемы в территориальных сетях все типы данных передавались в аналоговой форме, при этом дискретные по своему характеру компьютерные данные преобразовывались в аналоговую форму с помощью модемов.

Однако по мере развития техники съема и передачи аналоговых данных выяснилось, что передача их в аналоговой форме не позволяет улучшить качество принятых на другом конце линии данных, если они существенно исказились при передаче. Сам аналоговый сигнал не дает никаких указаний ни о том, что произошло искажение, ни о том, как его исправить, поскольку форма сигнала может быть любой, в том числе и такой, которую зафиксировал приемник. Улучшение же качества линий, особенно территориальных, требует огромных усилий и капиталовложений. Поэтому на смену аналоговой технике записи и передачи звука и изображения пришла цифровая техника. Эта техника использует так называемую дискретную модуляцию исходных непрерывных во времени аналоговых процессов.

Дискретные способы модуляции основаны на дискретизации непрерывных процессов как по амплитуде, так и по времени (рис. 2.19). Рассмотрим принципы искретной модуляции на примере *импульсно-кодовой модуляции, ИКМ* (*Pulse Amplitude Modulation, PAM*), которая широко применяется в цифровой телефонии.

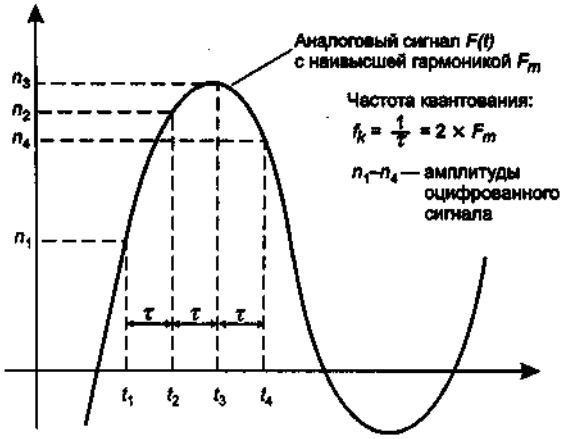


Рис. 2.19. Дискретная модуляция непрерывного процесса

Амплитуда исходной непрерывной функции измеряется с заданным периодом – за счет этого происходит дискретизация по времени. Затем каждый замер представляется в виде двоичного числа определенной разрядности, что означает дискретизацию по значениям функции – непрерывное множество возможных значений амплитуды заменяется дискретным множеством ее значений. Устройство, которое выполняет подобную функцию, называется *аналого-цифровым преобразователем* (*АЦП*). После этого замеры передаются по каналам связи в виде последовательности единиц и нулей. При этом применяются те же методы кодирования, что и в случае передачи изначально дискретной информации, то есть, например, методы, основанные на коде B8ZS или 2B1Q.

На приемной стороне линии коды преобразуются в исходную последовательность бит, а специальная аппаратура, называемая *цифро-аналоговым преобразователем* (*ЦАП*), производит демодуляцию оцифрованных амплитуд непрерывного сигнала, восстанавливая исходную непрерывную функцию времени.

Дискретная модуляции основана на *теории отображения Найквиста – Котельникова*. В соответствии с этой теорией, аналоговая непрерывная функция, переданная в виде последовательности ее дискретных по времени значений, может быть точно восстановлена, если частота дискретизации была в два или более раз выше, чем частота самой высокой гармоники спектра исходной функции.

Если это условие не соблюдается, то восстановленная функция будет существенно отличаться от исходной.

Преимуществом цифровых методов записи, воспроизведения и передачи аналоговой информации является возможность контроля достоверности считанных с носителя или полученных по линии связи данных. Для этого можно применять те же методы, которые применяются для компьютерных данных (и

рассматриваются более подробно далее), – вычисление контрольной суммы, повторная передача искаженных кадров, применение самокорректирующихся кодов.

Для качественной передачи голоса в методе ИКМ используется частота квантования амплитуды звуковых колебаний в 8000 Гц. Это связано с тем, что в аналоговой телефонии для передачи голоса был выбран диапазон от 300 до 3400 Гц, который достаточно качественно передает все основные гармоники собеседников. В соответствии с *теоремой Найквиста – Котельникова* для качественной передачи голоса достаточно выбрать частоту дискретизации, в два раза превышающую самую высокую гармонику непрерывного сигнала, то есть 2 × 3400 = 6800 Гц. Выбранная в действительности частота дискретизации 8000 Гц обеспечивает некоторый запас качества. В методе ИКМ обычно используется 7 или 8 бит кода для представления амплитуды одного замера. Соответственно это дает 127 или 256 градаций звукового сигнала, что оказывается вполне достаточным для качественной передачи голоса.

При использовании метода ИКМ для передачи одного голосового канала необходима пропускная способность 56 или 64 Кбит/с в зависимости от того, каким количеством бит представляется каждый замер. Если для этих целей используется 7 бит, то при частоте передачи замеров в 8000 Гц получаем:

8000 × 7 = 56000 бит/с или 56 Кбит/с;

а для случая 8-ми бит:

8000 × 8 = 64000 бит/с или 64 Кбит/с.

Стандартным является цифровой канал 64 Кбит/с, который также называется *элементарным каналом цифровых телефонных сетей*.

Передача непрерывного сигнала в дискретном виде требует от сетей жесткого соблюдения временного интервала в 125 мкс (соответствующего частоте дискретизации 8000 Гц) между соседними замерами, то есть требует синхронной передачи данных между узлами сети. При несоблюдении синхронности прибывающих замеров исходный сигнал восстанавливается неверно, что приводит к искажению голоса, изображения или другой мультимедийной информации, передаваемой по цифровым сетям. Так, искажение синхронизации в 10 мс может привести к эффекту «эха», а сдвиги между замерами в 200 мс приводят к потере распознаваемости произносимых слов. В то же время потеря одного замера при соблюдении синхронности между остальными замерами практически не сказывается на воспроизводимом звуке. Это происходит за счет сглаживающих устройств в цифро-аналоговых преобразователях, которые основаны на свойстве инерционности любого физического сигнала – амплитуда звуковых колебаний не может мгновенно измениться на бóльшую величину.

На качество сигнала после ЦАП влияет не только синхронность поступления на его вход замеров, но и погрешность дискретизации амплитуд этих замеров.

В теореме Найквиста – Котельникова предполагается, что амплитуды функции измеряются точно, в то же время использование для их хранения двоичных чисел с ограниченной разрядностью несколько искажает эти амплитуды. Соответственно искажается восстановленный непрерывный сигнал, что называется шумом дискретизации (по амплитуде).

Существуют и другие методы дискретной модуляции, позволяющие представить замеры голоса в более компактной форме, например в виде последовательности 4-битных или 2-битных чисел. При этом один голосовой канал требует меньшей пропускной способности, например 32 Кбит/с, 16 Кбит/с или еще меньше. С 1985 года применяется стандарт CCITT кодирования голоса, называемый Adaptive Differential Pulse Code Modulation (ADPCM). Коды ADPCM основаны на нахождении разностей между последовательными замерами голоса, которые затем и передаются по сети. В коде ADPCM для хранения одной разности используются 4 бит и голос передается со скоростью 32 Кбит/с. Более современный метод, Linear Predictive Coding (LPC), делает замеры исходной функции более редко, но использует методы прогнозирования направления изменения амплитуды сигнала. При помощи этого метода можно понизить скорость передачи голоса до 9600 бит/с.

Представленные в цифровой форме непрерывные данные легко можно передать через компьютерную сеть. Для этого достаточно поместить несколько замеров в кадр какой-нибудь стандартной сетевой технологии, снабдить кадр правильным адресом назначения и отправить адресату. Адресат должен извлечь из кадра замеры и подать их с частотой квантования (для голоса – с частотой 8000 Гц) на цифро-аналоговый преобразователь. По мере поступления следующих кадров с замерами голоса операция должна повториться. Если кадры будут прибывать достаточно синхронно, то качество голоса может быть достаточно высоким. Однако, как мы уже знаем, кадры в компьютерных сетях могут задерживаться как в конечных узлах (при ожидании доступа к разделяемой среде), так и в промежуточных коммуникационных устройствах – мостах, коммутаторах и маршрутизаторах. Поэтому

качество голоса при передаче в цифровой форме через компьютерные сети обычно бывает невысоким. Для качественной передачи оцифрованных непрерывных сигналов – голоса, изображения – сегодня используют специальные цифровые сети, такие как ISDN, ATM, и сети цифрового телевидения. Тем не менее для передачи внутрикорпоративных телефонных разговоров сегодня характерны сети frame relay, задержки передачи кадров которых укладываются в допустимые пределы.

2.2.5. Асинхронная и синхронная передачи

При обмене данными на физическом уровне единицей информации является бит, поэтому средства физического уровня всегда поддерживают побитовую синхронизацию между приемником и передатчиком.

Канальный уровень оперирует кадрами данных и обеспечивает синхронизацию между приемником и передатчиком на уровне кадров. В обязанности приемника входит распознавание начала первого байта кадра, распознавание границ полей кадра и распознавание признака окончания кадра.

Обычно достаточно обеспечить синхронизацию на указанных двух уровнях – битовом и кадровом, – чтобы передатчик и приемник смогли обеспечить устойчивый обмен информацией. Однако при плохом качестве линии связи (обычно это относится к телефонным коммутируемым каналам) для удешевления аппаратуры и повышения надежности передачи данных вводят дополнительные средства синхронизации на уровне байт.

Такой режим работы называется *асинхронным* или *старт-стопным*. Другой причиной использования такого режима работы является наличие устройств, которые генерируют байты данных в случайные моменты времени. Так работает клавиатура дисплея или другого терминального устройства, с которого человек вводит данные для обработки их компьютером.

В асинхронном режиме каждый байт данных сопровождается специальными сигналами «старт» и «стоп» (рис. 2.20, *а*). Назначение этих сигналов состоит в том, чтобы, во-первых, известить приемник о приходе данных и, во-вторых, чтобы дать приемнику достаточно времени для выполнения некоторых функций, связанных с синхронизацией, до поступления следующего байта. Сигнал «старт» имеет продолжительность в один тактовый интервал, а сигнал «стоп» может длиться один, полтора или два такта, поэтому говорят, что используется один, полтора или два бита в качестве стопового сигнала, хотя пользовательские биты эти сигналы не представляют.

Асинхронным описанный режим называется потому, что каждый байт может быть несколько смещен во времени относительно побитовых тактов предыдущего байта. Такая асинхронность передачи байт не влияет на корректность принимаемых данных, так как в начале каждого байта происходит дополнительная синхронизация приемника с источником за счет битов «старт». Более «свободные» временны́е допуски определяют низкую стоимость оборудования асинхронной системы.

При синхронном режиме передачи старт-стопные биты между каждой парой байт отсутствуют. Пользовательские данные собираются в кадр, который предваряется байтами синхронизации (рис. 2.20, *б*). Байт синхронизации - это байт, содержащий заранее известный код, например 0111110, который оповещает приемник о приходе кадра данных. При его получении приемник должен войти в байтовый синхронизм с передатчиком, то есть правильно понимать начало очередного байта кадра. Иногда применяется несколько синхробайт для обеспечения более надежной синхронизации приемника и передатчика. Так как при передаче длинного кадра у приемника могут появиться проблемы с синхронизацией бит, то в этом случае используются самосинхронизирующиеся коды.

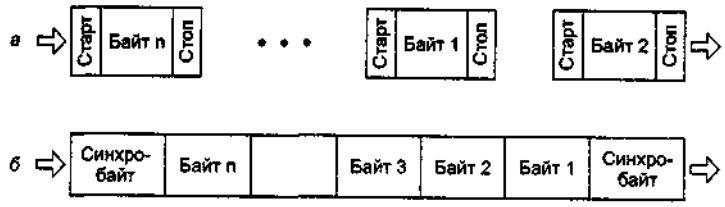


Рис. 2.20. Асинхронная (*а*) и синхронная (*б*) передачи на уровне байт

Выводы

- При передаче дискретных данных по узкополосному каналу тональной частоты,

используемому в телефонии, наиболее подходящими оказываются способы аналоговой модуляции, при которых несущая синусоида модулируется исходной последовательностью двоичных цифр. Эта операция осуществляется специальными устройствами – модемами.

- Для низкоскоростной передачи данных применяется изменение частоты несущей синусоиды. Более высокоскоростные модемы работают на комбинированных способах квадратурной амплитудной модуляции (QAM), для которой характерны 4 уровня амплитуды несущей синусоиды и 8 уровней фазы. Не все из возможных 32 сочетаний метода QAM используются для передачи данных, запрещенные сочетания позволяют распознавать искаженные данные на физическом уровне.
- На широкополосных каналах связи применяются потенциальные и импульсные методы кодирования, в которых данные представлены различными уровнями постоянного потенциала сигнала либо полярностями импульса или его фронта.
- При использовании потенциальных кодов особое значение приобретает задача синхронизации приемника с передатчиком, так как при передаче длинных последовательностей нулей или единиц сигнал на входе приемника не изменяется и приемнику сложно определить момент съема очередного бита данных.
- Наиболее простым потенциальным кодом является код без возвращения к нулю (NRZ), однако он не является самосинхронизирующимся и создает постоянную составляющую.
- Наиболее популярным импульсным кодом является манчестерский код, в котором информацию несет направление перепада сигнала в середине каждого такта. Манчестерский код применяется в технологиях Ethernet и Token Ring.
- Для улучшения свойств потенциального кода NRZ используются методы логического кодирования, исключающие длинные последовательности нулей. Эти методы основаны:
- на введении избыточных бит в исходные данные (коды типа 4B/5B);
- скрэмблировании исходных данных (коды типа 2B1Q).
- Улучшенные потенциальные коды обладают более узким спектром, чем импульсные, поэтому они находят применение в высокоскоростных технологиях, таких как FDDI, Fast Ethernet, Gigabit Ethernet.

2.3. Методы передачи данных канального уровня

Канальный уровень обеспечивает передачу пакетов данных, поступающих от протоколов верхних уровней, узлу назначения, адрес которого также указывает протокол верхнего уровня. Протоколы канального уровня оформляют переданные им пакеты в кадры собственного формата, помещая указанный адрес назначения в одно из полей такого кадра, а также сопровождая кадр контрольной суммой. Протокол канального уровня имеет локальный смысл, он предназначен для доставки кадров данных, как правило, в пределах сетей с простой топологией связей и однотипной или близкой технологией, например в односегментных сетях Ethernet или же в многосегментных сетях Ethernet и Token Ring иерархической топологии, разделенных только мостами и коммутаторами. Во всех этих конфигурациях адрес назначения имеет локальный смысл для данной сети и не изменяется при прохождении кадра от узла-источника к узлу назначения. Возможность передавать данные между локальными сетями разных технологий связана с тем, что в этих технологиях используются адреса одинакового формата, к тому же производители сетевых адаптеров обеспечивают уникальность адресов независимо от технологии.

Другой областью действия протоколов канального уровня являются связи типа «точка-точка» глобальных сетей, когда протокол канального уровня ответственен за доставку кадра непосредственному соседу. Адрес в этом случае не имеет принципиального значения, а на первый план выходит способность протокола восстанавливать искаженные и утерянные кадры, так как плохое качество территориальных каналов, особенно коммутируемых телефонных, часто требует выполнения подобных действий.

Если же перечисленные выше условия не соблюдаются, например связи между сегментами Ethernet имеют петлевидную структуру, либо объединяемые сети используют различные способы адресации, как это имеет место в сетях Ethernet и X.25, то протокол канального уровня не может в одиночку справиться с задачей передачи кадра между узлами и требует помощи протокола сетевого уровня.

Наиболее существенными характеристиками метода передачи, а значит, и протокола, работающего на канальном уровне, являются следующие:

- асинхронный/синхронный;
- символьно-ориентированный/бит-ориентированный;
- с предварительным установлением соединения/дейтаграммный;
- с обнаружением искаженных данных/без обнаружения;
- с обнаружением потерянных данных/без обнаружения;
- с восстановлением искаженных и потерянных данных/без восстановления;
- с поддержкой динамической компрессии данных/без поддержки.

Многие из этих свойств характерны не только для протоколов канального уровня, но и для протоколов более высоких уровней

.

2.3.1. Асинхронные протоколы

Асинхронные протоколы представляют собой наиболее старый способ связи. Эти протоколы оперируют не с кадрами, а с отдельными символами, которые представлены байтами со старт-стоповыми символами. Асинхронные протоколы ведут свое происхождение от тех времен, когда два человека связывались с помощью телетайпов по каналу «точка-точка». С развитием техники асинхронные протоколы стали применяться для связи телетайпов, разного рода клавиатур и дисплеев с вычислительными машинами. Единицей передаваемых данных был не кадр данных, а отдельный символ. Некоторые символы имели управляющий характер, например символ <CR> предписывал телетайпу или дисплею выполнить возврат каретки на начало строки. В этих протоколах существуют управляющие последовательности, обычно начинающиеся с символа <ESC>. Эти последовательности вызывали на управляемом устройстве достаточно сложные действия – например, загрузку нового шрифта на принтер.

В асинхронных протоколах применяются стандартные наборы символов, чаще всего ASCII или EBCDIC. Так как первые 32 или 27 кодов в этих наборах являются специальными кодами, которые не отображаются на дисплее или принтере, то они использовались асинхронными протоколами для управления режимом обмена данными. В самих пользовательских данных, которые представляли собой буквы, цифры, а также такие знаки, как @, %, $ и т. п., специальные символы никогда не встречались, так что проблемы их отделения от пользовательских данных не существовало.

Постепенно асинхронные протоколы усложнялись и стали наряду с отдельными символами использовать целые блоки данных, то есть кадры. Например, популярный протокол XMODEM передает файлы между двумя компьютерами по асинхронному модему. Начало приема очередного блока файла инициируется символьной командой – принимающая сторона постоянно передает символ ASCII NAK. Передающая сторона, приняв NAK, отправляет очередной блок файла, состоящий из 128 байт данных, заголовка и концевика. Заголовок состоит из специального символа SOH (Start Of Header) и номера блока. Концевик содержит контрольную сумму блока данных. Приемная сторона, получив новый блок, проверяла его номер и контрольную сумму. В случае совпадения этих параметров с ожидаемыми приемник отправлял символ АСК, а в противном случае – символ NAK, после чего передатчик должен был повторить передачу данного блока. В конце передачи файла передавался символ EOX.

Как видно из описания протокола XMODEM, часть управляющих операций выполнялась в асинхронных протоколах посылкой в асинхронном режиме отдельных символов, в то же время часть данных пересылалась блоками, что более характерно для синхронных протоколов.

2.3.2. Синхронные символьно-ориентированные и бит-ориентированные протоколы

В синхронных протоколах между пересылаемыми символами (байтами) нет стартовых и стоповых сигналов, поэтому отдельные символы в этих протоколах пересылать нельзя. Все обмены данными осуществляются кадрами, которые имеют в общем случае заголовок, поле данных и концевик (рис. 2.21). Все биты кадра передаются непрерывным синхронным потоком, что значительно ускоряет передачу данных.

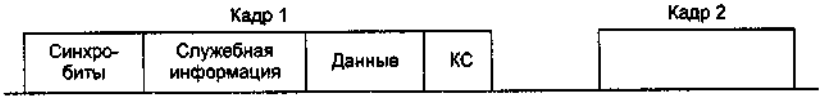


Рис. 2.21. Кадры синхронных протоколов

Так как байты в этих протоколах не отделяются друг от друга служебными сигналами, то одной из первых задач приемника является распознавание границы байт. Затем приемник должен найти начало и конец кадра, а также определить границы каждого поля кадра – адреса назначения, адреса источника, других служебных полей заголовка, поля данных и контрольной суммы, если она имеется.

Большинство протоколов допускает использование в кадре поля данных переменной длины. Иногда и заголовок может иметь переменную длину. Обычно протоколы определяют максимальное значение, которое может иметь длина поля данных. Эта величина называется *максимальной единицей передачи данных* (*Maximum Transfer Unit, MTU*). В некоторых протоколах задается также минимальное значение, которое может иметь длина поля данных. Например, протокол Ethernet требует, чтобы поле данных содержало по крайней мере 46 байт данных (если приложение хочет отправить меньшее количество байт, то оно обязано дополнить их до 46 байт любыми значениями). Другие протоколы разрешают использовать поле данных нулевой длины, например FDDI.

Существуют также протоколы с кадрами фиксированной длины, например, в протоколе ATM кадры фиксированного размера 53 байт, включая служебную информацию. Для таких протоколов необходимо решить только первую часть задачи – распознать начало кадра.

Синхронные протоколы канального уровня бывают двух типов: символьно-ориентированные (байт-ориентированные) и бит-ориентированные. Для обоих характерны одни и те же методы синхронизации бит. Главное различие между ними заключается в методе синхронизации символов и кадров.

Символьно-ориентированные протоколы

Символьно-ориентированные протоколы используются в основном для передачи блоков отображаемых символов, например текстовых файлов. Так как при синхронной передаче нет стоповых и стартовых битов, для синхронизации символов необходим другой метод. Синхронизация достигается за счет того, что передатчик добавляет два или более управляющих символа, называемых символами SYN, перед каждым блоком символов. В коде ASCII символ SYN имеет двоичное значение 0010110, это несимметричное относительно начала символа значение позволяет легко разграничивать отдельные символы SYN при их последовательном приеме. Символы SYN выполняют две функции: во-первых, они обеспечивают приемнику побитную синхронизацию, во-вторых, как только битовая синхронизация достигается, они позволяют приемнику начать распознавание границ символов SYN. После того как приемник начал отделять один символ от другого, можно задавать границы начала кадра с помощью другого специального символа. Обычно в символьных протоколах для этих целей используется символ STX (Start of TeXt, ASCII 0000010). Другой символ отмечает окончание кадра – ETX (End of TeXt, ASCII 0000011).

Однако такой простой способ выделения начала и конца кадра хорошо работал только в том случае, если внутри кадра не было символов STX и ETX. При подключении к компьютеру алфавитно-цифровых терминалов такая задача действительно не возникала. Тем не менее синхронные символьно-ориентированные протоколы позднее стали использоваться и для связи компьютера с компьютером, а в этом случае данные внутри кадра могут быть любые, если, например, между компьютерами передается программа. Наиболее популярным протоколом такого типа был протокол BSC компании IBM. Он работал в двух режимах – непрозрачном, в котором некоторые специальные символы внутри кадра запрещались, и прозрачном, в котором разрешалась передачи внутри кадра любых символов, в том числе и ETX. Прозрачность достигалась за счет того, что перед управляющими символами STX и ETX всегда вставлялся символ DLE (Data Link Escape). Такая процедура называется *стаффингом* символов (stuff – всякая всячина, заполнитель). А если в поле данных кадра встречалась последовательность DLE ETX, то передатчик удваивал символ DLE, то есть порождал последовательность DLE DLE ETX. Приемник, встретив подряд два символа DLE DLE, всегда удалял первый, но оставшийся DLE уже не рассматривал как начало управляющей последовательности, то есть оставшиеся символы DLE ETX считал просто пользовательскими данными.

Бит-ориентированные протоколы

Потребность в паре символов в начале и конце каждого кадра вместе с дополнительными символами DLE означает, что символьно-ориентированная передача не эффективна для передачи двоичных данных, так как приходится в поле данных кадра добавлять достаточно много избыточных данных. Кроме того, формат управляющих символов для разных кодировок различен, например, в коде ASCII символ SYN равен 0010110, а в коде EBCDIC - 00110010. Так что этот метод допустим только с определенным типом кодировки, даже если кадр содержит чисто двоичные данные. Чтобы преодолеть эти проблемы, сегодня почти всегда используется более универсальный метод, называемый бит-ориентированной передачей. Этот метод сейчас применяется при передаче как двоичных, так и символьных данных. На рис. 2.22 показаны 3 различные схемы бит-ориентированной передачи. Они отличаются способом обозначения начала и конца каждого кадра.

Первая схема, показанная на рис. 2.22, *а*, похожа на схему с символами STX и ETX в символьно-ориентированных протоколах. Начало и конец каждого кадра отмечается одной и той же 8-битовой последовательностью – 01111110, называемой флагом. Термин «бит-ориентированный» используется потому, что принимаемый поток бит сканируется приемником на побитовой основе для обнаружения стартового флага, а затем во время приема для обнаружения стопового флага. Поэтому длина кадра в этом случае не обязательно должна быть кратна 8 бит.

Чтобы обеспечить синхронизацию приемника, передатчик посылает последовательность байтов простоя (каждый состоит из 11111111), предшествующую стартовому флагу.

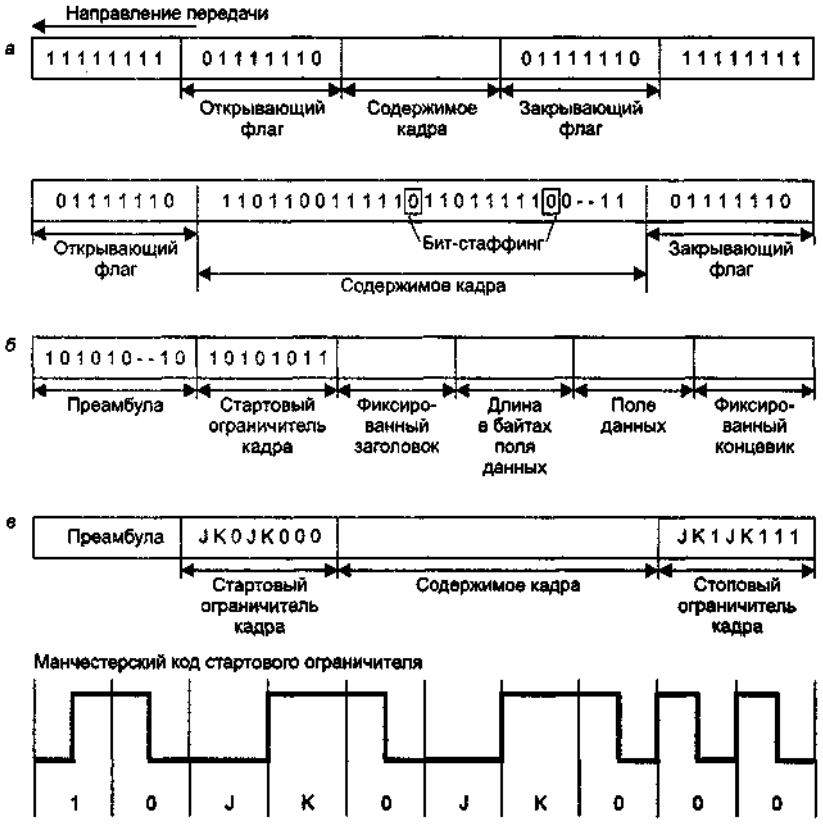


Рис. 2.22. Способы выделения начала и конца кадра при синхронной передаче

Для достижения прозрачности данных в этой схеме необходимо, чтобы флаг не присутствовал в поле данных кадра. Это достигается с помощью приема, известного как вставка 0 бита, – *бит-стаффинга*. Схема вставки бита работает только во время передачи поля данных кадра. Если эта схема обнаруживает, что подряд передано пять 1, то она автоматически вставляет дополнительный 0 (даже если после этих пяти 1 шел 0). Поэтому последовательность 01111110 никогда не появится в поле данных кадра. Аналогичная схема работает в приемнике и выполняет обратную функцию. Когда после пяти 1 обнаруживается 0, он автоматически удаляется из поля данных кадра. Бит-стаффинг гораздо более экономичен, чем байт-стаффинг, так как вместо лишнего байта вставляется один бит, следовательно, скорость передачи пользовательских данных в этом случае замедляется в меньшей степени.

Во второй схеме (см. рис. 2.22, *б*) для обозначения начала кадра имеется только стартовый флаг, а для определения конца кадра используется поле длины кадра, которое при фиксированных размерах заголовка и концевика чаще всего имеет смысл длины поля данных кадра. Эта схема наиболее применима в локальных сетях. В этих сетях для обозначения факта незанятости среды в исходном состоянии по среде вообще не передается никаких символов. Чтобы все остальные станции вошли в битовую синхронизацию, посылающая станция предваряет содержимое кадра последовательностью бит, известной как преамбула, которая состоит из чередования единиц и нулей 101010... Войдя в битовую синхронизацию, приемник исследует входной поток на побитовой основе, пока не обнаружит байт начала кадра 10101011, который выполняет роль символа STX. За этим байтом следует заголовок кадра, в котором в определенном месте находится поле длины поля данных. Таким образом, в этой схеме приемник просто отсчитывает заданное количество байт, чтобы определить окончание кадра.

Третья схема (см. рис. 2.22, *в*) использует для обозначения начала и конца кадра флаги, которые включают запрещенные для данного кода сигналы (code violations, V). Например, при манчестерском кодировании вместо обязательного изменения полярности сигнала в середине тактового интервала уровень сигнала остается неизменным и низким (запрещенный сигнал J) или неизменным и высоким (запрещенный сигнал К). Начало кадра отмечается последовательностью JK0JK000, а конец - последовательностью JK1JK100. Этот способ очень экономичен, так как не требует ни бит-стаффинга, ни поля длины, но его недостаток заключается в зависимости от принятого метода физического кодирования. При использовании избыточных кодов роль сигналов J и К играют запрещенные символы, например, в коде 4В/5В этими символами являются коды 11000 и 10001.

Каждая из трех схем имеет свои преимущества и недостатки. Флаги позволяют отказаться от специального дополнительного поля, но требуют специальных мер: либо по разрешению размещения флага в поле данных за счет бит-стаффинга, либо по использованию в качестве флага запрещенных сигналов, что делает эту схему зависимой от способа кодирования.

Протоколы с гибким форматом кадра

Для большей части протоколов характерны кадры, состоящие из служебных полей фиксированной длины. Исключение делается только для поля данных, с целью экономной пересылки как небольших квитанций, так и больших файлов. Способ определения окончания кадра путем задания длины поля данных, рассмотренный выше, как раз рассчитан на такие кадры с фиксированной структурой и фиксированными размерами служебных полей.

Однако существует ряд протоколов, в которых кадры имеют гибкую .структуру. Например, к таким протоколам относятся очень популярный прикладной протокол управления сетями SNMP, а также протокол канального уровня PPP, используемый для соединений типа «точка-точка». Кадры таких протоколов состоят из неопределенного количества полей, каждое из которых может иметь переменную

длину. Начало такого кадра отмечается некоторым стандартным образом, например с помощью флага, а затем протокол последовательно просматривает поля кадра и определяет их количество и размеры. Каждое поле обычно описывается двумя дополнительными полями фиксированного размера. Например, если в кадре встречается поле, содержащее некоторую символьную строку, то в кадр вставляются три поля.

Тип	Длина	Значение
string	б	public

Дополнительные поля «Тип» и «Длина» имеют фиксированный размер в один байт, поэтому протокол легко находит границы поля «Значение». Так как количество таких полей также неизвестно, для определения общей длины кадра используется либо общее поле «Длина», которое помещается в начале кадра и относится ко всем полям данных, либо закрывающий флаг.

2.3.3. Передача с установлением соединения и без установления соединения

При передаче кадров данных на канальном уровне используются как дейтаграммные процедуры, работающие без установления соединения (*connectionless*), так и процедуры с предварительным

установлением логического соединения (*connection-oriented*).

При дейтаграммной передаче кадр посылается в сеть «без предупреждения», и никакой ответственности за его утерю протокол не несет (рис. 2.23, *а*). Предполагается, что сеть всегда готова принять кадр от конечного узла. Дейтаграммный метод работает быстро, так как никаких предварительных действий перед отправкой данных не выполняется. Однако при таком методе трудно организовать в рамках протокола отслеживание факта доставки кадра узлу назначения. Этот метод не гарантирует доставку пакета.

Передача с установлением соединения более надежна, но требует больше времени для передачи данных и вычислительных затрат от конечных узлов.

В этом случае узлу-получателю отправляется служебный кадр специального формата с предложением установить соединение (рис. 2.23, *б*). Если узел-получатель согласен с этим, то он посылает в ответ другой служебный кадр, подтверждающий установление соединения и предлагающий для данного логического соединения некоторые параметры, например идентификатор соединения, максимальное значение поля данных кадров, которые будут использоваться в рамках данного соединения, и т. п. Узел-инициатор соединения может завершить процесс установления соединения отправкой третьего служебного кадра, в котором сообщит, что предложенные параметры ему подходят. На этом логическое соединение считается установленным, и в его рамках можно передавать информационные кадры с пользовательскими данными. После передачи некоторого законченного набора данных, например определенного файла, узел инициирует разрыв данного логического соединения, посылая соответствующий служебный кадр.

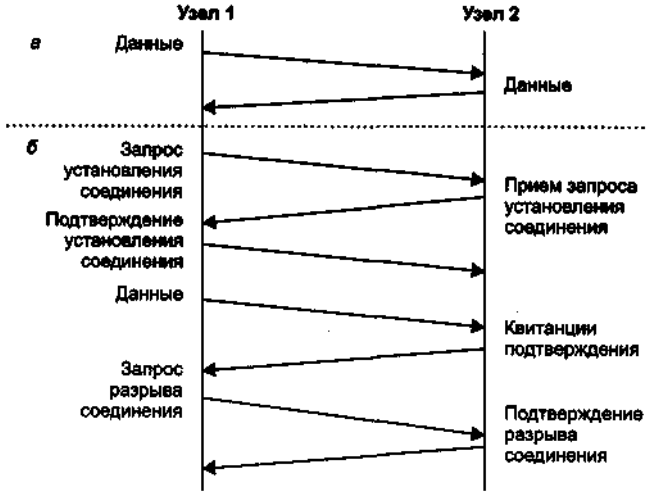


Рис. 2.23. Протоколы без установления соединения (*а*) и с установлением соединения (*б*)

Заметим, что, в отличие от протоколов дейтаграммного типа, которые поддерживают только один тип кадра – информационный, протоколы, работающие по процедуре с установлением соединения, должны поддерживать несколько типов кадров – служебные, для установления (и разрыва) соединения, и информационные, переносящие собственно пользовательские данные.

Логическое соединение обеспечивает передачу данных как в одном направлении – от инициатора соединения, так и в обоих направлениях.

Процедура установления соединения может использоваться для достижения различных целей.

- Для взаимной аутентификации либо пользователей, либо оборудования (маршрутизаторы тоже могут иметь имена и пароли, которые нужны для уверенности в том, что злоумышленник не подменил корпоративный маршрутизатор и не отвел поток данных в свою сеть для анализа).
- Для согласования изменяемых параметров протокола: MTU, различных тайм-аутов и т. п.
- Для обнаружения и коррекции ошибок. Установление логического соединения дает точку отсчета для задания начальных значений номеров кадров. При потере нумерованного кадра приемник, во-первых, получает возможность обнаружить этот факт, а во-вторых, он может сообщить передатчику, какой в точности кадр нужно передать повторно.
- В некоторых технологиях процедуру установления логического соединения используют при

динамической настройке коммутаторов сети для маршрутизации всех последующих кадров, которые будут проходить через сеть в рамках данного логического соединения. Так работают сети технологий X.25, frame relay и ATM.

Как видно из приведенного списка, при установлении соединения могут преследоваться разные цели, в некоторых случаях – несколько одновременно. В этой главе мы рассмотрим использование логического соединения для обнаружения и коррекции ошибок, а остальные случаи будут рассматриваться в последующих главах по мере необходимости.

2.3.4. Обнаружение и коррекция ошибок

Канальный уровень должен обнаруживать ошибки передачи данных, связанные с искажением бит в принятом кадре данных или с потерей кадра, и по возможности их корректировать.

Большая часть протоколов канального уровня выполняет только первую задачу – обнаружение ошибок, считая, что корректировать ошибки, то есть повторно передавать данные, содержавшие искаженную информацию, должны протоколы верхних уровней. Так работают такие популярные протоколы локальных сетей, как Ethernet, Token Ring, FDDI и другие. Однако существуют протоколы канального уровня, например LLC2 или LAP-B, которые самостоятельно решают задачу восстановления искаженных или потерянных кадров.

Очевидно, что протоколы должны работать наиболее эффективно в типичных условиях работы сети. Поэтому для сетей, в которых искажения и потери кадров являются очень редкими событиями, разрабатываются протоколы типа Ethernet, в которых не предусматриваются процедуры устранения ошибок. Действительно, наличие процедур восстановления данных потребовало бы от конечных узлов дополнительных вычислительных затрат, которые в условиях надежной работы сети являлись бы избыточными.

Напротив, если в сети искажения и потери случаются часто, то желательно уже на канальном уровне использовать протокол с коррекцией ошибок, а не оставлять эту работу протоколам верхних уровней. Протоколы верхних уровней, например транспортного или прикладного, работая с большими тайм-аутами, восстановят потерянные данные с большой задержкой. В глобальных сетях первых поколений, например сетях X.25, которые работали через ненадежные каналы связи, протоколы канального уровня всегда выполняли процедуры восстановления потерянных и искаженных кадров.

Поэтому нельзя считать, что один протокол лучше другого потому, что он восстанавливает ошибочные кадры, а другой протокол – нет. Каждый протокол должен работать в тех условиях, для которых он разработан.

Методы обнаружения ошибок

Все методы обнаружения ошибок основаны на передаче в составе кадра данных служебной избыточной информации, по которой можно судить с некоторой степенью вероятности о достоверности принятых данных. Эту служебную информацию принято называть *контрольной суммой* (*или последовательностью контроля кадра – Frame Check Sequence, FCS*). Контрольная сумма вычисляется как функция от основной информации, причем необязательно только путем суммирования. Принимающая сторона повторно вычисляет контрольную сумму кадра по известному алгоритму и в случае ее совпадения с контрольной суммой, вычисленной передающей стороной, делает вывод о том, что данные были переданы через сеть корректно.

Существует несколько распространенных алгоритмов вычисления контрольной суммы, отличающихся вычислительной сложностью и способностью обнаруживать ошибки в данных.

Контроль по паритету представляет собой наиболее простой метод контроля данных. В то же время это наименее мощный алгоритм контроля, так как с его помощью можно обнаружить только одиночные ошибки в проверяемых данных. Метод заключается в суммировании по модулю 2 всех бит контролируемой информации. Например, для данных 100101011 результатом контрольного суммирования будет значение 1. Результат суммирования также представляет собой один бит данных, который пересылается вместе с контролируемой информацией. При искажении при пересылке любого одного бита исходных данных (или контрольного разряда) результат суммирования будет отличаться от принятого контрольного разряда, что говорит об ошибке. Однако двойная ошибка, например 110101010, будет неверно принята за корректные данные. Поэтому контроль по паритету применяется к небольшим

порциям данных, как правило, к каждому байту, что дает коэффициент избыточности для этого метода 1/8. Метод редко применяется в вычислительных сетях из-за его большой избыточности и невысоких диагностических способностей.

Вертикальный и горизонтальный контроль по паритету представляет собой модификацию описанного выше метода. Его отличие состоит в том, что исходные данные рассматриваются в виде матрицы, строки которой составляют байты данных. Контрольный разряд подсчитывается отдельно для каждой строки и для каждого столбца матрицы. Этот метод обнаруживает большую часть двойных ошибок, однако обладает еще большей избыточностью. На практике сейчас также почти не применяется.

Циклический избыточный контроль (*Cyclic Redundancy Check, CRC*) является в настоящее время наиболее популярным методом контроля в вычислительных сетях (и не только в сетях, например, этот метод широко применяется при записи данных на диски и дискеты). Метод основан на рассмотрении исходных данных в виде одного многоразрядного двоичного числа. Например, кадр стандарта Ethernet, состоящий из 1024 байт, будет рассматриваться как одно число, состоящее из 8192 бит. В качестве контрольной информации рассматривается остаток от деления этого числа на известный делитель R. Обычно в качестве делителя выбирается семнадцати- или тридцати трехразрядное число, чтобы остаток от деления имел длину 16 разрядов (2 байт) или 32 разряда (4 байт). При получении кадра данных снова вычисляется остаток от деления на тот же делитель R, но при этом к данным кадра добавляется и содержащаяся в нем контрольная сумма. Если остаток от деления на R равен нулю*, то делается вывод об отсутствии ошибок в полученном кадре, в противном случае кадр считается искаженным.

* Существует несколько модифицированная процедура вычисления остатка, приводящая к получению в случае отсутствия ошибок известного ненулевого остатка, что является более надежным показателем корректности.

Этот метод обладает более высокой вычислительной сложностью, но его диагностические возможности гораздо выше, чем у методов контроля по паритету. Метод CRC обнаруживает все одиночные ошибки, двойные ошибки и ошибки в нечетном числе бит. Метод обладает также невысокой степенью избыточности. Например, для кадра Ethernet размером в 1024 байт контрольная информация длиной в 4 байт составляет только 0,4 %.

Методы восстановления искаженных и потерянных кадров

Методы коррекции ошибок в вычислительных сетях основаны на повторной передаче кадра данных в том случае, если кадр теряется и не доходит до адресата или приемник обнаружил в нем искажение информации. Чтобы убедиться в необходимости повторной передачи данных, отправитель нумерует отправляемые кадры и для каждого кадра ожидает от приемника так называемой *положительной квитанции* – служебного кадра, извещающего о том, что исходный кадр был получен и данные в нем оказались корректными. Время этого ожидания ограничено – при отправке каждого кадра передатчик запускает таймер, и, если по его истечении положительная квитанция на получена, кадр считается утерянным. Приемник в случае получения кадра с искаженными данными может отправить *отрицательную квитанцию* – явное указание на то, что данный кадр нужно передать повторно.

Существуют два подхода к организации процесса обмена квитанциями: с простоями и с организацией «окна».

Метод с простоями (*Idle Source*) требует, чтобы источник, пославший кадр, ожидал получения квитанции (положительной или отрицательной) от приемника и только после этого посылал следующий кадр (или повторял искаженный). Если же квитанция не приходит в течение тайм-аута, то кадр (или квитанция) считается утерянным и его передача повторяется. На рис. 2.24, *а* видно, что в этом случае производительность обмена данными существенно снижается, – хотя передатчик и мог бы послать следующий кадр сразу же после отправки предыдущего, он обязан ждать прихода квитанции. Снижение производительности этого метода коррекции особенно заметно на низкоскоростных каналах связи, то есть в территориальных сетях.

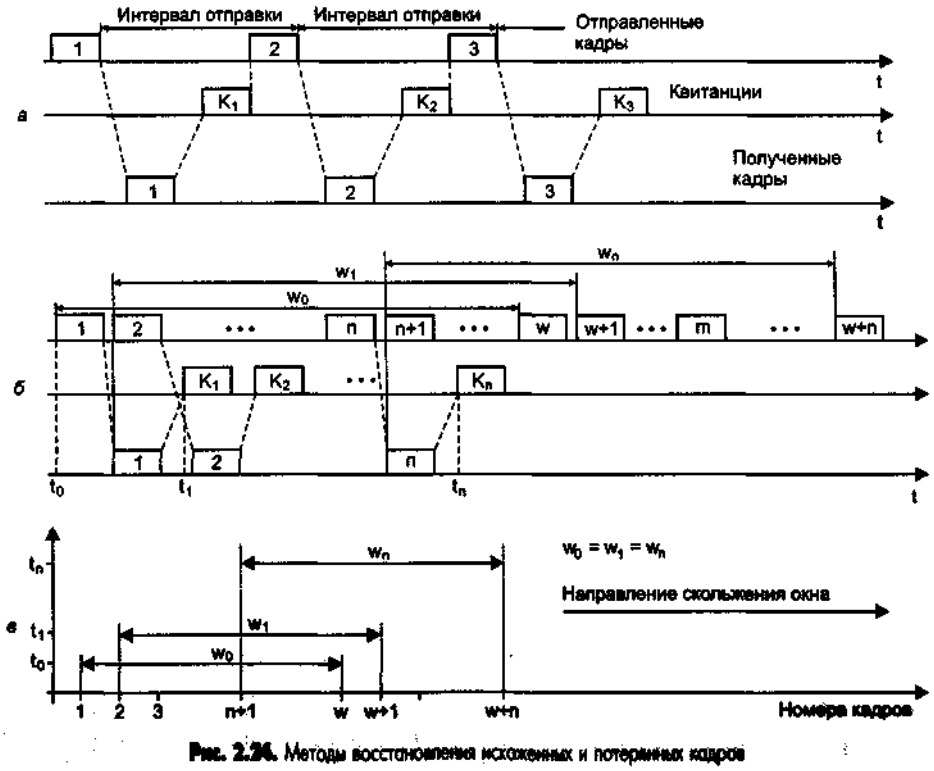


Рис. 2.24. Методы восстановления искаженных и потерянных кадров

Второй метод называется методом *«скользящего окна»* (*sliding window*). В этом методе для повышения коэффициента использования линии источнику разрешается передать некоторое количество кадров в непрерывном режиме, то есть в максимально возможном для источника темпе, без получения на эти кадры положительных ответных квитанций. (Далее, где это не искажает существо рассматриваемого вопроса, положительные квитанции для краткости будут называться просто «квитанциями».) Количество кадров, которые разрешается передавать таким образом, называется размером окна. Рисунок 2.24, *б* иллюстрирует данный метод для окна размером в W кадров.

В начальный момент, когда еще не послано ни одного кадра, окно определяет диапазон кадров с номерами от 1 до W включительно. Источник начинает передавать кадры и получать в ответ квитанции. Для простоты предположим, что квитанции поступают в той же последовательности, что и кадры, которым они соответствуют. В момент t_1 при получении первой квитанции К1 окно сдвигается на одну позицию, определяя новый диапазон от 2 до (W+1).

Процессы отправки кадров и получения квитанций идут достаточно независимо друг от друга. Рассмотрим произвольный момент времени t_n, когда источник получил квитанцию на кадр с номером n. Окно сдвинулось вправо и определило диапазон разрешенных к передаче кадров от (n+1) до (W+n). Все множество кадров, выходящих из источника, можно разделить на перечисленные ниже группы (рис. 2.24,*б*).

- *Кадры с номерами от 1 до n* уже были отправлены и квитанции на них получены, то есть они находятся за пределами окна слева.
- *Кадры, начиная с номера (n+1) и кончая номером (W+n),* находятся в пределах окна и потому могут быть отправлены не дожидаясь прихода какой-либо квитанции. Этот диапазон может быть разделен еще на два поддиапазона:
- кадры с номерами от (n+1) до m, которые уже отправлены, но квитанции на них еще не получены;
- кадры с номерами от m до (W+n), которые пока не отправлены, хотя запрета на это нет.
- *Все кадры с номерами, большими или равными (W+n+1),* находятся за пределами окна справа и поэтому пока не могут быть отправлены.

Перемещение окна вдоль последовательности номеров кадров показано на рис. 2.24, *в*. Здесь t_0 – исходный момент, t_1 и t_n – моменты прихода квитанций на первый и n-й кадр соответственно. Каждый раз, когда приходит квитанция, окно сдвигается влево, но его размер при этом не меняется и остается равным W. Заметим, что хотя в данном примере размер окна в процессе передачи остается постоянным, в реальных протоколах (например, TCP) можно встретить варианты данного алгоритма с изменяющимся размером окна.

Итак, при отправке кадра с номером n источнику разрешается передать еще W-1 кадров до получения квитанции на кадр n, так что в сеть последним уйдет кадр с номером (W+n-1). Если же за это время квитанция на кадр n так и не пришла, то процесс передачи приостанавливается, и по истечении некоторого тайм-аута кадр n (или квитанция на него) считается утерянным, и он передается снова.

Если же поток квитанций поступает более-менее регулярно, в пределах допуска в W кадров, то скорость обмена достигает максимально возможной величины для данного канала и принятого протокола.

Метод скользящего окна более сложен в реализации, чем метод с простоями, так как передатчик должен хранить в буфере все кадры, на которые пока не получены положительные квитанции. Кроме того, требуется отслеживать несколько параметров алгоритма: размер окна W, номер кадра, на который получена квитанция, номер кадра, который еще можно передать до получения новой квитанции.

Приемник может не посылать квитанции на каждый принятый корректный кадр. Если несколько кадров пришли почти одновременно, то приемник может послать квитанцию только на последний кадр. При этом подразумевается, что все предыдущие кадры также дошли благополучно.

Некоторые методы используют отрицательные квитанции. Отрицательные квитанции бывают двух типов – групповые и избирательные. Групповая квитанция содержит номер кадра, начиная с которого нужно повторить передачу всех кадров, отправленных передатчиком в сеть. Избирательная отрицательная квитанция требует повторной передачи только одного кадра.

Метод скользящего окна реализован во многих протоколах: LLC2, LAP-B, X.25, TCP, Novell NCP Burst Mode.

Метод с простоями является частным случаем метода скользящего окна, когда размер окна равен единице.

Метод скользящего окна имеет два параметра, которые могут заметно влиять на эффективность передачи данных между передатчиком и приемником, – размер окна и величина тайм-аута ожидания квитанции. В надежных сетях, когда кадры искажаются и теряются редко, для повышения скорости обмена данными размер окна нужно увеличивать, так как при этом передатчик будет посылать кадры с меньшими паузами. В ненадежных сетях размер окна следует уменьшать, так как при частых потерях и искажениях кадров резко возрастает объем вторично передаваемых через сеть кадров, а значит, пропускная способность сети будет расходоваться во многом вхолостую – полезная пропускная способность сети будет падать.

Выбор тайм-аута зависит не от надежности сети, а от задержек передачи кадров сетью.

Во многих реализациях метода скользящего окна величина окна и тайм-аут выбираются адаптивно, в зависимости от текущего состояния сети.

2.3.5. Компрессия данных

Компрессия (сжатие) данных применяется для сокращения времени их передачи. Так как на компрессию данных передающая сторона тратит дополнительное время, к которому нужно еще прибавить аналогичные затраты времени на декомпрессию этих данных принимающей стороной, то выгоды от сокращения времени на передачу сжатых данных обычно бывают заметны только для низкоскоростных каналов. Этот порог скорости для современной аппаратуры составляет около 64 Кбит/с. Многие программные и аппаратные средства сети способны выполнять *динамическую компрессию* данных в отличие от статической, когда данные предварительно компрессируются (например, с помощью популярных архиваторов типа WinZip), а уже затем отсылаются в сеть.

На практике может использоваться ряд алгоритмов компрессии, каждый из которых применим к определенному типу данных. Некоторые модемы (называемые интеллектуальными) предлагают *адаптивную компрессию*, при которой в зависимости от передаваемых данных выбирается определениый алгоритм компрессии. Рассмотрим некоторые из общих алгоритмов компрессии данных.

Десятичная упаковка. Когда данные состоят только из чисел, значительную экономию можно получить путем уменьшения количества используемых на цифру бит с 7 до 4, используя простое двоичное кодирование десятичных цифр вместо кода ASCII. Просмотр таблицы ASCII показывает, что старшие три бита всех кодов десятичных цифр содержат комбинацию 011. Если все данные в кадре информации состоят из десятичных цифр, то, поместив в заголовок кадра соответствующий управляющий символ, можно существенно сократить длину кадра.

Относительное кодирование. Альтернативой десятичной упаковке при передаче числовых данных с

небольшими отклонениями между последовательными цифрами является передача только этих отклонений вместе с известным опорным значением. Такой метод используется, в частности, в рассмотренном выше методе цифрового кодирования голоса ADPCM, передающем в каждом такте только разницу между соседними замерами голоса.

Символьное подавление. Часто передаваемые данные содержат большое количество повторяющихся байт. Например, при передаче черно-белого изображения черные поверхности будут порождать большое количество нулевых значений, а максимально освещенные участки изображения – большое количество байт, состоящих из всех единиц. Передатчик сканирует последовательность передаваемых байт и, если обнаруживает последовательность из трех или более одинаковых байт, заменяет ее специальной трехбайтовой последовательностью, в которой указывает значение байта, количество его повторений, а также отмечает начало этой последовательности специальным управляющим символом.

Коды переменной длины. В этом методе кодирования используется тот факт, что не все символы в передаваемом кадре встречаются с одинаковой частотой. Поэтому во многих схемах кодирования коды часто встречающихся символов заменяют кодами меньшей длины, а редко встречающихся – кодами большей длины. Такое кодирование называется также статистическим кодированием. Из-за того, что символы имеют различную длину, для передачи кадра возможна только бит-ориентированная передача.

При *статистическом кодировании* коды выбираются таким образом, чтобы при анализе последовательности бит можно было бы однозначно определить соответствие определенной порции бит тому или иному символу или же запрещенной комбинации бит. Если данная последовательность бит представляет собой запрещенную комбинацию, то необходимо к ней добавить еще один бит и повторить анализ. Например, если при неравномерном кодировании для наиболее часто встречающегося символа «Р» выбран код 1, состоящий из одного бита, то значение 0 однобитного кода будет запрещенным. Иначе мы сможем закодировать только два символа. Для другого часто встречающегося символа «О» можно использовать код 01, а код 00 оставить как запрещенный. Тогда для символа «А» можно выбрать код 001, для символа «П» – код 0001 и т. п.

Вообще, неравномерное кодирование наиболее эффективно, когда неравномерность распределения частот передаваемых символов достаточна велика, как при передаче длинных текстовых строк. Напротив, при передаче двоичных данных, например кодов программ, оно малоэффективно, так как 8-битовые коды при этом распределены почти равномерно.

Одним из наиболее распространенных алгоритмов, на основе которых строятся неравномерные коды, является алгоритм Хафмана, позволяющий строить коды автоматически, на основании известных частот символов. Существуют адаптивные модификации метода Хафмана, которые позволяют строить дерево кодов «на ходу», по мере поступления данных от источника.

Многие модели коммуникационного оборудования, такие как модемы, мосты, коммутаторы и маршрутизаторы, поддерживают протоколы динамической компрессии, позволяющие сократить объем передаваемой информации в 4, а иногда и в 8 раз. В таких случаях говорят, что протокол обеспечивает коэффициент сжатия 1:4 или 1:8. Существуют стандартные протоколы компрессии, например V.42bis, а также большое количество нестандартных, фирменных протоколов. Реальный коэффициент компрессии зависит от типа передаваемых данных, так, графические и текстовые данные обычно сжимаются хорошо, а коды программ – хуже.

Выводы

- Основной задачей протоколов канального уровня является доставка кадра узлу назначения в сети определенной технологии и достаточно простой топологии.
- Асинхронные протоколы разрабатывались для обмена данными между низкоскоростными старт-стопными устройствами: телетайпами, алфавитно-цифровыми терминалами и т. п. В этих протоколах для управления обменом данными используются не кадры, а отдельные символы из нижней части кодовых таблиц ASCII или EBCDIC. Пользовательские данные могут оформляться в кадры, но байты в таких кадрах всегда отделяются друг от друга стартовыми и стоповыми сигналами.
- Синхронные протоколы посылают кадры как для отправки пользовательских данных, так и для управления обменом.
- В зависимости от способа выделения начала и конца кадра синхронные протоколы делятся на символьно-ориентированные и бит-ориентированные. В первых для этой цели используются

символы кодов ASCII или EBCDIC, а в последних – специальный набор бит, называемый флагом. Бит-ориентированные протоколы более рационально расходуют поле данных кадра, так как для исключения из него значения, совпадающего с флагом, добавляют к нему только один дополнительный бит, а символьно-ориентированные протоколы добавляют целый символ.

- В дейтаграммных протоколах отсутствует процедура предварительного установления соединения, и за счет этого срочные данные отправляются в сеть без задержек.
- Протоколы с установлением соединения могут обладать многими дополнительными свойствами, отсутствующими у дейтаграммных протоколов. Наиболее часто в них реализуется такое свойство, как способность восстанавливать искаженные и потерянные кадры.
- Для обнаружения искажений наиболее популярны методы, основанные на циклических избыточных кодах (CRC), которые выявляют многократные ошибки.
- Для восстановления кадров используется метод повторной передачи на основе квитанций. Этот метод работает по алгоритму с простоями источника, а также по алгоритму скользящего окна.
- Для повышения полезной скорости передачи данных в сетях применяется динамическая компрессия данных на основе различных алгоритмов. Коэффициент сжатия зависит от типа данных и применяемого алгоритма и может колебаться в пределах от 1:2 до 1:8.

2.4. Методы коммутации

Любые сети связи поддерживают некоторый способ коммутации своих абонентов между собой. Этими абонентами могут быть удаленные компьютеры, локальные сети, факс-аппараты или просто собеседники, общающиеся с помощью телефонных аппаратов. Практически невозможно предоставить каждой паре взаимодействующих абонентов свою собственную некоммутируемую физическую линию связи, которой они могли бы монопольно «владеть» в течение длительного времени. Поэтому в любой сети всегда применяется какой-либо способ коммутации абонентов, который обеспечивает доступность имеющихся физических каналов одновременно для нескольких сеансов связи между абонентами сети. На рис. 2.25 показана типичная структура сети с коммутацией абонентов.

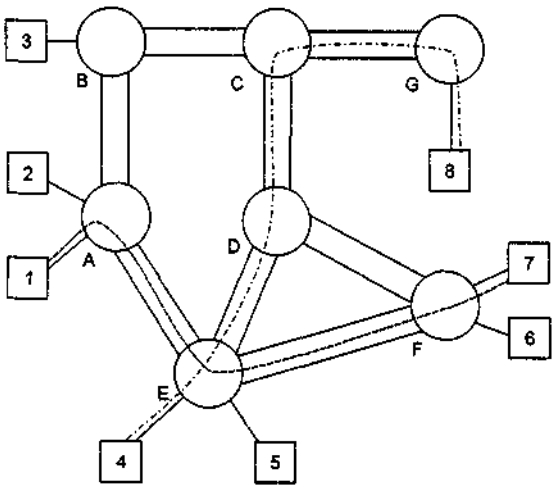


Рис. 2.25. Общая структура сети с коммутацией абонентов

Абоненты соединяются с коммутаторами индивидуальными линиями связи, каждая из которых используется в любой момент времени только одним, закрепленным за этой линией абонентом. Между коммутаторами линии связи разделяются несколькими абонентами, то есть используются совместно.

Существуют три принципиально различные схемы коммутации абонентов в сетях: *коммутация каналов* (*circuit switching*), *коммутация пакетов* (*packet switching*) и *коммутация сообщений* (*message switching*). Внешне все эти схемы соответствуют приведенной на рис. 2.25 структуре сети, однако возможности и свойства их различны. Сети с коммутацией каналов имеют более богатую историю, они ведут свое происхождение от первых телефонных сетей. Сети с коммутацией пакетов сравнительно

молоды, они появились в конце 60-х годов как результат экспериментов с первыми глобальными компьютерными сетями. Сети с коммутацией сообщений послужили прототипом современных сетей с коммутацией пакетов и сегодня они в чистом виде практически не существуют.

Каждая из этих схем имеет свои преимущества и недостатки, но по долгосрочным прогнозам многих специалистов будущее принадлежит технологии коммутации пакетов, как более гибкой и универсальной.

Как сети с коммутацией пакетов, так и сети с коммутацией каналов можно разделить на два класса по другому признаку – на сети с *динамической коммутацией* и сети с *постоянной коммутацией*.

В первом случае сеть разрешает устанавливать соединение по инициативе пользователя сети. Коммутация выполняется на время сеанса связи, а затем (опять же по инициативе одного из взаимодействующих пользователей) связь разрывается. В общем случае любой пользователь сети может соединиться с любым другим пользователем сети. Обычно период соединения между парой пользователей при динамической коммутации составляет от нескольких секунд до нескольких часов и завершается при выполнении определенной работы – передачи файла, просмотра страницы текста или изображения и т. п.

Во втором случае сеть не предоставляет пользователю возможность выполнить динамическую коммутацию с другим произвольным пользователем сети. Вместо этого сеть разрешает паре пользователей заказать соединение на длительный период времени. Соединение устанавливается не пользователями, а персоналом, обслуживающим сеть. Время, на которое устанавливается постоянная коммутация, измеряется обычно несколькими месяцами. Режим постоянной коммутации в сетях с коммутацией каналов часто называется сервисом *выделенных* (*dedicated*) или *арендуемых* (*leased*) *каналов*.

Примерами сетей, поддерживающих режим динамической коммутации, являются телефонные сети общего пользования, локальные сети, сети TCP/IP.

Наиболее популярными сетями, работающими в режиме постоянной коммутации, сегодня являются сети технологии SDH, на основе которых строятся выделенные каналы связи с пропускной способностью в несколько гигабит в секунду.

Некоторые типы сетей поддерживают оба режима работы. Например, сети X.25 и ATM могут предоставлять пользователю возможность динамически связаться с любым другим пользователем сети и в то же время отправлять данные по постоянному соединению одному вполне определенному абоненту.

2.4.1. Коммутация каналов

Коммутация каналов подразумевает образование непрерывного составного физического канала из последовательно соединенных отдельных канальных участков для прямой передачи данных между узлами. Отдельные каналы соединяются между собой специальной аппаратурой – коммутаторами, которые могут устанавливать связи между любыми конечными узлами сети. В сети с коммутацией каналов перед передачей данных всегда необходимо выполнить процедуру установления соединения, в процессе которой и создается составной канал.

Например, если сеть, изображенная на рис. 2.25, работает по технологии коммутации каналов, то узел 1, чтобы передать данные узлу 7, прежде всего должен передать специальный запрос на установление соединения коммутатору A, указав адрес назначения 7. Коммутатор A должен выбрать маршрут образования составного канала, а затем передать запрос следующему коммутатору, в данном случае E. Затем коммутатор E передает запрос коммутатору F, а тот, в свою очередь, передает запрос узлу 7. Если узел 7 принимает запрос на установление соединения, он направляет по уже установленному каналу ответ исходному узлу, после чего составной канал считается скоммутированным и узлы 1 и 7 могут обмениваться по нему данными, например вести телефонный разговор.

Коммутаторы, а также соединяющие их каналы должны обеспечивать одновременную передачу данных нескольких абонентских каналов. Для этого они должны быть высокоскоростными и поддерживать какую-либо технику мультиплексирования абонентских каналов.

В настоящее время для мультиплексирования абонентских каналов используются две техники:

- техника частотного мультиплексирования (Frequency Division Multiplexing, FDM);
- техника мультиплексирования с разделением времени (Time Division Multiplexing, TDM).

Коммутация каналов на основе частотного мультиплексирования

Техника частотного мультиплексирования каналов (FDM) была разработана для телефонных сетей, но применяется она и для других видов сетей, например сетей кабельного телевидения.

Рассмотрим особенности этого вида мультиплексирования на примере телефонной сети.

Речевые сигналы имеют спектр шириной примерно в 10 000 Гц, однако основные гармоники укладываются в диапазон от 300 до 3400 Гц. Поэтому для качественной передачи речи достаточно образовать между двумя собеседниками канал с полосой пропускания в 3100 Гц, который и используется в телефонных сетях для соединения двух абонентов. В то же время полоса пропускания кабельных систем с промежуточными усилителями, соединяющих телефонные коммутаторы между собой, обычно составляет сотни килогерц, а иногда и сотни мегагерц. Однако непосредственно передавать сигналы нескольких абонентских каналов по широкополосному каналу невозможно, так как все они работают в одном и том же диапазоне частот и сигналы разных абонентов смешаются между собой так, что разделить их будет невозможно.

Для разделения абонентских каналов характерна техника модуляции высокочастотного несущего синусоидального сигнала низкочастотным речевым сигналом (рис. 2.26). Эта техника подобна технике аналоговой модуляции при передаче дискретных сигналов модемами, только вместо дискретного исходного сигнала используются непрерывные сигналы, порождаемые звуковыми колебаниями. В результате спектр модулированного сигнала переносится в другой диапазон, который симметрично располагается относительно несущей частоты и имеет ширину, приблизительно совпадающую с шириной модулирующего сигнала.

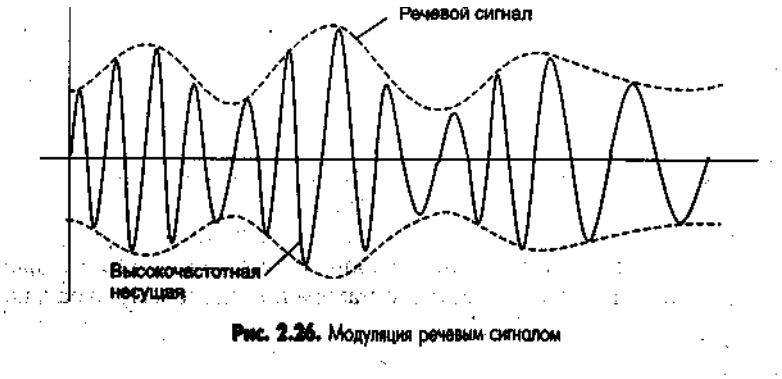


Рис. 2.26. Модуляция речевым сигналом

Если сигналы каждого абонентского канала перенести в свой собственный диапазон частот, то в одном широкополосном канале можно одновременно передавать сигналы нескольких абонентских каналов.

На входы FDM-коммутатора поступают исходные сигналы от абонентов телефонной сети. Коммутатор выполняет перенос частоты каждого канала в свой диапазон частот. Обычно высокочастотный диапазон делится на полосы, которые отводятся для передачи данных абонентских каналов (рис. 2.27). Чтобы низкочастотные составляющие сигналов разных каналов не смешивались между собой, полосы делают шириной в 4 кГц, а не в 3,1 кГц, оставляя между ними страховой промежуток в 900 Гц. В канале между двумя FDM-коммутаторами одновременно передаются сигналы всех абонентских каналов, но каждый из них занимает свою полосу частот. Такой канал называют *уплотненным*.

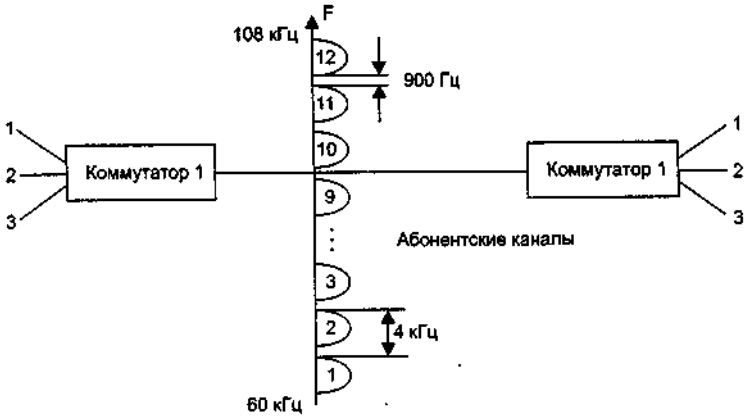


Рис. 2.27. Коммутация на основе частотного уплотнения

Выходной FDM-коммутатор выделяет модулированные сигналы каждой несущей частоты и передает их на соответствующий выходной канал, к которому непосредственно подключен абонентский телефон.

В сетях на основе FDM-коммутации принято несколько уровней иерархии уплотненных каналов. Первый уровень уплотнения образуют 12 абонентских каналов, которые составляют *базовую группу* каналов, занимающую полосу частот шириной в 48 кГц с границами от 60 до 108 кГц. Второй уровень уплотнения образуют 5 базовых групп, которые составляют *супергруппу*, с полосой частот шириной в 240 кГц и границами от 312 до 552 кГц. Супергруппа передает данные 60 абонентских каналов тональной частоты. Десять супергрупп образуют *главную группу*, которая используется для связи между коммутаторами на больших расстояниях. Главная группа передает данные 600 абонентов одновременно и требует от канала связи полосу пропускания шириной не менее 2520 кГц с границами от 564 до 3084 кГц.

Коммутаторы FDM могут выполнять как динамическую, так и постоянную коммутацию. При динамической коммутации один абонент инициирует соединение с другим абонентом, посылая в сеть номер вызываемого абонента. Коммутатор динамически выделяет данному абоненту одну из свободных полос своего уплотненного канала. При постоянной коммутации за абонентом полоса в 4 кГц закрепляется на длительный срок путем настройки коммутатора по отдельному входу, недоступному пользователям.

Принцип коммутации на основе разделения частот остается неизменным и в сетях другого вида, меняются только границы полос, выделяемых отдельному абонентскому каналу, а также количество низкоскоростных каналов в уплотненном высокоскоростном.

Коммутация каналов на основе разделения времени

Коммутация на основе техники разделения частот разрабатывалась в расчете на передачу непрерывных сигналов, представляющих голос. При переходе к цифровой форме представления голоса была разработана новая техника мультиплексирования, ориентирующаяся на дискретный характер передаваемых данных.

Эта техника носит название *мультиплексирования с разделением времени* (*Time Division Multiplexing, TDM*). Реже используется и другое ее название – техника *синхронного режима передачи* (*Synchronous Transfer Mode, STM*). Рисунок 2.28 поясняет принцип коммутации каналов на основе техники TDM.

Аппаратура TDM-сетей – мультиплексоры, коммутаторы, демультиплексоры – работает в режиме разделения времени, поочередно обслуживая в течение цикла своей работы все абонентские каналы. Цикл работы оборудования TDM равен 125 мкс, что соответствует периоду следования замеров голоса в цифровом абонентском канале. Это значит, что мультиплексор или коммутатор успевает вовремя обслужить любой абонентский канал и передать его очередной замер далее по сети. Каждому соединению выделяется один квант времени цикла работы аппаратуры, называемый также тайм-слотом. Длительность тайм-слота зависит от числа абонентских каналов, обслуживаемых мультиплексором TDM или коммутатором.

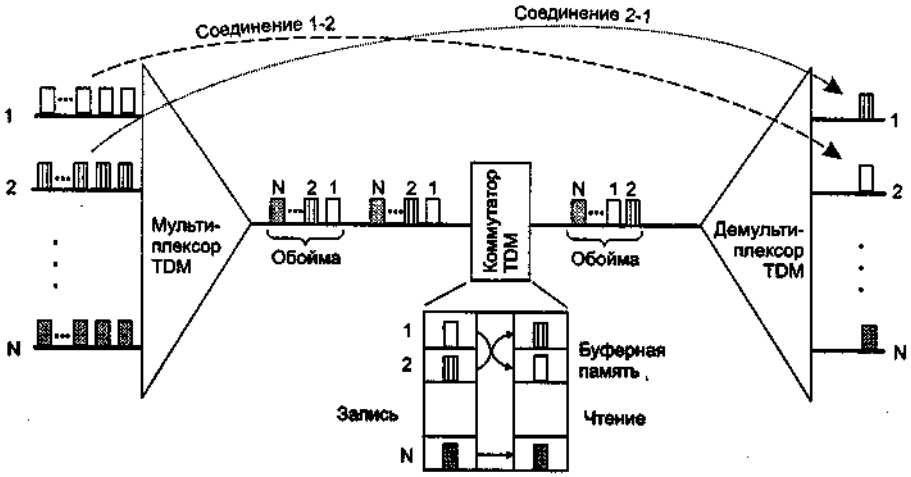


Рис. 2.28. Коммутация на основе разделения канала во времени

Мультиплексор принимает информацию по N входным каналам от конечных абонентов, каждый из которых передает данные по абонентскому каналу со скоростью 64 Кбит/с – 1 байт каждые 125 мкс. В каждом цикле мультиплексор выполняет следующие действия:

- прием от каждого канала очередного байта данных;
- составление из принятых байтов уплотненного кадра, называемого также обоймой;
- передача уплотненного кадра на выходной канал с битовой скоростью, равной Nx64 Кбит/с.

Порядок байт в обойме соответствует номеру входного канала, от которого этот байт получен. Количество обслуживаемых мультиплексором абонентских каналов зависит от его быстродействия. Например, мультиплексор T1, представляющий собой первый промышленный мультиплексор, работавший по технологии TDM, поддерживает 24 входных абонентских канала, создавая на выходе обоймы стандарта T1, передаваемые с битовой скоростью 1,544 Мбит/с.

Демультиплексор выполняет обратную задачу – он разбирает байты уплотненного кадра и распределяет их по своим нескольким выходным каналам, при этом он считает, что порядковый номер байта в обойме соответствует номеру выходного канала.

Коммутатор принимает уплотненный кадр по скоростному каналу от мультиплексора и записывает каждый байт из него в отдельную ячейку своей буферной памяти, причем в том порядке, в котором эти байты были упакованы в уплотненный кадр. Для выполнения операции коммутации байты извлекаются из буферной памяти не в порядке поступления, а в таком порядке, который соответствует поддерживаемым в сети соединениям абонентов. Так, например, если первый абонент левой части сети рис. 2.28 должен соединиться со вторым абонентом в правой части сети, то байт, записанный в первую ячейку буферной памяти, будет извлекаться из нее вторым. «Перемешивая» нужным образом байты в обойме, коммутатор обеспечивает соединение конечных абонентов в сети.

Однажды выделенный номер тайм-слота остается в распоряжении соединения «входной канал–выходной слот» в течение всего времени существования этого соединения, даже если передаваемый трафик является пульсирующим и не всегда требует захваченного количества тайм-слотов. Это означает, что соединение в сети TDM всегда обладает известной и фиксированной пропускной способностью, кратной 64 Кбит/с.

Работа оборудования TDM напоминает работу сетей с коммутацией пакетов, так как каждый байт данных можно считать некоторым элементарным пакетом. Однако, в отличие от пакета компьютерной сети, «пакет» сети TDM не имеет индивидуального адреса. Его адресом является порядковый номер в обойме или номер выделенного тайм-слота в мультиплексоре или коммутаторе. Сети, использующие технику TDM, требуют синхронной работы всего оборудования, что и определило второе название этой техники – синхронный режим передач (STM). Нарушение синхронности разрушает требуемую коммутацию абонентов, так как при этом теряется адресная информация. Поэтому перераспределение тайм-слотов между различными каналами в оборудовании TDM невозможно, даже если в каком-то цикле работы мультиплексора тайм-слот одного из каналов оказывается избыточным, так как на входе этого канала в этот момент нет данных для передачи (например, абонент телефонной сети молчит).

Существует модификация техники TDM, называемая *статистическим разделением канала во времени* (*Statistical TDM, STDM*). Эта техника разработана специально для того, чтобы с помощью временно свободных тайм-слотов одного канала можно было увеличить пропускную способность

остальных. Для решения этой задачи каждый байт данных дополняется полем адреса небольшой длины, например в 4 или 5 бит, что позволяет мультиплексировать 16 или 32 канала. Однако техника STDM не нашла широкого применения и используется в основном в нестандартном оборудовании подключения терминалов к мэйнфреймам. Развитием идей статистического мультиплексирования стала технология асинхронного режима передачи – ATM, которая вобрала в себя лучшие черты техники коммутации каналов и пакетов.

Сети TDM могут поддерживать либо режим динамической коммутации, либо режим постоянной коммутации, а иногда и оба эти режима. Так, например, основным режимом цифровых телефонных сетей, работающих на основе технологии TDM, является динамическая коммутация, но они поддерживают также и постоянную коммутацию, предоставляя своим абонентам службу выделенных каналов.

Существует аппаратура, которая поддерживает только режим постоянной коммутации. К ней относится оборудование типа T1/E1, а также высокоскоростное оборудование SDH. Такое оборудование используется для построения первичных сетей, основной функцией которых является создание выделенных каналов между коммутаторами, поддерживающими динамическую коммутацию.

Сегодня практически все данные – голос, изображение, компьютерные данные – передаются в цифровой форме. Поэтому выделенные каналы TDM-технологии, которые обеспечивают нижний уровень для передачи цифровых данных, являются универсальными каналами для построения сетей любого типа: телефонных, телевизионных и компьютерных.

Общие свойства сетей с коммутацией каналов

Сети с коммутацией каналов обладают несколькими важными общими свойствами независимо от того, какой тип мультиплексирования в них используется.

Сети с динамической коммутацией требуют предварительной процедуры установления соединения между абонентами. Для этого в сеть передается адрес вызываемого абонента, который проходит через коммутаторы и настраивает их на последующую передачу данных. Запрос на установление соединения маршрутизируется от одного коммутатора к другому и в конце концов достигает вызываемого абонента. Сеть может отказать в установлении соединения, если емкость требуемого выходного канала уже исчерпана. Для FDM-коммутатора емкость выходного канала равна количеству частотных полос этого канала, а для TDM-коммутатора – количеству тайм-слотов, на которые делится цикл работы канала. Сеть отказывает в соединении также в том случае, если запрашиваемый абонент уже установил соединение с кем-нибудь другим: В первом случае говорят, что занят коммутатор, а во втором – абонент. Возможность отказа в соединении является недостатком метода коммутации каналов.

Если соединение может быть установлено, то ему выделяется фиксированная полоса частот в FDM-сетях или же фиксированная пропускная способность в TDM-сетях. Эти величины остаются неизменными в течение всего периода соединения. Гарантированная пропускная способность сети после установления соединения является важным свойством, необходимым для таких приложений, как передача голоса, изображения или управления объектами в реальном масштабе времени. Однако динамически изменять пропускную способность канала по требованию абонента сети с коммутацией каналов не могут, что делает их неэффективными в условиях пульсирующего трафика.

Недостатком сетей с коммутацией каналов, является невозможность, применения пользовательской аппаратуры, работающей с разной скоростью. Отдельные части составного канала работают с одинаковой скоростью, так как сети с коммутацией каналов не буферизуют данные пользователей.

Сети с коммутацией каналов хорошо приспособлены для коммутации потоков данных постоянной скорости, когда единицей коммутации является не отдельный байт или пакет данных, а долговременный синхронный поток данных между двумя абонентами. Для таких потоков сети с коммутацией каналов добавляют минимум служебной информации для маршрутизации данных через сеть, используя временную позицию каждого бита потока в качестве его адреса назначения в коммутаторах сети.

Обеспечение дуплексного режима работы на основе технологий FDM, TDM и WDM

В зависимости от направления возможной передачи данных способы передачи данных по линии связи делятся на следующие типы:

- *симплексный* – передача осуществляется по линии связи только в одном направлении;
- *полудуплексный* – передача ведется в обоих направлениях, но попеременно во времени. Примером такой передачи служит технология Ethernet;
- *дуплексный* – передача ведется одновременно в двух направлениях.

Дуплексный режим – наиболее универсальный и производительный способ работы канала. Самым простым вариантом организации дуплексного режима является использование двух независимых физических каналов (двух пар проводников или двух световодов) в кабеле, каждый из которых работает в симплексном режиме, то есть передает данные в одном направлении. Именно такая идея лежит в основе реализации дуплексного режима работы во многих сетевых технологиях, например Fast Ethernet или ATM.

Иногда такое простое решение оказывается недоступным или неэффективным. Чаще всего это происходит в тех случаях, когда для дуплексного обмена данными имеется всего один физический канал, а организация второго связана с большими затратами. Например, при обмене данными с помощью модемов через телефонную сеть у пользователя имеется только один физический канал связи с АТС – двухпроводная линия, и приобретать второй вряд ли целесообразно. В таких случаях дуплексный режим работы организуется на основе разделения канала на два логических подканала с помощью техники FDM или TDM.

Модемы для организации дуплексного режима работы на двухпроводной линии применяют технику FDM. Модемы, использующие частотную модуляцию, работают на четырех частотах: две частоты – для кодирования единиц и нулей в одном направлении, а остальные две частоты – для передачи данных в обратном направлении.

При цифровом кодировании дуплексный режим на двухпроводной линии организуется с помощью техники TDM. Часть тайм-слотов используется для передачи данных в одном направлении, а часть – для передачи в другом направлении. Обычно тайм-слоты противоположных направлений чередуются, из-за чего такой способ иногда называют «пинг-понговой» передачей. TDM-разделение линии характерно, например, для цифровых сетей с интеграцией услуг (ISDN) на абонентских двухпроводных окончаниях.

В волоконно-оптических кабелях при использовании одного оптического волокна для организации дуплексного режима работы применяется передача данных в одном направлении с помощью светового пучка одной длины волны, а в обратном – другой длины волны. Такая техника относится к методу FDM, однако для оптических кабелей она получила название *разделения по длине волны* (*Wave Division Multiplexing, WDM*). WDM применяется и для повышения скорости передачи данных в одном направлении, обычно используя от 2 до 16 каналов.

2.4.2. Коммутация пакетов

Принципы коммутации пакетов

Коммутация пакетов – это техника коммутации абонентов, которая была специально разработана для эффективной передачи компьютерного трафика. Эксперименты по созданию первых компьютерных сетей на основе техники коммутации каналов показали, что этот вид коммутации не позволяет достичь высокой общей пропускной способности сети. Суть проблемы заключается в пульсирующем характере трафика, который генерируют типичные сетевые приложения. Например, при обращении к удаленному файловому серверу пользователь сначала просматривает содержимое каталога этого сервера, что порождает передачу небольшого объема данных. Затем он открывает требуемый файл в текстовом редакторе, и эта операция может создать достаточно интенсивный обмен данными, особенно если файл содержит объемные графические включения. После отображения нескольких страниц файла пользователь некоторое время работает с ними локально, что вообще не требует передачи данных по сети, а затем возвращает модифицированные копии страниц на сервер – и это снова порождает интенсивную передачу данных по сети.

Коэффициент пульсации трафика отдельного пользователя сети, равный отношению средней интенсивности обмена данными к максимально возможной, может составлять 1:50 или 1:100. Если для описанной сессии организовать коммутацию канала между компьютером пользователя и сервером, то большую часть времени канал будет простаивать. В то же время коммутационные возможности сети будут использоваться – часть тайм-слотов или частотных полос коммутаторов будет занята и

недоступна другим пользователям сети.

При коммутации пакетов все передаваемые пользователем сети сообщения разбиваются в исходном узле на сравнительно небольшие части, называемые пакетами. Напомним, что сообщением называется логически завершенная порция данных – запрос на передачу файла, ответ на этот запрос, содержащий весь файл, и т. п. Сообщения могут иметь произвольную длину, от нескольких байт до многих мегабайт. Напротив, пакеты обычно тоже могут иметь переменную длину, но в узких пределах, например от 46 до 1500 байт. Каждый пакет снабжается заголовком, в котором указывается адресная информация, необходимая для доставки пакета узлу назначения, а также номер пакета, который будет использоваться узлом назначения для сборки сообщения (рис. 2.29). Пакеты транспортируются в сети как независимые информационные блоки. Коммутаторы сети принимают пакеты от конечных узлов и на основании адресной информации передают их друг другу, а в конечном итоге – узлу назначения.

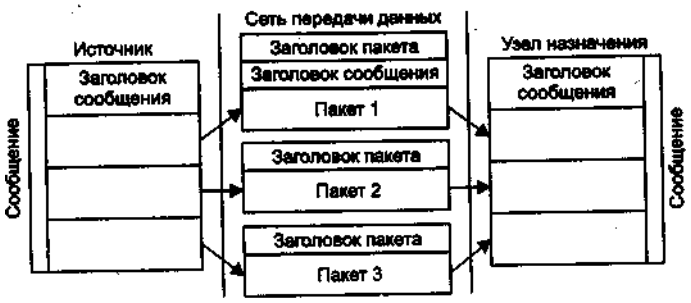


Рис. 2.29. Разбиение сообщения на пакеты

Коммутаторы пакетной сети отличаются от коммутаторов каналов тем, что они имеют внутреннюю буферную память для временного хранения пакетов, если выходной порт коммутатора в момент принятия пакета занят передачей другого пакета (рис. 2.30). В этом случае пакет находится некоторое время в очереди пакетов в буферной памяти выходного порта, а когда до него дойдет очередь, то он передается следующему коммутатору. Такая схема передачи данных позволяет сглаживать пульсации трафика на магистральных связях между коммутаторами и тем самым использовать их наиболее эффективным образом для повышения пропускной способности сети в целом.

Действительно, для пары абонентов наиболее эффективным было бы предоставление им в единоличное пользование скоммутированного канала связи, как это делается в сетях с коммутацией каналов. При этом способе время взаимодействия этой пары абонентов было бы минимальным, так как данные без задержек передавались бы от одного абонента другому. Простои канала во время пауз передачи абонентов не интересуют, для них важно быстрее решить свою собственную задачу. Сеть с коммутацией пакетов замедляет процесс взаимодействия конкретной пары абонентов, так как их пакеты могут ожидать в коммутаторах, пока по магистральным связям передаются другие пакеты, пришедшие в коммутатор ранее.

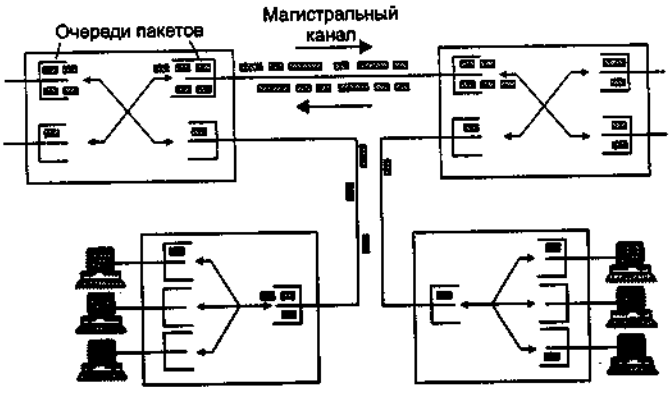


Рис. 2.30. Сглаживание пульсаций трафика в сети с коммутацией пакетов

Тем не менее общий объем передаваемых сетью компьютерных данных в единицу времени при технике коммутации пакетов будет выше, чем при технике коммутации каналов. Это происходит потому, что пульсации отдельных абонентов в соответствии с законом больших чисел распределяются во времени. Поэтому коммутаторы постоянно и достаточно равномерно загружены работой, если число

обслуживаемых ими абонентов действительно велико. На рис. 2.30 показано, что трафик, поступающий от конечных узлов на коммутаторы, очень неравномерно распределен во времени. Однако коммутаторы более высокого уровня иерархии, которые обслуживают соединения между коммутаторами нижнего уровня, загружены более равномерно, и поток пакетов в магистральных каналах, соединяющих коммутаторы верхнего уровня, имеет почти максимальный коэффициент использования.

Более высокая эффективность сетей с коммутацией пакетов по сравнению с сетями с коммутацией каналов (при равной пропускной способности каналов связи) была доказана в 60-е годы как экспериментально, так и с помощью имитационного моделирования. Здесь уместна аналогия с мультипрограммными операционными системами. Каждая отдельная программа в такой системе выполняется дольше, чем в однопрограммной системе, когда программе выделяется все процессорное время, пока она не завершит свое выполнение. Однако общее число программ, выполняемых за единицу времени, в мультипрограммной системе больше, чем в однопрограммной.

Виртуальные каналы в сетях с коммутацией пакетов

Описанный выше режим передачи пакетов между двумя конечными узлами сети предполагает независимую маршрутизацию каждого пакета. Такой режим работы сети называется дейтаграммным, и при его использовании коммутатор может изменить маршрут какого-либо пакета в зависимости от состояния сети – работоспособности каналов и других коммутаторов, длины очередей пакетов в соседних коммутаторах и т. п.

Существует и другой режим работы сети – передача пакетов по *виртуальному каналу* (*virtual circuit* или *virtual channel*). В этом случае перед тем, как начать передачу данных между двумя конечными узлами, должен быть установлен виртуальный канал, который представляет собой единственный маршрут, соединяющий эти конечные узлы. Виртуальный канал может быть динамическим или постоянным. Динамический виртуальный канал устанавливается при передаче в сеть специального пакета – запроса на установление соединения. Этот пакет проходит через коммутаторы и «прокладывает» виртуальный канал. Это означает, что коммутаторы запоминают маршрут для данного соединения и при поступлении последующих пакетов данного соединения отправляют их всегда по проложенному маршруту. Постоянные виртуальные каналы создаются администраторами сети путем ручной настройки коммутаторов.

При отказе коммутатора или канала на пути виртуального канала соединение разрывается, и виртуальный канал нужно прокладывать заново. При этом он, естественно, обойдет отказавшие участки сети.

Каждый режим передачи пакетов имеет свои преимущества и недостатки. Дейтаграммный метод не требует предварительного установления соединения и поэтому работает без задержки перед передачей данных. Это особенно выгодно для передачи небольшого объема данных, когда время установления соединения может быть соизмеримым со временем передачи данных. Кроме того, дейтаграммный метод быстрее адаптируется к изменениям в сети.

При использовании метода виртуальных каналов время, затраченное на установление виртуального канала, компенсируется последующей быстрой передачей всего потока пакетов. Коммутаторы распознают принадлежность пакета к виртуальному каналу по специальной метке – номеру виртуального канала, а не анализируют адреса конечных узлов, как это делается при дейтаграммном методе.

Пропускная способность сетей с коммутацией пакетов

Одним из отличий метода коммутации пакетов от метода коммутации каналов является неопределенность пропускной способности соединения между двумя абонентами. В методе коммутации каналов после образования составного канала пропускная способность сети при передаче данных между конечными узлами известна – это пропускная способность канала. Данные после задержки, связанной с установлением канала, начинают передаваться на максимальной для канала скорости (рис. 2.31, *а*). Время передачи сообщения в сети с коммутацией каналов $T_{к.к}$. равно сумме задержки распространения сигнала по линии связи $t_{з.р}$. и задержки передачи сообщения $t_{з.п}$.. Задержка распространения сигнала зависит от скорости распространения электромагнитных волн в конкретной физической среде, которая колеблется от 0,6 до 0,9 скорости света в вакууме. Время передачи сообщения равно V/C, где V – объем

сообщения в битах, а С – пропускная способность канала в битах в секунду.

В сети с коммутацией пакетов наблюдается принципиально другая картина.

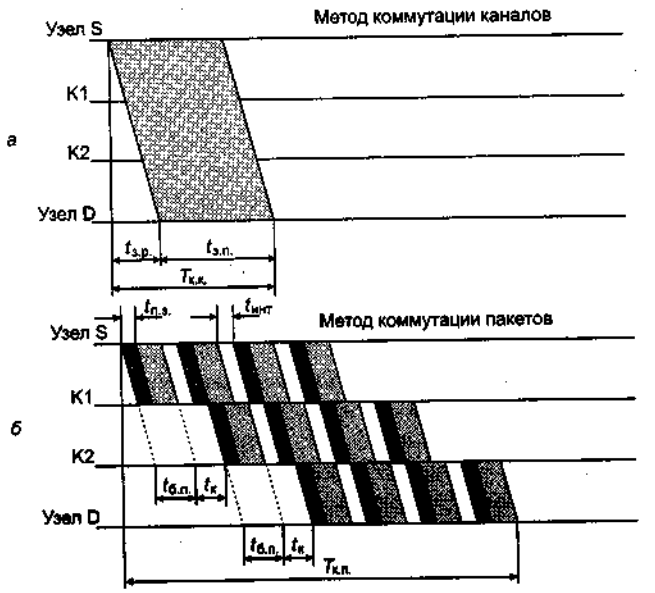


Рис. 2.31. Задержки передачи данных в сетях с коммутацией каналов и пакетов

Процедура установления соединения в этих сетях, если она используется, занимает примерно такое же время, как и в сетях с коммутацией каналов, поэтому будем сравнивать только время передачи данных.

На рис. 2.31, *б* показан пример передачи в сети с коммутацией пакетов. Предполагается, что в сеть передается сообщение того же объема, что и сообщение, иллюстрируемое рис. 2.31, *а*, однако оно разделено на пакеты, каждый из которых снабжен заголовком. Время передачи сообщения в сети с коммутацией пакетов обозначено на рисунке Тк.п. При передаче этого сообщения, разбитого на пакеты, по сети с коммутацией пакетов возникают дополнительные временные задержки. Во-первых, это задержки в источнике передачи, который, помимо передачи собственно сообщения, тратит дополнительное время на передачу заголовков $t_{п.з}$, плюс к этому добавляются задержки $t_{инт,}$ вызванные интервалами между передачей каждого следующего пакета (это время уходит на формирование очередного пакета стеком протоколов).

Во-вторых, дополнительное время тратится в каждом коммутаторе. Здесь задержки складываются из времени буферизации пакета $t_{б.п.}$ (коммутатор не может начать передачу пакета, не приняв его полностью в свой буфер) и времени коммутации t_k. Время буферизации равно времени приема пакета с битовой скоростью протокола. Время коммутации складывается из времени ожидания пакета в очереди и времени перемещения пакета в выходной порт. Если время перемещения пакета фиксировано и обычно невелико (от нескольких микросекунд до нескольких десятков микросекунд), то время ожидания пакета в очереди колеблется в очень широких пределах и заранее неизвестно, так как зависит от текущей загрузки сети пакетами.

Проведем грубую оценку задержки в передаче данных в сетях с коммутацией пакетов по сравнению с сетями с коммутацией каналов на простейшем примере. Пусть тестовое сообщение, которое нужно передать в обоих видах сетей, составляет 200 Кбайт. Отправитель находится от получателя на расстоянии 5000 км. Пропускная способность линий связи составляет 2 Мбит/с.

Время передачи данных по сети с коммутацией каналов складывается из времени распространения сигнала, которое для расстояния 5000 км можно оценить примерно в 25 мс, и времени передачи сообщения, которое при пропускной способности 2 Мбит/с и длине сообщения 200 Кбайт равно примерно 800 мс, то есть всего передача данных заняла 825 мс.

Оценим дополнительное время, которое потребуется для передачи этого сообщения по сети с коммутацией пакетов. Будем считать, что путь от отправителя до получателя пролегает через 10 коммутаторов. Исходное сообщение разбивается на пакеты в 1 Кбайт, всего 200 пакетов. Вначале оценим задержку, которая возникает в исходном узле. Предположим, что доля служебной информации, размещенной в заголовках пакетов, по отношению к общему объему сообщения составляет 10 %. Следовательно, дополнительная задержка, связанная с передачей заголовков пакетов, составляет 10 %

от времени передачи целого сообщения, то есть 80 мс. Если принять интервал между отправкой пакетов равным 1 мс, тогда дополнительные потери за счет интервалов составят 200 мс. Итого, в исходном узле из-за пакетирования сообщения при передаче возникла дополнительная задержка в 280 мс.

Каждый из 10 коммутаторов вносит задержку коммутации, которая может иметь большой разброс, от долей до тысяч миллисекунд. В данном примере примем, что на коммутацию в среднем тратится 20 мс. Кроме того, при прохождении сообщений через коммутатор возникает задержка буферизации пакета. Эта задержка при величине пакета 1 Кбайт и пропускной способности линии 2 Мбит/с равна 4 мс. Общая задержка, вносимая 10 коммутаторами, составит примерно 240 мс. В результате дополнительная задержка, созданная сетью с коммутацией пакетов, составила 520 мс. Учитывая, что вся передача данных в сети с коммутацией каналов заняла 825 мс, эту дополнительную задержку можно считать существенной.

Хотя приведенный расчет носит очень приблизительный характер, но он делает более понятными те причины, которые приводят к тому, что процесс передачи для определенной пары абонентов в сети с коммутацией пакетов является более медленным, чем в сети с коммутацией каналов.

Неопределенная пропускная способность сети с коммутацией пакетов – это плата за ее общую эффективность при некотором ущемлении интересов отдельных абонентов. Аналогично, в мультипрограммной операционной системе время выполнения приложения предсказать заранее невозможно, так как оно зависит от количества других приложений, с которыми делит процессор данное приложение.

На эффективность работы сети существенно влияют размеры пакетов, которые передает сеть. Слишком большие размеры пакетов приближают сеть с коммутацией пакетов к сети с коммутацией каналов, поэтому эффективность сети при этом падает. Слишком маленькие пакеты заметно увеличивают долю служебной информации, так как каждый пакет несет с собой заголовок фиксированной длины, а количество пакетов, на которые разбиваются сообщения, будет резко расти при уменьшении размера пакета. Существует некоторая золотая середина, которая обеспечивает максимальную эффективность работы сети, однако ее трудно определить точно, так как она зависит от многих факторов, некоторые из них к тому же постоянно меняются в процессе работы сети. Поэтому разработчики протоколов для сетей с коммутацией пакетов выбирают пределы, в которых может находиться длина пакета, а точнее его поле данных, так как заголовок, как правило, имеет фиксированную длину. Обычно нижний предел поля данных выбирается равным нулю, что разрешает передавать служебные пакеты без пользовательских данных, а верхний предел не превышает 4-х килобайт. Приложения при передаче данных пытаются занять максимальный размер поля данных, чтобы быстрее выполнить обмен данными, а небольшие пакеты обычно используются для квитанций о доставке пакета.

При выборе размера пакета необходимо учитывать также и интенсивность битовых ошибок канала. На ненадежных каналах необходимо уменьшать размеры пакетов, так как это уменьшает объем повторно передаваемых данных при искажениях пакетов.

2.4.3. Коммутация сообщений

Под *коммутацией сообщений* понимается передача единого блока данных между транзитными компьютерами сети с временной буферизацией этого блока на диске каждого компьютера (рис. 2.32). Сообщение в отличие от пакета имеет произвольную длину, которая определяется не технологическими соображениями, а содержанием информации, составляющей сообщение. Например, сообщением может быть текстовый документ, файл с кодом программы, электронное письмо.

Транзитные компьютеры могут соединяться между собой как сетью с коммутацией пакетов, так и сетью с коммутацией каналов. Сообщение хранится в транзитном компьютере на диске, причем время хранения может быть достаточно большим, если компьютер загружен другими работами или сеть временно перегружена.

По такой схеме обычно передаются сообщения, не требующие немедленного ответа, чаще всего сообщения электронной почты. Режим передачи с промежуточным хранением на диске называется режимом <*хранение-и-передача*> (*store-and-forward*).

Режим коммутации сообщений разгружает сеть для передачи трафика, требующего быстрого ответа, например трафика службы WWW или файловой службы.

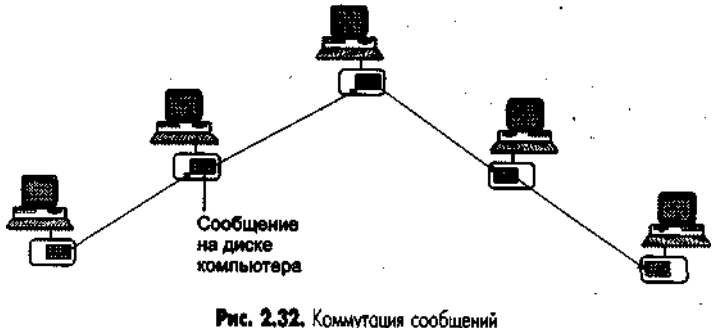


Рис. 2.32. Коммутация сообщений

Количество транзитных компьютеров стараются по возможности уменьшить. Если компьютеры подключены к сети с коммутацией пакетов, то число промежуточных компьютеров обычно уменьшается до двух. Например, пользователь передает почтовое сообщение своему серверу исходящей почты, а тот сразу старается передать сообщение серверу входящей почты адресата. Но если компьютеры связаны между собой телефонной сетью, то часто используется несколько промежуточных серверов, так как прямой доступ к конечному серверу может быть невозможен в данный момент из-за перегрузки телефонной сети (абонент занят) или экономически невыгоден из-за высоких тарифов на дальнюю телефонную связь.

Техника коммутации сообщений появилась в компьютерных сетях раньше техники коммутации пакетов, но потом была вытеснена последней, как более эффективной по критерию пропускной способности сети. Запись сообщения на диск занимает достаточно много времени, кроме того, наличие дисков предполагает специализированные компьютеры в качестве коммутаторов, что удорожает сеть.

Сегодня коммутация сообщений работает только для некоторых не оперативных служб, причем чаще всего поверх сети с коммутацией пакетов, как служба прикладного уровня.

Выводы

- В сетях для соединения абонентов используются три метода коммутации: коммутация каналов, коммутация пакетов и коммутация сообщений.
- Как коммутация каналов, так и коммутация пакетов может быть либо динамической, либо постоянной.
- В сетях с коммутацией каналов абонентов соединяет составной канал, образуемый коммутаторами сети по запросу одного из абонентов.
- Для совместного разделения каналов между коммутаторами сети несколькими абонентскими каналами используются две технологии: частотного разделения канала (FDM) и разделения канала во времени (TDM). Частотное разделение характерно для аналоговой модуляции сигналов, а временное – для цифрового кодирования.
- Сети с коммутацией каналов хорошо коммутируют потоки данных постоянной интенсивности, например потоки данных, создаваемые разговаривающими по телефону собеседниками, но не могут перераспределять пропускную способность магистральных каналов между потоками абонентских каналов динамически.
- Сети с коммутацией пакетов были специально разработаны для эффективной передачи пульсирующего компьютерного трафика. Буферизация пакетов разных абонентов в коммутаторах позволяет сгладить неравномерности интенсивности трафика каждого абонента и равномерно загрузить каналы связи между коммутаторами.
- Сети с коммутацией пакетов эффективно работают в том отношении, что объем передаваемых данных от всех абонентов сети в единицу времени больше, чем при использовании сети с коммутацией каналов. Однако для каждой пары абонентов пропускная способность сети может оказаться ниже, чем у сети с коммутацией каналов, за счет очередей пакетов в коммутаторах.
- Сети с коммутацией пакетов могут работать в одном из двух режимов: дейтаграммном режиме или режиме виртуальных каналов.
- Размер пакета существенно влияет на производительность сети. Обычно пакеты в сетях имеют максимальный размер в 1-4 Кбайт.
- Коммутация сообщений предназначена для организации взаимодействия пользователей в

режиме off-line, когда не ожидается немедленной реакции на сообщение. При этом методе коммутации сообщение передается через несколько транзитных компьютеров, где оно целиком буферизуется на диске.

Вопросы и упражнения

1. Могут ли цифровые линии связи передавать аналоговые данные?
2. Каким будет теоретический предел скорости передачи данных в битах в секунду по каналу с шириной полосы пропускания в 20 кГц, если мощность передатчика составляет 0,01 мВт, а мощность шума в канале равна 0,0001 мВт?
3. Определите пропускную способность канала связи для каждого из направлений дуплексного режима, если известно, что его полоса пропускания равна 600 кГц, а в методе кодирования используется 10 состояний сигнала.
4. Рассчитайте задержку распространения сигнала и задержку передачи данных для случая передачи пакета в 128 байт:
   - по кабелю витой пары длиной в 100 м при скорости передачи 100 Мбит/с;
   - коаксиальному кабелю длиной в 2 км при скорости передачи в 10 Мбит/с;
   - спутниковому геостационарному каналу протяженностью в 72 000 км при скорости передачи 128 Кбит/с.

   Считайте скорость распространения сигнала равной скорости света в вакууме 300 000 км/с.
5. Какой кадр передаст на линию передатчик, если он работает с использованием техники бит-стаффинга с флагом 7E, а на вход передатчика поступила последовательность 24 A5 7E 56 8C (все значения - шестнадцатеричные)?
6. Поясните, из каких соображений выбрана пропускная способность 64 Кбит/с элементарного канала цифровых телефонных сетей?
7. Назовите методы компрессии, наиболее подходящие для текстовой информации. Почему они неэффективны для сжатия двоичных данных?
8. Предложите коды неравной длины для каждого из символов A, B, C, D, F и O, если нужно передать сообщение BDDACAAFOOOAOOOO. Будет ли достигнута компрессия данных по сравнению с использованием:
   - традиционных кодов ASCII?
   - кодов равной длины, учитывающих наличие только данных символов?
9. Как передатчик определяет факт потери положительной квитанции в методе скользящего окна?
10. Сеть с коммутацией пакетов испытывает перегрузку. Для устранения этой ситуации размер окна в протоколах компьютеров сети нужно увеличить или уменьшить?
11. Как влияет надежность линий связи в сети на выбор размера окна?
12. В чем проявляется избыточность TDM-технологии?
13. Какой способ коммутации более эффективен: коммутация каналов или коммутация пакетов?
14. Объясните разницу между тремя понятиями:
    - логические соединения, на которых основаны некоторые протоколы;
    - виртуальные каналы в сетях с коммутацией пакетов;
    - составные каналы в сетях с коммутацией каналов.

ГЛАВА 3. БАЗОВЫЕ ТЕХНОЛОГИИ ЛОКАЛЬНЫХ СЕТЕЙ

3.1, Протоколы и стандарты локальных сетей

3.1.1. Общая характеристика протоколов локальных сетей

При организации взаимодействия узлов в локальных сетях основная роль отводится протоколу канального уровня. Однако для того, чтобы канальный уровень мог справиться с этой задачей, структура локальных сетей должна быть вполне определенной, так, например, наиболее популярный протокол канального уровня – Ethernet – рассчитан на параллельное подключение всех узлов сети к общей для них шине – отрезку коаксиального кабеля или иерархической древовидной структуре

сегментов, образованных повторителями. Протокол Token Ring также рассчитан на вполне определенную конфигурацию – соединение компьютеров в виде логического кольца.

Подобный подход, заключающийся в использовании простых структур кабельных соединений между компьютерами локальной сети, соответствовал основной цели, которую ставили перед собой разработчики первых локальных сетей во второй половине 70-х годов. Эта цель заключалась в нахождении простого и дешевого решения для объединения в вычислительную сеть нескольких десятков компьютеров, находящихся в пределах одного здания. Решение должно было быть недорогим, поскольку в сеть объединялись недорогие компьютеры – появившиеся и быстро распространившиеся тогда мини-компьютеры стоимостью в 10 000-20 000 долларов. Количество их в одной организации было небольшим, поэтому предел в несколько десятков (максимум – до сотни) компьютеров представлялся вполне достаточным для роста практически любой локальной сети.

Для упрощения и, соответственно, удешевления аппаратных и программных решений разработчики первых локальных сетей остановились на совместном использовании кабелей всеми компьютерами сети в режиме разделения времени, то есть режиме TDM. Наиболее явным образом режим совместного использования кабеля проявляется в классических сетях Ethernet, где коаксиальный кабель физически представляет собой неделимый отрезок кабеля, общий для всех узлов сети. Но и в сетях Token Ring и FDDI, где каждая соседняя пара компьютеров соединена, казалось бы, своими индивидуальными отрезками кабеля с концентратором, эти отрезки не могут использоваться компьютерами, которые непосредственно к ним подключены, в произвольный момент времени. Эти отрезки образуют логическое кольцо, доступ к которому как к единому целому может быть получен только по вполне определенному алгоритму, в котором участвуют все компьютеры сети. Использование кольца как общего разделяемого ресурса упрощает алгоритмы передачи по нему кадров, так как в каждый конкретный момент времени кольцо занято только одним компьютером.

Использование разделяемых сред (shared media) позволяет упростить логику работы сети. Например, отпадает необходимость контроля переполнения узлов сети кадрами от многих станций, решивших одновременно обменяться информацией. В глобальных сетях, где отрезки кабелей, соединяющих отдельные узлы, не рассматриваются как общий ресурс, такая необходимость возникает, и для решения этой проблемы в протоколы обмена информацией вводятся весьма сложные процедуры управления потоком кадров, предотвращающие переполнение каналов связи и узлов сети.

Использование в локальных сетях очень простых конфигураций (общая шина и кольцо) наряду с положительными имело и отрицательные последствия, из которых наиболее неприятными были ограничения по производительности и надежности. Наличие только одного пути передачи информации, разделяемого всеми узлами сети, в принципе ограничивало пропускную способность сети пропускной способностью этого пути (которая делилась в среднем на число компьютеров сети), а надежность сети – надежностью этого пути. Поэтому по мере повышения популярности локальных сетей и расширения их сфер применения все больше стали применяться специальные коммуникационные устройства – мосты и маршрутизаторы, – которые в значительной мере снимали ограничения единственной разделяемой среды передачи данных. Базовые конфигурации в форме общей шины и кольца превратились в элементарные структуры локальных сетей, которые можно теперь соединять друг с другом более сложным образом, образуя параллельные основные или резервные пути между узлами.

Тем не менее внутри базовых структур по-прежнему работают все те же протоколы разделяемых единственных сред передачи данных, которые были разработаны более 15 лет назад. Это связано с тем, что хорошие скоростные и надежностные характеристики кабелей локальных сетей удовлетворяли в течение всех этих лет пользователей небольших компьютерных сетей, которые могли построить сеть без больших затрат только с помощью сетевых адаптеров и кабеля. К тому же колоссальная инсталляционная база оборудования и программного обеспечения для технологий Ethernet и Token Ring способствовала тому, что сложился следующий подход: в пределах небольших сегментов используются старые протоколы в их неизменном виде, а объединение таких сегментов в общую сеть происходит с помощью дополнительного и достаточно сложного оборудования.

В последние несколько лет наметилось движение к отказу от разделяемых сред передачи данных в локальных сетях и переходу к применению активных коммутаторов, к которым конечные узлы присоединяются индивидуальными линиями связи. В чистом виде такой подход предлагается в технологии ATM (Asynchronous Transfer Mode), а в технологиях, носящих традиционные названия с приставкой switched (коммутируемый): switched Ethernet, switched Token Ring, switched FDDI, обычно используется смешанный подход, сочетающий разделяемые и индивидуальные среды передачи данных.

Чаще всего конечные узлы соединяются в небольшие разделяемые сегменты с помощью повторителей, а сегменты соединяются друг с другом с помощью индивидуальных коммутируемых связей.

Существует и достаточно заметная тенденция к использованию в традиционных технологиях так называемой микросегментации, когда даже конечные узлы сразу соединяются с коммутатором индивидуальными каналами. Такие сети получаются дороже разделяемых или смешанных, но производительность их выше.

При использовании коммутаторов у традиционных технологий появился новый режим работы – *полнодуплексный* (*full-duplex*). В разделяемом сегменте станции всегда работают в *полудуплексном режиме* (*half-duplex*), так как в каждый момент времени сетевой адаптер станции либо передает свои данные, либо принимает чужие, но никогда не делает это одновременно. Это справедливо для всех технологий локальных сетей, так как разделяемые среды поддерживаются не только классическими технологиями локальных сетей Ethernet, Token Ring, FDDI, но и всеми новыми - Fast Ethernet, lOOVG-AnyLAN, Gigabit Ethernet.

В полнодуплексном режиме сетевой адаптер может одновременно передавать свои данные в сеть и принимать из сети чужие данные. Такой режим несложно обеспечивается при прямом соединение с мостом/коммутатором или маршрутизатором, так как вход и выход каждого порта такого устройства работают независимо друг от друга, каждый со своим буфером кадров.

Сегодня каждая технология локальных сетей приспособлена для работы как в полудуплексном, так и полнодуплексном режимах. В этих режимах ограничения, накладываемые на общую длину сети, существенно отличаются, так что одна и та же технология может позволять строить весьма различные сети в зависимости от выбранного режима работы (который зависит от того, какие устройства используются для соединения узлов – повторители или коммутаторы). Например, технология Fast Ethernet позволяет для полудуплексного режима строить сети диаметром не более 200 метров, а для полнодуплексного режима ограничений на диаметр сети не существует. Поэтому при сравнении различных технологий необходимо обязательно принимать во внимание возможность их работы в двух режимах. В данной главе изучается в основном полудуплексный режим работы протоколов, а полнодуплексный режим рассматривается в следующей главе, совместно с изучением коммутаторов.

Несмотря на появление новых технологий, классические протоколы локальных сетей Ethernet и Token Ring по прогнозам специалистов будут повсеместно использоваться еще по крайней мере лет 5-10, в связи с чем знание их деталей необходимо для успешного применения современной коммуникационной аппаратуры. Кроме того, некоторые современные высокопроизводительные технологии, такие как Fast Ethernet, Gigabit Ethernetj в значительной степени сохраняют преемственность со своими предшественниками. Это еще раз подтверждает важность изучения классических протоколов локальных сетей, естественно, наряду с изучением новых технологий.

3.1.2. Структура стандартов IEEE 802.x

В 1980 году в институте IEEE был организован комитет 802 по стандартизации локальных сетей, в результате работы которого было принято семейство стандартов IEEE 802.x, которые содержат рекомендации по проектированию нижних уровней локальных сетей. Позже результаты работы этого комитета легли в основу комплекса международных стандартов ISO 8802-1...5. Эти стандарты были созданы на основе очень распространенных фирменных стандартов сетей Ethernet, ArcNet и Token Ring.

Помимо IEEE в работе по стандартизации протоколов локальных сетей принимали участие и другие организации. Так, для сетей, работающих на оптоволокне, американским институтом по стандартизации ANSI был разработан стандарт FDDI, обеспечивающий скорость передачи данных 100 Мб/с. Работы по стандартизации протоколов ведутся также ассоциацией ECMA, которой приняты стандарты ECMA-80, 81, 82 для локальной сети типа Ethernet и впоследствии стандарты ECMA-89, 90 по методу передачи маркера.

Стандарты семейства IEEE 802.x охватывают только два нижних уровня семиуровневой модели OSI – физический и канальный. Это связано с тем, что именно эти уровни в наибольшей степени отражают специфику локальных сетей. Старшие же уровни, начиная с сетевого, в значительной степени имеют общие черты как для локальных, так и для глобальных сетей.

Специфика локальных сетей также нашла свое отражение в разделении канального уровня на два подуровня, которые часто называют также уровнями. Канальный уровень (Data Link Layer) делится в локальных сетях на два подуровня:

- логической передачи данных (Logical Link Control, LLC);
- управления доступом к среде (Media Access Control, MAC).

Уровень MAC появился из-за существования в локальных сетях разделяемой среды передачи данных. Именно этот уровень обеспечивает корректное совместное использование общей среды, предоставляя ее в соответствии с определенным алгоритмом в распоряжение той или иной станции сети. После того как доступ к среде получен, ею может пользоваться более высокий уровень – уровень LLC, организующий передачу логических единиц данных, кадров информации, с различным уровнем качества транспортных услуг. В современных локальных сетях получили распространение несколько протоколов уровня MAC, реализующих различные алгоритмы доступа к разделяемой среде. Эти протоколы полностью определяют специфику таких технологий, как Ethernet, Fast Ethernet, Gigabit Ethernet, Token Ring, FDDI, lOOVG-AnyLAN.

Уровень LLC отвечает за передачу кадров данных между узлами с различной степенью надежности, а также реализует функции интерфейса с прилегающим к нему сетевым уровнем. Именно через уровень LLC сетевой протокол запрашивает у канального уровня нужную ему транспортную операцию с нужным качеством. На уровне LLC существует несколько режимов работы, отличающихся наличием или отсутствием на этом уровне процедур восстановления кадров в случае их потери или искажения, то есть отличающихся качеством транспортных услуг этого уровня.

Протоколы уровней MAC и LLC взаимно независимы – каждый протокол уровня MAC может применяться с любым протоколом уровня LLC, и наоборот.

Стандарты IEEE 802 имеют достаточно четкую структуру, приведенную на рис. 3.1:

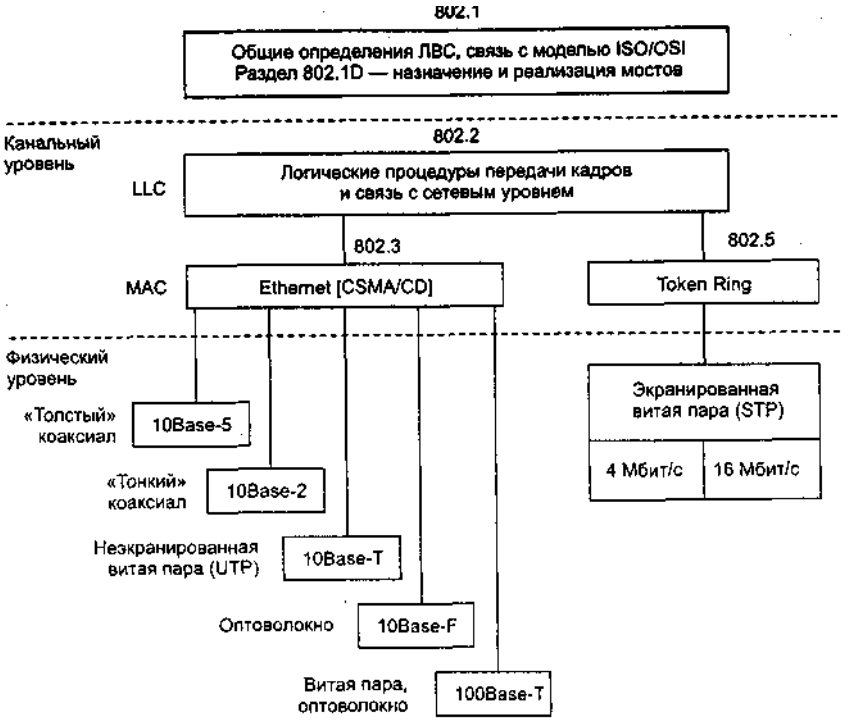


Рис. 3.1. Структура стандартов IEEE 802.x

Эта структура появилась в результате большой работы, проведенной комитетом 802 по выделению в разных фирменных технологиях общих подходов и общих функций, а также согласованию стилей их описания. В результате канальный уровень был разделен на два упомянутых подуровня. Описание каждой технологии разделено на две части: описание уровня MAC и описание физического уровня. Как видно из рисунка, практически у каждой технологии единственному протоколу уровня MAC соответствует несколько вариантов протоколов физического уровня (на рисунке в целях экономии места приведены только технологии Ethernet и Token Ring, но все сказанное справедливо также и для остальных технологий, таких как ArcNet, FDDI, lOOVG-AnyLAN).

Над канальным уровнем всех технологий изображен общий для них протокол LLC, поддерживающий несколько режимов работы, но независимый от выбора конкретной технологии. Стандарт LLC курирует подкомитет 802.2. Даже технологии, стандартизованные не в рамках комитета 802, ориентируются на использование протокола LLC, определенного стандартом 802.2, например

протокол FDDI, стандартизованный ANSI.

Особняком стоят стандарты, разрабатываемые подкомитетом 802.1. Эти стандарты носят общий для всех технологий характер. В подкомитете 802.1 были разработаны общие определения локальных сетей и их свойств, определена связь трех уровней модели IEEE 802 с моделью OSI. Но наиболее практически важными являются стандарты 802.1, которые описывают взаимодействие между собой различных технологий, а также стандарты по построению более сложных сетей на основе базовых топологий. Эта группа стандартов носит общее название стандартов межсетевого взаимодействия (internetworking). Сюда входят такие важные стандарты, как стандарт 802.1D, описывающий логику работы моста/коммутатора, стандарт 802.1H, определяющий работу транслирующего моста, который может без маршрутизатора объединять сети Ethernet и FDDI, Ethernet и Token Ring и т. п. Сегодня набор стандартов, разработанных подкомитетом 802.1, продолжает расти. Например, недавно он пополнился важным стандартом 802.1Q, определяющим способ построения виртуальных локальных сетей VLAN в сетях на основе коммутаторов.

Стандарты 802.3, 802.4, 802.5 и 802.12 описывают технологии локальных сетей, которые появились в результате улучшений фирменных технологий, легших в их основу. Так, основу стандарта 802.3 составила технология Ethernet, разработанная компаниями Digital, Intel и Xerox (или Ethernet DIX), стандарт 802.4 появился как обобщение технологии ArcNet компании Datapoint Corporation, а стандарт 802.5 в основном соответствует технологии Token Ring компании IBM.

Исходные фирменные технологии и их модифицированные варианты – стандарты 802.x в ряде случаев долгие годы существовали параллельно. Например, технология ArcNet так до конца не была приведена в соответствие со стандартом 802.4 (теперь это делать поздно, так как где-то примерно с 1993 года производство оборудования ArcNet было свернуто). Расхождения между технологией Token Ring и стандартом 802.5 тоже периодически возникают, так как компания IBM регулярно вносит усовершенствования в свою технологию и комитет 802.5 отражает эти усовершенствования в стандарте с некоторым запозданием. Исключение составляет технология Ethernet. Последний фирменный стандарт Ethernet DIX был принят в 1980 году, и с тех пор никто больше не предпринимал попыток фирменного развития Ethernet. Все новшества в семействе технологий Ethernet вносятся только в результате принятия открытых стандартов комитетом 802.3.

Более поздние стандарты изначально разрабатывались не одной компанией, а группой заинтересованных компаний, а потом передавались в соответствующий подкомитет IEEE 802 для утверждения. Так произошло с технологиями Fast Ethernet, lOOVG-AnyLAN, Gigabit Ethernet. Группа заинтересованных компаний образовывала сначала небольшое объединение, а затем по мере развития работ к нему присоединялись другие компании, так что процесс принятия стандарта носил открытый характер.

Сегодня комитет 802 включает следующий ряд подкомитетов, в который входят как уже упомянутые, так и некоторые другие:

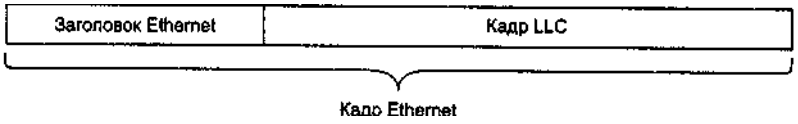
- 802.1 – Internetworking – объединение сетей;
- 802.2 – Logical Link Control, LLC – управление логической передачей данных;
- 802.3 – Ethernet с методом доступа CSMA/CD;
- 802.4 – Token Bus LAN – локальные сети с методом доступа Token Bus;
- 802.5 – Token Ring LAN – локальные сети с методом доступа Token Ring;
- 802.6 – Metropolitan Area Network, MAN – сети мегаполисов;
- 802.7 – Broadband Technical Advisory Group – техническая консультационная группа по широкополосной передаче;
- 802.8 – Fiber Optic Technical Advisory Group – техническая консультационная группа по волоконно-оптическим сетям;
- 802.9 – Integrated Voice and data Networks - интегрированные сети передачи голоса и данных;
- 802.10 – Network Security – сетевая безопасность;
- 802.11 – Wireless Networks – беспроводные сети;
- 802.12 – Demand Priority Access LAN, lOOVG-AnyLAN – локальные сети с методом доступа по требованию с приоритетами.

Выводы

- При организации взаимодействия узлов в локальных сетях основная роль отводится классическим технологиям Ethernet, Token Ring, FDDI, разработанным более 15 лет назад и основанным на использовании разделяемых сред.
- Разделяемые среды поддерживаются не только классическими технологиями локальных сетей Ethernet, Token Ring, FDDI, но и новыми – Fast Ethernet, lOOVG-AnyLAN, Gigabit Ethernet.
- Современной тенденцией является частичный или полный отказ от разделяемых сред: соединение узлов индивидуальными связями (например, в технологии ATM), широкое использование коммутируемых связей и микросегментации. Еще одна важная тенденция – появление полнодуплексного режима работы практически для всех технологий локальных сетей.
- Комитет IEEE 802.x разрабатывает стандарты, которые содержат рекомендации для проектирования нижних уровней локальных сетей – физического и канального. Специфика локальных сетей нашла свое отражение в разделении канального уровня на два подуровня – LLC и MAC.
- Стандарты подкомитета 802.1 носят общий для всех технологий характер и постоянно пополняются. Наряду с определением локальных сетей и их свойств, стандартами межсетевого взаимодействия, описанием логики работы моста/коммутатора к результатам работы комитета относится и стандартизация сравнительно новой технологии виртуальных локальных сетей VLAN.
- Подкомитет 802.2 разработал и поддерживает стандарт LLC. Стандарты 802.3, 802.4, 802.5 описывают технологии локальных сетей, которые появились в результате улучшений фирменных технологий, легших в их основу, соответственно Ethernet, ArcNet, Token Ring.
- Более поздние стандарты изначально разрабатывались не одной компанией, а группой заинтересованных компаний, а потом передавались в соответствующий подкомитет IEEE 802 для утверждения.

3.2. Протокол LLC уровня управления логическим каналом (802.2)

Протокол LLC обеспечивает для технологий локальных сетей нужное качество услуг транспортной службы, передавая свои кадры либо дейтаграммным способом, либо с помощью процедур с установлением соединения и восстановлением кадров. Протокол LLC занимает уровень между сетевыми протоколами и протоколами уровня MAC. Протоколы сетевого уровня передают через межуровневый интерфейс данные для протокола LLC – свой пакет (например, пакет IP, IPX или NetBEUI), адресную информацию об узле назначения, а также требования к качеству транспортных услуг, которое протокол LLC должен обеспечить. Протокол LLC помещает пакет протокола верхнего уровня в свой кадр, который дополняется необходимыми служебными полями. Далее через межуровневый интерфейс протокол LLC передает свой кадр вместе с адресной информацией об узле назначения соответствующему протоколу уровня MAC, который упаковывает кадр LLC в свой кадр (например, кадр Ethernet).



В основу протокола LLC положен протокол HDLC (High-level Data Link Control Procedure), являющийся стандартом ISO. Собственно стандарт HDLC представляет собой обобщение нескольких близких стандартов, характерных для различных технологий: протокола LAP-B сетей X.25 (стандарта, широко распространенного в территориальных сетях), LAP-D, используемого в сетях ISDN, LAP-M, работающего в современных модемах. В спецификации IEEE 802.2 также имеется несколько небольших отличий от стандарта HDLC.

Первоначально в фирменных технологиях подуровень LLC не выделялся в самостоятельный подуровень, да и его функции растворялись в общих функциях протокола канального уровня. Из-за больших различий в функциях протоколов фирменных технологий, которые можно отнести к уровню LLC, на уровне LLC пришлось ввести три типа процедур. Протокол сетевого уровня может обращаться

к одной из этих процедур.

3.2.1. Три типа процедур уровня LLC

В соответствии со стандартом 802.2 уровень управления логическим каналом 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.

Этот набор процедур является общим для всех методов доступа к среде, определенных стандартами 802.3 - 802.5, а также стандартом FDDI и стандартом 802.12 на технологию lOOVG-AnyLAN.

Процедура без установления соединения и без подтверждения LLC1 дает пользователю средства для передачи данных с минимумом издержек. Это дейтаграммный режим работы. Обычно этот вид процедуры используется, когда такие функции, как восстановление данных после ошибок и упорядочивание данных, выполняются протоколами вышележащих уровней, поэтому нет нужды дублировать их на уровне LLC.

Процедура с установлением соединений и подтверждением LLC2 дает пользователю возможность установить логическое соединение перед началом передачи любого блока данных и, если это требуется, выполнить процедуры восстановления после ошибок и упорядочивание потока этих блоков в рамках установленного соединения. Протокол LLC2 во многом аналогичен протоколам семейства HDLC (LAP-B, LAP-D, LAP-M), которые применяются в глобальных сетях для обеспечения надежной передачи кадров на зашумленных линиях. Протокол LLC2 работает в режиме скользящего окна.

В некоторых случаях (например, при использовании сетей в системах реального времени, управляющих промышленными объектами), когда временные издержки установления логического соединения перед отправкой данных неприемлемы, а подтверждение о корректности приема переданных данных необходимо, базовая процедура без установления соединения и без подтверждения не подходит. Для таких случаев предусмотрена дополнительная процедура, называемая процедурой без установления соединения, но с подтверждением LLC3.

Использование одного из трех режимов работы уровня LLC зависит от стратегии разработчиков конкретного стека протоколов. Например, в стеке TCP/IP уровень LLC всегда работает в режиме LLC1, выполняя простую работу извлечения из кадра и демультиплексирования пакетов различных протоколов – IP, ARP, RARP. Аналогично используется уровень LLC стеком IPX/SPX.

А вот стек Microsoft/IBM, основанный на протоколе NetBIOS/NetBEUI, часто использует режим LLC2. Это происходит тогда, когда сам протокол NetBIOS/NetBEUI должен работать в режиме с восстановлением потерянных и искаженных данных. В этом случае эта работа перепоручается уровню LLC2. Если же протокол NetBIOS/NetBEUI работает в дейтаграммном режиме, то протокол LLC работает в режиме LLC1.

Режим LLC2 используется также стеком протоколов SNA в том случае, когда на нижнем уровне применяется технология Token Ring.

3.2.2. Структура кадров LLC. Процедура с восстановлением кадров LLC2

По своему назначению все кадры уровня LLC (называемые в стандарте 802.2 блоками данных – Protocol Data Unit, PDU) подразделяются на три типа – информационные, управляющие и ненумерованные.

- *Информационные кадры* (*Information*) предназначены для передачи информации в процедурах с установлением логического соединения LLC2 и должны обязательно содержать поле информации. В процессе передачи информационных блоков осуществляется их нумерация в режиме скользящего окна.
- *Управляющие кадры* (*Supervisory*) предназначены для передачи команд и ответов в процедурах с установлением логического соединения LLC2, в том числе запросов на повторную передачу искаженных информационных блоков.
- *Ненумерованные кадры* (*Unnumbered*) предназначены для передачи ненумерованных команд и ответов, выполняющих в процедурах без установления логического соединения передачу информации, идентификацию и тестирование LLC-уровня, а в процедурах с установлением

логического соединения LLC2 – установление и разъединение логического соединения, а также информирование об ошибках.

Все типы кадров уровня LLC имеют единый формат:

Флаг 01111110	Адрес точки входа службы назначения (DSAP)	Адрес точки входа службы источника (SSAP)	Управляющее поле (Control)	Данные (Data)	Флаг 01111110

Кадр LLC обрамляется двумя однобайтовыми полями «Флаг», имеющими значение 01111110. Флаги используются на уровне MAC для определения границ кадра LLC. В соответствии с многоуровневой структурой протоколов стандартов IEEE 802, кадр LLC вкладывается в кадр уровня MAC: кадр Ethernet, Token Ring, FDDI и т. д. При этом флаги кадра LLC отбрасываются.

Кадр LLC содержит поле данных и заголовок, который состоит из трех полей:

- адрес точки входа службы назначения (Destination Service Access Point, DSAP);
- адрес точки входа службы источника (Source Service Access Point, SSAP);
- управляющее поле (Control).

Поле данных кадра LLC предназначено для передачи по сети пакетов протоколов вышележащих уровней – сетевых протоколов IP, IPX, AppleTalk, DECnet, в редких случаях – прикладных протоколов, когда те вкладывают свои сообщения непосредственно в кадры канального уровня. Поле данных может отсутствовать в управляющих кадрах и некоторых ненумерованных кадрах.

Адресные поля DSAP и *SSAP* занимают по 1 байту. Они позволяют указать, какая служба верхнего уровня пересылает данные с помощью этого кадра. Программному обеспечению узлов сети при получении кадров канального уровня необходимо распознать, какой протокол вложил свой пакет в поле данных поступившего кадра, чтобы передать извлеченный из кадра пакет нужному протоколу верхнего уровня для последующей обработки. Для идентификации этих протоколов вводятся так называемые адреса точки входа службы (Service Access Point, SAP). Значения адресов SAP приписываются протоколам в соответствии со стандартом 802.2. Например, для протокола IP значение SAP равно 0x6, для протокола NetBIOS – OxFO. Для одних служб определена только одна точка входа и, соответственно, только один SAP, а для других – несколько, когда адреса DSAP и SSAP совпадают. Например, если в кадре LLC значения DSAP и SSAP содержат код протокола IPX, то обмен кадрами осуществляется между двумя IPX-модулями, выполняющимися в разных узлах. Но в некоторых случаях в кадре LLC указываются различающиеся DSAP и SSAP. Это возможно только в тех случаях, когда служба имеет несколько адресов SAP, что может быть использовано протоколом узла отправителя в специальных целях, например для уведомления узла получателя о переходе протокола-отправителя в некоторый специфический режим работы. Этим свойством протокола LLC часто пользуется протокол NetBEUI.

Поле управления (1 или 2 байта) имеет сложную структуру при работе в режиме LLC2 и достаточно простую структуру при работе в режиме LLC1 (рис. 3.2).

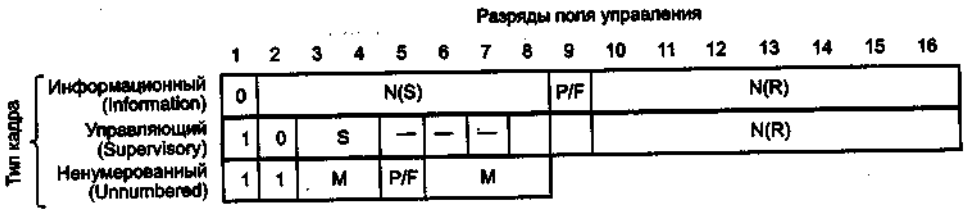


Рис. 3.2. Структура поля управления

В режиме LLC1 используется только один тип кадра – ненумерованный. У этого кадра поле управления имеет длину в один байт. Все подполя поля управления ненумерованных кадров принимают нулевые значения, так что значимыми остаются только первые два бита поля, используемые как признак типа кадра. Учитывая, что в протоколе Ethernet при записи реализован обратный порядок бит в байте, то запись поля управления кадра LLC1, вложенного в кадр протокола Ethernet, имеет значение 0x03 (здесь и далее префикс Ox обозначает шестнадцатеричное представление).

В режиме LLC2 используются все три типа кадров. В этом режиме кадры делятся на команды и ответы на эти команды. Бит P/F (Poll/Final) имеет следующее значение: в командах он называется битом Poll и требует, чтобы на команду был дан ответ, а в ответах он называется битом Final и говорит о том, что ответ состоит из одного кадра.

Ненумерованные кадры используются на начальной стадии взаимодействия двух узлов, а именно стадии установления соединения по протоколу LLC2. Поле M ненумерованных кадров определяет несколько типов команд, которыми пользуются два узла на этапе установления соединения. Ниже приведены примеры некоторых команд.

- *Установить сбалансированный асинхронный расширенный режим* (*SABME*). Эта команда является запросом на установление соединения. Она является одной из команд полного набора команд такого рода протокола HDLC. Расширенный режим означает использование двухбайтных полей управления для кадров остальных двух типов.
- *Ненумерованное подтверждение* (*UA*). Служит для подтверждения установления или разрыва соединения.
- *Сброс соединения* (*REST*). Запрос на разрыв соединения.

После установления соединения данные и положительные квитанции начинают передаваться в информационных кадрах. Логический канал протокола LLC2 является дуплексным, так что данные могут передаваться в обоих направлениях. Если поток дуплексный, то положительные квитанции на кадры также доставляются в информационных кадрах. Если же потока кадров в обратном направлении нет или же нужно передать отрицательную квитанцию, то используются супервизорные кадры.

В информационных кадрах имеется поле N(S) для указания номера отправленного кадра, а также поле N(R) для указания номера кадра, который приемник ожидает получить от передатчика следующим, При работе протокола LLC2 используется скользящее окно размером в 127 кадров, а для их нумерации циклически используется 128 чисел, от 0 до 127.

Приемник всегда помнит номер последнего кадра, принятого от передатчика, и поддерживает переменную с указанным номером кадра, который он ожидает принять от передатчика следующим. Обозначим его через V(R). Именно это значение передается в поле N(R) кадра, посылаемого передатчику. Если в ответ на этот кадр приемник принимает кадр, в котором номер посланного кадра N(S) совпадает с номером ожидаемого кадра V(R), то такой кадр считается корректным (если, конечно, корректна его контрольная сумма). Если приемник принимает кадр с номером N(S), неравным V(R), то этот кадр отбрасывается и посылается отрицательная квитанция *Отказ* (*REJ*) с номером V(R). При приеме отрицательной квитанции передатчик обязан повторить передачу кадра с номером V(R), а также всех кадров с большими номерами, которые он уже успел отослать, пользуясь механизмом окна в 127 кадров.

В состав супервизорных кадров входят следующие:

- *Отказ* (*REJect*);
- *Приемник не готов* (*Receiver Not Ready, RNR*);
- *Приемник готов* (*Receiver Ready, RR*).

Команда RR с номером N(R) часто используется как положительная квитанция, когда поток данных от приемника к передатчику отсутствует, а команда RNR – для замедления потока кадров, поступающих на приемник. Это может быть необходимо, если приемник не успевает обработать поток кадров, присылаемых ему с большой скоростью за счет механизма окна. Получение кадра RNR требует от передатчика полной приостановки передачи, до получения кадра RR. С помощью этих кадров осуществляется управление потоком данных, что особенно важно для коммутируемых сетей, в которых нет разделяемой среды, автоматически тормозящей работу передатчика за счет того, что новый кадр нельзя передать, пока приемник не закончил прием предыдущего.

Выводы

- Протокол LLC обеспечивает для технологий локальных сетей нужное качество транспортной службы, передавая свои кадры либо дейтаграммным способом, либо с помощью процедур с установлением соединения и восстановлением кадров.
- LLC предоставляет верхним уровням три типа процедур: процедуру без установления соединения и без подтверждения; процедуру с установлением соединения и подтверждением;

процедуру без установления соединения, но с подтверждением.

- Логический канал протокола LLC2 является дуплексным, так что данные могут передаваться в обоих направлениях.
- Протокол LLC в режиме с установлением соединения использует алгоритм скользящего окна.
- *Протокол LLC* с помощью управляющих кадров имеет возможность регулировать поток данных, поступающих от узлов сети. Это особенно важно для коммутируемых сетей, в которых нет разделяемой среды, автоматически тормозящей работу передатчика при высокой загрузке сети.

3.3. Технология Ethernet (802.3)

Ethernet – это самый распространенный на сегодняшний день стандарт локальных сетей. Общее количество сетей, работающих по протоколу Ethernet в настоящее время, оценивается в 5 миллионов, а количество компьютеров с установленными сетевыми адаптерами Ethernet – в 50 миллионов.

Когда говорят Ethernet, то под этим обычно понимают любой из вариантов этой технологии. В более узком смысле Ethernet – это сетевой стандарт, основанный на экспериментальной сети Ethernet Network, которую фирма Xerox разработала и реализовала в 1975 году. Метод доступа был опробован еще раньше: во второй половине 60-х годов в радиосети Гавайского университета использовались различные варианты случайного доступа к общей радиосреде, получившие общее название Aloha. В 1980 году фирмы DEC, Intel и Xerox совместно разработали и опубликовали стандарт Ethernet версии II для сети, построенной на основе коаксиального кабеля, который стал последней версией фирменного стандарта Ethernet. Поэтому фирменную версию стандарта Ethernet называют стандартом Ethernet DIX или Ethernet II.

На основе стандарта Ethernet DIX был разработан стандарт IEEE 802.3, который во многом совпадает со своим предшественником, но некоторые различия все же имеются. В то время как в стандарте IEEE 802.3 различаются уровни MAC и LLC, в оригинальном Ethernet оба эти уровня объединены в единый канальный уровень. В Ethernet DIX определяется протокол тестирования конфигурации (Ethernet Configuration Test Protocol), который отсутствует в IEEE 802.3. Несколько отличается и формат кадра, хотя минимальные и максимальные размеры кадров в этих стандартах совпадают. Часто для того, чтобы отличить Ethernet, определенный стандартом IEEE, и фирменный Ethernet DIX, первый называют технологией 802.3, а за фирменным оставляют название Ethernet без дополнительных обозначений.

В зависимости от типа физической среды стандарт IEEE 802.3 имеет различные модификации - 10Base-5, 10Base-2,10Base-T, 10Base-FL, l0Base-FB.

В 1995 году был принят стандарт Fast Ethernet, который во многом не является самостоятельным стандартом, о чем говорит и тот факт, что его описание просто является дополнительным разделом к основному стандарту 802.3 – разделом 802.3u. Аналогично, принятый в 1998 году стандарт Gigabit Ethernet описан в разделе 802.3z основного документа.

Для передачи двоичной информации по кабелю для всех вариантов физического уровня технологии Ethernet, обеспечивающих пропускную способность 10 Мбит/с, используется манчестерский код.

Все виды стандартов Ethernet (в том числе Fast Ethernet и Gigabit Ethernet) используют один и тот же метод разделения среды передачи данных – метод CSMA/CD.

3.3.1. Метод доступа CSMA/CD

В сетях Ethernet используется метод доступа к среде передачи данных, называемый методом коллективного доступа с опознаванием несущей и обнаружением коллизий (carrier-sense-multiply-access with collision detection, CSMA/CD).

Этот метод применяется исключительно в сетях с логической общей шиной (к которым относятся и радиосети, породившие этот метод). Все компьютеры такой сети имеют непосредственный доступ к общей шине, поэтому она может быть использована для передачи данных между любыми двумя узлами сети. Одновременно все компьютеры сети имеют возможность немедленно (с учетом задержки распространения сигнала по физической среде) получить данные, которые любой из компьютеров начал передавать на общую шину (рис. 3.3). Простота схемы подключения – это один из факторов, определивших успех стандарта Ethernet. Говорят, что кабель, к которому подключены все станции, работает в режиме коллективного доступа (Multiply Access, MA).



Рис. 3.3. Метод случайного доступа CSMA/CD

Этапы доступа к среде

Все данные, передаваемые по сети, помещаются в кадры определенной структуры и снабжаются уникальным адресом станции назначения.

Чтобы получить возможность передавать кадр, станция должна убедиться, что разделяемая среда свободна. Это достигается прослушиванием основной гармоники сигнала, которая также называется несущей частотой (carrier-sense, CS). Признаком незанятости среды является отсутствие на ней несущей частоты, которая при манчестерском способе кодирования равна 5-10 МГц, в зависимости от последовательности единиц и нулей, передаваемых в данный момент.

Если среда свободна, то узел имеет право начать передачу кадра. Этот кадр изображен на рис. 3.3 первым. Узел 1 обнаружил, что среда свободна, и начал передавать свой кадр. В классической сети Ethernet на коаксиальном кабеле сигналы передатчика узла 1 распространяются в обе стороны, так что все узлы сети их получают. Кадр данных всегда сопровождается *преамбулой* (*preamble*), которая состоит из 7 байт, состоящих из значений 10101010, и 8-го байта, равного 10101011. Преамбула нужна для вхождения приемника в побитовый и побайтовый синхронизм с передатчиком.

Все станции, подключенные к кабелю, могут распознать факт передачи кадра, и та станция, которая узнает собственный адрес в заголовках кадра, записывает его содержимое в свой внутренний буфер, обрабатывает полученные данные, передает их вверх по своему стеку, а затем посылает по кабелю кадр-ответ. Адрес станции-источника содержится в исходном кадре, поэтому станция-получатель знает, кому нужно послать ответ.

Узел 2 во время передачи кадра узлом 1 также пытался начать передачу своего кадра, однако обнаружил, что среда занята – на ней присутствует несущая частота, – поэтому узел 2 вынужден ждать, пока узел 1 не прекратит передачу кадра.

После окончания передачи кадра все узлы сети обязаны выдержать технологическую паузу (Inter Packet Gap) в 9,6 мкс. Эта пауза, называемая также межкадровым интервалом, нужна для приведения сетевых адаптеров в исходное состояние, а также для предотвращения монопольного захвата среды одной станцией. После окончания технологической паузы узлы имеют право начать передачу своего кадра, так как среда свободна. Из-за задержек распространения сигнала по кабелю не все узлы строго одновременно фиксируют факт окончания передачи кадра узлом 1.

В приведенном примере узел 2 дождался окончания передачи кадра узлом 1, сделал паузу в 9,6 мкс и начал передачу своего кадра.

Возникновение коллизии

При описанном подходе возможна ситуация, когда две станции одновременно пытаются передать кадр данных по общей среде. Механизм прослушивания среды и пауза между кадрами не гарантируют от возникновения такой ситуации, когда две или более станции одновременно решают, что среда свободна, и начинают передавать свои кадры. Говорят, что при этом происходит *коллизия* (*collision*), так как содержимое обоих кадров сталкивается на общем кабеле и происходит искажение информации – методы кодирования, используемые в Ethernet, не позволяют выделять сигналы каждой станции из

общего сигнала.

ПРИМЕЧАНИЕ Заметим, что этот факт отражен в составляющей «Base(band)», присутствующей в названиях всех физических протоколов технологии Ethernet (например, 10Base-2, 10Base-T и т. п.). Baseband network означает сеть с немодулированной передачей, в которой сообщения пересылаются в цифровой форме по единственному каналу, без частотного разделения.

Коллизия – это нормальная ситуация в работе сетей Ethernet. В примере, изображенном на рис. 3.4, коллизию породила одновременная передача данных узлами 3 и 1. Для возникновения коллизии не обязательно, чтобы несколько станций начали передачу абсолютно одновременно, такая ситуация маловероятна. Гораздо вероятней, что коллизия возникает из-за того, что один узел начинает передачу раньше другого, но до второго узла сигналы первого просто не успевают дойти к тому времени, когда второй узел решает начать передачу своего кадра. То есть коллизии – это следствие распределенного характера сети.

Чтобы корректно обработать коллизию, все станции одновременно наблюдают за возникающими на кабеле сигналами. Если передаваемые и наблюдаемые сигналы отличаются, то фиксируется *обнаружение коллизии* (*collision detection, CD*). Для увеличения вероятности скорейшего обнаружения коллизии всеми станциями сети станция, которая обнаружила, коллизию, прерывает передачу своего кадра (в произвольном месте, возможно, и не на границе байта) и усиливает ситуацию коллизии посылкой в сеть специальной последовательности из 32 бит, называемой *jam-последовательностью*.

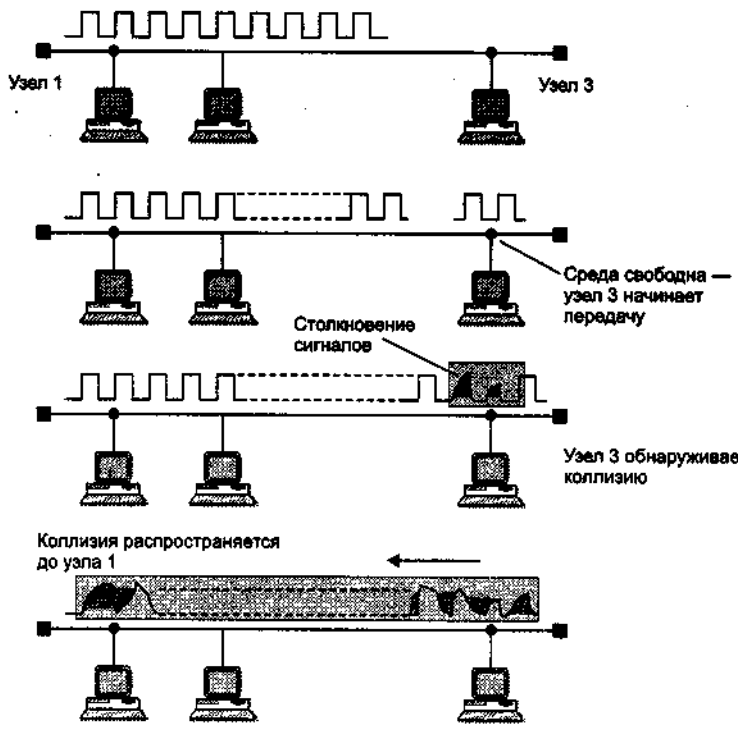


Рис. 3.4. Схема возникновения и распространения коллизии

После этого обнаружившая коллизию передающая станция обязана прекратить передачу и сделать паузу в течение короткого случайного интервала времени. Затем она может снова предпринять попытку захвата среды и передачи кадра. Случайная пауза выбирается по следующему алгоритму:

Пауза = L × (интервал отсрочки),

где интервал отсрочки равен 512 битовым интервалам (в технологии Ethernet принято все интервалы измерять в битовых интервалах; битовый интервал обозначается как bt и соответствует времени между появлением двух последовательных бит данных на кабеле; для скорости 10 Мбит/с величина битового интервала равна 0,1 мкс или 100 нс);

L представляет собой целое число, выбранное с равной вероятностью из диапазона [O, 2^N], где N – номер повторной попытки передачи данного кадра: 1,2,..., 10.

После 10-й попытки интервал, из которого выбирается пауза, не увеличивается. Таким образом, случайная пауза может принимать значения от 0 до 52,4 мс.

Если 16 последовательных попыток передачи кадра вызывают коллизию, то передатчик должен прекратить попытки и отбросить этот кадр.

Из описания метода доступа видно, что он носит вероятностный характер, и вероятность успешного получения в свое распоряжение общей среды зависит от загруженности сети, то есть от интенсивности возникновения в станциях потребности в передаче кадров. При разработке этого метода в конце 70-х годов предполагалось, что скорость передачи данных в 10 Мбит/с очень высока по сравнению с потребностями компьютеров во взаимном обмене данными, поэтому загрузка сети будет всегда небольшой. Это предположение остается иногда справедливым и по сей день, однако уже появились приложения, работающие в реальном масштабе времени с мультимедийной информацией, которые очень загружают сегменты Ethernet. При этом коллизии возникают гораздо чаще. При значительной интенсивности коллизий полезная пропускная способность сети Ethernet резко падает, так как сеть почти постоянно занята повторными попытками передачи кадров. Для уменьшения интенсивности возникновения коллизий нужно либо уменьшить трафик, сократив, например, количество узлов в сегменте или заменив приложения, либо повысить скорость протокола, например перейти на Fast Ethernet.

Следует отметить, что метод доступа CSMA/CD вообще не гарантирует станции, что она когда-либо сможет получить доступ к среде. Конечно, при небольшой загрузке сети вероятность такого события невелика, но при коэффициенте использования сети, приближающемся к 1, такое событие становится очень вероятным. Этот недостаток метода случайного доступа – плата за его чрезвычайную простоту, которая сделала технологию Ethernet самой недорогой. Другие методы доступа – маркерный доступ сетей Token Ring и FDDI, метод Demand Priority сетей 100VG-AnyLAN – свободны от этого недостатка.

Время двойного оборота и распознавание коллизий

Четкое распознавание коллизий всеми станциями сети является необходимым условием корректной работы сети Ethernet. Если какая-либо передающая станция не распознает коллизию и решит, что кадр данных ею передан верно, то этот кадр данных будет утерян. Из-за наложения сигналов при коллизии информация кадра исказится, и он будет отбракован принимающей станцией (возможно, из-за несовпадения контрольной суммы). Скорее всего, искаженная информация будет повторно передана каким-либо протоколом верхнего уровня, например транспортным или прикладным, работающим с установлением соединения. Но повторная передача сообщения протоколами верхних уровней произойдет через значительно более длительный интервал времени (иногда даже через несколько секунд) по сравнению с микросекундными интервалами, которыми оперирует протокол Ethernet. Поэтому если коллизии не будут надежно распознаваться узлами сети Ethernet, то это приведет к заметному снижению полезной пропускной способности данной сети.

Для надежного распознавания коллизий должно выполняться следующее соотношение:

$$T_{min} \geq PDV,$$

где T_{min} – время передачи кадра минимальной длины, а PDV – время, за которое сигнал коллизии успевает распространиться до самого дальнего узла сети. Так как в худшем случае сигнал должен пройти дважды между наиболее удаленными друг от друга станциями сети (в одну сторону проходит неискаженный сигнал, а на обратном пути распространяется уже искаженный коллизией сигнал), то это время называется *временем двойного оборота* (*Path Delay Value, PDV*).

При выполнении этого условия передающая станция должна успевать обнаружить коллизию, которую вызвал переданный ее кадр, еще до того, как она закончит передачу этого кадра.

Очевидно, что выполнение этого условия зависит, с одной стороны, от длины минимального кадра и пропускной способности сети, а с другой, стороны, от длины кабельной системы сети и скорости распространения сигнала в кабеле (для разных типов кабеля эта скорость несколько отличается).

Все параметры протокола Ethernet подобраны таким образом, чтобы при нормальной работе узлов сети коллизии всегда четко распознавались. При выборе параметров, конечно, учитывалось и приведенное выше соотношение, связывающее между собой минимальную длину кадра и максимальное расстояние между станциями в сегменте сети.

В стандарте Ethernet принято, что минимальная длина поля данных кадра составляет 46 байт (что вместе со служебными полями дает минимальную длину кадра 64 байт, а вместе с преамбулой – 72 байт

или 576 бит). Отсюда может быть определено ограничение на расстояние между станциями.

Итак, в 10-мегабитном Ethernet время передачи кадра минимальной длины равно 575 битовых интервалов, следовательно, время двойного оборота должно быть меньше 57,5 мкс. Расстояние, которое сигнал может пройти за это время, зависит от типа кабеля и для толстого коаксиального кабеля равно примерно 13 280 м. Учитывая, что за это время сигнал должен пройти по линии связи дважды, расстояние между двумя узлами не должно быть больше 6 635 м. В стандарте величина этого расстояния выбрана существенно меньше, с учетом других, более строгих ограничений.

Одно из таких ограничений связано с предельно допустимым затуханием сигнала. Для обеспечения необходимой мощности сигнала при его прохождении между наиболее удаленными друг от друга станциями сегмента кабеля максимальная длина непрерывного сегмента толстого коаксиального кабеля с учетом вносимого им затухания выбрана в 500 м. Очевидно, что на кабеле в 500 м условия распознавания коллизий будут выполняться с большим запасом для кадров любой стандартной длины, в том числе и 72 байт (время двойного оборота по кабелю 500 м составляет всего 43,3 битовых интервала). Поэтому минимальная длина кадра могла бы быть установлена еще меньше. Однако разработчики технологии не стали уменьшать минимальную длину кадра, имея в виду многосегментные сети, которые строятся из нескольких сегментов, соединенных повторителями.

Повторители увеличивают мощность передаваемых с сегмента на сегмент сигналов, в результате затухание сигналов уменьшается и можно использовать сеть гораздо большей длины, состоящую из нескольких сегментов. В коаксиальных реализациях Ethernet разработчики ограничили максимальное количество сегментов в сети пятью, что в свою очередь ограничивает общую длину сети 2500 метрами. Даже в такой многосегментной сети условие обнаружения коллизий по-прежнему выполняется с большим запасом (сравним полученное из условия допустимого затухания расстояние в 2500 м с вычисленным выше максимально возможным по времени распространения сигнала расстоянием 6635 м). Однако в действительности временной запас является существенно меньше, поскольку в многосегментных сетях сами повторители вносят в распространение сигнала дополнительную задержку в несколько десятков битовых интервалов. Естественно, небольшой запас был сделан также для компенсации отклонений параметров кабеля и повторителей.

В результате учета всех этих и некоторых других факторов было тщательно подобрано соотношение между минимальной длиной кадра и максимально возможным расстоянием между станциями сети, которое обеспечивает надежное распознавание коллизий. Это расстояние называют также максимальным диаметром сети.

С увеличением скорости передачи кадров, что имеет место в новых стандартах, базирующихся на том же методе доступа CSMA/CD, например Fast Ethernet, максимальное расстояние между станциями сети уменьшается пропорционально увеличению скорости передачи. В стандарте Fast Ethernet оно составляет около 210 м, а в стандарте Gigabit Ethernet оно было бы ограничено 25 метрами, если бы разработчики стандарта не предприняли некоторых мер по увеличению минимального размера пакета.

В табл. 3.1 приведены значения основных параметров процедуры передачи кадра стандарта 802.3, которые не зависят от реализации физической среды. Важно отметить, что каждый вариант физической среды технологии Ethernet добавляет к этим ограничениям свои, часто более строгие ограничения, которые также должны выполняться и которые будут рассмотрены ниже.

Таблица 3.1. Параметры уровня MAC Ethernet

Параметры	Значения
Битовая скорость	10Мбит/с
Интервал отсрочки	512 битовых интервала
Межкадровый интервал (IPG)	9,6 мкс
Максимальное число попыток передачи	16
Максимальное число возрастания диапазона паузы	10
Длина jam-последовательности	32 бита
Максимальная длина кадра (без преамбулы)	1518 байт
Минимальная длина кадра (без преамбулы)	64 байт (512 бит)
Длина преамбулы	64 бит
Минимальная длина случайной паузы после коллизии	0 битовых интервалов
Максимальная длина случайной паузы после коллизии	524000 битовых интервала
Максимальное расстояние между станциями сети	2500 м

Максимальное число станций в сети	1024

3.3.2. Максимальная производительность сети Ethernet

Количество обрабатываемых кадров Ethernet в секунду часто указывается производителями мостов/коммутаторов и маршрутизаторов как основная характеристика производительности этих устройств. В свою очередь, интересно знать чистую максимальную пропускную способность сегмента Ethernet в кадрах в секунду в идеальном случае, когда в сети нет коллизий и нет дополнительных задержек, вносимых мостами и маршрутизаторами. Такой показатель помогает оценить требования к производительности коммуникационных устройств, так как в каждый порт устройства не может поступать больше кадров в единицу времени, чем позволяет это сделать соответствующий протокол.

Для коммуникационного оборудования наиболее тяжелым режимом является обработка кадров минимальной длины. Это объясняется тем, что на обработку каждого кадра мост, коммутатор или маршрутизатор тратит примерно одно и то же время, связанное с просмотром таблицы продвижения пакета, формированием нового кадра (для маршрутизатора) и т. п. А количество кадров минимальной длины, поступающих на устройство в единицу времени, естественно больше, чем кадров любой другой длины. Другая характеристика производительности коммуникационного оборудования — бит в секунду используется реже, так как она не говорит о том, какого размера кадры при этом обрабатывало устройство, а на кадрах максимального размера достичь высокой производительности, измеряемой в битах в секунду гораздо легче.

Используя параметры, приведенные в табл. 3.1, рассчитаем максимальную производительность сегмента Ethernet в таких единицах, как число переданных кадров (пакетов) минимальной длины в секунду

ПРИМЕЧАНИЕ При указании пропускной способности сетей термины кадр и пакет обычно используются как синонимы. Соответственно, аналогичными являются и единицы измерения производительности frames-per-second, fps и packets-per-second, pps.

Для расчета максимального количества кадров минимальной длины, проходящих по сегменту Ethernet, заметим, что размер кадра минимальной длины вместе с преамбулой составляет 72 байт или 576 бит (рис. 3.5.), поэтому на его передачу затрачивается 57,5 мкс. Прибавив межкадровый интервал в 9,6 мкс, получаем, что период следования кадров минимальной длины составляет 67,1 мкс. Отсюда максимально возможная пропускная способность сегмента Ethernet составляет 14 880 кадр/с.



Рис. 3.5. К расчету пропускной способности протокола Ethernet

Естественно, что наличие в сегменте нескольких узлов снижает эту величину за счет ожидания доступа к среде, а также за счет коллизий, приводящих к необходимости повторной передачи кадров.

Кадры максимальной длины технологии Ethernet имеют поле длины 1500 байт, что вместе со служебной информацией дает 1518 байт, а с преамбулой составляет 1526 байт или 12 208 бит. Максимально возможная пропускная способность сегмента Ethernet для кадров максимальной длины составляет 813 кадр/с. Очевидно, что при работе с большими кадрами нагрузка на мосты, коммутаторы и маршрутизаторы довольно ощутимо снижается.

Теперь рассчитаем, какой максимальной полезной пропускной способностью в бит в секунду обладают сегменты Ethernet при использовании кадров разного размера.

Под *полезной пропускной способностью протокола* понимается скорость передачи пользовательских данных, которые переносятся полем данных кадра. Эта пропускная способность всегда меньше номинальной битовой скорости протокола Ethernet за счет нескольких факторов:

- служебной информации кадра;
- межкадровых интервалов (IPG);

- ожидания доступа к среде.

Для кадров минимальной длины полезная пропускная способность равна:

$C_п = 14880 \times 46 \times 8 = 5{,}48$ Мбит/с.

Это намного меньше 10 Мбит/с, но следует учесть, что кадры минимальной длины используются в основном для передачи квитанций, так что к передаче собственно данных файлов эта скорость отношения не имеет.

Для кадров максимальной длины полезная пропускная способность равна:

$C_п = 813 \times 1500 \times 8 = 9{,}76$ Мбит/с,

что весьма близко к номинальной скорости протокола.

Еще раз подчеркнем, что такой скорости можно достигнуть только в том случае, когда двум взаимодействующим узлам в сети Ethernet другие узлы не мешают, что бывает крайне редко.

При использовании кадров среднего размера с полем данных в 512 байт пропускная способность сети составит 9,29 Мбит/с, что тоже достаточно близко к предельной пропускной способности в 10 Мбит/с.

ВНИМАНИЕ Отношение текущей пропускной способности сети к ее максимальной пропускной способности называется коэффициентом использования сети (network utilization). При этом при определении текущей пропускной способности принимается во внимание передача по сети любой информации, как пользовательской, так и служебной. Коэффициент является важным показателем для технологий разделяемых сред, так как при случайном характере метода доступа высокое значение коэффициента использования часто говорит о низкой полезной пропускной способности сети (то есть скорости передачи пользовательских данных) – слишком много времени узлы тратят на процедуру получения доступа и повторные передачи кадров после коллизий.

При отсутствии коллизий и ожидания доступа коэффициент использования сети зависит от размера поля данных кадра и имеет максимальное значение 0,976 при передаче кадров максимальной длины. Очевидно, что в реальной сети Ethernet среднее значение коэффициента использования сети может значительно отличаться от этой величины. Более сложные случаи определения пропускной способности сети с учетом ожидания доступа и отработки коллизий будут рассмотрены ниже.

3.3.3. Форматы кадров технологии Ethernet

Стандарт технологии Ethernet, описанный в документе IEEE 802.3, дает описание единственного формата кадра уровня MAC. Так как в кадр уровня MAC должен вкладываться кадр уровня LLC, описанный в документе IEEE 802.2, то по стандартам IEEE в сети Ethernet может использоваться только единственный вариант кадра канального уровня, заголовок которого является комбинацией заголовков MAC и LLC подуровней.

Тем не менее на практике в сетях Ethernet на канальном уровне используются кадры 4-х различных форматов (типов). Это связано с длительной историей развития технологии Ethernet, насчитывающей период существования до принятия стандартов IEEE 802, когда подуровень LLC не выделялся из общего протокола и, соответственно, заголовок LLC не применялся.

Консорциум трех фирм Digital, Intel и Xerox в 1980 году представил на рассмотрение комитету 802.3 свою фирменную версию стандарта Ethernet (в которой был, естественно, описан определенный формат кадра) в качестве проекта международного стандарта, но комитет 802.3 принял стандарт, отличающийся в некоторых деталях от предложения DIX. Отличия касались и формата кадра, что породило существование двух различных типов кадров в сетях Ethernet.

Еще один формат кадра появился в результате усилий компании Novell по ускорению работы своего стека протоколов в сетях Ethernet.

И наконец, четвертый формат кадра стал результатом деятельности комитета 802.2 по приведению предыдущих форматов кадров к некоторому общему стандарту.

Различия в форматах кадров могут приводить к несовместимости в работе аппаратуры и сетевого программного обеспечения, рассчитанного на работу только с одним стандартом кадра Ethernet. Однако сегодня практически все сетевые адаптеры, драйверы сетевых адаптеров, мосты/коммутаторы и маршрутизаторы умеют работать со всеми используемыми на практике форматами кадров технологии Ethernet, причем распознавание типа кадра выполняется автоматически.

Ниже приводится описание всех четырех типов кадров Ethernet (здесь под кадром понимается весь набор полей, которые относятся к канальному уровню, то есть поля MAC и LLC уровней). Один и тот же тип кадра может иметь разные названия, поэтому ниже для каждого типа кадра приведено по нескольку наиболее употребительных названий:

- кадр 802.3/LLC (кадр 802.3/802.2 или кадр Novell 802.2);
- кадр Raw 802.3 (или кадр Novell 802.3);
- кадр Ethernet DIX (или кадр Ethernet II);
- кадр Ethernet SNAP.

Форматы всех этих четырех типов кадров Ethernet приведены на рис. 3.6.

Кадр 802.3/LLC

Заголовок кадра 802.3/LLC является результатом объединения полей заголовков кадров, определенных в стандартах IEEE 802.3 и 802.2.

Стандарт 802.3 определяет восемь полей заголовка (рис. 3.6; поле преамбулы и начальный ограничитель кадра на рисунке не показаны).

- *Поле преамбулы* (*Preamble*) состоит из семи синхронизирующих байт 10101010. При манчестерском кодировании эта комбинация представляется в физической среде периодическим волновым сигналом с частотой 5 МГц.
- *Начальный ограничитель кадра* (*Start-of-frame-delimiter, SFD*) состоит из одного байта 10101011. Появление этой комбинации бит является указанием на то, что следующий байт – это первый байт заголовка кадра.
- *Адрес назначения* (*Destination Address, DA*) может быть длиной 2 или 6 байт. На практике всегда используются адреса из 6 байт. Первый бит старшего байта адреса назначения является признаком того, является адрес индивидуальным или групповым. Если он равен 0, то адрес является *индивидуальным* (*unicast*), а если 1, то это *групповой адрес* (*multicast*). Групповой адрес может предназначаться всем узлам сети или же определенной группе узлов сети. Если адрес состоит из всех единиц, то есть имеет шестнадцатеричное представление OxFFFFFFFFFFFF, то он предназначается всем узлам сети и называется *широковещательным адресом* (*broadcast*). В остальных случаях групповой адрес связан только с теми узлами, которые сконфигурированы (например, вручную) как члены группы, номер которой указан в групповом адресе. Второй бит, старшего байта адреса определяет способ назначения адреса – централизованный или локальный. Если этот бит равен 0 (что бывает почти всегда в стандартной аппаратуре Ethernet), то адрес назначен централизованно, с помощью комитета IEEE. Комитет IEEE распределяет между производителями оборудования так называемые организационно уникальные идентификаторы (Organizationally Unique Identifier, OUI). Этот идентификатор помещается в 3 старших байта адреса (например, идентификатор 000081 определяет компанию Bay Networks). За уникальность младших 3-х байт адреса отвечает производитель оборудования. Двадцать четыре бита, отводимые производителю для адресации интерфейсов его продукции, позволяют выпустить 16 миллионов интерфейсов под одним идентификатором организации. Уникальность централизованно распределяемых адресов распространяется на все основные технологии локальных сетей – Ethernet, Token Ring, FDDI и т. д.

ВНИМАНИЕ В стандартах IEEE Ethernet младший бит байта изображается в самой левой позиции поля, а старший бит -в самой правой. Этот нестандартный способ отображения порядка бит в байте соответствует порядку передачи бит в линию связи передатчиком Ethernet. В стандартах других организаций, например RFC IETF, ITU-T, ISO, используется традиционное представление байта, когда младший бит считается самым правым битом байта, а старший - самым левым. При этом порядок следования байтов остается традиционным Поэтому при чтении стандартов, опубликованных этими организациями, а также чтении данных, отображаемых на экране операционной системой или анализатором протоколов, значения каждого байта кадра Ethernet нужно зеркально отобразить, чтобы получить правильное представление о значении разрядов

этого байта в соответствии с документами IEEE. Например, групповой адрес, имеющийся в нотации IEEE вид 1000 0000 0000 0000 10100111 1111 0000 0000 0000 0000 0000 или в шестнадцатеричной записи 80-00-A7-FO-00-00, будет, скорее всего, отображен анализатором протоколов в традиционном виде как 01-00-5E-OF-00-00.

- *Адрес источника* (*Source Address, SA*) – это 2- или 6-байтовое поле, содержащее адрес узла – отправителя кадра. Первый бит адреса всегда имеет значение 0.
- *Длина* (*Length, L*) – 2-байтовое поле, которое определяет длину поля данных в кадре.
- *Поле данных* (*Data*) может содержать от 0 до 1500 байт. Но если длина поля меньше 46 байт, то используется следующее поле – поле заполнения, – чтобы дополнить кадр до минимально допустимого значения в 46 байт.
- *Поле заполнения* (*Padding*) состоит из такого количества байт заполнителей, которое обеспечивает минимальную длину поля данных в 46 байт. Это обеспечивает корректную работу механизма обнаружения коллизий. Если длина поля данных достаточна, то поле заполнения в кадре не появляется.
- *Поле контрольной суммы* (*Frame Check Sequence, FCS*) состоит из 4 байт, содержащих контрольную сумму. Это значение вычисляется по алгоритму CRC-32. После получения кадра рабочая станция выполняет собственное вычисление контрольной суммы для этого кадра, сравнивает полученное значение со значением поля контрольной суммы и, таким образом, определяет, не искажен ли полученный кадр.

Кадр 802.3 является кадром MAC-подуровня, поэтому в соответствии со стандартном 802.2 в его поле данных вкладывается кадр подуровня LLC с удаленными флагами начала и конца кадра. Формат кадра LLC был описан выше. Так как кадр LLC имеет заголовок длиной 3 (в режиме LLC1) или 4 байт (в режиме LLC2), то максимальный размер поля данных уменьшается до 1497 или 1496 байт:

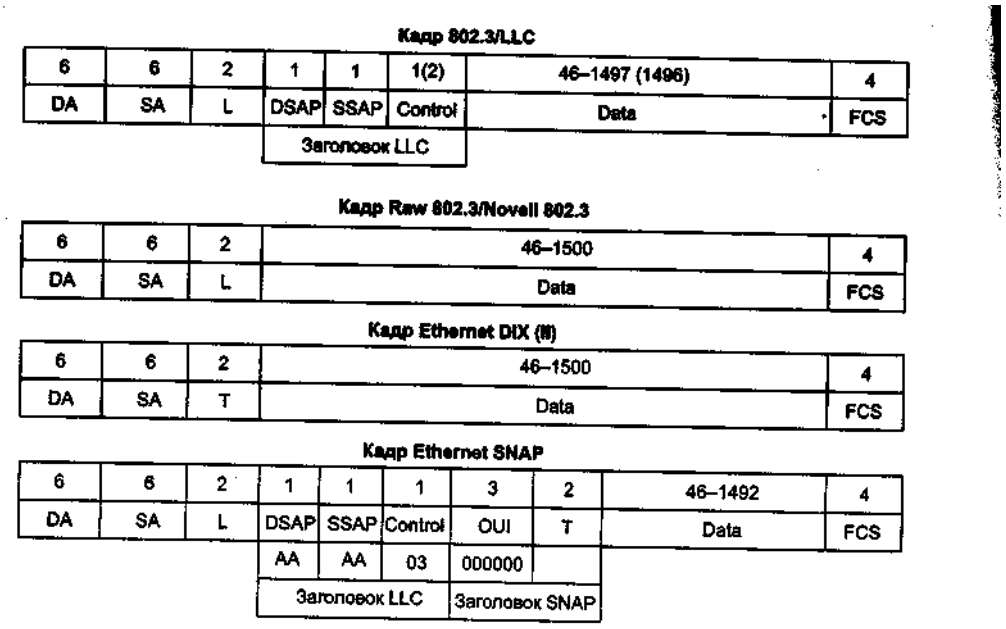


Рис. 3.6. Форматы кадров Ethernet

Кадр Raw 802.3/Novell 802.3

Кадр Raw 802.3, называемый также кадром *Novell 802.3*, представлен на рис. 3.6. Из рисунка видно, что это кадр подуровня MAC стандарта 802.3, но без вложенного кадра подуровня LLC. Компания Novell долгое время не использовала служебные поля кадра LLC в своей операционной системе NetWare из-за отсутствия необходимости идентифицировать тип информации, вложенной в поле данных, – там всегда находился пакет протокола IPX, долгое время бывшего единственным протоколом сетевого уровня в ОС NetWare.

Теперь, когда необходимость идентификации протокола верхнего уровня появилась, компания Novell стала использовать возможность инкапсуляции в кадр подуровня MAC кадра LLC, то есть использовать

стандартные кадры 802.3/LLC. Такой кадр компания обозначает теперь в своих операционных системах как кадр 802.2, хотя он является комбинацией заголовков 802.3 и 802.2.

Кадр Ethernet DIX/Ethernet II

Кадр Ethernet DIX, называемый также *кадром Ethernet II*, имеет структуру (см. рис. 3.6), совпадающую со структурой кадра Raw 802.3. Однако 2-байтовое поле *Длина*(*L*) кадра Raw 802.3 в кадре *Ethernet DIX* используется в качестве поля типа протокола. Это поле, теперь получившее название Type (T) или EtherType, предназначено для тех же целей, что и поля DSAP и SSAP кадра LLC – для указания типа протокола верхнего уровня, вложившего свой пакет в поле данных этого кадра.

В то время как коды протоколов в полях SAP имеют длину в один байт, в поле Type для кода протокола отводятся 2 байта. Поэтому один и тот же протокол в поле SAP и поле Type будет кодироваться в общем случае разными числовыми значениями. Например, протокол IP имеет код 2048_{10} (0x0800) для поля EtherType и значение 6 для поля SAP. Значения кодов протоколов для поля EtherType появились раньше значений SAP, так как фирменная версия Ethernet DIX существовала до появления стандарта 802.3, и ко времени распространения оборудования 802.3 уже стали стандартами де-факто для многих аппаратных и программных продуктов. Так как структуры кадров Ethernet DIX и Raw 802.3 совпадают, то поле длины/типа часто в документации обозначают как поле L/T.

Кадр Ethernet SNAP

Для устранения разнобоя в кодировках типов протоколов, сообщения которых вложены в поле данных кадров Ethernet, комитетом 802.2 была проведена работа по дальнейшей стандартизации кадров Ethernet. В результате появился кадр Ethernet SNAP (SNAP – SubNetwork Access Protocol, протокол доступа к подсетям). Кадр Ethernet SNAP (см. рис. 3.6) представляет собой расширение кадра 802.3/LLC за счет введения дополнительного заголовка протокола SNAP, состоящего из двух полей: OUI и Type. Поле Type состоит из 2-х байт и повторяет по формату и назначению поле Type кадра Ethernet II (то есть в нем используются те же значения кодов протоколов). Поле OUI (Organizationally Unique Identifier) определяет идентификатор организации, которая контролирует коды протоколов в поле Type. С помощью заголовка SNAP достигнута совместимость с кодами протоколов в кадрах Ethernet II, а также создана универсальная схема кодирования протоколов. Коды протоколов для технологий 802 контролирует IEEE, которая имеет OUI, равный 000000. Если в будущем потребуются другие коды протоколов для какой-либо новой технологии, для этого достаточно указать другой идентификатор организации, назначающей эти коды, а старые значения кодов останутся в силе (в сочетании с другим идентификатором OUI).

Так как SNAP представляет собой протокол, вложенный в протокол LLC, то в полях DSAP и SSAP записывается код OxAA, отведенный для протокола SNAP. Поле Control заголовка LLC устанавливается в 0x03, что соответствует использованию ненумерованных кадров.

Заголовок SNAP является дополнением к заголовку LLC, поэтому он допустим не только в кадрах Ethernet, но и в кадрах протоколов других технологий 802. Например, протокол IP всегда использует структуру заголовков LLC/SNAP при инкапсуляции в кадры всех протоколов локальных сетей: FDDI, Token Ring, 100VG-AnyLAN, Ethernet, Fast Ethernet, Gigabit Ethernet.

Правда, при передаче пакетов IP через сети Ethernet, Fast Ethernet и Gigabit Ethernet протокол IP использует кадры Ethernet DIX.

Использование различных типов кадров Ethernet

Автоматическое распознавание типов кадров Ethernet выполняется достаточно несложно. Для кодирования типа протокола в поле EtherType указываются значения, превышающие значение максимальной длины поля данных, равное 1500, поэтому кадры Ethernet II легко отличить от других типов кадров по значению поля L/T. Дальнейшее распознавание типа кадра проводится по наличию или отсутствию полей LLC. Поля LLC могут отсутствовать только в том случае, если за полем длины идет начало пакета IPX, а именно 2-байтовое поле контрольной суммы пакета, которое всегда заполняется единицами, что дает значение в 255 байт. Ситуация, когда поля DSAP и SSAP одновременно содержат такие значения, возникнуть не может, поэтому наличие двух байт 255 говорит о том, что это кадр Raw

802.3. В остальных случаях дальнейший анализ проводится в зависимости от значений полей DSAP и SSAP. Если они равны OxAA, то это кадр Ethernet SNAP, а если нет, то 802.3/LLC.

В табл. 3.2 приведены данные о том, какие типы кадров Ethernet обычно поддерживают реализации популярных протоколов сетевого уровня.

Таблица 3.2. Типы кадров Ethernet, поддерживающие реализации популярных протоколов сетевого уровня

Тип кадра	Сетевые протоколы
Ethernet II	IPX, IP, AppleTalk Phase I
Ethernet 802.3	IPX
Ethernet 802.2	IPX FTAM
Ethernet SNAP	IPX, IP, AppleTalk Phase II

3.3.4. Спецификации физической среды Ethernet

Исторически первые сети технологии Ethernet были созданы на коаксиальном кабеле диаметром 0,5 дюйма. В дальнейшем были определены и другие спецификации физического уровня для стандарта Ethernet, позволяющие использовать различные среды передачи данных. Метод доступа CSMA/CD и все временные параметры остаются одними и теми же для любой спецификации физической среды технологии Ethernet 10 Мбит/с.

Физические спецификации технологии Ethernet на сегодняшний день включают следующие среды передачи данных.

- 10Base-5 – коаксиальный кабель диаметром 0,5 дюйма, называемый «толстым» коаксиалом. Имеет волновое сопротивление 50 Ом. Максимальная длина сегмента – 500 метров (без повторителей).
- 10Base-2 – коаксиальный кабель диаметром 0,25 дюйма, называемый «тонким» коаксиалом. Имеет волновое сопротивление 50 Ом. Максимальная длина сегмента – 185 метров (без повторителей).
- 10Base-T – кабель на основе неэкранированной витой пары (Unshielded Twisted Pair, UTP). Образует звездообразную топологию на основе концентратора. Расстояние между концентратором и конечным узлом – не более 100 м.
- 10Base-F – волоконно-оптический кабель. Топология аналогична топологии стандарта 10Base-T. Имеется несколько вариантов этой спецификации – FOIRL (расстояние до 1000 м), 10Base-FL (расстояние до 2000 м), 10Base-FB (расстояние до 2000 м).

Число 10 в указанных выше названиях обозначает битовую скорость передачи данных этих стандартов – 10 Мбит/с, а слово Base – метод передачи на одной базовой частоте 10 МГц (в отличие от методов, использующих несколько несущих частот, которые называются Broadband – широкополосными). Последний символ в названии стандарта физического уровня обозначает тип кабеля.

Стандарт 10Base-5

Стандарт 10Base-5 в основном соответствует экспериментальной сети Ethernet фирмы Xerox и может считаться классическим Ethernet. Он использует в качестве среды передачи данных коаксиальный кабель с волновым сопротивлением 50 Ом, диаметром центрального медного провода 2,17 мм и внешним диаметром около 10 мм («толстый» Ethernet). Такими характеристиками обладают кабели марок RG-8 и RG-11.

Различные компоненты сети, состоящей из трех сегментов, соединенных повторителями, выполненной на толстом коаксиале, показаны на рис. 3.7.

Кабель используется как моноканал для всех станций. Сегмент кабеля имеет максимальную длину 500 м (без повторителей) и должен иметь на концах согласующие *терминаторы*, сопротивлением 50 Ом, поглощающие распространяющиеся по кабелю сигналы и препятствующие возникновению отраженных сигналов. При отсутствии терминаторов («заглушек») в кабеле возникают стоячие волны,

так что одни узлы получают мощные сигналы, а другие – настолько слабые, что их прием становится невозможным.

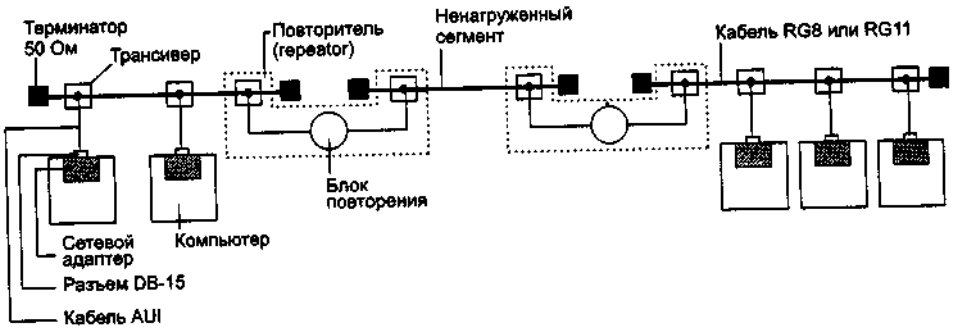


Рис. 3.7. Компоненты физического уровня сети стандарта 10 Base-5, состоящей из трех сегментов

Станция должна подключаться к кабелю при помощи приемопередатчика – *трансивера* (*transmitter+receiver = transceiver*). Трансивер устанавливается непосредственно на кабеле и питается от сетевого адаптера компьютера. Трансивер может подсоединяться к кабелю как методом прокалывания, обеспечивающим непосредственный физический контакт, так и бесконтактным методом.

Трансивер соединяется с сетевым адаптером интерфейсным кабелем *AUI* (*Attachment Unit Interface*) длиной до 50 м, состоящим из 4 витых пар (адаптер должен иметь разъем AUI). Наличие стандартного интерфейса между трансивером и остальной частью сетевого адаптера очень полезно при переходе с одного типа кабеля на другой. Для этого достаточно только заменить трансивер, а остальная часть сетевого адаптера остается неизменной, так как она отрабатывает протокол уровня MAC. При этом необходимо только, чтобы новый трансивер (например, трансивер для витой пары) поддерживал стандартный интерфейс AUI. Для присоединения к интерфейсу AUI используется разъем DB-15.

Допускается подключение к одному сегменту не более 100 трансиверов, причем расстояние между подключениями трансиверов не должно быть меньше 2,5 м. На кабеле имеется разметка через каждые 2,5 м, которая обозначает точки подключения трансиверов. При подсоединении компьютеров в соответствии с разметкой влияние стоячих волн в кабеле на сетевые адаптеры сводится к минимуму.

Трансивер – это часть сетевого адаптера, которая выполняет следующие функции:

- прием и передача данных с кабеля на кабель;
- определение коллизий на кабеле;
- электрическая развязка между кабелем и остальной частью адаптера;
- защита кабеля от некорректной работы адаптера.

Последнюю функцию иногда называют «*контролем болтливости*», что является буквальным переводом соответствующего английского термина (*jabber control*). При возникновении неисправностей в адаптере может возникнуть ситуация, когда на кабель будет непрерывно выдаваться последовательность случайных сигналов. Так как кабель - это общая среда для всех станций, то работа сети будет заблокирована одним неисправным адаптером. Чтобы этого не случилось, на выходе передатчика ставится схема, которая проверяет время передачи кадра. Если максимально возможное время передачи пакета превышается (с некоторым запасом), то эта схема просто отсоединяет выход передатчика от кабеля. Максимальное время передачи кадра (вместе с преамбулой) равно 1221 мкс, а время jabber-контроля устанавливается равным 4000 мкс (4 мс).

Упрощенная структурная схема трансивера показана на рис. 3.8. Передатчик и приемник присоединяются к одной точке кабеля с помощью специальной схемы, например трансформаторной, позволяющей организовать одновременную передачу и прием сигналов с кабеля.

Детектор коллизий определяет наличие коллизии в коаксиальном кабеле по повышенному уровню постоянной составляющей сигналов. Если постоянная составляющая превышает определенный порог (около 1,5 В), значит, на кабель работает более одного передатчика. Развязывающие элементы (РЭ) обеспечивают гальваническую развязку трансивера от остальной части сетевого адаптера и тем самым защищают адаптер и компьютер от значительных перепадов напряжения, возникающих на кабеле при его повреждении.

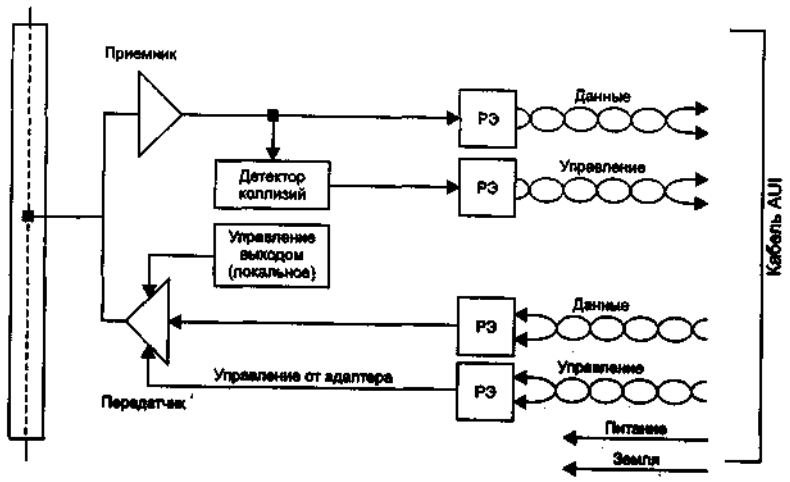


Рис. 3.8. Структурная схема трансивера

Стандарт 10Base-5 определяет возможность использования в сети специального устройства - *повторителя* (*repeator*). Повторитель служит для объединения в одну сеть нескольких сегментов кабеля и увеличения тем самым общей длины сети. Повторитель принимает сигналы из одного сегмента кабеля и побитно синхронно повторяет их в другом сегменте, улучшая форму и мощность импульсов, а также синхронизируя импульсы. Повторитель состоит из двух (или нескольких) трансиверов, которые присоединяются к сегментам кабеля, а также блока повторения со своим тактовым генератором. Для лучшей синхроннизации передаваемых бит повторитель задерживает передачу нескольких первых бит преамбулы кадра, за счет чего увеличивается задержка передачи кадра с сегмента на сегмент, а также несколько уменьшается межкадровый интервал IPG.

Стандарт разрешает использование в сети не более 4 повторителей и, соответственно, не более 5 сегментов кабеля. При максимальной длине сегмента кабеля в 500 м это дает максимальную длину сети 10Base-5 в 2500 м. Только 3 сегмента из 5 могут быть нагруженными, то есть такими, к которым подключаются конечные узлы. Между нагруженными сегментами должны быть ненагруженные сегменты, так что максимальная конфигурация сети представляет собой два нагруженных крайних сегмента, которые соединяются ненагруженными сегментами еще с одним центральным нагруженным сегментом. На рис. 3.7 был приведен пример сети Ethernet, состоящей из трех сегментов, объединенных двумя повторителями. Крайние сегменты являются нагруженными, а промежуточный – ненагруженным.

Правило применения повторителей в сети Ethernet 10Base-5 носит название «*правило 5-4-3*»: 5 сегментов, 4 повторителя, 3 нагруженных сегмента. Ограниченное число повторителей объясняется дополнительными задержками распространения сигнала, которые они вносят. Применение повторителей увеличивает время двойного распространения сигнала, которое для надежного распознавания коллизий не должно превышать время передачи кадра минимальной длины, то есть кадра в 72 байт или 576 бит.

Каждый повторитель подключается к сегменту одним своим трансивером, поэтому к нагруженным сегментам можно подключить не более 99 узлов. Максимальное число конечных узлов в сети 10Base-5 таким образом составляет 99x3 = 297 узлов.

К *достоинствам* стандарта 10Base-5 относятся:

- хорошая защищенность кабеля от внешних воздействий;
- сравнительно большое расстояние между узлами;
- возможность простого перемещения рабочей станции в пределах длины кабеля AUI.

Недостатками 10Base-5 являются:

- высокая стоимость кабеля;
- сложность его прокладки из-за большой жесткости;
- потребность в специальном инструменте для заделки кабеля;
- останов работы всей сети при повреждении кабеля или плохом соединении;
- необходимость заранее предусмотреть подводку кабеля ко всем возможным местам установки компьютеров.

Стандарт 10 Base-2

Стандарт 10Base-2 использует в качестве передающей среды коаксиальный кабель с диаметром центрального медного провода 0,89 мы и внешним диаметром около 5 мм («тонкий» Ethernet). Кабель имеет волновое сопротивление 50 Ом. Такими характеристиками обладают кабели марок RG-58 /U, RG-58 A/U, RG-58 C/U.

Максимальная длина сегмента без повторителей составляет 185 м, сегмент должен иметь на концах согласующие терминаторы 50 Ом. Тонкий коаксиальный кабель дешевле толстого, из-за чего сети 10Base-2 иногда называют сетями Cheapernet (от cheaper – более дешевый). Но за дешевизну кабеля приходится расплачиваться качеством – «тонкий» коаксиал обладает худшей помехозащищенностью, худшей механической прочностью и более узкой полосой пропускания.

Станции подключаются к кабелю с помощью высокочастотного BNC Т-коннектора, который представляет собой тройник, один отвод которого соединяется с сетевым адаптером, а два других – с двумя концами разрыва кабеля. Максимальное количество станций, подключаемых к одному сегменту, – 30. Минимальное расстояние между станциями –1м. Кабель «тонкого» коаксиала имеет разметку для подключения узлов с шагом в 1 м.

Стандарт 10Base-2 также предусматривает использование повторителей, применение которых также должно соответствовать «правилу 5-4-3». В этом случае сеть будет иметь максимальную длину в 5x185 = 925 м. Очевидно, что это ограничение является более сильным, чем общее ограничение в 2500 метров.

ВНИМАНИЕ Для построения корректной сети Ethernet нужно соблюсти много ограничений, причем некоторые из них относятся к одним и тем же параметрам сети - например, максимальная длина или максимальное количество компьютеров в сети должны удовлетворять одновременно нескольким разным условиям. Корректная сеть Ethernet должна соответствовать всем требованиям, но на практике нужно удовлетворить только наиболее жесткие. Так, если в сети Ethernet не должно быть более 1024 узлов, а стандарт lOBase-2 ограничивает число нагруженных сегментов тремя, то общее количество узлов в сети 10Base-2 не должно превышать 29x3 = 87. Менее жесткое ограничение в 1024 конечных узла в сети 10Base-2 никогда не достигается.

Стандарт 10Base-2 очень близок к стандарту 10Base-5. Но трансиверы в нем объединены с сетевыми адаптерами за счет того, что более гибкий тонкий коаксиальный кабель может быть подведен непосредственно к выходному разъему платы сетевого адаптера, установленной в шасси компьютера. Кабель в данном случае «висит» на сетевом адаптере, что затрудняет физическое перемещение компьютеров.

Типичный состав сети стандарта 10Base-2, состоящей из одного сегмента кабеля, показан на рис. 3.9.

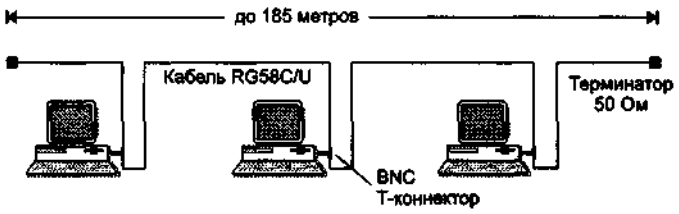


Рис. 3.9. Сеть стандарта 10Base-2

Реализация этого стандарта на практике приводит к наиболее простому решению для кабельной сети, так как для соединения компьютеров требуются только сетевые адаптеры, Т-коннекторы и терминаторы 50 Ом. Однако этот вид кабельных соединений наиболее сильно подвержен авариям и сбоим: кабель более восприимчив к помехам, чем «толстый» коаксиал, в моноканале имеется большое количество механических соединений (каждый Т-коннектор дает три механических соединения, два из которых имеют жизненно важное значение для всей сети), пользователи имеют доступ к разъемам и могут нарушить целостность моноканала. Кроме того, эстетика и эргономичность этого решения оставляют желать лучшего, так как от каждой станции через Т-коннектор отходят два довольно заметных провода, которые под столом часто образуют моток кабеля – запас, необходимый на случай даже небольшого

перемещения рабочего места.

Общим недостатком стандартов 10Base-5 и 10Base-2 является отсутствие оперативной информации о состоянии моноканала. Повреждение кабеля обнаруживается сразу же (сеть перестает работать), но для поиска отказавшего отрезка кабеля необходим специальный прибор – кабельный тестер.

Стандарт 10 Base-T

Стандарт принят в 1991 году, как дополнение к существующему набору стандартов Ethernet, и имеет обозначение 802.3i.

Сети 10Base-T используют в качестве среды две *неэкранированные витые пары* (*Unshielded Twisted Pair, UTP*). Многопарный кабель на основе неэкранированной витой пары категории 3 (категория определяет полосу пропускания кабеля, величину перекрестных наводок NEXT и некоторые другие параметры его качества) телефонные компании уже достаточно давно использовали для подключения телефонных аппаратов внутри зданий. Этот кабель носит также название Voice Grade, говорящее о том, что он предназначен для передачи голоса.

Идея приспособить этот популярный вид кабеля для построения локальных сетей оказалась очень плодотворной, так как многие здания уже были оснащены нужной кабельной системой. Оставалось разработать способ подключения сетевых адаптеров и прочего коммуникационного оборудования к витой паре таким образом, чтобы изменения в сетевых адаптерах и программном обеспечении сетевых операционных систем были бы минимальными по сравнению с сетями Ethernet на коаксиале. Это удалось, поэтому переход на витую пару требует только замены трансивера сетевого адаптера или порта маршрутизатора, а метод доступа и все протоколы канального уровня остались теми же, что и в сетях Ethernet на коаксиале.

Конечные узлы соединяются по топологии «точка-точка» со специальным устройством – многопортовым повторителем с помощью двух витых пар. Одна витая пара требуется для передачи данных от станции к повторителю (выход Tx сетевого адаптера), а другая – для передачи данных от повторителя к станции (вход R_x сетевого адаптера). На рис. 3.10 показан пример трехпортового повторителя. Повторитель принимает сигналы от одного из конечных узлов и синхронно передает их на все свои остальные порты, кроме того, с которого поступили сигналы.

Многопортовые повторители в данном случае обычно называются концентраторами (англоязычные термины - hub или concentrator). Концентратор осуществляет функции повторителя сигналов на всех отрезках витых пар, подключенных к его портам, так что образуется единая среда передачи данных – логический моноканал (логическая общая шина). Повторитель обнаруживает коллизию в сегменте в случае одновременной передачи сигналов по нескольким своим Rx-входам и посылает jam-последовательность на все свои Tx-выходы. Стандарт определяет битовую скорость передачи данных 10 Мбит/с и максимальное расстояние отрезка витой пары между двумя непосредственно связанными узлами (станциями и концентраторами) не более 100 м при наличии витой пары качества не ниже категории 3. Это расстояние определяется полосой пропускания витой пары – на длине 100 м она позволяет передавать данные со скоростью 10 Мбит/с при использовании манчестерского кода.

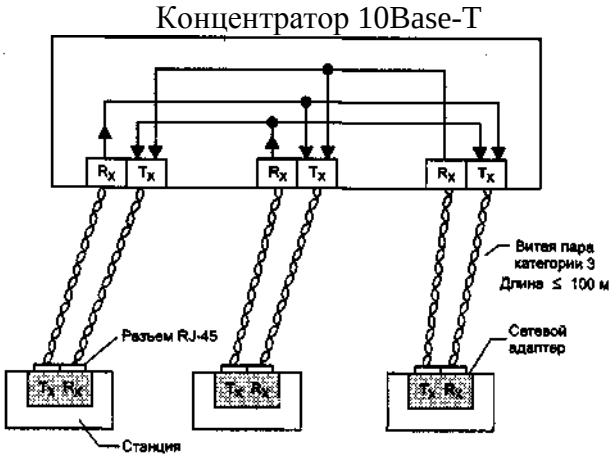


Рис. 3.10. Сеть стандарта 10Base-T: T_x – передатчик; R_x – приемник

Концентраторы 10Base-T можно соединять друг с другом с помощью тех же портов, которые предназначены для подключения конечных узлов. При этом нужно позаботиться о том, чтобы передатчик и приемник одного порта были соединены соответственно с приемником и передатчиком другого порта.

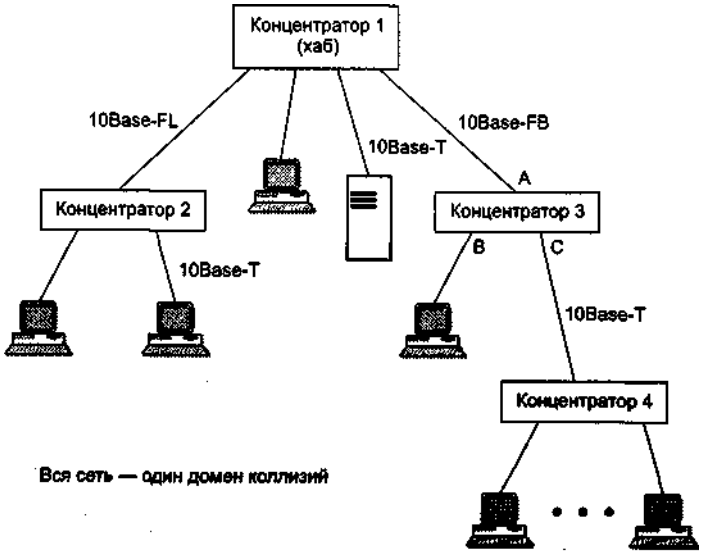


Рис. 3.11. Иерархическое соединение концентраторов Ethernet

Для обеспечения синхронизации станций при реализации процедур доступа CSMA/CD и надежного распознавания станциями коллизий в стандарте определено максимально число концентраторов между любыми двумя станциями сети, а именно 4. Это правило носит название «правила 4-х хабов» и оно заменяет «правило 5-4-3», применяемое к коаксиальным сетям. При создании сети 10Base-T с большим числом станций концентраторы можно соединять друг с другом иерархическим способом, образуя древовидную структуру (рис. 3.11).

ВНИМАНИЕ Петлевидное соединение концентраторов в стандарте 10Base-T запрещено, так как оно приводит к некорректной работе сети. Это требование означает, что в сети 10Base-T не разрешается создавать параллельные каналы связи между критически важными концентраторами для резервирования связей на случай отказа порта, концентратора или кабеля. Резервирование связей возможно только за счет перевода одной из параллельных связей в неактивное (заблокированное) состояние.

Общее количество станций в сети 10Base-T не должно превышать общего предела в 1024, и для данного типа физического уровня это количество действительно можно достичь. Для этого достаточно создать двухуровневую иерархию концентраторов, расположив на нижнем уровне достаточное количество концентраторов с общим количеством портов 1024 (рис. 3.12). Конечные узлы нужно подключить к портам концентраторов нижнего уровня. Правило 4-х хабов при этом выполняется – между любыми конечными узлами будет ровно 3 концентратора.

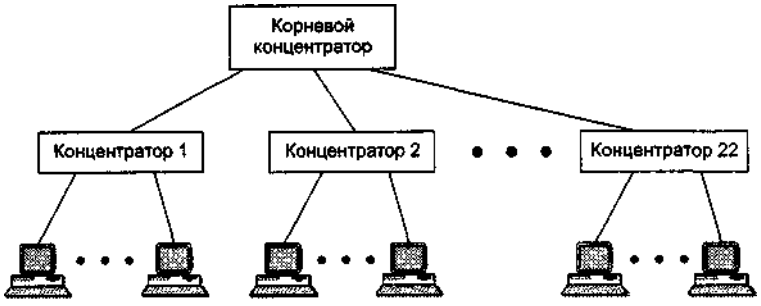


Рис. 3.12. Схема с максимальным количеством станций

Максимальная длина сети в 2500 м здесь понимается как максимальное расстояние между любыми

двумя конечными узлами сети (часто применяется также термин «максимальный диаметр сети»). Очевидно, что если между любыми двумя узлами сети не должно быть больше 4-х повторителей, то максимальный диаметр сети 10Base-T составляет 5x100 = 500 м.

Сети, построенные на основе стандарта 10Base-T, обладают по сравнению с коаксиальными вариантами Ethernet многими преимуществами. Эти преимущества связаны с разделением общего физического кабеля на отдельные кабельные отрезки, подключенные к центральному коммуникационному устройству. И хотя логически эти отрезки по-прежнему образуют общую разделяемую среду, их физическое разделение позволяет контролировать их состояние и отключать в случае обрыва, короткого замыкания или неисправности сетевого адаптера на индивидуальной основе. Это обстоятельство существенно облегчает эксплуатацию больших сетей Ethernet, так как концентратор обычно автоматически выполняет такие функции, уведомляя при этом администратора сети о возникшей проблеме.

В стандарте 10Base-T определена процедура тестирования физической работоспособности двух отрезков витой пары, соединяющих трансивер конечного узла и порт повторителя. Эта процедура называется *тестом связанности* (*tink test*), и она основана на передаче каждые 16 мс специальных импульсов J и K манчестерского кода между передатчиком и приемником каждой витой пары. Если тест не проходит, то порт блокируется и отключает проблемный узел от сети. Так как коды J и K являются запрещенными при передаче кадров, то тестовые последовательности не влияют на работу алгоритма доступа к среде.

Появление между конечными узлами активного устройства, которое может контролировать работу узлов и изолировать от сети некорректно работающие, является главным преимуществом технологии 10Base-T по сравнению со сложными в эксплуатации коаксиальными сетями. Благодаря концентраторам сеть Ethernet приобрела некоторые черты отказоустойчивой системы.

Оптоволоконный Ethernet

В качестве среды передачи данных 10 мегабитный Ethernet использует оптическое волокно. Оптоволоконные стандарты в качестве основного типа кабеля рекомендуют достаточно дешевое многомодовое оптическое волокно, обладающее полосой пропускания 500-800 МГц при длине кабеля 1 км. Допустимо и более дорогое одномодовое оптическое волокно с полосой пропускания в несколько гигагерц, но при этом нужно применять специальный тип трансивера.

Функционально сеть Ethernet на оптическом кабеле состоит из тех же элементов, что и сеть стандарта 10Base-T – сетевых адаптеров, многопортового повторителя и отрезков кабеля, соединяющих адаптер с портом повторителя. Как и в случае витой пары, для соединения адаптера с повторителем используются два оптоволокна – одно соединяет выход Tx адаптера со входом R_x повторителя, а другое – вход R_x адаптера с выходом T_x повторителя.

Стандарт FOIRL (*Fiber Optic Inter-Repeater Link*) представляет собой первый стандарт комитета 802.3 для использования оптоволокна в сетях Ethernet. Он гарантирует длину оптоволоконной связи между повторителями до 1 км при общей длине сети не более 2500 м. Максимальное число повторителей между любыми узлами сети – 4. Максимального диаметра в 2500 м здесь достичь можно, хотя максимальные отрезки кабеля между всеми 4 повторителями, а также между повторителями и конечными узлами недопустимы – иначе получится сеть длиной 5000 м.

Стандарт 10Base-FL представляет собой незначительное улучшение стандарта FOIRL. Увеличена мощность передатчиков, поэтому максимальное расстояние между узлом и концентратором увеличилось до 2000 м. Максимальное число повторителей между узлами осталось равным 4, а максимальная длина сети – 2500 м.

Стандарт 10Base-FB предназначен только для соединения повторителей. Конечные узлы не могут использовать этот стандарт для присоединения к портам концентратора. Между узлами сети можно установить до 5 повторителей 10Base-FB при максимальной длине одного сегмента 2000 м и максимальной длине сети 2740 м.

Повторители, соединенные по стандарту 10Base-FB, при отсутствии кадров для передачи постоянно обмениваются специальными последовательностями сигналов, отличающимися от сигналов кадров данных, для поддержания синхронизации. Поэтому они вносят меньшие задержки при передаче данных из одного сегмента в другой, и это является главной причиной, по которой кбличество повторителей удалось увеличить до 5. В качестве специальных сигналов используются манчестерские коды J и K в

следующей последовательности: J-J-K-K-J-J-... Эта последовательность порождает импульсы частоты 2,5 МГц, которые и поддерживают синхронизацию приемника одного концентратора с передатчиком другого. Поэтому стандарт 10Base-FB имеет также название *синхронный Ethernet*.

Как и в стандарте 10Base-T, оптоволоконные стандарты Ethernet разрешают соединять концентраторы только в древовидные иерархические структуры. Любые петли между портами концентраторов не допускаются.

Домен коллизий

В технологии Ethernet, независимо от применяемого стандарта физического уровня, существует понятие домена коллизий.

Домен коллизий (*collision domain*) – это часть сети Ethernet, все узлы которой распознают коллизию независимо от того, в какой части этой сети коллизия возникла. Сеть Ethernet, построенная на повторителях, всегда образует один домен коллизий. Домен коллизий соответствует одной разделяемой среде. Мосты, коммутаторы и маршрутизаторы делят сеть Ethernet на несколько доменов коллизий.

Приведенная на рис. 3.11 сеть представляет собой один домен коллизий. Если, например, столкновение кадров произошло в концентраторе 4, то в соответствии с логикой работы концентраторов 10Base-T сигнал коллизии распространится по всем портам всех концентраторов.

Если же вместо концентратора 3 поставить в сеть мост, то его порт С, связанный с концентратором 4, воспримет сигнал коллизии, но не передаст его на свои остальные порты, так как это не входит в его обязанности. Мост просто отработает ситуацию коллизии средствами порта С, который подключен к общей среде, где эта коллизия возникла. Если коллизия возникла из-за того, что мост пытался передать через порт С кадр в концентратор 4, то, зафиксировав сигнал коллизии, порт С приостановит передачу кадра и попытается передать его повторно через случайный интервал времени. Если порт С принимал в момент возникновения коллизии кадр, то он просто отбросит полученное начало кадра и будет ожидать, когда узел, передававший кадр через концентратор 4, не сделает повторную попытку передачи. После успешного принятия данного кадра в свой буфер мост передаст его на другой порт в соответствии с таблицей продвижения, например на порт А. Все события, связанные с обработкой коллизий портом С, для остальных сегментов сети, которые подключены к другим портам моста, останутся просто неизвестными.

Узлы, образующие один домен коллизий, работают синхронно, как единая распределенная электронная схема

Общие характеристики стандартов Ethernet 10 Мбит/с

В табл. 3.3 и 3.4 сведены основные ограничения и характеристики стандартов Ethernet.

Таблица 3.3. Общие ограничения для всех стандартов Ethernet

Номинальная пропускная способность	10 Мбит/с
Максимальное число станций в сети	1024
Максимальное расстояние между узлами в сети	2500 м (в 10Base-FB 2750 м)
Максимальное число коаксиальных сегментов в сети	5

Таблица 3.4. Параметры спецификаций физического уровня для стандарта Ethernet

	10Base-5	10Base-2	10Base-T	10Base-F
Кабель	Толстый коаксиальный кабель RG-8 или RG-11	Тонкий коаксиальный кабель RG-58	Неэкраниро-ванная витая пара категорий 3, 4, 5	Многомодовый волоконно-оптический кабель

Максимальная длина сегмента, м	500	185	100	2000
Максимальное расстояние между узлами сети (при использо-вании повторителей), м	2500	925	500	2500 (2740 для 10Base-FB)
Максимальное число станций в сегменте	100	30	1024	1024
Максимальное число повторителей между любыми станциями сети	4	4	4	4 (5 для 10 Base-FB)

3.3.5. Методика расчета конфигурации сети Ethernet

Соблюдение многочисленных ограничений, установленных для различных стандартов физического уровня сетей Ethernet, гарантирует корректную работу сети (естественно, при исправном состоянии всех элементов физического уровня).

Наиболее часто приходится проверять ограничения, связанные с длиной отдельного сегмента кабеля, а также количеством повторителей и общей длиной сети. Правила «5-4-3» для коаксиальных сетей и «4-х хабов» для сетей на основе витой пары и оптоволокна не только дают гарантии работоспособности сети, но и оставляют большой «запас прочности» сети. Например, если посчитать время двойного оборота в сети, состоящей из 4-х повторителей 10Base-5 и 5-ти сегментов максимальный длины 500 м, то окажется, что оно составляет 537 битовых интервала. А так как время передачи кадра минимальной длины, состоящего вместе с преамбулой 72 байт, равно 575 битовым интервалам, то видно, что разработчики стандарта Ethernet оставили 38 битовых интервала в качестве запаса для надежности. Тем не менее комитет 802.3 говорит, что и 4 дополнительных битовых интервала создают достаточный запас надежности.

Комитет IEEE 802.3 приводит исходные данные о задержках, вносимых повторителями и различными средами передачи данных, для тех специалистов, которые хотят самостоятельно рассчитывать максимальное количество повторителей и максимальную общую длину сети, не довольствуясь теми значениями, которые приведены в правилах «5-4-3» и «4-х хабов». Особенно такие расчеты полезны для сетей, состоящих из смешанных кабельных систем, например коаксиала и оптоволокна, на которые правила о количестве повторителей не рассчитаны. При этом максимальная длина каждого отдельного физического сегмента должна строго соответствовать стандарту, то есть 500 м для «толстого» коаксиала, 100 м для витой пары и т. д.

Чтобы сеть Ethernet, состоящая из сегментов различной физической природы, работала корректно, необходимо выполнение четырех основных условий:

- количество станций в сети не более 1024;
- максимальная длина каждого физического сегмента не более величины, определенной в соответствующем стандарте физического уровня;
- время двойного оборота сигнала (Path Delay Value, PDV) между двумя самыми удаленными друг от друга станциями сети не более 575 битовых интервала;
- сокращение межкадрового интервала IPG (Path Variability Value, PVV) при прохождении

последовательности кадров через все повторители должно быть не больше, чем 49 битовых интервала. Так как при отправке кадров конечные узлы обеспечивают начальное межкадровое расстояние в 96 битовых интервала, то после прохождения повторителя оно должно быть не меньше, чем 96 - 49 = 47 битовых интервала.

Соблюдение этих требований обеспечивает корректность работы сети даже в случаях, когда нарушаются простые правила конфигурирования, определяющие максимальное количество повторителей и общую длину сети в 2500 м.

Расчет PDV

Для упрощения расчетов обычно используются справочные данные IEEE, содержащие значения задержек распространения сигналов в повторителях, приемопередатчиках и различных физических средах. В табл. 3.5 приведены данные, необходимые для расчета значения PDV для всех физических стандартов сетей Ethernet. Битовый интервал обозначен как bt.

Таблица 3.5. Данные для расчета значения PDV

Тип сегмента	База левого сегмента, bt	База промежуточного сегмента, bt	База правого сегмента, bt	Задержка среды на 1 м, bt	Максимальная длина сегмента, м
10Base-5	11,8	46,5	169,5	0,0866	500
10Base-2	11,8	46,5	169,5	0,1026	185
10Base-T	15,3	42,0	165,0	0,113	100
10Base-FB	–	24,0	–	0,1	2000
lOBase-FL	12,3	33,5	156,5	0,1	2000
FOIRL	7,8	29,0	152,0	0,1	1000
AUI (> 2 м)	0	0	0	0,1026	2+48

Комитет 802.3 старался максимально упростить выполнение расчетов, поэтому данные, приведенные в таблице, включают сразу несколько этапов прохождения сигнала. Например, задержки, вносимые повторителем, состоят из задержки входного трансивера, задержки блока повторения и задержки выходного трансивера. Тем не менее в таблице все эти задержки представлены одной величиной, названной базой сегмента.

Чтобы не нужно было два раза складывать задержки, вносимые кабелем, в таблице даются удвоенные величины задержек для каждого типа кабеля.

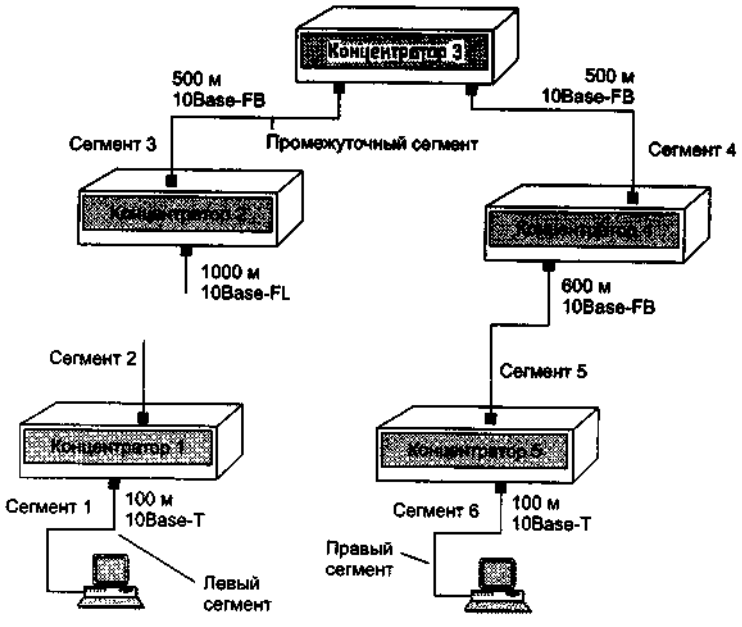


Рис. 3.13. Пример сети Ethernet, состоящей из сегментов различных физических стандартов

В таблице используются также такие понятия, как левый сегмент, правый сегмент и промежуточный сегмент. Поясним эти термины на примере сети, приведенной на рис. 3.13. Левым сегментом называется сегмент, в котором начинается путь сигнала от выхода передатчика (выход Tx на рис. 3.10) конечного узла. На примере это сегмент 1. Затем сигнал проходит через промежуточные сегменты 2-5 и доходит до приемника (вход R_x на рис. 3.10) наиболее удаленного узла наиболее удаленного сегмента 6, который называется правым. Именно здесь в худшем случае происходит столкновение кадров и возникает коллизия, что и подразумевается в таблице.

С каждым сегментом связана постоянная задержка, названная базой, которая зависит только от типа сегмента и от положения сегмента на пути сигнала (левый, промежуточный или правый). База правого сегмента, в котором возникает коллизия, намного превышает базу левого и промежуточных сегментов.

Кроме этого, с каждым сегментом связана задержка распространения сигнала вдоль кабеля сегмента, которая зависит от длины сегмента и вычисляется путем умножения времени распространения сигнала по одному метру кабеля (в битовых интервалах) на длину кабеля в метрах.

Расчет заключается в вычислении задержек, вносимых каждым отрезком кабеля (приведенная в таблице задержка сигнала на 1 м кабеля умножается на длину сегмента), а затем суммировании этих задержек с базами левого, промежуточных и правого сегментов. Общее значение PDV не должно превышать 575.

Так как левый и правый сегменты имеют различные величины базовой задержки, то в случае различных типов сегментов на удаленных краях сети необходимо выполнить расчеты дважды: один раз принять в качестве левого сегмента сегмент одного типа, а во второй – сегмент другого типа. Результатом можно считать максимальное значение PDV. В нашем примере крайние сегменты сети принадлежат к одному типу – стандарту 10Base-T, поэтому двойной расчет не требуется, но если бы они были сегментами разного типа, то в первом случае нужно было бы принять в качестве левого сегмент между станцией и концентратором 1, а во втором считать левым сегмент между станцией и концентратором 5.

Приведенная на рисунке сеть в соответствии с правилом 4-х хабов не является корректной – в сети между узлами сегментов 1 и 6 имеется 5 хабов, хотя не все сегменты являются сегментами 10Base-FB. Кроме того, общая длина сети равна 2800 м, что нарушает правило 2500 м. Рассчитаем значение PDV для нашего примера.

Левый сегмент *1*: 15,3 (база) + 100 × 0,113 = 26,6.

Промежуточный сегмент *2*: 33,5 + 1000 × 0,1 = 133,5

Промежуточный сегмент *3*: 24 + 500 × 0,1 = 74,0.

Промежуточный сегмент *4*: 24 + 500 × 0,1 = 74,0.

Промежуточный сегмент *5*: 24 + 600 × 0,1 = 84,0.

Правый сегмент *6*: 165 + 100 × 0,113 = 176,3.

Сумма всех составляющих дает значение PDV, равное 568,4.

Так как значение PDV меньше максимально допустимой величины 575, то эта сеть проходит по критерию времени двойного оборота сигнала несмотря на то, что ее общая длина составляет больше 2500 м, а количество повторителей – больше 4-х.

Расчет PVV

Чтобы признать конфигурацию сети корректной, нужно рассчитать также уменьшение межкадрового интервала повторителями, то есть величину PVV.

Для расчета PVV также можно воспользоваться значениями максимальных величин уменьшения межкадрового интервала при прохождении повторителей различных физических сред, рекомендованными IEEE и приведенными в табл. 3.6.

Таблица 3.6. Сокращение межкадрового интервала повторителями

Тип сегмента	Передающий сегмент, bt	Промежуточный сегмент, bt

10Base-5 или 10Base-2	16	11
10Base-FB	–	2
10Base-FL	10,5	8
10Base-T	10,5	8

В соответствии с этими данными рассчитаем значение PVV для нашего примера.

Левый сегмент *1* 10Base-T: сокращение в 10,5 bt.

Промежуточный сегмент *2* 10Base-FL: 8.

Промежуточный сегмент *3* 10Base-FB: 2.

Промежуточный сегмент *4* 10Base-FB: 2.

Промежуточный сегмент *5* lOBaserFB: 2.

Сумма этих величин дает значение PVV, равное 24,5, что меньше предельного значения в 49 битовых интервала.

В результате приведенная в примере сеть соответствует стандартам Ethernet по всем параметрам, связанным и с длинами сегментов, и с количеством повторителей.

Выводы

- Ethernet – это самая распространенная на сегодняшний день технология локальных сетей. В широком смысле Ethernet – это целое семейство технологий, включающее различные фирменные и стандартные варианты, из которых наиболее известны фирменный вариант Ethernet DIX, 10-мегабитные варианты стандарта IEEE 802.3, а также новые высокоскоростные технологии Fast Ethernet и Gigabit Ethernet. Почти все виды технологий Ethernet используют один и тот же метод разделения среды передачи данных – метод случайного доступа CSMA/CD, который определяет облик технологии в целом.
- В узком смысле Ethernet – это 10-мегабитная технология, описанная в стандарте IEEE 802.3.
- Важным явлением в сетях Ethernet является коллизия – ситуация, когда две станции одновременно пытаются передать кадр данных по общей среде. Наличие коллизий – это неотъемлемое свойство сетей Ethernet, являющееся следствием принятого случайного метода доступа. Возможность четкого распознавания коллизий обусловлена правильным выбором параметров сети, в частности соблюдением соотношения между минимальной длиной кадра и максимально возможным диаметром сети.
- На характеристики производительности сети большое значение оказывает коэффициент использования сети, который отражает ее загруженность. При значениях этого коэффициента свыше 50 % полезная пропускная способность сети резко падает: из-за роста интенсивности коллизий, а также увеличения времени ожидания доступа к среде.
- Максимально возможная пропускная способность сегмента Ethernet в кадрах в секунду достигается при передаче кадров минимальной длины и составляет 14 880 кадр/с. При этом полезная пропускная способность сети составляет всего 5,48 Мбит/с, что лишь ненамного превышает половину номинальной пропускной способности – 10 Мбит/с.
- Максимально возможная полезная пропускная способность сети Ethernet составляет 9,75 Мбит/с, что соответствует использованию кадров максимальной длины в 1518 байт, которые передаются по сети со скоростью 513 кадр/с.
- При отсутствии коллизий и ожидания доступа *коэффициент использования* сети зависит от размера поля данных кадра и имеет максимальное значение 0,96.
- Технология Ethernet поддерживает 4 разных типа кадров, которые имеют общий формат адресов узлов. Существуют формальные признаки, по которым сетевые адаптеры автоматически распознают тип кадра.
- В зависимости от типа физической среды стандарт IEEE 802.3 определяет различные спецификации: 10Base-5, 10Base-2, 10Base-T, FOIRL, lOBase-FL, lOBase-FB. Для каждой спецификации определяются тип кабеля, максимальные длины непрерывных отрезков кабеля, а также правила использования повторителей для увеличения диаметра сети: правило «5-4-3» для коаксиальных вариантов сетей, и правило «4-х хабов» для витой пары и оптоволокна.

- Для «смешанной» сети, состоящей из физических сегментов различного типа, полезно проводить расчет общей длины сети и допустимого количества повторителей. Комитет IEEE 802.3 приводит исходные данные для таких расчетов, в которых указываются задержки, вносимые повторителями различных спецификаций физической среды, сетевыми адаптерами и сегментами кабеля.

3.4. Технология Token Ring (802.5)

3.4.1. Основные характеристики технологии

Сети Token Ring, так же как и сети Ethernet, характеризует разделяемая среда передачи данных, которая в данном случае состоит из отрезков кабеля, соединяющих все станции сети в кольцо. Кольцо рассматривается как общий разделяемый ресурс, и для доступа к нему требуется не случайный алгоритм, как в сетях Ethernet, а детерминированный, основанный на передаче станциям права на использование кольца в определенном порядке. Это право передается с помощью кадра специального формата, называемого *маркером* или *токеном* (*token*).

Технология Token Ring был разработана компанией IBM в 1984 году, а затем передана в качестве проекта стандарта в комитет IEEE 802, который на ее основе принял в 1985 году стандарт 802.5. Компания IBM использует технологию Token Ring в качестве своей основной сетевой технологии для построения локальных сетей на основе компьютеров различных классов – мэйнфреймов, мини-компьютеров и персональных компьютеров. В настоящее время именно компания IBM является основным законодателем моды технологии Token Ring, производя около 60 % сетевых адаптеров этой технологии.

Сети Token Ring работают с двумя битовыми скоростями – 4 и 16 Мбит/с. Смешение станций, работающих на различных скоростях, в одном кольце не допускается. Сети Token Ring, работающие со скоростью 16 Мбит/с, имеют некоторые усовершенствования в алгоритме доступа по сравнению со стандартом 4 Мбит/с.

Технология Token Ring является более сложной технологией, чем Ethernet. Она обладает свойствами отказоустойчивости. В сети Token Ring определены процедуры контроля работы сети, которые используют обратную связь кольцеобразной структуры – посланный кадр всегда возвращается в станцию-отправитель. В некоторых случаях обнаруженные ошибки в работе сети устраняются автоматически, например может быть восстановлен потерянный маркер. В других случаях ошибки только фиксируются, а их устранение выполняется вручную обслуживающим персоналом.

Для контроля сети одна из станций выполняет роль так называемого *активного монитора*. Активный монитор выбирается во время инициализации кольца как станция с максимальным значением MAC-адреса. Если активный монитор выходит из строя, процедура инициализации кольца повторяется и выбирается новый активный монитор. Чтобы сеть могла обнаружить отказ активного монитора, последний в работоспособном состоянии каждые 3 секунды генерирует специальный кадр своего присутствия. Если этот кадр не появляется в сети более 7 секунд, то остальные станции сети начинают процедуру выборов нового активного монитора.

3.4.2. Маркерный метод доступа к разделяемой среде

В сетях с *маркерным методом доступа* (а к ним, кроме сетей Token Ring, относятся сети FDDI а также сети, близкие к стандарту 802.4, - ArcNet, сети производственного назначения MAP) право на доступ к среде передается циклически от станции к станции по логическому кольцу.

В сети Token Ring кольцо образуется отрезками кабеля, соединяющими соседние станции. Таким образом, каждая станция связана со своей предшествующей и последующей станцией и может непосредственно обмениваться данными только с ними. Для обеспечения доступа станций к физической среде по кольцу циркулирует кадр специального формата и назначения - маркер. В сети Token Ring любая станция всегда непосредственно получает данные только от одной станции - той, которая является предыдущей в кольце. Такая станция называется *ближайшим активным, соседом, расположенным выше по потоку* (данных) - *Nearest Active Upstream Neighbor, NAUN*. Передачу же данных станция всегда осуществляет своему ближайшему соседу вниз по потоку данных.

Получив маркер, станция анализирует его и при отсутствии у нее данных для передачи обеспечивает

его продвижение к следующей станции. Станция, которая имеет данные для передачи, при получении маркера изымает его из кольца, что дает ей право доступа к физической среде и передачи своих данных. Затем эта станция выдает в кольцо кадр данных установленного формата последовательно по битам. Переданные данные проходят по кольцу всегда в одном направлении от одной станции к другой. Кадр снабжен адресом назначения и адресом источника.

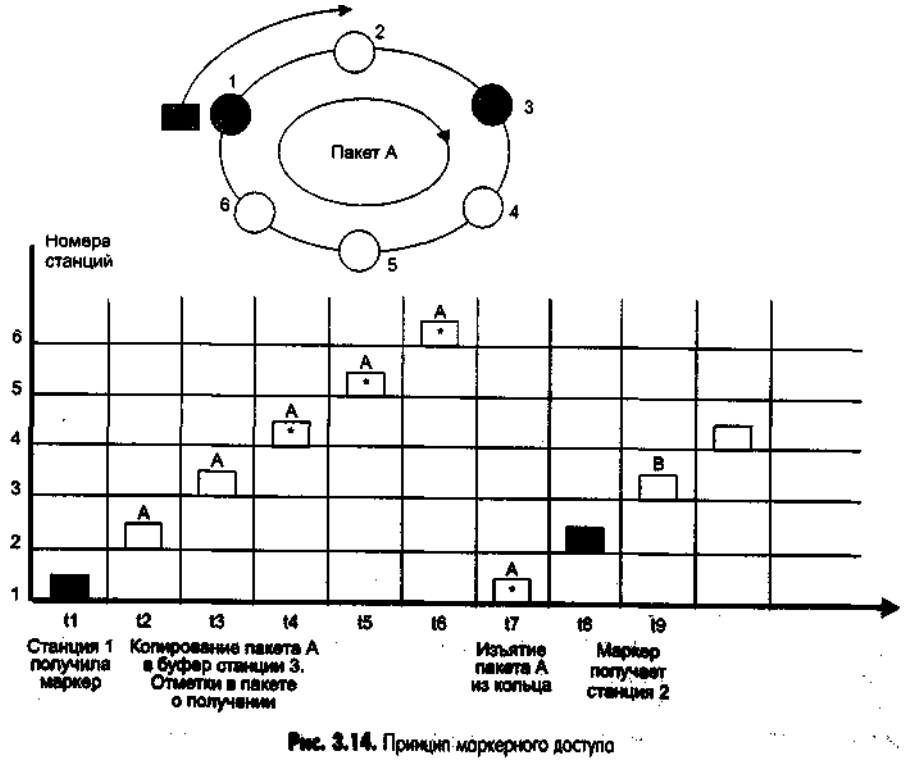


Рис. 3.14. Принцип маркерного доступа

Все станции кольца ретранслируют кадр побитно, как повторители. Если кадр проходит через станцию назначения, то, распознав свой адрес, эта станция копирует кадр в свой внутренний буфер и вставляет в кадр признак подтверждения приема. Станция, выдавшая кадр данных в кольцо, при обратном его получении с подтверждением приема изымает этот кадр из кольца и передает в сеть новый маркер для обеспечения возможности другим станциям сети передавать данные. Такой алгоритм доступа применяется в сетях Token Ring со скоростью работы 4 Мбит/с, описанных в стандарте 802.5.

На рис. 3.14 описанный алгоритм доступа к среде иллюстрируется временной диаграммой. Здесь показана передача пакета А в кольце, состоящем из 6 станций, от станции 1 к станции 3. После прохождения станции назначения 3 в пакете А устанавливаются два признака – признак распознавания адреса и признак копирования пакета в буфер (что на рисунке отмечено звездочкой внутри пакета). После возвращения пакета в станцию 1 отправитель распознает свой пакет по адресу источника и удаляет пакет из кольца. Установленные станцией 3 признаки говорят станции-отправителю о том, что пакет дошел до адресата и был успешно скопирован им в свой буфер.

Время владения разделяемой средой в сети Token Ring ограничивается *временем удержания маркера* (*token holding time*), после истечения которого станция обязана прекратить передачу собственных данных (текущий кадр разрешается завершить) и передать маркер далее по кольцу. Станция может успеть передать за время удержания маркера один или несколько кадров в зависимости от размера кадров и величины времени удержания маркера. Обычно время удержания маркера по умолчанию равно 10 мс, а максимальный размер кадра в стандарте 802.5 не определен. Для сетей 4 Мбит/с он обычно равен 4 Кбайт, а для сетей 16 Мбит/с – 16 Кбайт. Это связано с тем, что за время удержания маркера станция должна успеть передать хотя бы один кадр. При скорости 4 Мбит/с за время 10 мс можно передать 5000 байт, а при скорости 16 Мбит/с – соответственно 20 000 байт. Максимальные размеры кадра выбраны с некоторым запасом.

В сетях Token Ring 16 Мбит/с используется также несколько другой алгоритм доступа к кольцу, называемый алгоритмом *раннего освобождения маркера* (*Early Token Release*). В соответствии с ним станция передает маркер доступа следующей станции сразу же после окончания передачи последнего бита кадра, не дожидаясь возвращения по кольцу этого кадра с битом подтверждения приема. В этом

случае пропускная способность кольца используется более эффективно, так как по кольцу одновременно продвигаются кадры нескольких станций. Тем не менее свои кадры в каждый момент времени может генерировать только одна станция – та, которая в данный момент владеет маркером доступа. Остальные станции в это время только повторяют чужие кадры, так что принцип разделения кольца во времени сохраняется, ускоряется только процедура передачи владения кольцом.

Для различных видов сообщений, передаваемым кадрам, могут назначаться различные *приоритеты*: от 0 (низший) до 7 (высший). Решение о приоритете конкретного кадра принимает передающая станция (протокол Token Ring получает этот параметр через межуровневые интерфейсы от протоколов верхнего уровня, например прикладного). Маркер также, всегда имеет некоторый уровень текущего приоритета. Станция имеет право захватить переданный ей маркер только в том случае, если приоритет кадра, который она хочет передать, выше (или равен) приоритета маркера. В противном случае станция обязана передать маркер следующей по кольцу станции.

За наличие в сети маркера, причем единственной его копии, отвечает активный монитор. Если активный монитор не получает маркер в течение длительного времени (например, 2,6 с), то он порождает новый маркер.

3.4.3. Форматы кадров Token Ring

В Token Ring существуют три различных формата кадров:

- маркер;
- кадр данных;
- прерывающая последовательность.

Маркер

Кадр маркера состоит из трех полей, каждое длиной в один байт.

- *Начальный ограничитель* (*Start Delimiter, SD*) появляется в начале маркера, а также в начале любого кадра, проходящего по сети. Поле представляет собой следующую уникальную последовательность символов манчестерского кода: JKOJKOOO. Поэтому начальный ограничитель нельзя спутать ни с какой битовой последовательностью внутри кадра.
- *Управление доступом* (*Access Control*) состоит из четырех подполей: PPP, T, M и RRR, где PPP – биты приоритета, T – бит маркера, M – бит монитора, RRR – резервные биты приоритета. Бит T, установленный в 1, указывает на то, что данный кадр является маркером доступа. Бит монитора устанавливается в 1 активным монитором и в 0 любой другой станцией, передающей маркер или кадр. Если активный монитор видит маркер или кадр, содержащий бит монитора со значением 1, то активный монитор знает, что этот кадр или маркер уже однажды обошел кольцо и не был обработан станциями. Если это кадр, то он удаляется из кольца. Если это маркер, то активный монитор передает его дальше по кольцу. Использование полей приоритетов будет рассмотрено ниже.
- *Конечный ограничитель* (*End Delimeter, ED*) – последнее поле маркера. Так же как и поле начального ограничителя, это поле содержит уникальную последовательность манчестерских кодов JK1JK1, а также два однобитовых признака: I и E. Признак I (Intermediate) показывает, является ли кадр последним в серии кадров (I=0) или промежуточным (I=1). Признак E (Error) – это признак ошибки. Он устанавливается в 0 станцией-отправителем, и любая станция кольца, через которую проходит кадр, должна установить этот признак в 1, если она обнаружит ошибку по контрольной сумме или другую некорректность кадра.

Кадр данных и прерывающая последовательность

Кадр данных включает те же три поля, что и маркер, и имеет кроме них еще несколько дополнительных полей: Таким образом, кадр данных состоит из следующих полей:

- начальный ограничитель (Start Delimiter, SD);
- управление кадром (Frame Control, FC);
- адрес назначения (Destination Address, DA);

- адрес источника (Source Address, SA);
- данные (INFO);
- контрольная сумма (Frame Check Sequence, FCS);
- конечный ограничитель (End Delimeter, ED);
- статус кадра (Frame Status, FS).

Кадр данных может переносить либо служебные данные для управления кольцом (данные MAC-уровня), либо пользовательские данные (LLC-уровня). Стандарт Token Ring определяет 6 типов управляющих кадров MAC-уровня. Поле FC определяет тип кадра (MAC или LLC), и если он определен как MAC, то поле также указывает, какой из шести типов кадров представлен данным кадром.

Назначение этих шести типов кадров описано ниже.

- Чтобы удостовериться, что ее адрес уникальный, станция, когда впервые присоединяется к кольцу, посылает кадр *Тест дублирования адреса* (*Duplicate Address Test, DAT*).
- Чтобы сообщить другим станциям, что он работоспособен, активный монитор периодически посылает в кольцо кадр *Существует активный монитор* (*Active Monitor Present, AMP*).
- Кадр *Существует резервный монитор* (*Standby Monitor Present, SMP*) отправляется любой станцией, не являющейся активным монитором.
- Резервный монитор отправляет кадр *Маркер заявки* (*Claim Token, CT*), когда подозревает, что активный монитор отказал, затем резервные мониторы договариваются между собой, какой из них станет новым активным монитором.
- Станция отправляет кадр *Сигнал* (*Beacon, BCN*) в случае возникновения серьезных сетевых проблем, таких как обрыв кабеля, обнаружение станции, передающей кадры без ожидания маркера, выход станции из строя. Определяя, какая станция отправляет кадр сигнала, диагностирующая программа (ее существование и функции не определяются стандартами Token Ring) может локализовать проблему. Каждая станция периодически передает кадры BCN до тех пор, пока не примет кадр BCN от своего предыдущего (NAUN) соседа. В результате в кольце только одна станция продолжает передавать кадры BCN – та, у которой имеются проблемы с предыдущим соседом. В сети Token Ring каждая станция знает MAC-адрес своего предыдущего соседа, поэтому Beacon-процедура приводит к выявлению адреса некорректно работающей станции.
- Кадр *Очистка* (*Purge, PRG*) используется новым активным монитором для того, чтобы перевести все станции в исходное состояние и очистить кольцо от всех ранее посланных кадров.

В стандарте 802.5 используются адреса той же структуры, что и в стандарте 802.3. Адреса назначения и источника могут иметь длину либо 2, либо 6 байт. Первый бит адреса назначения определяет групповой или индивидуальный адрес как для 2-байтовых, так и для 6-байтовых адресов. Второй бит в 6-байтовых адресах говорит о том, назначен адрес локально или глобально. Адрес, состоящий из всех единиц, является широковещательным.

Адрес источника имеет тот же размер и формат, что и адрес назначения. Однако признак группового адреса используется в нем особым способом. Так как адрес источника не может быть групповым, то наличие единицы в этом разряде говорит о том, что в кадре имеется специальное *поле маршрутной информации* (*Routing Information Field, RIF*). Эта информация требуется при работе мостов, связывающих несколько колец Token Ring, в режиме маршрутизации от источника.

Поле данных INFO кадра может содержать данные одного из описанных управляющих кадров уровня MAC или пользовательские данные, упакованные в кадр уровня LLC. Это поле, как уже отмечалось, не имеет определенной стандартом максимальной длины, хотя существуют практические ограничения на его размер, основанные на временных соотношениях между временем удержания маркера и временем передачи кадра.

Поле статуса FS имеет длину 1 байт и содержит 4 резервных бита и 2 подполя: бит распознавания адреса A и бит копирования кадра C. Так как это поле не сопровождается вычисляемой суммой CRC, то используемые биты для надежности дублируются: поле статуса FS имеет вид ACxxACxx. Если бит распознавания адреса не установлен во время получения кадра, это означает, что станция назначения больше не присутствует в сети (возможно, вследствие неполадок, а возможно, станция находится в другом кольце, связанном с данным с помощью моста). Если оба бита опознавания адреса и копирования кадра установлены и бит обнаружения ошибки также установлен, то исходная станция

знает, что ошибка случилась после того, как этот кадр был корректно получен.

Прерывающая последовательность состоит из двух байтов, содержащих начальный и конечный ограничители. Прерывающая последовательность может появиться в любом месте потока битов и сигнализирует о том, что текущая передача кадра или маркера отменяется.

Приоритетный доступ к кольцу

Каждый кадр данных или маркер имеет приоритет, устанавливаемый битами приоритета (значение от 0 до 7, причем 7 – наивысший приоритет). Станция может воспользоваться маркером, если только у нее есть кадры для передачи с приоритетом равным или большим, чем приоритет маркера. Сетевой адаптер станции с кадрами, у которых приоритет ниже, чем приоритет маркера, не может захватить маркер, но может поместить наибольший приоритет своих ожидающих передачи кадров в резервные биты маркера, но только в том случае, если записанный в резервных битах приоритет ниже его собственного. В результате в резервных битах приоритета устанавливается наивысший приоритет станции, которая пытается получить доступ к кольцу, но не может этого сделать из-за высокого приоритета маркера.

Станция, сумевшая захватить маркер, передает свои кадры с приоритетом маркера, а затем передает маркер следующему соседу. При этом она переписывает значение резервного приоритета в поле приоритета маркера, а резервный приоритет обнуляется. Поэтому при следующем проходе маркера по кольцу его захватит станция, имеющая наивысший приоритет.

При инициализации кольца основной и резервный приоритет маркера устанавливаются в 0.

Хотя механизм приоритетов в технологии Token Ring имеется, но он начинает работать только в том случае, когда приложение или прикладной протокол решают его использовать. Иначе все станции будут иметь равные права доступа к кольцу, что в основном происходит на практике, так как большая часть приложений этим механизмом не пользуется. Это связано с тем, что приоритеты кадров поддерживаются не во всех технологиях, например в сетях Ethernet они отсутствуют, поэтому приложение будет вести себя по-разному, в зависимости от технологии нижнего уровня, что нежелательно. В современных сетях приоритетность обработки кадров обычно обеспечивается коммутаторами или маршрутизаторами, которые поддерживают их независимо от используемых протоколов канального уровня.

3.4.4. Физический уровень технологии Token Ring

Стандарт Token Ring фирмы IBM изначально предусматривал построение связей в сети с помощью концентраторов, называемых MAU (Multistation Access Unit) или MSAU (Multi-Station Access Unit), то есть устройствами многостанционного доступа (рис. 3.15). Сеть Token Ring может включать до 260 узлов.

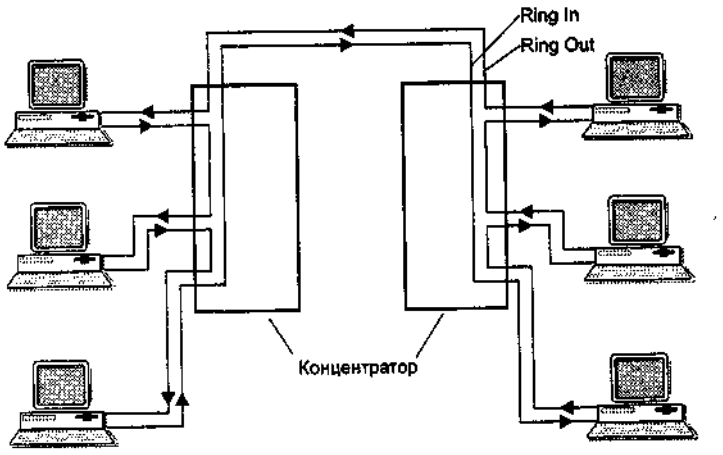


Рис. 3.15. Физическая конфигурация сети Token Ring

Концентратор Token Ring может быть активным или пассивным. Пассивный концентратор просто

соединяет порты внутренними связями так, чтобы станции, подключаемые к этим портам, образовали кольцо. Ни усиление сигналов, ни их ресинхронизацию пассивный MSAU не выполняет. Такое устройство можно считать простым кроссовым блоком за одним исключением – MSAU обеспечивает обход какого-либо порта, когда присоединенный к этому порту компьютер выключают. Такая функция необходима для обеспечения связности кольца вне зависимости от состояния подключенных компьютеров. Обычно обход порта выполняется за счет релейных схем, которые питаются постоянным током от сетевого адаптера, а при выключении сетевого адаптера нормально замкнутые контакты реле соединяют вход порта с его выходом.

Активный концентратор выполняет функции регенерации сигналов и поэтому иногда называется повторителем, как в стандарте Ethernet.

Возникает вопрос – если концентратор является пассивным устройством, то каким образом обеспечивается качественная передача сигналов на большие расстояния, которые возникают при включении в сеть нескольких сот компьютеров? Ответ состоит в том, что роль усилителя сигналов в этом случае берет на себя каждый сетевой адаптер, а роль ресинхронизирующего блока выполняет сетевой адаптер активного монитора кольца. Каждый сетевой адаптер Token Ring имеет блок повторения, который умеет регенерировать и ресинхронизировать сигналы, однако последнюю функцию выполняет в кольце только блок повторения активного монитора.

Блок ресинхронизации состоит из 30-битного буфера, который принимает манчестерские сигналы с несколько искаженными за время оборота по кольцу интервалами следования. При максимальном количестве станций в кольце (260) вариация задержки циркуляции бита по кольцу может достигать 3-битовых интервалов. Активный монитор «вставляет» свой буфер в кольцо и синхронизирует битовые сигналы, выдавая их на выход с требуемой частотой.

В общем случае сеть Token Ring имеет комбинированную звездно-кольцевую конфигурацию. Конечные узлы подключаются к MSAU по топологии звезды, а сами MSAU объединяются через специальные порты Ring In (RI) и Ring Out (RO) для образования магистрального физического кольца.

Все станции в кольце должны работать на одной скорости – либо 4 Мбит/с, либо 16 Мбит/с. Кабели, соединяющие станцию с концентратором, называются ответвительными (lobe cable), а кабели, соединяющие концентраторы, – магистральными (trunk cable).

Технология Token Ring позволяет использовать для соединения конечных станций и концентраторов различные типы кабеля: STP Type 1, UTP Type 3, UTP Type 6, а также волоконно-оптический кабель.

При использовании экранированной витой пары STP Type 1 из номенклатуры кабельной системы IBM в кольцо допускается объединять до 260 станций при длине ответвительных кабелей до 100 метров, а при использовании неэкранированной витой пары максимальное количество станций сокращается до 72 при длине ответвительных кабелей до 45 метров.

Расстояние между пассивными MSAU может достигать 100 м при использовании кабеля STP Type 1 и 45 м при использовании кабеля UTP Type 3. Между активными MSAU максимальное расстояние увеличивается соответственно до 730 м или 365 м в зависимости от типа кабеля.

Максимальная длина кольца Token Ring составляет 4000 м. Ограничения на максимальную длину кольца и количество станций в кольце в технологии Token Ring не являются такими жесткими, как в технологии Ethernet. Здесь эти ограничения во многом связаны со временем оборота маркера по кольцу (но не только – есть и другие соображения, диктующие выбор ограничений). Так, если кольцо состоит из 260 станций, то при времени удержания маркера в 10 мс маркер вернется в активный монитор в худшем случае через 2,6 с, а это время как раз составляет тайм-аут контроля оборота маркера. В принципе, все значения тайм-аутов в сетевых адаптерах узлов сети Token Ring можно настраивать, поэтому можно построить сеть Token Ring с большим количеством станций и с большей длиной кольца.

Существует большое количество аппаратуры для сетей Token Ring, которая улучшает некоторые стандартные характеристики этих сетей: максимальную длину сети, расстояние между концентраторами, надежность (путем использования двойных колец).

Недавно компания IBM предложила новый вариант технологии Token Ring, названный High-Speed Token Ring, HSTR. Эта технология поддерживает битовые скорости в 100 и 155 Мбит/с, сохраняя основные особенности технологии Token Ring 16 Мбит/с.

Выводы

- Технология Token Ring развивается в основном компанией IBM и имеет также статус

стандарта IEEE 802.5, который отражает наиболее важные усовершенствования, вносимые в технологию IBM.

- В сетях Token Ring используется маркерный метод доступа, который гарантирует каждой станции получение доступа к разделяемому кольцу в течение времени оборота маркера. Из-за этого свойства этот метод иногда называют детерминированным.
- Метод доступа основан на приоритетах: от 0 (низший) до 7 (высший). Станция сама определяет приоритет текущего кадра и может захватить кольцо только в том случае, когда в кольце нет более приоритетных кадров.
- Сети Token Ring работают на двух скоростях: 4 и 16 Мбит/с и могут использовать в качестве физической среды экранированную витую пару, неэкранированную витую пару, а также волоконно-оптический кабель. Максимальное количество станций в кольце – 260, а максимальная длина кольца – 4 км.
- Технология Token Ring обладает элементами отказоустойчивости. За счет обратной связи кольца одна из станций – активный монитор – непрерывно контролирует наличие маркера, а также время оборота маркера и кадров данных. При некорректной работе кольца запускается процедура его повторной инициализации, а если она не помогает, то для локализации неисправного участка кабеля или неисправной станции используется процедура beaconing.
- Максимальный размер поля данных кадра Token Ring зависит от скорости работы кольца. Для скорости 4 Мбит/с он равен около 5000 байт, а при скорости 16 Мбит/с – около 16 Кбайт. Минимальный размер поля данных кадра не определен, то есть может быть равен 0.
- В сети Token Ring станции в кольцо объединяют с помощью концентраторов, называемых MSAU. Пассивный концентратор MSAU выполняет роль кроссовой панели, которая соединяет выход предыдущей станции в кольце со входом последующей. Максимальное расстояние от станции до MSAU – 100 м для STP и 45 м для UTP.
- Активный монитор выполняет в кольце также роль повторителя – он ресинхронизирует сигналы, проходящие по кольцу.
- Кольцо может быть построено на основе активного концентратора MSAU, который в этом случае называют повторителем.
- Сеть Token Ring может строиться на основе нескольких колец, разделенных мостами, маршрутизирующими кадры по принципу «от источника», для чего в кадр Token Ring добавляется специальное поле с маршрутом прохождения колец.

3.5. Технология FDDI

Технология FDDI (*Fiber Distributed Data Interface*) – оптоволоконный интерфейс распределенных данных – это первая технология локальных сетей, в которой средой передачи данных является волоконно-оптический кабель. Работы по созданию технологий и устройств для использования волоконно-оптических каналов в локальных сетях начались в 80-е годы, вскоре после начала промышленной эксплуатации подобных каналов в территориальных сетях. Проблемная группа X3T9.5 института ANSI разработала в период с 1986 по 1988 гг. начальные версии стандарта FDDI, который обеспечивает передачу кадров со скоростью 100 Мбит/с по двойному волоконно-оптическому кольцу длиной до 100 км.

3.5.1. Основные характеристики технологии

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

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

Сеть FDDI строится на основе двух оптоволоконных колец, которые образуют основной и резервный пути передачи данных между узлами сети. Наличие двух колец – это основной способ повышения отказоустойчивости в сети FDDI, и узлы, которые хотят воспользоваться этим повышенным потенциалом надежности, должны быть подключены к обоим кольцам.

В нормальном режиме работы сети данные проходят через все узлы и все участки кабеля только первичного (Primary) кольца, этот режим назван режимом *Thru* – «сквозным» или «транзитным». Вторичное кольцо (Secondary) в этом режиме не используется.

В случае какого-либо вида отказа, когда часть первичного кольца не может передавать данные (например, обрыв кабеля или отказ узла), первичное кольцо объединяется со вторичным (рис. 3.16), вновь образуя единое кольцо. Этот режим работы сети называется Wrap, то есть «свертывание» или «сворачивание» колец. Операция свертывания производится средствами концентраторов и/или сетевых адаптеров FDDI. Для упрощения этой процедуры данные по первичному кольцу всегда передаются в одном направлении (на диаграммах это направление изображается против часовой стрелки), а по вторичному – в обратном (изображается по часовой стрелке). Поэтому при образовании общего кольца из двух колец передатчики станций по-прежнему остаются подключенными к приемникам соседних станций, что позволяет правильно передавать и принимать информацию соседними станциями.

В стандартах FDDI много внимания отводится различным процедурам, которые позволяют определить наличие отказа в сети, а затем произвести необходимую реконфигурацию. Сеть FDDI может полностью восстанавливать свою работоспособность в случае единичных отказов ее элементов. При множественных отказах сеть распадается на несколько не связанных сетей. Технология FDDI дополняет механизмы обнаружения отказов технологии Token Ring механизмами реконфигурации пути передачи данных в сети, основанными на наличии резервных связей, обеспечиваемых вторым кольцом.

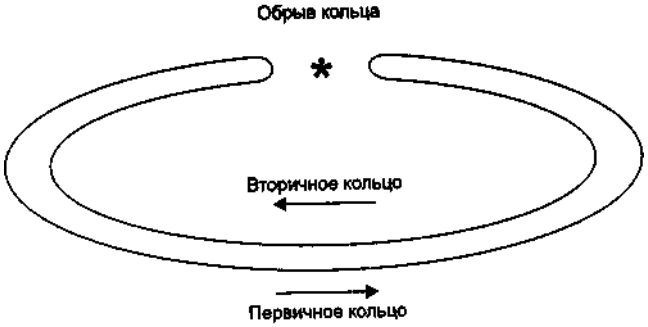


Рис. 3.16. Реконфигурация колец FDDI при отказе

Кольца в сетях FDDI рассматриваются как общая разделяемая среда передачи данных, поэтому для нее определен специальный метод доступа. Этот метод очень близок к методу доступа сетей Token Ring и также называется методом маркерного (или токенного) кольца – token ring.

Отличия метода доступа заключаются в том, что время удержания маркера в сети FDDI не является постоянной величиной, как в сети Token Ring. Это время зависит от загрузки кольца – при небольшой загрузке оно увеличивается, а при больших перегрузках может уменьшаться до нуля. Эти изменения в методе доступа касаются только асинхронного трафика, который не критичен к небольшим задержкам передачи кадров. Для синхронного трафика время удержания маркера по-прежнему остается фиксированной величиной. Механизм приоритетов кадров, аналогичный принятому в технологии Token Ring, в технологии FDDI отсутствует. Разработчики технологии решили, что деление трафика на 8 уровней приоритетов избыточно и достаточно разделить трафик на два класса – асинхронный и синхронный, последний из которых обслуживается всегда, даже при перегрузках кольца.

В остальном пересылка кадров между станциями кольца на уровне MAC полностью соответствует технологии Token Ring. Станции FDDI применяют алгоритм раннего освобождения маркера, как и сети Token Ring со скоростью 16 Мбит/с.

Адреса уровня MAC имеют стандартный для технологий IEEE 802 формат. Формат кадра FDDI близок к формату кадра Token Ring, основные отличия заключаются в отсутствии полей приоритетов. Признаки распознавания адреса, копирования кадра и ошибки позволяют сохранить имеющиеся в сетях Token Ring процедуры обработки кадров станцией-отправителем, промежуточными станциями и станцией-получателем.

На рис. 3.17 приведено соответствие структуры протоколов технологии FDDI семиуровневой модели

OSI. FDDI определяет протокол физического уровня и протокол подуровня доступа к среде (MAC) канального уровня. Как и во многих других технологиях локальных сетей, в технологии FDDI используется протокол подуровня управления каналом данных LLC, определенный в стандарте IEEE 802.2. Таким образом, несмотря на то что технология FDDI была разработана и стандартизована институтом ANSI, а не комитетом IEEE, она полностью вписывается в структуру стандартов 802.

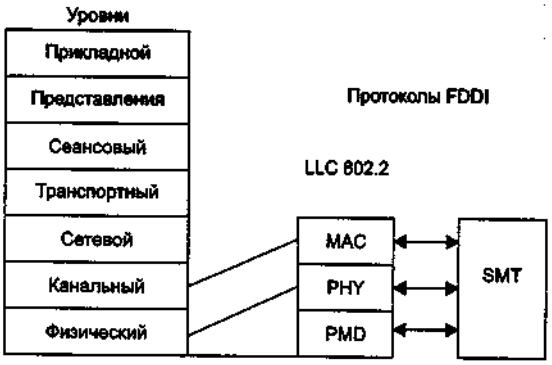


Рис. 3.17. Структура протоколов технологии FDDI

Отличительной особенностью технологии FDDI является уровень управления станцией – *Station Management* (*SMT*). Именно уровень SMT выполняет все функции по управлению и мониторингу всех остальных уровней стека протоколов FDDI. В управлении кольцом принимает участие каждый узел сети FDDI. Поэтому все узлы обмениваются специальными кадрами SMT для управления сетью.

Отказоустойчивость сетей FDDI обеспечивается протоколами и других уровней: с помощью физического уровня устраняются отказы сети по физическим причинам, например из-за обрыва кабеля, а с помощью уровня MAC – логические отказы сети, например потеря нужного внутреннего пути передачи маркера и кадров данных между портами концентратора.

3.5.2. Особенности метода доступа FDDI

Для передачи синхронных кадров станция всегда имеет право захватить маркер при его поступлении. При этом время удержания маркера имеет заранее заданную фиксированную величину.

Если же станции кольца FDDI нужно передать асинхронный кадр (тип кадра определяется протоколами верхних уровней), то для выяснения возможности захвата маркера при его очередном появлении станция должна измерить интервал времени, который прошел с момента предыдущего прихода маркера. Этот интервал называется *временем оборота маркера* (*Token Rotation Time, TRT*). Интервал TRT сравнивается с другой величиной – *максимально допустимым временем оборота маркера по кольцу T_Opr*. Если в технологии Token Ring максимально допустимое время оборота маркера является фиксированной величиной (2,6 с из расчета 260 станций в кольце), то в технологии FDDI станции договариваются о величине T_Opr во время инициализации кольца. Каждая станция может предложить свое значение T_Opr, в результате для кольца устанавливается минимальное из предложенных станциями времен. Это позволяет учитывать потребности приложений, работающих на станциях. Обычно синхронным приложениям (приложениям реального времени) нужно чаще передавать данные в сеть небольшими порциями, а асинхронным приложениям лучше получать доступ к сети реже, но большими порциями. Предпочтение отдается станциям, передающим синхронный трафик.

Таким образом, при очередном поступлении маркера для передачи асинхронного кадра сравнивается фактическое время оборота маркера TRT с максимально возможным T_Opr. Если кольцо не перегружено, то маркер приходит раньше, чем истекает интервал T_Opr, то есть TRT < T_Opr. В этом случае станции разрешается захватить маркер и передать свой кадр (или кадры) в кольцо. Время удержания маркера THT равно разности T_Opr - TRT, и в течение этого времени станция передает в кольцо столько асинхронных кадров, сколько успеет.

Если же кольцо перегружено и маркер опоздал, то интервал TRT будет больше T_Opr. В этом случае станция не имеет права захватить маркер для асинхронного кадра. Если все станции в сети хотят передавать только асинхронные кадры, а маркер сделал оборот по кольцу слишком медленно, то все станции пропускают маркер в режиме повторения, маркер быстро делает очередной оборот и на

следующем цикле работы станции уже имеют право захватить маркер и передать свои кадры.

Метод доступа FDDI для асинхронного трафика является адаптивным и хорошо регулирует временные перегрузки сети.

3.5.3. Отказоустойчивость технологии FDDI

Для обеспечения отказоустойчивости в стандарте FDDI предусмотрено создание двух оптоволоконных колец – первичного и вторичного. В стандарте FDDI допускаются два вида подсоединения станций к сети. Одновременное подключение к первичному и вторичному кольцам называется двойным подключением – Dual Attachment, DA. Подключение только к первичному кольцу называется одиночным подключением – Single Attachment, SA.

В стандарте FDDI предусмотрено наличие в сети конечных узлов – станций (Station), а также концентраторов (Concentrator). Для станций и концентраторов допустим любой вид подключения к сети – как одиночный, так и двойной. Соответственно такие устройства имеют соответствующие названия: SAS (Single Attachment Station), DAS (Dual Attachment Station), SAC (Single Attachment Concentrator) и DAC (Dual Attachment Concentrator).

Обычно концентраторы имеют двойное подключение, а станции – одинарное, как это показано на рис. 3.18, хотя это и не обязательно. Чтобы устройства легче было правильно присоединять к сети, их разъемы маркируются. Разъемы типа A и B должны быть у устройств с двойным подключением, разъем M (Master) имеется у концентратора для одиночного подключения станции, у которой ответный разъем должен иметь тип S (Slave).

В случае однократного обрыва кабеля между устройствами с двойным подключением сеть FDDI сможет продолжить нормальную работу за счет автоматической реконфигурации внутренних путей передачи кадров между портами концентратора (рис. 3.19). Двукратный обрыв кабеля приведет к образованию двух изолированных сетей FDDI. При обрыве кабеля, идущего к станции с одиночным подключением, она становится отрезанной от сети, а кольцо продолжает работать за счет реконфигурации внутреннего пути в концентраторе – порт M, к которому была подключена данная станция, будет исключен из общего пути.

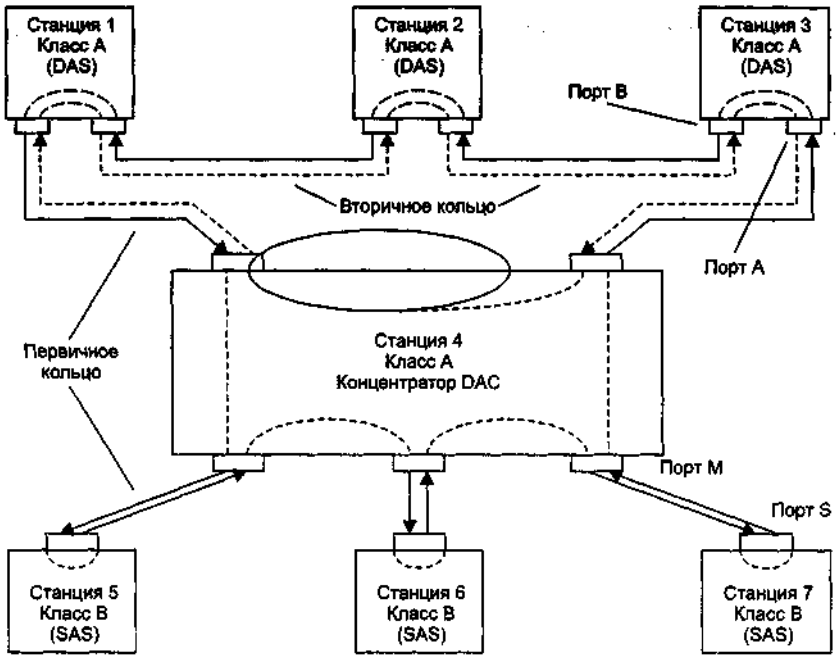


Рис. 3.18. Подключение узлов к кольцам FDDI

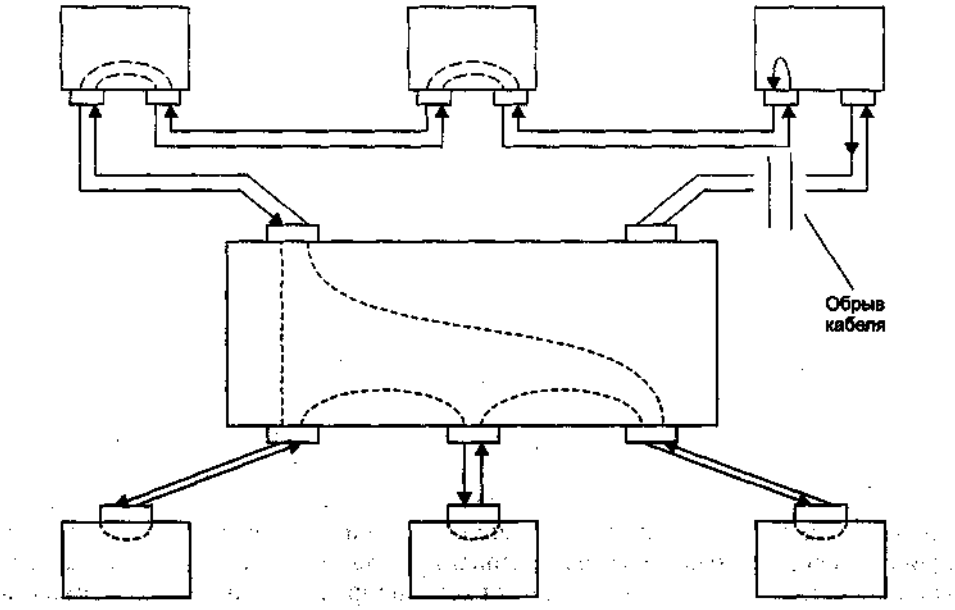


Рис. 3.19. Реконфигурация сети FDDI при обрыве провода

Для сохранения работоспособности при отключении питания в станциях с двойным подключением, то есть станциях DAS, последние должны быть оснащены оптическими обходными переключателями (Optical Bypass Switch), которые создают обходной путь для световых потоков при исчезновении питания, которое они получают от станции.

И наконец, станции DAS или концентраторы DAC можно подключать к двум портам M одного или двух концентраторов, создавая древовидную структуру с основными и резервными связями. По умолчанию порт B поддерживает основную связь, а порт A – резервную. Такая конфигурация называется подключением Dual Homing.

Отказоустойчивость поддерживается за счет постоянного слежения уровня SMT концентраторов и станций за временными интервалами циркуляции маркера и кадров, а также за наличием физического соединения между соседними портами в сети. В сети FDDI нет выделенного активного монитора – все станции и концентраторы равноправны, и при обнаружении отклонений от нормы они начинают процесс повторной инициализации сети, а затем и ее реконфигурации.

Реконфигурация внутренних путей в концентраторах и сетевых адаптерах выполняется специальными оптическими переключателями, которые перенаправляют световой луч и имеют достаточно сложную конструкцию.

3.5.4 Физический уровень технологии FDDI

В технологии FDDI для передачи световых сигналов по оптическим волокнам реализовано логическое кодирование 4B/5B в сочетании с физическим кодированием NRZI. Эта схема приводит к передаче по линии связи сигналов с тактовой частотой 125 МГц.

Так как из 32 комбинаций 5- битных символов для кодирования исходных 4- битных символов нужно только 16 комбинаций, то из оставшихся 16 выбрано несколько кодов, которые используются как служебные. К наиболее важным служебным символам относиться символ Idle- простой, который постоянно передается между портами в течении пауз между передачей кадров данных. За счет этого станции и концентраторы сети FDDI имеют постоянную информацию о состоянии физических соединений своих портов. В случае отсутствия потока символов Idle фиксируется отказ физической связи и производиться реконфигурация внутреннего пути концентратора или станции, если это возможно.

При первоначальном соединении кабелем двух узлов их порты сначала выполняют процедуру установления физического соединения. В этой процедуре используются последовательности служебных символов кода 4B/5B, с помощью которых создается некоторый язык команд физического уровня. Эти команды позволяют портам выяснить друг у друга типы портов (A, B, M илиS) и решить, корректно ли данное соединение (например, соединение S-S является некорректным и т. п.). Если соединение

корректно, то далее выполняется тест качества канала при передаче символов кодов 4B/5B, а затем проверяется работоспособность уровня MAC соединенных устройств путем передачи нескольких кадров MAC. Если все тесты прошли успешно, то физическое соединение считается установленным. Работу по установлению физического соединения контролирует протокол управления станцией SMT.

Физический уровень разделен на два подуровня: независимый от среды подуровень PHY (Physical) и зависящий от среды подуровень PMD (Physical Media Dependent) (см. рис. 3.17).

Технология FDDI в настоящее время поддерживает два подуровня PMD: для волоконно-оптического кабеля и для неэкранированной витой пары категории 5. Последний стандарт появился позже оптического и носит название TP-PMD.

Оптоволоконный подуровень PMD обеспечивает необходимые средства для передачи данных от одной станции к другой по оптическому волокну. Его спецификация определяет:

- использование в качестве основной физической среды многомодового волоконно-оптического кабеля 62,5/125 мкм;
- требования к мощности оптических сигналов и максимальному затуханию между узлами сети. Для стандартного многомодового кабеля эти требования приводят к предельному расстоянию между узлами в 2 км, а для одномодового кабеля расстояние увеличивается до 10-40 км в зависимости от качества кабеля;
- требования к оптическим обходным переключателям (optical bypass switches) и оптическим приемопередатчикам;
- параметры оптических разъемов MIC (Media Interface Connector), их маркировку;
- использование для передачи света с длиной волны в 1300 нм;
- представление сигналов в оптических волокнах в соответствии с методом NRZI.

Подуровень TP-PMD определяет возможность передачи данных между станциями по витой паре в соответствии с методом физического кодирования MLT-3, использующего два уровня потенциала: +V и -V для представления данных в кабеле. Для получения равномерного по мощности спектра сигнала данные перед физическим кодированием проходят через скрэмблер. Максимальное расстояние между узлами в соответствии со стандартом TP-PMD равно 100 м.

Максимальная общая длина кольца FDDI составляет 100 километров, максимальное число станций с двойным подключением в кольце – 500.

3.5.5. Сравнение FDDI с технологиями Ethernet и Token Ring

В табл. 3.7 представлены результаты сравнения технологии FDDI с технологиями Ethernet и Token Ring.

Таблица 3.7. Характеристики технологий FDDI, Ethernet, Token Ring

Характеристика	FDDI	Ethernet	Token Ring

Битовая скорость	100 Мбит/с	10 Мбит/с	16 Мбит/с
Топология	Двойное кольцо деревьев	Шина/звезда	Звезда/кольцо
Метод доступа	Доля от времени оборота маркера	CSMA/CD	Приоритетная система, резервирования
Среда передачи данных	Оптоволокно, неэкранированная витая пара категории 5	Толстый коаксиал, тонкий коаксиал, витая пара категории 3, оптоволокно	Экранированная и неэкранированная витая пара, оптоволокно
Максимальная длина сети (без мостов)	200 км (100 км на кольцо)	2500м	4000м
Максимальное расстояние между узлами	2 км (не больше 11дБ потерь между узлами)	2500м	100м
Максимальное количество узлов	500 (1000 соединений)	1024	260 для экранированной витой пары, 72 для неэкранированной витой пары
Тактирование и восстановление после отказов	Распределенная реализация тактирования и восстановления после отказов	Не определены	Активный монитор

Технология FDDI разрабатывалась для применения в ответственных участках сетей – на магистральных соединениях между крупными сетями, например сетями зданий, а также для подключения к сети высокопроизводительных серверов. Поэтому главным для разработчиков было обеспечить высокую скорость передачи данных, отказоустойчивость на уровне протокола и большие расстояния между узлами сети. Все эти цели были достигнуты. В результате технология FDDI получилась качественной, но весьма дорогой. Даже появление более дешевого варианта для витой пары не намного снизило стоимость подключения одного узла к сети FDDI. Поэтому практика показала, что основной областью применения технологии FDDI стали магистрали сетей, состоящих из нескольких зданий, а также сети масштаба крупного города, то есть класса MAN. Для подключения клиентских компьютеров и даже небольших серверов технология оказалась слишком дорогой. А поскольку оборудование FDDI выпускается уже около 10 лет, значительного снижения его стоимости ожидать не приходится.

В результате сетевые специалисты с начала 90-х годов стали искать пути создания сравнительно недорогих и в то же время высокоскоростных технологий, которые бы так же успешно работали на всех этажах корпоративной сети, как это делали в 80-е годы технологии Ethernet и Token Ring.

Выводы

- Технология FDDI первой использовала волоконно-оптический кабель в локальных сетях, а также работу на скорости 100 Мбит/с.

- Существует значительная преемственность между технологиями Token Ring и FDDI: для обеих характерны кольцевая топология и маркерный метод доступа.
- Технология FDDI является наиболее отказоустойчивой технологией локальных сетей. При однократных отказах кабельной системы или станции сеть, за счет «сворачивания» двойного кольца в одинарное, остается вполне работоспособной.
- Маркерный метод доступа FDDI работает по-разному для синхронных и асинхронных кадров (тип кадра определяет станция). Для передачи синхронного кадра станция всегда может захватить пришедший маркер на фиксированное время. Для передачи асинхронного кадра станция может захватить маркер только в том случае, когда маркер выполнил оборот по кольцу достаточно быстро, что говорит об отсутствии перегрузок кольца. Такой метод доступа, во-первых, отдает предпочтение синхронным кадрам, а во-вторых, регулирует загрузку кольца, притормаживая передачу несрочных асинхронных кадров.
- В качестве физической среды технология FDDI использует волоконно-оптические кабели и UTP категории 5 (этот вариант физического уровня называется TP-PMD).
- Максимальное количество станций двойного подключения в кольце – 500, максимальный диаметр двойного кольца – 100 км. Максимальные расстояния между соседними узлами для многомодового кабеля равны 2 км, для витой пары UPT категории 5 – 100 м, а для одномодового оптоволокна зависят от его качества.

3.6. Fast Ethernet и lOOVG-AnyLAN как развитие технологии Ethernet

Классический 10-мегабитный Ethernet устраивал большинство пользователей на протяжении около 15 лет. Однако в начале 90-х годов начала ощущаться его недостаточная пропускная способность. Для компьютеров на процессорах Intel 80286 или 80386 с шинами ISA (8 Мбайт/с) или EISA (32 Мбайт/с) пропускная способность сегмента Ethernet составляла 1/8 или 1/32 канала «память-диск», и это хорошо согласовывалось с соотношением объемов данных, обрабатываемых локально, и данных, передаваемых по сети. Для более мощных клиентских станций с шиной PCI (133 Мбайт/с) эта доля упала до 1/133, что было явно недостаточно. Поэтому многие сегменты 10-мегабитного Ethernet стали перегруженными, реакция серверов в них значительно упала, а частота возникновения коллизий существенно возросла, еще более снижая полезную пропускную способность.

Назрела необходимость в разработке «нового» Ethernet, то есть технологии, которая была бы такой же эффективной по соотношению цена/качество при производительности 100 Мбит/с. В результате поисков и исследований специалисты разделились на два лагеря, что в конце концов привело к появлению двух новых технологий – Fast Ethernet и lOOVG-AnyLAN. Они отличаются степенью преемственности с классическим Ethernet.

В 1992 году группа производителей сетевого оборудования, включая таких лидеров технологии Ethernet, как SynOptics, 3Com и ряд других, образовали некоммерческое объединение Fast Ethernet Alliance для разработки стандарта новой технологии, которая должна была в максимально возможной степени сохранить особенности технологии Ethernet.

Второй лагерь возглавили компании Hewlett-Packard и AT&T, которые предложили воспользоваться удобным случаем для устранения некоторых известных недостатков технологии Ethernet. Через некоторое время к этим компаниям присоединилась компания IBM, которая внесла свой вклад предложением обеспечить в новой технологии некоторую совместимость с сетями Token Ring.

В комитете 802 института IEEE в это же время была сформирована исследовательская группа для изучения технического потенциала новых высокоскоростных технологий. За период с конца 1992 года и по конец 1993 года группа IEEE изучила 100-мегабитные решения, предложенные различными производителями. Наряду с предложениями Fast Ethernet Alliance группа рассмотрела также и высокоскоростную технологию, предложенную компаниями Hewlett-Packard и AT&T.

В центре дискуссий была проблема сохранения случайного метода доступа CSMA/CD. Предложение Fast Ethernet Alliance сохраняло этот метод и тем самым обеспечивало преемственность и согласованность сетей 10 Мбит/с и 100 Мбит/с. Коалиция HP и AT&T, которая имела поддержку значительно меньшего числа производителей в сетевой индустрии, чем Fast Ethernet Alliance, предложила совершенно новый метод доступа, названный *Demand Priority* – приоритетный доступ по требованию. Он существенно менял картину поведения узлов в сети, поэтому не смог вписаться в технологию Ethernet и стандарт 802.3, и для его стандартизации был организован новый комитет IEEE

802.12.

Осенью 1995 года обе технологии стали стандартами IEEE. Комитет IEEE 802.3 принял спецификацию Fast Ethernet в качестве стандарта 802.3u, который не является самостоятельным стандартом, а представляет собой дополнение к существующему стандарту 802.3 в виде глав с 21 по 30. Комитет 802.12 принял технологию lOOVG-AnyLAN, которая использует новый метод доступа Demand Priority и поддерживает кадры двух форматов – Ethernet и Token Ring.

3.6.1. Физический уровень технологии Fast Ethernet

Все отличия технологии Fast Ethernet от Ethernet сосредоточены на физическом уровне (рис. 3.20). Уровни MAC и LLC в Fast Ethernet остались абсолютно теми же, и их описывают прежние главы стандартов 802.3 и 802.2. Поэтому рассматривая технологию Fast Ethernet, мы будем изучать только несколько вариантов ее физического уровня.

Более сложная структура физического уровня технологии Fast Ethernet вызвана тем, что в ней используются три варианта кабельных систем:

- волоконно-оптический многомодовый кабель, используются два волокна;
- витая пара категории 5, используются две пары;
- витая пара категории 3, используются четыре пары.

Коаксиальный кабель, давший миру первую сеть Ethernet, в число разрешенных сред передачи данных новой технологии Fast Ethernet не попал. Это общая тенденция многих новых технологий, поскольку на небольших расстояниях витая пара категории 5 позволяет передавать данные с той же скоростью, что и коаксиальный кабель, но сеть получается более дешевой и .удобной в эксплуатации. На больших расстояниях оптическое волокно обладает гораздо более широкой полосой пропускания, чем коаксиал, а стоимость сети получается ненамного выше, особенно если учесть высокие затраты на поиск и устранение неисправностей в крупной кабельной коаксиальной системе.

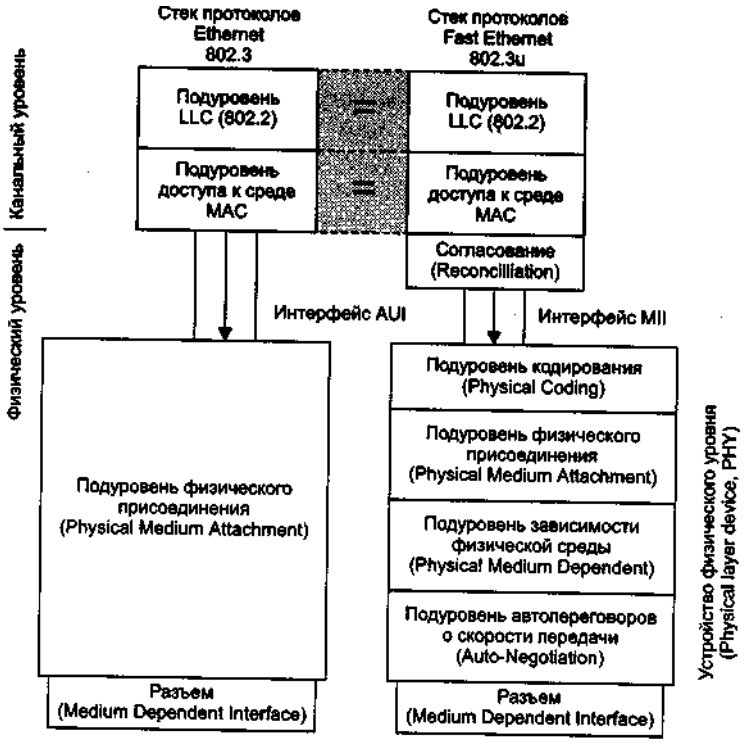


Рис. 3.20. Отличия технологии Fast Ethernet от технологии Ethernet

Отказ от коаксиального кабеля привел к тому, что сети Fast Ethernet всегда имеют иерархическую древовидную структуру, построенную на концентраторах, как и сети lOBase-T/lOBase-F. Основным отличием конфигураций сетей Fast Ethernet является сокращение диаметра сети примерно до 200 м, что объясняется уменьшением времени передачи кадра минимальной длины в 10 раз за счет увеличения скорости передачи в 10 раз по сравнению с 10-мегабитным Ethernet.

Тем не менее это обстоятельство не очень препятствует построению крупных сетей на технологии Fast Ethernet. Дело в том, что середина 90-х годов отмечена не только широким распространением

недорогих высокоскоростных технологий, но и бурным развитием локальных сетей на основе коммутаторов. При использовании коммутаторов протокол Fast Ethernet может работать в полнодуплексном режиме, в котором нет ограничений на общую длину сети, а остаются только ограничения на длину физических сегментов, соединяющих соседние устройства (адаптер - коммутатор или коммутатор - коммутатор). Поэтому при создании магистралей локальных сетей большой протяженности технология Fast Ethernet также активно применяется, но только в полнодуплексном варианте, совместно с коммутаторами.

В данном разделе рассматривается полудуплексный вариант работы технологии Fast Ethernet, который полностью соответствует определению метода доступа, описанному в стандарте 802.3. Особенности полнодуплексного режима Fast Ethernet описаны в главе 4.

По сравнению с вариантами физической реализации Ethernet (а их насчитывается шесть), в Fast Ethernet отличия каждого варианта от других глубже – меняется как количество проводников, так и методы кодирования. А так как физические варианты Fast Ethernet создавались одновременно, а не эволюционно, как для сетей Ethernet, то имелась возможность детально определить те подуровни физического уровня, которые не изменяются от варианта к варианту, и те подуровни, которые специфичны для каждого варианта физической среды.

Официальный стандарт 802.3u установил три различных спецификации для физического уровня Fast Ethernet и дал им следующие названия (рис. 3.21):

- 100Base-TX для двухпарного кабеля на неэкранированной витой паре UTP категории 5 или экранированной витой паре STP Type 1;
- 100Base-T4 для четырехпарного кабеля на неэкранированной витой паре UTP
- категории 3, 4 или 5;
- 100Base-FX для многомодового оптоволоконного кабеля, используются два волокна.

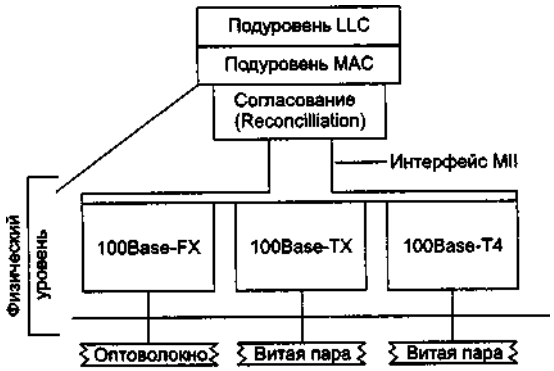


Рис. 3.21. Структура физического уровня Fast Ethernet

Для всех трех стандартов справедливы следующие утверждения и характеристики.

- Форматы кадров технологии Fast Ethernet не отличаются от форматов кадров технологий 10-мегабитного Ethernet.
- Межкадровый интервал (IPG) равен 0,96 мкс, а битовый интервал равен 10 нс. Все временные параметры алгоритма доступа (интервал отсрочки, ‘ время передачи кадра минимальной длины и т. п.), измеренные в битовых интервалах, остались прежними, поэтому изменения в разделы стандарта, касающиеся уровня MAC, не вносились.
- Признаком свободного состояния среды является передача по ней символа Idle соответствующего избыточного кода (а не отсутствие сигналов, как в стандартах Ethernet 10 Мбит/с).

Физический уровень включает три элемента:

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

Уровень согласования нужен для того, чтобы уровень MAC, рассчитанный на интерфейс AUI, смог работать с физическим уровнем через интерфейс MII.

Устройство физического уровня (PHY) состоит, в свою очередь, из нескольких подуровней (см. рис. 3.20):

- подуровня логического кодирование данных, преобразующего поступающие от уровня MAC байты в символы кода 4B/5B или 8B/6T (оба кода используются в технологии Fast Ethernet);
- подуровней физического присоединения и подуровня зависимости от физической среды (PMD), которые обеспечивают формирование сигналов в соответствии с методом физического кодирования, например NRZI или MLT-3;
- подуровня автопереговоров, который позволяет двум взаимодействующим портам автоматически выбрать наиболее эффективный режим работы, например, полудуплексный или полнодуплексный (этот подуровень является факультативным).

Интерфейс MII поддерживает независимый от физической среды способ обмена данными между подуровнем MAC и подуровнем PHY. Этот интерфейс аналогичен по назначению интерфейсу AUI классического Ethernet за исключением того, что интерфейс AUI располагался между подуровнем физического кодирования сигнала (для любых вариантов кабеля использовался одинаковый метод физического кодирования – манчестерский код) и подуровнем физического присоединения к среде, а интерфейс MII располагается между подуровнем MAC и подуровнями кодирования сигнала, которых в стандарте Fast Ethernet три – FX, TX и T4.

Разъем MII в отличие от разъема AUI имеет 40 контактов, максимальная длина кабеля MII составляет один метр. Сигналы, передаваемые по интерфейсу MII, имеют амплитуду 5 В.

Физический уровень 100Base-FX – многомодовое оптоволокно, два волокна

Эта спецификация определяет работу протокола Fast Ethernet по многомодовому оптоволокну в полудуплексном и полнодуплексном режимах на основе хорошо проверенной схемы кодирования FDDI. Как и в стандарте FDDI, каждый узел соединяется с сетью двумя оптическими волокнами, идущими от приемника (Rx) и от передатчика (Tx).

Между спецификациями 100Base-FX и 100Base-TX есть много общего, поэтому общие для двух спецификаций свойства будут даваться под обобщенным названием 100Base-FX /TX.

В то время как Ethernet со скоростью передачи 10 Мбит/с использует манчестерское кодирование для представления данных при передаче по кабелю, в стандарте Fast Ethernet определен другой метод кодирования – 4B/5B. Этот метод уже показал свою эффективность в стандарте FDDI и без изменений перенесен в спецификацию 100Base-FX/TX. При этом методе каждые 4 бита данных подуровня MAC (называемых символами) представляются 5 битами. Избыточный бит позволяет применить потенциальные коды при представлении каждого из пяти бит в виде электрических или оптических импульсов. Существование запрещенных комбинаций символов позволяет отбраковывать ошибочные символы, что повышает устойчивость работы сетей с 100BaseFX/TX.



Рис. 3.22. Непрерывный поток данных спецификаций 100Base-FX/TX

Для отделения кадра Ethernet от символов Idle используется комбинация символов Start Delimiter (пара символов J (11000) и K (10001) кода 4B/5B, а после завершения кадра перед первым символом Idle вставляется символ T (рис. 3.22).

После преобразования 4-битовых порций кодов MAC в 5-битовые порции физического уровня их необходимо представить в виде оптических или электрических сигналов в кабеле, соединяющем узлы сети. Спецификации 100Base-FX и 100Base-TX используют для этого различные методы физического кодирования – NRZI и MLT-3 соответственно (как и в технологии FDDI при работе через оптоволокно и витую пару).

Физический уровень 1 OOBase-TX - витая пара UTP Cat 5 или STP Type 1, две пары

В качестве среды передачи данных спецификация 100Base-TX использует кабель UTP категории 5 или кабель STP Type 1. Максимальная длина кабеля в обоих случаях – 100 м.

Основные отличия от спецификации 100Base-FX – использование метода MLT-3 для передачи

сигналов 5-битовых порций кода 4B/5B по витой паре, а также наличие функции автопереговоров (Auto-negotiation) для выбора режима работы порта. Схема автопереговоров позволяет двум соединенным физически устройствам, которые поддерживают несколько стандартов физического уровня, отличающихся битовой скоростью и количеством витых пар, выбрать наиболее выгодный режим работы. Обычно процедура автопереговоров происходит при подсоединении сетевого адаптера, который может работать на скоростях 10 и 100 Мбит/с, к концентратору или коммутатору.

Описанная ниже схема Auto-negotiation сегодня является стандартом технологии 100Base-T. До этого производители применяли различные собственные схемы автоматического определения скорости работы взаимодействующих портов, которые не были совместимы. Принятую в качестве стандарта схему Auto-negotiation предложила первоначально компания National Semiconductor под названием NWay.

Всего в настоящее время определено 5 различных режимов работы, которые могут поддерживать устройства 100Base-TX или 100Base-T4 на витых парах:

- 10Base-T – 2 пары категории 3;
- 10Base-T full-duplex – 2 пары категории 3;
- 100Base-TX - 2 пары категории 5 (или Type 1A STP);
- 100Base-T4 – 4 пары категории 3;
- 100Base-TX full-duplex - 2 пары категории 5 (или Type 1A STP).

Режим 10Base-T имеет самый низкий приоритет при переговорном процессе, а полнодуплексный режим 100Base-T4 – самый высокий. Переговорный процесс происходит при включении питания устройства, а также может быть инициирован в любой момент модулем управления устройства.

Устройство, начавшее процесс auto-negotiation, посылает своему партнеру пачку специальных импульсов *Fast Link Pulse burst* (*FLP*), в котором содержится 8-битное слово, кодирующее предлагаемый режим взаимодействия, начиная с самого приоритетного, поддерживаемого данным узлом.

Если узел-партнер поддерживает функцию auto-negotuiation и также может поддерживать предложенный режим, он отвечает пачкой импульсов FLP, в которой подтверждает данный режим, и на этом переговоры заканчиваются. Если же узел-партнер может поддерживать менее приоритетный режим, то он указывает его в ответе, и этот режим выбирается в качестве рабочего. Таким образом, всегда выбирается наиболее приоритетный общий режим узлов.

Узел, который поддерживает только технологию 10Base-T, каждые 16 мс посылает манчестерские импульсы для проверки целостности линии, связывающей его с соседним узлом. Такой узел не понимает запрос FLP, который делает ему узел с функцией Auto-negotiation, и продолжает посылать свои импульсы. Узел, получивший в ответ на запрос FLP только импульсы проверки целостности линии, понимает, что его партнер может работать только по стандарту 10Base-T, и устанавливает этот режим работы и для себя.

Физический уровень 100Base-T4 – витая пара UTP Cat 3, четыре пары

Спецификация 100Base-T4 была разработана для того, чтобы можно было использовать для высокоскоростного Ethernet имеющуюся проводку на витой паре категории 3. Эта спецификация позволяет повысить общую пропускную способность за счет одновременной передачи потоков бит по всем 4 парам кабеля.

Спецификация 100Base-T4 появилась позже других спецификаций физического уровня Fast Ethernet. Разработчики этой технологии в первую очередь хотели создать физические спецификации, наиболее близкие к спецификациям 10Base-T и 10Base-F, которые работали на двух линиях передачи данных: двух парах или двух волокнах. Для реализации работы по двум витым парам пришлось перейти на более качественный кабель категории 5.

В то же время разработчики конкурирующей технологии lOOVG-AnyLAN изначально сделали ставку на работу по витой паре категории 3; самое главное преимущество состояло не столько в стоимости, а в том, что она была уже проложена в подавляющем числе зданий. Поэтому после выпуска спецификаций 100Base-TX и 100Base-FX разработчики технологии Fast Ethernet реализовали свой вариант физического уровня для витой пары категории 3.

Вместо кодирования 4B/5B в этом методе используется кодирование 8B/6T, которое обладает более узким спектром сигнала и при скорости 33 Мбит/с укладывается в полосу 16 МГц витой пары категории

3 (при кодировании 4B/5B спектр сигнала в эту полосу не укладывается). Каждые 8 бит информации уровня MAC кодируются 6-ю троичными цифрами (ternary symbols), то есть цифрами, имеющими три состояния. Каждая троичная цифра имеет длительность 40 нс. Группа из 6-ти троичных цифр затем передается на одну из трех передающих витых пар, независимо и последовательно.

Четвертая пара всегда используется для прослушивания несущей частоты в целях обнаружения коллизии. Скорость передачи данных по каждой из трех передающих пар равна 33,3 Мбит/с, поэтому общая скорость протокола 100Base-T4 составляет 100 Мбит/с. В то же время из-за принятого способа кодирования скорость изменения сигнала на каждой паре равна всего 25 Мбод, что и позволяет использовать витую пару категории 3.

На рис. 3.23 показано соединение порта MDI сетевого адаптера 100Base-T4 с портом MDI-X концентратора (приставка X говорит о том, что у этого разъема присоединения приемника и передатчика меняются парами кабеля по сравнению с разъемом сетевого адаптера, что позволяет проще соединять пары проводов в кабеле – без перекрещивания). Пара 1-2 всегда требуется для передачи данных от порта MDI к порту MDI-X, пара 3–6 –для приема данных портом MDI от порта MDI-X, а пары 4–5 и 7-8 являются двунаправленными и используются как для приема, так и для передачи, в зависимости от потребности.

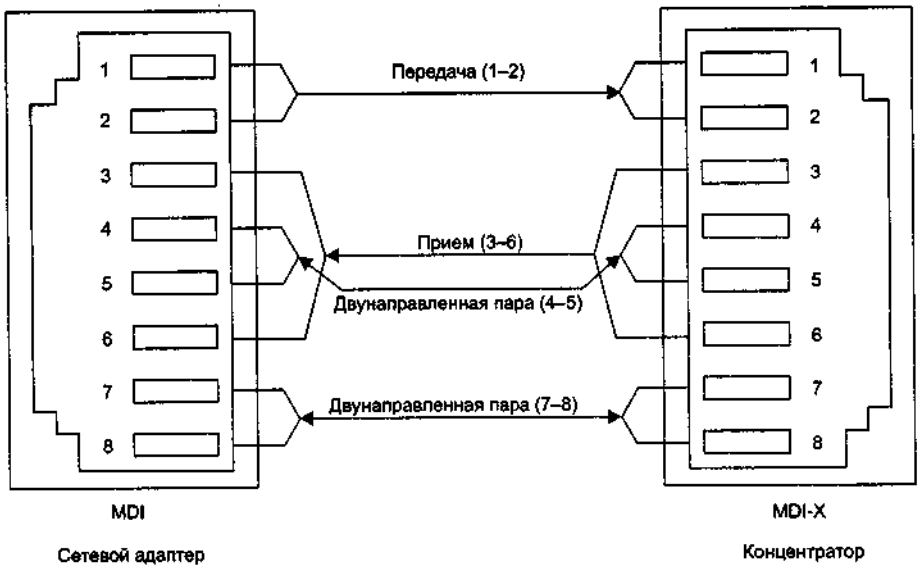


Рис. 3.23. Соединение узлов по спецификации 100Base-T4

3.6.2. Правила построения сегментов Fast Ethernet при использовании повторителей

Технология Fast Ethernet, как и все некоаксиальные варианты Ethernet, рассчитана на использование концентраторов-повторителей для образования связей в сети.

Правила корректного построения сегментов сетей Fast Ethernet включают:

- ограничения на максимальные длины сегментов, соединяющих DTE с DTE;
- ограничения на максимальные длины сегментов, соединяющих DTE с портом повторителя;
- ограничения на максимальный диаметр сети;
- ограничения на максимальное число повторителей и максимальную длину сегмента, соединяющего повторители.

Ограничения длин сегментов DTE-DTE

В качестве DTE (Data Terminal Equipment) может выступать любой источник кадров данных для сети: сетевой адаптер, порт моста, порт маршрутизатора, модуль управления сетью и другие подобные устройства. Отличительной особенностью DTE является то, что он вырабатывает новый кадр для разделяемого сегмента (мост или коммутатор, хотя и передают через выходной порт кадр, который выработал в свое время сетевой адаптер, но для сегмента сети, к которому подключен выходной порт, этот кадр является новым). Порт повторителя не является DTE, так как он побитно повторяет уже

появившийся в сегменте кадр.

В типичной конфигурации сети Fast Ethernet несколько DTE подключается к портам повторителя, образуя сеть звездообразной топологии. Соединения DTE–DTE в разделяемых сегментах не встречаются (если исключить экзотическую конфигурацию, когда сетевые адаптеры двух компьютеров соединены прямо друг с другом кабелем), а вот для мостов/коммутаторов и маршрутизаторов такие соединения являются нормой – когда сетевой адаптер прямо соединен с портом одного из этих устройств, либо эти устройства соединяются друг с другом.

Спецификация IEEE 802.3u определяет следующие максимальные длины сегментов DTE-DTE, приведенные в табл. 3.8.

Таблица 3.8. Максимальные длины сегментов DTE-DTE

Стандарт	Тип кабеля	Максимальная длина сегмента
100Base-TX	Категория 5 UTP	100м
100Base-FX	Многомодовое оптоволокно 62,5/125 мкм	412 м (полудуплекс) 2 км (полный дуплекс)
100Base-T4	Категория 3, 4 или 5 UTP	100м

Ограничения сетей Fast Ethernet, построенных на повторителях

Повторители Fast Ethernet делятся на два класса. Повторители класса I поддерживают все типы логического кодирования данных: как 4B/5B, так и 8B/6T. Повторители класса II поддерживают только какой-либо один тип логического кодирования – либо 4B/5B, либо 8B/6T. То есть повторители класса I позволяют выполнять трансляцию логических кодов с битовой скоростью 100 Мбит/с, а повторителям класса II эта операция недоступна.

Поэтому повторители класса I могут иметь порты всех трех типов физического уровня: 100Base-TX, 100Base-FX и 100Base-T4. Повторители класса II имеют либо все порты 100Base-T4, либо порты 100Base-TX и 100Base-FX, так как последние используют один логический код 4B/5B.

В одном домене коллизий допускается наличие только одного повторителя класса I. Это связано с тем, что такой повторитель вносит большую задержку при распространении сигналов из-за необходимости трансляции различных систем сигнализации – 70 bt.

Повторители класса II вносят меньшую задержку при передаче сигналов; 46 bt для портов TX/FX и 33,5 bt для портов, T4. Поэтому максимальное число повторителей класса II в домене коллизий – 2, причем они должны быть соединены между собой кабелем не длиннее 5 метров.

Небольшое количество повторителей Fast Ethernet не является серьезным препятствием при построении больших сетей, так как применение коммутаторов и маршрутизаторов делит сеть на несколько доменов коллизий, каждый из которых будет строиться на одном или двух повторителях. Общая длина сети не будет иметь в этом случае ограничений.

В табл. 3.9 приведены правила построения сети на основе повторителей класса I.

Таблица 3.9. Параметры сетей на основе повторителей класса I

Тип кабелей	Максимальный диаметр сети, м	Максимальная длина сегмента, м
Только витая пара (TX)	200	100
Только оптоволокно (FX)	272	136
Несколько сегментов на витой паре и один на оптоволокне	260	100 (TX) 160 (FX)

Несколько сегментов на витой паре и несколько сегментов на оптоволокне	272	100 (TX) 136 (FX)

Эти ограничения проиллюстрированы типовыми конфигурациями сетей, показанными на рис. 3.24.

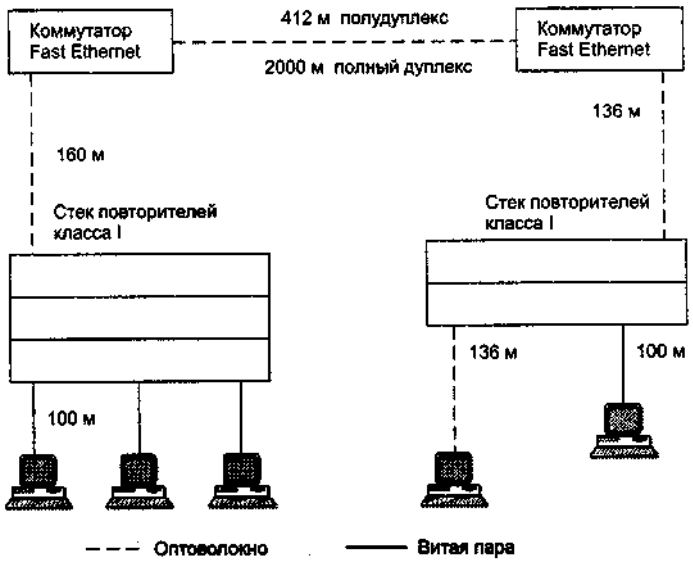


Рис. 3.24. Примеры построения сети Fast Ethernet с помощью повторителей класса I

Таким образом, правило 4-х хабов превратилось для технологии Fast Ethernet в правило одного или двух хабов, в зависимости от класса хаба.

При определении корректности конфигурации сети можно не руководствоваться правилами одного или двух хабов, а рассчитывать время двойного оборота сети, как это было показано выше для сети Ethernet 10 Мбит/с.

Как и для технологии Ethernet 10 Мбит/с, комитет 802.3 дает исходные данные для расчета времени двойного оборота сигнала. Однако при этом сама форма представления этих данных и методика расчета несколько изменились. Комитет предоставляет данные об удвоенных задержках, вносимых каждым элементом сети, не разделяя сегменты сети на левый, правый и промежуточный. Кроме того, задержки, вносимые сетевыми адаптерами, учитывают преамбулы кадров, поэтому время двойного оборота нужно сравнивать с величиной 512 битовых интервала (bt), то есть со временем передачи кадра минимальной длины без преамбулы.

Для повторителей класса I время двойного оборота можно рассчитать следующим образом.

Задержки, вносимые прохождением сигналов по кабелю, рассчитываются на основании данных табл. 3.10, в которой учитывается удвоенное прохождение сигнала по кабелю.

Таблица 3.10. Задержки, вносимые кабелем

Тип кабеля	Удвоенная задержка в bt на 1 м	Удвоенная задержка на кабеле максимальной длины
UTP Cat 3	1,14 bt	114 bt (100 м)
UTP Cat 4	1,14 bt	114 bt (100 м)
UTP Cat 5	1,112 bt	111,2 bt (100 м)
STP	1,112 bt	111,2 bt (100 м)
Оптоволокно	1,0 bt	412 bt (412 м)

Задержки, которые вносят два взаимодействующих через повторитель сетевых адаптера (или порта коммутатора), берутся из табл. 3.11.

Таблица 3.11. Задержки, вносимые сетевыми адаптерами

Тип сетевых адаптеров	Максимальная задержка при двойном обороте
Два адаптера TX/FX	100 bt
Два адаптера T4	138 bt
Один адаптер TX/FX и один T4	127 bt

Учитывая, что удвоенная задержка, вносимая повторителем класса I, равна 140 bt, можно рассчитать время двойного оборота для произвольной конфигурации сети, естественно, учитывая максимально возможные длины непрерывных сегментов кабелей, приведенные в табл. 3.10. Если получившееся значение меньше 512, значит, по критерию распознавания коллизий сеть является корректной. Комитет 802.3 рекомендует оставлять запас в 4 bt для устойчиво работающей сети, но разрешает выбирать эту величину из диапазона от 0 до 5 bt.

Рассчитаем для примера рекомендуемую в таблице конфигурацию сети, состоящую из одного повторителя и двух оптоволоконных сегментов длиной по 136 метров.

Каждый сегмент вносит задержку по 136 bt, пара сетевых адаптеров FX даёт задержку в 100 bt, а сам повторитель вносит задержку в 140 bt. Сумма задержек равна 512 bt, что говорит о том, что сеть корректна, но запас принят равным 0.

3.6.3. Особенности технологии 100VG-AnyLAN

Технология lOOVG-AnyLAN отличается от классического Ethernet в значительно большей степени, чем Fast Ethernet. Главные отличия перечислены ниже.

- Используется другой метод доступа Demand Priority, который обеспечивает более справедливое распределение пропускной способности сети по сравнению с методом CSMA/CD. Кроме того, этот метод поддерживает приоритетный доступ для синхронных приложений.
- Кадры передаются не всем станциям сети, а только станции назначения.
- В сети есть выделенный арбитр доступа – концентратор, и это заметно отличает данную технологию от других, в которых применяется распределенный между станциями сети алгоритм доступа.
- Поддерживаются кадры двух технологий – Ethernet и Token Ring (именно это обстоятельство дало добавку AnyLAN в названии технологии).
- Данные передаются одновременно по 4 парам кабеля UTP категории 3. По каждой паре данные передаются со скоростью 25 Мбит/с, что в сумме дает 100 Мбит/с. В отличие от Fast Ethernet в сетях 100VG-AnyLAN нет коллизий, поэтому удалось использовать для передачи все четыре пары стандартного кабеля категории 3. Для кодирования данных применяется код 5B/6B, который обеспечивает спектр сигнала в диапазоне до 16 МГц (полоса пропускания UTP категории 3) при скорости передачи данных 25 Мбит/с.

Метод доступа Demand Priority основан на передаче концентратору функций арбитра, решающего проблему доступа к разделяемой среде. Сеть 100VG-AnyLAN состоит из центрального концентратора, называемого также корневым, и соединенных с ним конечных узлов и других концентраторов (рис. 3.25).

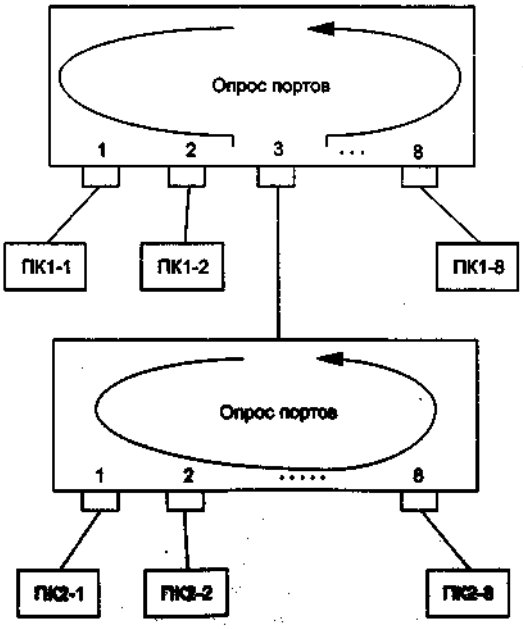


Рис. 3.25. Сеть 100VG-AnyLAN

Допускаются три уровня каскадирования. Каждый концентратор и сетевой адаптер lOOVG-AnyLAN должен быть настроен либо на работу с кадрами Ethernet либо с кадрами Token Ring, причем одновременно циркуляция обоих типов кадров не допускается.

Концентратор циклически выполняет опрос портов. Станция, желающая передать пакет, посылает специальный низкочастотный сигнал концентратору, запрашивая передачу кадра и указывая его приоритет. В сети lOOVG-AnyLAN используются два уровня приоритетов – низкий и высокий. Низкий уровень приоритета соответствует обычным данным (файловая служба, служба печати и т. п.), а высокий приоритет соответствует данным, чувствительным к временным задержкам (например, мультимедиа). Приоритеты запросов имеют статическую и динамическую составляющие, то есть станция с низким уровнем приоритета, долго не имеющая доступа к сети, получает высокий приоритет.

Если сеть свободна, то концентратор разрешает передачу пакета. После анализа адреса получателя в принятом пакете концентратор автоматически отправляет пакет станции назначения. Если сеть занята, концентратор ставит полученный запрос в очередь, которая обрабатывается в соответствии с порядком поступления запросов и с учетом приоритетов. Если к порту подключен другой концентратор, то опрос приостанавливается до завершения опроса концентратором нижнего уровня. Станции, подключенные к концентраторам различного уровня иерархии, не имеют преимуществ по доступу к разделяемой среде, так как решение о предоставлении доступа принимается после проведения опроса всеми концентраторами опроса всех своих портов.

Остается неясным вопрос – каким образом концентратор узнает, к какому порту подключена станция назначения? Во всех других технологиях кадр просто передавался всем станциям сети, а станция назначения, распознав свой адрес, копировала кадр в буфер. Для решения этой задачи концентратор узнает адрес MAC станции в момент физического присоединения ее к сети кабелем. Если в других технологиях процедура физического соединения выясняет связность кабеля (link test в технологии 10Base-T), тип порта (технология FDDI), скорость работы порта (процедура auto-negotiation в Fast Ethernet), то в технологии lOOVG-AnyLAN концентратор при установлении физического соединения выясняет адрес MAC станции. И запоминает его в таблице адресов MAC, аналогичной таблице моста/коммутатора. Отличие концентратора lOOVG-AnyLAN от моста/коммутатора в том, что у него нет внутреннего буфера для хранения кадров. Поэтому он принимает от станций сети только один кадр, отправляет его на порт назначения и, пока этот кадр не будет полностью принят станцией назначения, новые кадры концентратор не принимает. Так что эффект разделяемой среды сохраняется. Улучшается только безопасность сети – кадры не попадают на чужие порты, и их труднее перехватить.

Технология lOOVG-AnyLAN поддерживает несколько спецификаций физического уровня. Первоначальный вариант был рассчитан на четыре неэкранированные витые пары категорий 3,4,5. Позже появились варианты физического уровня, рассчитанные на две неэкранированные витые пары категории 5, две экранированные витые пары типа 1 или же два оптических многомодовых

оптоволокна.

Важная особенность технологии l00VG-AnyLAN – сохранение форматов кадров Ethernet и Token Ring. Сторонники l00VG-AnyLAN утверждают, что этот подход облегчит межсетевое взаимодействие через мосты и маршрутизаторы, а также обеспечит совместимость с существующими средствами сетевого управления, в частности с анализаторами протоколов.

Несмотря на много хороших технических решений, технология l00VG-AnyLAN не нашла большого количества сторонников и значительно уступает по популярности технологии Fast Ethernet. Возможно, это произошло из-за того, что технические возможности поддержки разных типов трафика у технологии ATM существенно шире, чем у l00VG-AnyLAN. Поэтому при необходимости тонкого обеспечения качества обслуживания применяют (или собираются применять) технологию ATM. А для сетей, в которых нет необходимости поддерживать качество обслуживания на уровне разделяемых сегментов, более привычной оказалась технология Fast Ethernet. Тем более что для поддержки очень требовательных к скорости передачи данных приложений имеется технология Gigabit Ethernet, которая, сохраняя преемственность с Ethernet и Fast Ethernet, обеспечивает скорость передачи данных 1000 Мбит/с.

Выводы

- Потребности в высокоскоростной и в то же время недорогой технологии для подключения к сети мощных рабочих станций привели в начале 90-х годов к созданию инициативной группы, которая занялась поисками нового Ethernet – такой же простой и эффективной технологии, но работающей на скорости 100 Мбит/с.
- Специалисты разбились на два лагеря, что в конце концов привело к появлению двух стандартов, принятых осенью 1995 года: комитет 802.3 утвердил стандарт Fast Ethernet, почти полностью повторяющий технологию Ethernet 10 Мбит/с, а специально созданный комитет 802.12 утвердил стандарт технологии l00VG-AnyLAN, которая сохраняла формат кадра Ethernet, но существенно изменяла метод доступа.
- Технология Fast Ethernet сохранила в неприкосновенности метод доступа CSMA/CD, оставив в нем тот же алгоритм и те же временные параметры в битовых интервалах (сам битовый интервал уменьшился в 10 раз). Все отличия Fast Ethernet от Ethernet проявляются на физическом уровне.
- В стандарте Fast Ethernet определены три спецификации физического уровня: 100Base-TX для 2-х пар UTP категории 5 или 2-х пар STP Type 1 (метод кодирования 4B/5B), 100Base-FX для многомодового волоконно-оптического кабеля с двумя оптическими волокнами (метод кодирования 4B/5B) и 100Base-T4, работающую на 4-х парах UTP категории 3, но использующую одновременно только три пары для передачи, а оставшуюся – для обнаружения коллизии (метод кодирования 8B/6T).
- Стандарты 100Base-TX/FX могут работать в полнодуплексном режиме.
- Максимальный диаметр сети Fast Ethernet равен приблизительно 200 м, а более точные значения зависят от спецификации физической среды. В домене коллизий Fast Ethernet допускается не более одного повторителя класса I (позволяющего транслировать коды 4B/5B в коды 8B/6T и обратно) и не более двух повторителей класса II (не позволяющих выполнять трансляцию кодов).
- Технология Fast Ethernet при работа на витой паре позволяет за счет процедуры автопереговоров двум портам выбирать наиболее эффективный режим работы – скорость 10 Мбит/с или 100 Мбит/с, а также полудуплексный или полнодуплекеный режим.
- В технологии l00VG-AnyLAN арбитром, решающим вопрос о предоставлении станциям доступа к разделяемой среде, является концентратор, поддерживающий метод Demand Priority – приоритетные требования. Метод Demand Priority оперирует с двумя уровнями приоритетов, выставляемыми станциями, причем приоритет станции, долго не получающей обслуживания, повышается динамически.
- Концентраторы VG могут объединяться в иерархию, причем порядок доступа к среде не зависит от того, к концентратору какого уровня подключена станция, а зависит только от приоритета кадра и времени подачи заявки на обслуживание.

- Технология lOOVG-AnyLAN поддерживает кабель UTP категории 3, причем для обеспечения скорости 100 Мбит/с передает данные одновременно по 4-м парам. Имеется также физический стандарт для кабеля UTP категории 5, кабеля STP Type 1 и волоконно-оптического кабеля.

3.7. Высокоскоростная технология Gigabit Ethernet

3.7.1. Общая характеристика стандарта

Достаточно быстро после появления на рынке продуктов Fast Ethernet сетевые интеграторы и администраторы почувствовали определенные ограничения при построении корпоративных сетей. Во многих случаях серверы, подключенные по 100-мегабитному каналу, перегружали магистрали сетей, работающие также на скорости 100 Мбит/с – магистрали FDDI и Fast Ethernet. Ощущалась потребность в следующем уровне иерархии скоростей. В 1995 году более высокий уровень скорости могли предоставить только коммутаторы ATM, а при отсутствии в то время удобных средств миграции этой технологии в локальные сети (хотя спецификация LAN Emulation – LANE была принята в начале 1995 года, практическая ее реализация была впереди) внедрять их в локальную сеть почти никто не решался. Кроме того, технология ATM отличалась очень высоким уровнем стоимости.

Поэтому логичным выглядел следующий шаг, сделанный IEEE, – через 5 месяцев после окончательного принятия стандарта Fast Ethernet в июне 1995 года исследовательской группе по изучению высокоскоростных технологий IEEE было предписано заняться рассмотрением возможности выработки стандарта Ethernet с еще более высокой битовой скоростью.

Летом 1996 года было объявлено о создании группы 802.3z для разработки протокола, максимально подобного Ethernet, но с битовой скоростью 1000 Мбит/с. Как и в случае Fast Ethernet, сообщение было воспринято сторонниками Ethernet с большим энтузиазмом.

Основной причиной энтузиазма была перспектива такого же плавного перевода магистралей сетей на Gigabit Ethernet, подобно тому, как были переведены на Fast Ethernet перегруженные сегменты Ethernet, расположенные на нижних уровнях иерархии сети. К тому же опыт передачи данных на гигабитных скоростях уже имелся, как в территориальных сетях (технология SDH), так и в локальных – технология Fibre Channel, которая используется в основном для подключения высокоскоростной периферии к большим компьютерам и передает данные по волоконно-оптическому кабелю со скоростью, близкой к гигабитной, посредством избыточного кода 8B/10B.

В образованный для согласования усилий в этой области Gigabit Ethernet Alliance с самого начала вошли такие флагманы отрасли, как Bay Networks, Cisco Systems и 3Com. За год своего существования количество участников Gigabit Ethernet Alliance существенно выросло и насчитывает сейчас более 100. В качестве первого варианта физического уровня был принят уровень технологии Fiber Channel, с ее кодом 8B/10B (как и в случае Fast Ethernet, когда для ускорения работ был принят отработанный физический уровень FDDI).

Первая версия стандарта была рассмотрена в январе 1997 года, а окончательно стандарт 802.3z был принят 29 июня 1998 года на заседании комитета IEEE 802.3. Работы по реализации Gigabit Ethernet на витой паре категории 5 были переданы специальному комитету 802.3aЬ, который уже рассмотрел несколько вариантов проекта этого стандарта, причем с июля 1998 года проект приобрел достаточно стабильный характер. Окончательное принятие стандарта 802.3ab ожидается в сентябре 1999 года.

Не дожидаясь принятия стандарта, некоторые компании выпустили первое оборудование Gigabit Ethernet на оптоволоконном кабеле уже к лету 1997 года.

Основная идея разработчиков стандарта Gigabit Ethernet состоит в максимальном сохранении идей классической технологии Ethernet при достижении битовой скорости в 1000 Мбит/с.

Так как при разработке новой технологии естественно ожидать некоторых технических новинок, идущих в общем русле развития сетевых технологий, то важно отметить, что Gigabit Ethernet, так же как и его менее скоростные собратья, на уровне протокола *не будет* поддерживать:

- качество обслуживания;
- избыточные связи;
- тестирование работоспособности узлов и оборудования (в последнем случае – за исключением тестирования связи порт – порт, как это делается для Ethernet 10Base-T и 10Base-F и Fast Ethernet).

Все три названных свойства считаются весьма перспективными и полезными в современных сетях, а

особенно в сетях ближайшего будущего. Почему же авторы Gigabit Ethernet отказываются от них?

По поводу качества обслуживания коротко можно ответить так: «сила есть – ума не надо». Если магистраль сети будет работать со скоростью в 20 000 раз превышающей среднюю скорость сетевой активности клиентского компьютера и в 100 раз превышающей среднюю сетевую активность сервера с сетевым адаптером 100 Мбит/с, то о задержках пакетах на магистрали во многих случаях можно не заботиться вообще. При небольшом коэффициенте загрузки магистрали 1000 Мбит/с очереди в коммутаторах Gigabit Ethernet будут небольшими, а время буферизации и коммутации на такой скорости составляет единицы и даже доли микросекунд.

Ну а если все же магистраль загрузится на достаточную величину, то приоритет чувствительному к задержкам или требовательному к средней скорости трафику можно предоставить с помощью техники приоритетов в коммутаторах – соответствующие стандарты для коммутаторов уже приняты (они будут рассматриваться в следующей главе). Зато можно будет пользоваться весьма простой (почти как Ethernet) технологией, принципы работы которой известны практически всем сетевым специалистам.

Главная идея разработчиков технологии Gigabit Ethernet состоит в том, что существует и будет существовать весьма много сетей, в которых высокая скорость магистрали и возможность назначения пакетам приоритетов в коммутаторах будут вполне достаточны для обеспечения качества транспортного обслуживания всех клиентов сети. И только в тех редких случаях, когда и магистраль достаточно загружена, и требования к качеству обслуживания очень жесткие, нужно применять технологию ATM, которая действительно за счет высокой технической сложности дает гарантии качества обслуживания для всех основных видов трафика.

Избыточные связи и тестирование оборудования не будут поддерживаться технологией Gigabit Ethernet из-за того, что с этими задачами хорошо справляются протоколы более высоких уровней, например Spanning Tree, протоколы маршрутизации и т. п. Поэтому разработчики технологии решили, что нижний уровень просто должен быстро передавать данные, а более сложные и более редко встречающиеся задачи (например, приоритезация трафика) должны передаваться верхним уровням.

Что же общего имеется в технологии Gigabit Ethernet по сравнению с технологиями Ethernet и Fast Ethernet?

- Сохраняются все форматы кадров Ethernet.
- По-прежнему будут существовать полудуплексная версия протокола, поддерживающая метод доступа CSMA/CD, и полнодуплексная версия, работающая с коммутаторами. По поводу сохранения полудуплексной версии протокола сомнения были еще у разработчиков Fast Ethernet, так как сложно заставить работать алгоритм CSMA/CD на высоких скоростях. Однако метод доступа остался неизменным в технологии Fast Ethernet, и его решили оставить в новой технологии Gigabit Ethernet. Сохранение недорогого решения для разделяемых сред позволит применить Gigabit Ethernet в небольших рабочих группах, имеющих быстрые серверы и рабочие станции.
- Поддерживаются все основные виды кабелей, используемых в Ethernet и Fast Ethernet: волоконно-оптический, витая пара категории 5, коаксиал.

Тем не менее разработчикам технологии Gigabit Ethernet для сохранения приведенных выше свойств пришлось внести изменения не только в физический уровень, как это было в случае Fast Ethernet, но и в уровень MAC.

Перед разработчиками стандарта Gigabit Ethernet стояло несколько трудно разрешимых проблем. Одной из них была задача обеспечения приемлемого диаметра сети для полудуплексного режима работы. В связи с ограничениями, накладываемыми методом CSMA/CD на длину кабеля, версия Gigabit Ethernet для разделяемой среды допускала бы длину сегмента всего в 25 метров при сохранении размера кадров и всех параметров метода CSMA/CD неизменными. Так как существует большое количество применений, когда нужно повысить диаметр сети хотя бы до 200 метров, необходимо было каким-то образом решить эту задачу за счет минимальных изменений в технологии Fast Ethernet.

Другой сложнейшей задачей было достижение битовой скорости 1000 Мбит/с на основных типах кабелей. Даже для оптоволокна достижение такой скорости представляет некоторые проблемы, так как технология Fibre Channel, физический уровень которой был взят за основу для оптоволоконной версии Gigabit Ethernet, обеспечивает скорость передачи данных всего в 800 Мбит/с (битовая скорость на линии равна в этом случае примерно 1000 Мбит/с, но при методе кодирования 8B/ЮВ полезная битовая скорость на 25 % меньше скорости импульсов на линии).

И наконец, самая сложная задача – поддержка кабеля на витой паре. Такая задача на первый взгляд

кажется неразрешимой – ведь даже для 100-мегабитных протоколов пришлось использовать достаточно сложные методы кодирования, чтобы уложить спектр сигнала в полосу пропускания кабеля. Однако успехи специалистов по кодированию, проявившиеся в последнее время в новых стандартах модемов, показали, что задача имеет шансы на решение. Чтобы не тормозить принятие основной версии стандарта Gigabit Ethernet, использующего оптоволокно и коаксиал, был создан отдельный комитет 802.3ab, который занимается разработкой стандарта Gigabit Ethernet на витой паре категории 5.

Все эти задачи были успешно решены.

3.7.2. Средства обеспечения диаметра сети в 200 м на разделяемой среде

Для расширения максимального диаметра сети Gigabit Ethernet в полудуплексном режиме до 200 м разработчики технологии предприняли достаточно естественные меры, основывающиеся на известном соотношения времени передачи кадра минимальной длины и временем двойного оборота.

Минимальный размер кадра был увеличен (без учета преамбулы) с 64 до 512 байт или до 4096 bt. Соответственно, время двойного оборота теперь также можно было увеличить до 4095 bt, что делает допустимым диаметр сети около 200 м при использовании одного повторителя. При двойной задержке сигнала в 10 bt/m оптоволоконные кабели длиной 100 м вносят вклад во время двойного оборота по 1000 bt, и если повторитель и сетевые адаптеры будут вносить такие же задержки, как в технологии Fast Ethernet (данные для которых приводились в предыдущем разделе), то задержка повторителя в 1000 bt и пары сетевых адаптеров в 1000 bt дадут в сумме время двойного оборота 4000 bt, что удовлетворяет условию распознавания коллизий. Для увеличения длины кадра до требуемой в новой технологии величины сетевой адаптер должен дополнить поле данных до длины 448 байт так называемым *расширением* (*extention*), представляющим собой поле, заполненное запрещенными символами кода 8B/10B, которые невозможно принять за коды данных.

Для сокращения накладных расходов при использовании слишком длинных кадров для передачи коротких квитанций разработчики стандарта разрешили конечным узлам передавать несколько кадров подряд, без передачи среды другим станциям. Такой режим получил название Burst Mode – монопольный пакетный режим. Станция может передать подряд несколько кадров с общей длиной не более 65 536 бит или 8192 байт. Если станции нужно передать несколько небольших кадров, то она может не дополнять их до размера в 512 байт, а передавать подряд до исчерпания предела в 8192 байт (в этот предел входят все байты кадра, в том числе преамбула, заголовок, данные и контрольная сумма). Предел 8192 бейт называется BurstLength. Если станция начала передавать кадр и предел BurstLength был достигнут в середине кадра, то кадр разрешается передать до конца.

Увеличение «совмещенного» кадра до 8192 байт несколько задерживает доступ к разделяемой среде других станций, но при скорости 1000 Мбит/с эта задержка не столь существенна.

3.7.3. Спецификации физической среды стандарта 802.3z

В стандарте 802.3z определены следующие типы физической среды:

- одномодовый волоконно-оптический кабель;
- многомодовый волоконно-оптический кабель 62,5/125;
- многомодовый волоконно-оптический кабель 50/125;
- двойной коаксиал с волновым сопротивлением 75 Ом.

Многомодовый кабель

Для передачи данных по традиционному для компьютерных сетей многомодовому волоконно-оптическому кабелю стандарт определяет применение излучателей, работающих на двух длинах волн: 1300 и 850 нм. Применение светодиодов с длиной волны 850 нм объясняется тем, что они намного дешевле, чем светодиоды, работающие на волне 1300 нм, хотя при этом максимальная длина кабеля уменьшается, так как затухание многомодового оптоволокна на волне 850 м более чем в два раза выше, чем на волне 1300 нм. Однако возможность удешевления чрезвычайно важна для такой в целом дорогой технологии, как Gigabit Ethernet.

Для многомодового оптоволокна стандарт 802.3z определил спецификации 1000Base-SX и lOOOBase-LX.

В первом случае используется длина волны 850 нм (S означает Short Wavelength, короткая волна), а во втором – 1300 нм (L – от Long Wavelength, длинная волна).

Для спецификации 1000Base-SX предельная длина оптоволоконного сегмента для кабеля 62,5/125 оставляет 220 м, а для кабеля 50/125 – 500 м. Очевидно, что эти максимальные значения могут достигаться только для полнодуплексной передачи данных, так как время двойного оборота сигнала на двух отрезках 220 м равно 4400 bt, что превосходит предел 4095 bt даже без учета повторителя и сетевых адаптеров. Для полудуплексной передачи максимальные значения сегментов оптоволоконного кабеля всегда должны быть меньше 100 м. Приведенные расстояния в 220 и 500 м рассчитаны для худшего по стандарту случая полосы пропускания многомодового кабеля, находящегося в пределах от 160 до 500 МГц/км. Реальные кабели обычно обладают значительно лучшими характеристиками, находящимися между 600 и 1000 МГц/км. В этом случае можно увеличить длину кабеля до примерно 800 м.

Одномодовый кабель

Для спецификации 1000Base-LX в качестве источника излучения всегда применяется полупроводниковый лазер с длиной волны 1300 нм.

Основная область применения стандарта 1000Base-LX – это одномодовое оптоволокно. Максимальная длина кабеля для одномодового волокна равна 5000 м.

Спецификация 1000Base-LX может работать и на многомодовом кабеле. В этом случае предельное расстояние получается небольшим - 550 м. Это связано с особенностями распространения когерентного света в широком канале многомодового кабеля. Для присоединения лазерного трансивера к многомодовому кабелю необходимо использовать специальный адаптер.

Твинаксиальный кабель

В качестве среды передачи данных используется высококачественный твинаксиальный кабель (Twinax) с волновым сопротивлением 150 Ом (2x75 Ом). Данные посылаются одновременно по паре проводников, каждый из которых окружен экранирующей оплеткой. При этом получается режим полудуплексной передачи. Для обеспечения полнодуплексной передачи необходимы еще две пары коаксиальных проводников. Начал выпускаться специальный кабель, который содержит четыре. коаксиальных проводника – так называемый Quad-кабель. Он внешне напоминает кабель категории 5 и имеет близкий к нему внешний диаметр и гибкость. Максимальная длина твинаксиального сегмента составляет всего 25 метров, поэтому это решение подходит для оборудования, расположенного в одной комнате.

3.7.4. Gigabit Ethernet на витой паре категории 5

Как известно, каждая пара кабеля категории 5 имеет гарантированную полосу пропускания до 100 МГц. Для передачи по такому кабелю данных со скоростью 1000 Мбит/с было решено организовать параллельную передачу одновременно по всем 4 парам кабеля (так же, как и в технологии l00VG-AnyLAN).

Это сразу уменьшило скорость передачи данных по каждой паре до 250 Мбит/с. Однако и для такой скорости необходимо было придумать метод кодирования, который имел бы спектр не выше 100 МГц. Кроме того, одновременное использование четырех пар на первый взгляд лишает сеть возможность распознавать коллизии.

На оба эти вопроса комитет 802.3ab нашел ответы.

Для кодирования данных был применен код PAM5, использующий 5 уровней потенциала: -2, -1,0, +1, +2. Поэтому за один такт по одной паре передается 2,322 бит информации. Следовательно, тактовую частоту вместо 250 МГц можно снизить до 125 МГц. При этом если использовать не все коды, а передавать 8 бит за такт (по 4 парам), то выдерживается требуемая скорость передачи в 1000 Мбит/с и еще остается запас неиспользуемых кодов, так как код PAM5 содержит $5^4 = 625$ комбинаций, а если передавать за один такт по всем четырем парам 8 бит данных, то для этого требуется всего $2^8 = 256$ комбинаций. Оставшиеся комбинации приемник может использовать для контроля принимаемой информации и выделения правильных комбинаций на фоне шума. Код PAM5 на тактовой частоте 125

МГц укладывается в полосу 100 МГц кабеля категории 5.

Для распознавания коллизий и организации полнодуплексного режима разработчики спецификации 802.3аЬ применили технику, используемую при организации дуплексного режима на одной паре проводов в современных модемах и аппаратуре передачи данных абонентских окончаний ISDN. Вместо передачи по разным парам проводов или разнесения сигналов двух одновременно работающих навстречу передатчиков по диапазону частот оба передатчика работают навстречу друг другу по каждой из 4-х пар в одном и том же диапазоне частот, так как используют один и тот же потенциальный код PAM5 (рис. 3.26). Схема гибридной развязки *H* позволяет приемнику и передатчику одного и того же узла использовать одновременно витую пару и для приема и для передачи (так же, как и в трансиверах коаксиального Ethernet).

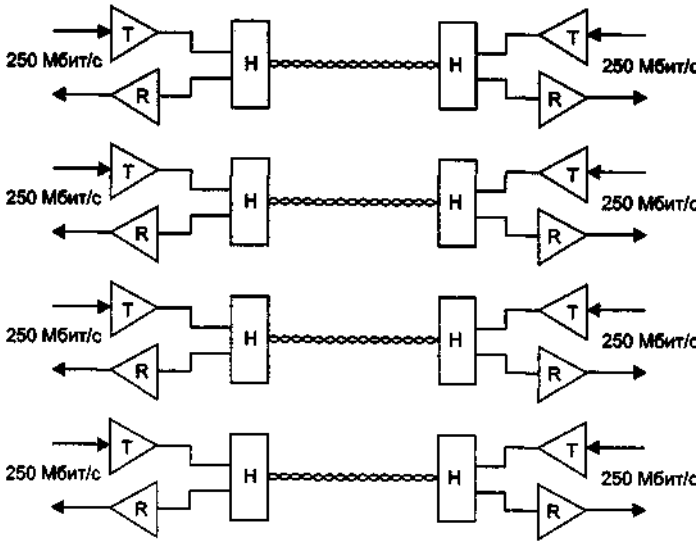


Рис. 3.26. Двунаправленная передача по четырем парам UTP категории 5

Для отделения принимаемого сигнала от своего собственного приемник вычитает из результирующего сигнала известный ему свой сигнал. Естественно, что это не простая операция и для ее выполнения используются специальные цифровые сигнальные процессоры - DSP (Digital Signal Processor). Такая техника уже прошла проверку практикой, но в модемах и сетях ISDN она применялась совсем на других скоростях.

При полудуплексном режиме работы получение встречного потока данных считается коллизией, а для полнодуплексного режима работы – нормальной ситуацией.

Ввиду того что работы по стандартизации спецификации Gigabit Ethernet на неэкранированной витой паре категории 5 подходят к концу, многие производители и потребители надеются на положительный исход этой работы, так как в этом случае для поддержки технологии Gigabit Ethernet не нужно будет заменять уже установленную проводку категории 5 на оптоволокно или проводку категории 7.

Выводы

- Технология Gigabit Ethernet добавляет новую, 1000 Мбит/с, ступень в иерархии скоростей семейства Ethernet. Эта ступень позволяет эффективно строить крупные локальные сети, в которых мощные серверы и магистрали нижних уровней сети работают на скорости 100 Мбит/с, а магистраль Gigabit Ethernet объединяет их, обеспечивая достаточно большой запас пропускной способности.
- Разработчики технологии Gigabit Ethernet сохранили большую степень преемственности с технологиями Ethernet и Fast Ethernet. Gigabit Ethernet использует те же форматы кадров, что и предыдущие версии Ethernet; работает в полнодуплексном и полудуплексном режимах, поддерживая на разделяемой среде тот же метод доступа CSMA/CD с минимальными изменениями.
- Для обеспечения приемлемого максимального диаметра сети в 200 м в полудуплексном режиме разработчики технологии пошли на увеличение минимального размера кадра с 64 до 512 байт. Разрешается также передавать несколько кадров подряд, не освобождая среду, на

интервале 8096 байт, тогда кадры не обязательно дополнять до 512 байт. Остальные параметры метода доступа и максимального размера кадра остались неизменными.

- Летом 1998 года был принят стандарт 802.3z, который определяет использование в качестве физической среды трех типов кабеля: многомодового оптоволоконного (расстояние до 500 м), одномодового оптоволоконного (расстояние до 5000 м) и двойного коаксиального (twinax), по которому данные передаются одновременно по двум медным экранированным проводникам на расстояние до 25 м.
- Для разработки варианта Gigabit Ethernet на UTP категории 5 была создана специальная группа 802.3aЬ, которая уже разработала проект стандарта для работы по 4-м парам UTP категории 5. Принятие этого стандарта ожидается в ближайшее время.

Вопросы и упражнения

1. Поясните разницу между расширяемостью и масштабируемостью на примере технологии Ethernet.
2. Что такое коллизия:
   - (A) ситуация, когда станция, желающая передать пакет, обнаруживает, что в данный момент другая станция уже заняла передающую среду;
   - (B) ситуация, когда две рабочие станции одновременно передают данные в разделяемую передающую среду.
3. Что такое домен коллизий? Являются ли доменами коллизий фрагменты сети, показанные на рис. 3.27?
4. В чем состоят функции преамбулы и начального ограничителя кадра в стандарте Ethernet?
5. Какие сетевые средства осуществляют jabber control?
6. Чему равны значения следующих характеристик стандарта 10Base-5:
   - номинальная пропускная способность (бит/с);
   - эффективная пропускная способность (бит/с);
   - пропускная способность (кадр/с);
   - внутрипакетная скорость передачи (бит/с);
   - межбитовый интервал (с).
7. Чем объясняется, что минимальный размер кадра в стандарте 10Base-5 был выбран равным 64 байт?

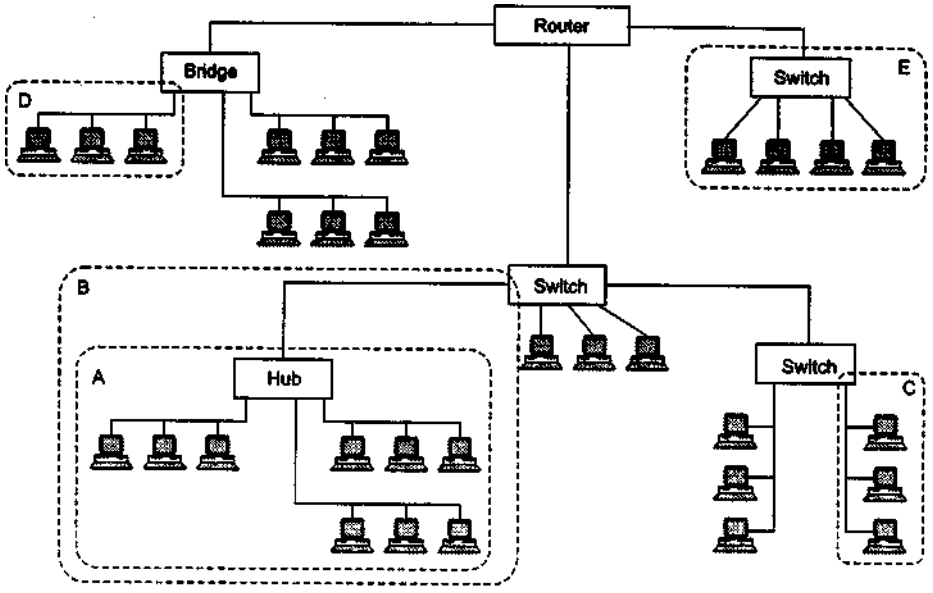


Рис. 3.27. Домены коллизий

8. Поясните смысл каждого поля кадра Ethernet.
9. Как известно, имеются 4 стандарта на формат кадров Ethernet. Выберите из ниже приведенного

списка названия для каждого из этих стандартов. Учтите, что некоторые стандарты имеют несколько названий:

- Novell 802.2;
- Ethernet II;
- 802.3/802.2
- Novell 802.3;
- Raw 802.3;
- Ethernet DIX;
- 802.3/LLC;
- Ethernet SNAP.

10. Что может произойти в сети, в которой передаются кадры Ethernet разных форматов?
11. При каких типах ошибок в сети Ethernet концентратор обычно отключает порт?
12. Как величина MTU влияет на работу сети? Какие проблемы несут слишком длинные кадры? В чем состоит неэффективность коротких кадров?
13. Как коэффициент использования влияет на производительность сети Ethernet?
1.4. Если один вариант технологии Ethernet имеет более высокую скорость передачи данных, чем другой (например, Fast Ethernet и Ethernet), то какая из них поддерживает большую максимальную длину сети?
15. Из каких соображений выбрана максимальная длина физического сегмента в стандартах Ethernet?
16. Проверьте корректность конфигурации сети Fast Ethernet, приведенной на рис. 3.28.

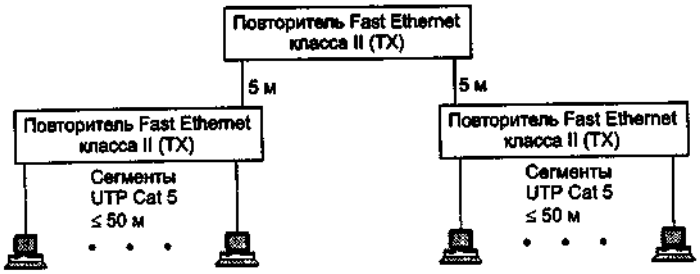


Рис. 3.28. Пример конфигурации сети

17. Укажите максимально допустимые значения MTU для:
    - Ethernet;
    - Token Ring;
    - FDDI;
    - ATM.
18. Опишите алгоритм доступа к среде технологии Token Ring.
19. Из каких соображений выбирается максимальное время оборота маркера по кольцу?
20. Если бы вам пришлось выбирать, какую из технологий – Ethernet или Token Ring – использовать в сети вашего предприятия, какое решение вы бы приняли? Какие соображения привели бы в качестве обоснования этого решения?
21. В чем состоит сходство и различие технологий FDDI и Token Ring?
22. Какие элементы сети FDDI обеспечивают отказоустойчивость?
23. Технология FDDI является отказоустойчивой. Означает ли это, что при любом однократном обрыве кабеля сеть FDDI будет продолжать нормально работать?
24. К каким последствиям может привести двукратный обрыв кабеля в кольце FDDI?
25. Что общего в работе концентратора lOOVG-AnyLAN и обычного моста?
26. Какие из ниже перечисленных пар сетевых технологий совместимы по форматам кадров и, следовательно, позволяют образовывать составную сеть без необходимости транслирования кадров:
    - (A) FDDI - Ethernet;
    - (B) Token Ring - Fast Ethernet;
    - (C) Token Ring – l00VG-AnyLAN;

- (D) Ethernet - Fast Ethernet;
- (E) Ethernet – l00VG-AnyLAN;
- (F) Token Ring - FDDI.

27. Из-за увеличения пропускной способности минимальный размер кадра в Gigabit Ethernet пришлось увеличить до 512 байт. В тех случаях, когда передаваемые данные не могут полностью заполнить поле данных кадра, оно дополняется до необходимой длины неким «заполнителем», который не несет полезной информации. Что предпринято в Gigabit Ethernet для сокращения накладных расходов, возникающих при передаче коротких данных?
28. С чем связано ограничение, известное как «правило 4-х хабов»?

ГЛАВА 4. ПОСТРОЕНИЕ ЛОКАЛЬНЫХ СЕТЕЙ ПО СТАНДАРТАМ ФИЗИЧЕСКОГО И КАНАЛЬНОГО УРОВНЕЙ

В данной главе рассматриваются вопросы, связанные с реализацией рассмотренных выше протоколов физического и канального уровней в сетевом коммуникационном оборудовании. Хотя на основе оборудования только этого уровня трудно построить достаточно крупную корпоративную сеть, именно кабельные системы, сетевые адаптеры, концентраторы, мосты и коммутаторы представляют наиболее массовый тип сетевых устройств.

За исключением кабельной системы, которая является протокольно независимой, устройство и функции коммуникационного оборудования остальных типов существенно зависят от того, какой конкретно протокол в них реализован. Концентратор Ethernet устроен не так, как концентратор Token Ring, а сетевой адаптер FDDI не сможет работать в сети Fast Ethernet. С другой стороны, даже в рамках одной технологии оборудование разных производителей может заметно отличаться друг от друга. В этой главе будут рассмотрены наиболее типичные варианты реализации основных и дополнительных устройств физического и канального уровней.

4.1. Структурированная кабельная система

Кабельная система является фундаментом любой сети. Как при строительстве нельзя создать хороший дом на плохо построенном фундаменте, так и сеть, отлично работающая на плохой кабельной системе, – это явление из области ненаучной фантастики. Если в кабелях ежедневно происходят короткие замыкания, контакты разъемов то отходят, то снова входят в плотное соединение, добавление новой станции приводит к необходимости тестирования десятка контактов разъемов из-за того, что документация на физические соединения не ведется, то ясно, что на основе такой кабельной системы любое, самое современное и производительное оборудование будет работать из рук вон плохо. Пользователи будут недовольны большими периодами простоев и низкой производительностью сети, а обслуживающий персонал будет в постоянной «запарке», разыскивая места коротких замыканий, обрывов и плохих контактов. Причем проблем с кабельной системой становится намного больше при увеличении размеров сети.

Ответом на высокие требования к качеству кабельной системы стали структурированные кабельные системы.

4.1.1. Иерархия в кабельной системе

Структурированная кабельная система (Structured Cabling System, SCS) – это набор коммутационных элементов (кабелей, разъемов, коннекторов, кроссовых панелей и шкафов), а также методика их совместного использования, которая позволяет создавать регулярные, легко расширяемые структуры связей в вычислительных сетях.

Структурированная кабельная система представляет своего рода «конструктор», с помощью 'которого проектировщик сети строит нужную ему конфигурацию из стандартных кабелей, соединенных стандартными разъемами и коммутируемых на стандартных кроссовых панелях. При необходимости конфигурацию связей можно легко изменить – добавить компьютер, сегмент, коммутатор, изъять ненужное оборудование, а также поменять соединения между компьютерами и концентраторами.

При построении структурированной кабельной системы подразумевается, что каждое рабочее место на предприятии должно быть оснащено розетками для подключения телефона и компьютера, даже если в данный момент этого не требуется. То есть хорошая структурированная кабельная система строится избыточной. В будущем это может сэкономить средства, так как изменения в подключении новых устройств можно производить за счет перекоммутации уже проложенных кабелей.

Структурированная кабельная система планируется и строится иерархически, с главной магистралью и многочисленными ответвлениями от нее (рис. 4.1).

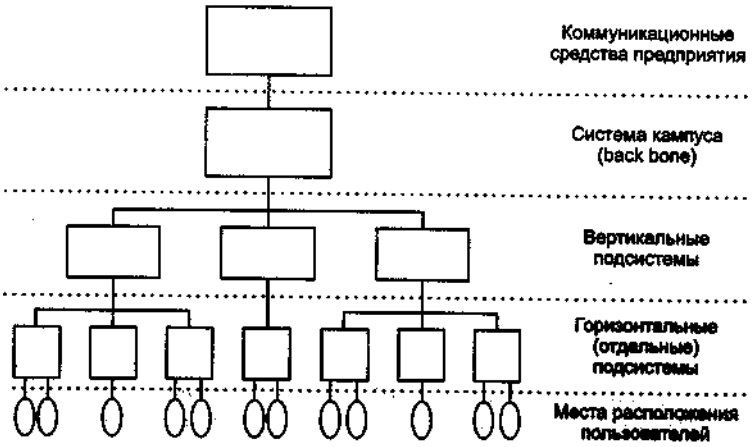


Рис. 4.1. Иерархия структурированной кабельной системы

Эта система может быть построена на базе уже существующих современных телефонных кабельных систем, в которых кабели, представляющие собой набор витых пар, прокладываются в каждом здании, разводятся между этажами, на каждом этаже используется специальный кроссовый шкаф, от которого провода в трубах и коробах подводятся к каждой комнате и разводятся по розеткам. К сожалению, в нашей стране далеко не во всех зданиях телефонные линии прокладываются витыми парами, поэтому они непригодны для создания компьютерных сетей, и кабельную систему в таком случае нужно строить заново.

Типичная иерархическая структура структурированной кабельной системы (рис. 4.2) включает:

- горизонтальные подсистемы (в пределах этажа);
- вертикальные подсистемы (внутри здания);
- подсистему кампуса (в пределах одной территории с несколькими зданиями).

Горизонтальная подсистема соединяет кроссовый шкаф этажа с розетками пользователей. Подсистемы этого типа соответствуют этажам здания. *Вертикальная подсистема* соединяет кроссовые шкафы каждого этажа с центральной аппаратной здания. Следующим шагом иерархии является *подсистема кампуса*, которая соединяет несколько зданий с главной аппаратной всего кампуса. Эта часть кабельной системы обычно называется магистралью (backbone).

Использование структурированной кабельной системы вместо хаотически проложенных кабелей дает предприятию много преимуществ.

- *Универсальность*. Структурированная кабельная система при продуманной организации может стать единой средой для передачи компьютерных данных в локальной вычислительной сети, организации локальной телефонной сети, передачи видеоинформации и даже передачи сигналов от датчиков пожарной безопасности или охранных систем. Это позволяет автоматизировать многие процессы контроля, мониторинга и управления хозяйственными службами и системами жизнеобеспечения предприятия.
- *Увеличение срока службы*. Срок морального старения хорошо структурированной кабельной системы может составлять 10-15 лет.

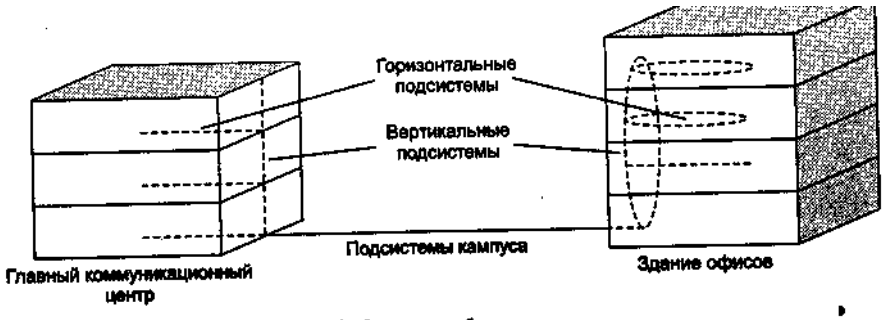


Рис. 4.2. Структура кабельных подсистем

- *Уменьшение стоимости добавления новых пользователей и изменения их мест размещения*. Известно, что стоимость кабельной системы значительна и определяется в основном не стоимостью кабеля, а стоимостью работ по его прокладке. Поэтому более выгодно провести однократную работу по прокладке кабеля, возможно, с большим запасом по длине, чем несколько раз выполнять прокладку, наращивая длину кабеля. При таком подходе все работы по добавлению или перемещению пользователя сводятся к подключению компьютера к уже имеющейся розетке.
- *Возможность легкого расширения сети*. Структурированная кабельная система является модульной, поэтому ее легко расширять. Например, к магистрали можно добавить новую подсеть, не оказывая никакого влияния на существующие подсети. Можно заменить в отдельной подсети тип кабеля независимо от остальной части сети. Структурированная кабельная система является основой для деления сети на легко управляемые логические сегменты, так как она сама уже разделена на физические сегменты.
- *Обеспечение более эффективного обслуживания*. Структурированная кабельная система облегчает обслуживание и поиск неисправностей по сравнению с шинной кабельной системой. При шинной организации кабельной системы отказ одного из устройств или соединительных элементов приводит к трудно локализуемому отказу всей сети. В структурированных кабельных системах отказ одного сегмента не действует на другие, так как объединение сегментов осуществляется с помощью концентраторов. Концентраторы диагностируют и локализуют неисправный участок.
- *Надежность*. Структурированная кабельная система имеет повышенную надежность, поскольку производитель такой системы гарантирует не только качество ее отдельных компонентов, но и их совместимость.

Первой структурированной кабельной системой, имеющий все современные черты такого типа систем, была система SYSTIMAX SCS компании Lucent Technologies (ранее – подразделение AT&T). И сегодня компании Lucent Technologies принадлежит основная доля мирового рынка. Многие другие компании также выпускают качественные структурированные кабельные системы, например AMP, BICC Brand-Rex, Siemens, Alcatel, MOD-TAP. На российском рынке успешно завоевывает себе место под солнцем отечественная структурированная кабельная система АйТи-СКС московской компании «АйТи».

4.1.2. Выбор типа кабеля для горизонтальных подсистем

Большинство проектировщиков начинает разработку структурированной кабельной системы с горизонтальных подсистем, так как именно к ним подключаются конечные пользователи. При этом они могут выбирать между экранированной витой парой, неэкранированной витой парой, коаксиальным кабелем и волоконно-оптическим кабелем. Возможно использование и беспроводных линий связи.

Горизонтальная подсистема характеризуется очень большим количеством ответвлений кабеля (рис. 4.3), так как его нужно провести к каждой пользовательской розетке, причем и в тех комнатах, где пока компьютеры в сеть не объединяются. Поэтому к кабелю, используемому в горизонтальной проводке, предъявляются повышенные требования к удобству выполнения ответвлений, а также удобству его прикладки в помещениях. На этаже обычно устанавливается кроссовая панель, которая позволяет с помощью коротких отрезков кабеля, оснащенного разъемами, провести перекоммутацию соединений между пользовательским оборудованием и концентраторами/коммутаторами.

Медный провод, в частности неэкранированная витая пара, является предпочтительной средой для горизонтальной кабельной подсистемы, хотя, если пользователям нужна очень высокая пропускная способность или кабельная система прокладывается в агрессивной среде, для нее подойдет и волоконно-оптический кабель. Коаксиальный кабель – это устаревшая технология, которой следует избегать, если только она уже широко не используется на предприятии. Беспроводная связь является новой и многообещающей технологией, однако из-за сравнительной новизны и низкой помехоустойчивости лучше ограничить масштабы ее использования неответственными областями.

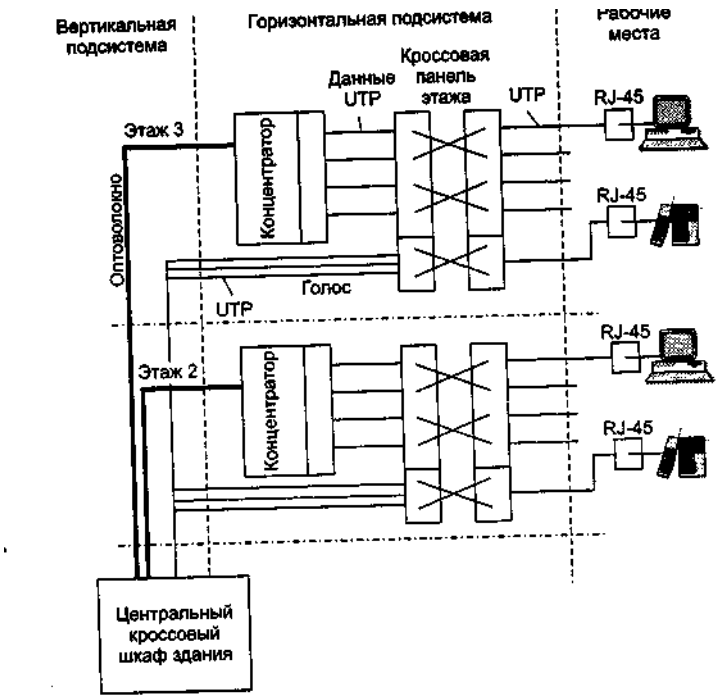


Рис. 4.3. Структура кабельной системы этажа и здания

При выборе кабеля принимаются во внимание следующие характеристики: полоса пропускания, расстояние, физическая защищенность, электромагнитная помехозащищенность, стоимость. Кроме того, при выборе кабеля нужно учитывать, какая кабельная система уже установлена на предприятии, а также какие тенденции и перспективы существуют на рынке в данный момент.

Экранированная витая пара, STP, позволяет передавать данные на большее расстояние и поддерживать больше узлов, чем неэкранированная. Наличие экрана делает ее более дорогой и не дает возможности передавать голос. Экранированная витая пара используется в основном в сетях, базирующихся на продуктах IBM и Token Ring, и редко подходит к остальному оборудованию локальных сетей.

Неэкранированная витая пара UTP по характеристикам полосы пропускания и поддерживаемым расстояниям также подходит для создания горизонтальных подсистем. Но так как она может передавать данные и голос, она используется чаще.

Однако и коаксиальный кабель все еще остается одним из возможных вариантов кабеля для горизонтальных подсистем. Особенно в случаях, когда высокий уровень электромагнитных помех не позволяет использовать витую пару или же небольшие размеры сети не создают больших проблем с эксплуатацией кабельной системы.

Толстый Ethernet обладает по сравнению с тонким большей полосой пропускания, он более стоек к повреждениям и передает данные на большие расстояния, однако к нему сложнее подсоединиться и он менее гибок. С толстым Ethernet сложнее работать, и он мало подходит для горизонтальных подсистем. Однако его можно использовать в вертикальной подсистеме в качестве магистрали, если оптоволоконный кабель по каким-то причинам не подходит.

Тонкий Ethernet – это кабель, который должен был решить проблемы, связанные с применением толстого Ethernet. До появления стандарта 10Base-T тонкий Ethernet был основным кабелем для горизонтальных подсистем. Тонкий Ethernet проще монтировать, чем толстый. Сети на тонком Ethernet можно быстро собрать, так как компьютеры соединяются друг с другом непосредственно.

Главный недостаток тонкого Ethernet – сложность его обслуживания. Каждый конец кабеля должен

завершаться терминатором 50 Ом. При отсутствии терминатора или утере им своих рабочих свойств (например, из-за отсутствия контакта) перестает работать весь сегмент сети, подключенный к этому кабелю. Аналогичные последствия имеет плохое соединение любой рабочей станции (осуществляемое через Т-коннектор). Неисправности в сетях на тонком Ethernet сложно локализовать. Часто приходится отсоединять Т-коннектор от сетевого адаптера, тестировать кабельный сегмент и затем последовательно повторять эту процедуру для всех присоединенных узлов. Поэтому стоимость эксплуатации сети на тонком Ethernet обычно значительно превосходит стоимость эксплуатации аналогичной сети на витой паре, хотя капитальные затраты на кабельную систему для тонкого Ethernet обычно ниже.

Основные области применения оптоволоконного кабеля – вертикальная подсистема и подсистемы кампусов. Однако, если нужна высокая степень защищенности данных, высокая пропускная способность или устойчивость к электромагнитным помехам, волоконно-оптический кабель может использоваться и в горизонтальных подсистемах. С волоконно-оптическим кабелем работают протоколы AppleTalk, ArcNet, Ethernet, FDDI и Token Ring, а также новые протоколы lOOVG-AnyLAN, Fast Ethernet, ATM.

Стоимость установки сетей на оптоволоконном кабеле для горизонтальной подсистемы оказывается весьма высокой. Эта стоимость складывается из стоимости сетевых адаптеров (около тысячи долларов каждый) и стоимости монтажных работ, которая в случае оптоволокна гораздо выше, чем при работе с другими видами кабеля.

Преобладающим кабелем для горизонтальной подсистемы является неэкранированная витая пара категории 5. Ее позиции еще более укрепятся с принятием спецификации 802.3ab для применения на этом виде кабеля технологии Gigabit Ethernet.

На рис. 4.4 показаны типовые коммутационные элементы структурированной кабельной системы, применяемые на этаже при прокладке неэкранированной витой пары. Для сокращения количества кабелей здесь установлен 25-парный кабель и разъем для такого типа кабеля Telco, имеющий 50 контактов.

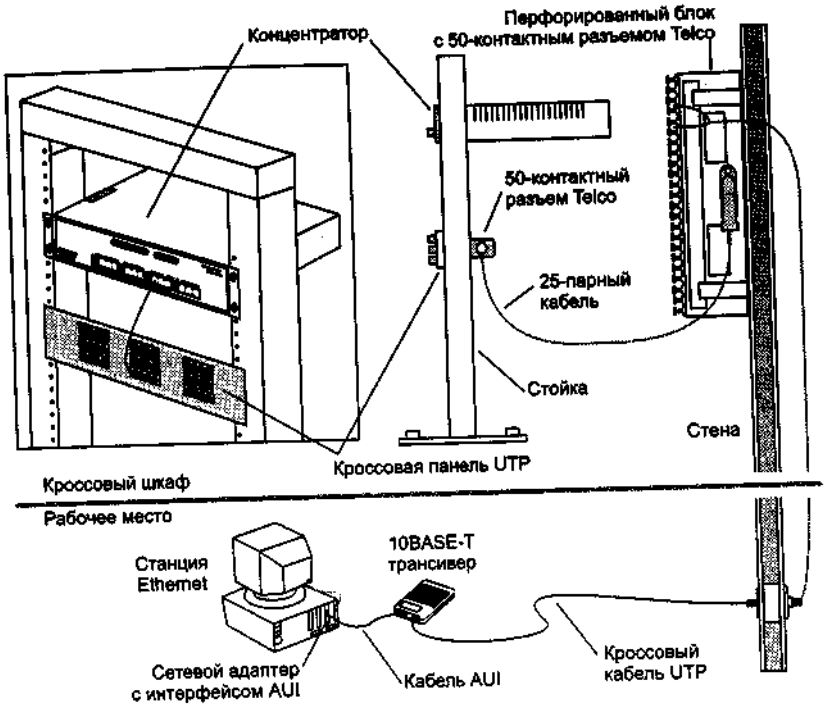


Рис. 4.4. Коммутоционные элементы горизонтальной кабельной подсистемы для UTP

4,1.3. Выбор типа кабеля для вертикальных подсистем

Кабель вертикальной (или магистральной) подсистемы, которая соединяет этажи здания, должен передавать данные на большие расстояния и с большей скоростью по сравнению с кабелем горизонтальной подсистемы. В прошлом основным видом кабеля для вертикальных подсистем был коаксиал. Теперь для этой цели все чаще используется оптоволоконный кабель.

Для вертикальной подсистемы выбор кабеля в настоящее время ограничивается тремя вариантами.

- Оптоволокно – отличные характеристики пропускной способности, расстояния и защиты данных; устойчивость к электромагнитным помехам; может передавать голос, видеоизображение и данные. Но сравнительно дорого, сложно выполнять ответвления.
- Толстый коаксиал – хорошие характеристики пропускной способности, расстояния и зашиты данных; может передавать данные. Но с ним сложно работать, хотя специалистов, имеющих подобный опыт работы, достаточно много.
- Широкополосный кабель, используемый в кабельном телевидении, – хорошие показатели пропускной способности и расстояния; может передавать голос, видео и данные. Но очень сложно работать и требуются большие затраты во время эксплуатации.

Применение волоконно-оптического кабеля в вертикальной подсистеме имеет ряд преимуществ. Он передает данные на значительно большие расстояния без необходимости регенерации сигнала. Он имеет сердечник меньшего диаметра, поэтому может быть проложен в более узких местах. Так как передаваемые по нему сигналы являются световыми, а неэлектрическими, оптоволоконный кабель не чувствителен к электромагнитным и радиочастотным помехам, в отличие от медного коаксиального кабеля. Это делает оптоволоконный кабель идеальной средой передачи данных для промышленных сетей. Оптоволоконному кабелю не страшна молния, поэтому он хорош для внешней прокладки. Он обеспечивает более высокую степень защиты от несанкционированного доступа, так как ответвление гораздо легче обнаружить, чем в случае медного кабеля (при ответвлении резко уменьшается интенсивность света).

Оптоволоконный кабель имеет и недостатки. Он дороже чем медный кабель, дороже обходится и его прокладка. Оптоволоконный кабель менее прочный, чем коаксиальный. Инструменты, применяемые при прокладке и тестировании оптоволоконного кабеля, имеют высокую стоимость и сложны в работе. Присоединение коннекторов к оптоволоконному кабелю требует большого искусства и времени, а следовательно, и денег.

Для уменьшения стоимости построения межэтажной магистрали на оптоволокне некоторые компании, например AMP, предлагают кабельную систему с одним коммутационным центром. Обычно, коммутационный центр есть на каждом этаже, а в здании имеется общий коммутационный центр (см. рис. 4.3.), соединяющий между собой коммутационные центры этажей. При такой традиционной схеме и использовании волоконно-оптического кабеля между этажами требуется выполнять достаточное большое число оптоволоконных соединений в коммутационных центрах этажей. Если же коммутационный центр в здании один, то все оптические кабели расходятся из единого кроссового шкафа прямо к разъемам конечного оборудования – коммутаторов, концентраторов или сетевых адаптеров с оптоволоконными трансиверами.

Толстый коаксиальный кабель также допустим в качестве магистрали сети, однако для новых кабельных систем более рационально использовать оптоволоконный кабель, так как он имеет больший срок службы и сможет в будущем поддерживать высокоскоростные и мультимедийные приложения. Но для уже существующих систем толстый коаксиальный кабель служил магистралью системы многие годы, и с этим нужно считаться. Причинами его повсеместного применения были широкая полоса пропускания, хорошая защищенность от электромагнитных помех и низкое радиоизлучение.

Хотя толстый коаксиальный кабель и дешевле, чем оптоволокно, но с ним гораздо сложнее работать. Он особенно чувствителен к различным уровням напряжения заземления, что часто бывает при переходе от одного этажа к другому. Эту проблему сложно разрешить. Поэтому кабелем номер 1 для горизонтальной подсистемы сегодня является волоконно-оптический кабель.

4.1.4. Выбор типа кабеля для подсистемы кампуса

Как и для вертикальных подсистем, оптоволоконный кабель является наилучшим выбором для подсистем нескольких зданий, расположенных в радиусе нескольких километров. Для этих подсистем также подходит толстый коаксиальный кабель.

При выборе кабеля для кампуса нужно учитывать воздействие среды на кабель вне помещения. Для предотвращения поражения молнией лучше выбрать для внешней проводки неметаллический оптоволоконный кабель. По многим причинам внешний кабель производится в полиэтиленовой защитной оболочке высокой плотности. При подземной прокладке кабель должен иметь специальную влагозащитную оболочку (от дождя и подземной влаги), а также металлический защитный слой от грызунов и вандалов. Влагозащитный кабель имеет прослойку из инертного газа между диэлектриком,

экраном и внешней оболочкой.

Кабель для внешней прокладки не подходит для прокладки внутри зданий, так как он выделяет при сгорании большое количество дыма.

Выводы

- Кабельная система составляет фундамент любой компьютерной сети. От ее качества зависят все основные свойства сети.
- Структурированная кабельная система представляет собой набор коммуникационных элементов – кабелей, разъемов, коннекторов, кроссовых панелей и шкафов, которые удовлетворяют стандартам и позволяют создавать регулярные, легко расширяемые структуры связей.
- Структурированная кабельная система состоит из трех подсистем: горизонтальной (в пределах этажа), вертикальной (между этажами) и подсистемы кампуса (в пределах одной территории с несколькими зданиями).
- Для горизонтальной подсистемы характерно наличие большого количества ответвлений и перекрестных связей. Наиболее подходящий тип кабеля – неэкранированная витая пара категории 5.
- Вертикальная подсистема состоит из более протяженных отрезков кабеля, количество ответвлений намного меньше, чем в горизонтальной подсистеме. Предпочтительный тип кабеля – волоконно-оптический.
- Для подсистемы кампуса характерна нерегулярная структура связей с центральным зданием. Предпочтительный тип кабеля – волоконно-оптический в специальной изоляции.
- Кабельная система здания строится избыточной, так как стоимость последующего расширения кабельной системы превосходит стоимость установки избыточных элементов.

4.2. Концентраторы и сетевые адаптеры

Концентраторы вместе с сетевыми адаптерами, а также кабельной системой представляют тот минимум оборудования, с помощью которого можно создать локальную сеть. Такая сеть будет представлять собой общую разделяемую среду. Понятно, что сеть не может быть слишком большой, так как при большом количестве узлов общая среда передачи данных быстро становится узким местом, снижающим производительность сети. Поэтому концентраторы и сетевые адаптеры позволяют строить небольшие базовые фрагменты сетей, которые затем должны объединяться друг с другом с помощью мостов, коммутаторов и маршрутизаторов.

4.2.1. Сетевые адаптеры

Функции и характеристики сетевых адаптеров

Сетевой адаптер (*Network Interface Card, NIC*) вместе со своим драйвером реализует второй, канальный уровень модели открытых систем в конечном узле сети – компьютере. Более точно, в сетевой операционной системе пара адаптер и драйвер выполняет только функции физического и MAC-уровней, в то время как LLC-уровень обычно реализуется модулем операционной системы, единым для всех драйверов и сетевых адаптеров. Собственно так оно и должно быть в соответствии с моделью стека протоколов IEEE 802. Например, в ОС Windows NT уровень LLC реализуется в модуле NDIS, общем для всех драйверов сетевых адаптеров, независимо от того, какую технологию поддерживает драйвер.

Сетевой адаптер совместно с драйвером выполняют две операции: передачу и прием кадра.

Передача кадра из компьютера в кабель состоит из перечисленных ниже этапов (некоторые могут отсутствовать, в зависимости от принятых методов кодирования).

- Прием кадра данных LLC через межуровневый интерфейс вместе с адресной информацией MAC-уровня. Обычно взаимодействие между протоколами внутри компьютера происходит через буферы, расположенные в оперативной памяти. Данные для передачи в сеть помещаются в эти буферы протоколами верхних уровней, которые извлекают их из дисковой

памяти либо из файлового кэша с помощью подсистемы ввода/вывода операционной системы.

- Оформление кадра данных MAC-уровня, в который инкапсулируется кадр LLC (с отброшенными флагами 01111110). Заполнение адресов назначения и источника, вычисление контрольной суммы.
- Формирование символов кодов при использовании избыточных кодов типа 4B/5B. Скрэмблирование кодов для получения более равномерного спектра сигналов. Этот этап используется не во всех протоколах – например, технология Ethernet 10 Мбит/с обходится без него.
- Выдача сигналов в кабель в соответствии с принятым линейным кодом – манчестерским, NRZI, MLT-3 и т. п.
- Прием кадра из кабеля в компьютер включает следующие действия.
- Прием из кабеля сигналов, кодирующих битовый поток.
- Выделение сигналов на фоне шума. Эту операцию могут выполнять различные специализированные микросхемы или сигнальные процессоры DSP. В результате в приемнике адаптера образуется некоторая битовая последовательность, с большой степенью вероятности совпадающая с той, которая была послана передатчиком.
- Если данные перед отправкой в кабель подвергались скрэмблированию, то они пропускаются через дескрэмблер, после чего в адаптере восстанавливаются символы кода, посланные передатчиком.
- Проверка контрольной суммы кадра. Если она неверна, то кадр отбрасывается, а через межуровневый интерфейс наверх, протоколу LLC передается соответствующий код ошибки. Если контрольная сумма верна, то из MAC-кадра извлекается кадр LLC и передается через межуровневый интерфейс наверх, протоколу LLC. Кадр LLC помещается в буфер оперативной памяти.

Распределение обязанностей между сетевым адаптером и его драйвером стандартами не определяется, поэтому каждый производитель решает этот вопрос самостоятельно. Обычно сетевые адаптеры делятся на адаптеры для клиентских компьютеров и адаптеры для серверов.

В адаптерах для клиентских компьютеров значительная часть работы перекладывается на драйвер, тем самым адаптер оказывается проще и дешевле. Недостатком такого подхода является высокая степень загрузки центрального процессора компьютера рутинными работами по передаче кадров из оперативной памяти компьютера в сеть. Центральный процессор вынужден заниматься этой работой вместо выполнения прикладных задач пользователя.

Поэтому адаптеры, предназначенные для серверов, обычно снабжаются собственными процессорами, которые самостоятельно выполняют большую часть работы по передаче кадров из оперативной памяти в сеть и в обратном направлении. Примером такого адаптера может служить сетевой адаптер SMS EtherPower со встроенным процессором Intel i960.

В зависимости от того, какой протокол реализует адаптер, адаптеры делятся на Ethernet-адаптеры, Token Ring-адаптеры, FDDI-адаптеры и т. д. Так как протокол Fast Ethernet позволяет за счет процедуры автопереговоров автоматически выбрать скорость работы сетевого адаптера в зависимости от возможностей концентратора, то многие адаптеры Ethernet сегодня поддерживают две скорости работы и имеют в своем названии приставку 10/100. Это свойство некоторые производители называют *авточувствительностью*.

Сетевой адаптер перед установкой в компьютер необходимо конфигурировать. При конфигурировании адаптера обычно задаются номер прерывания IRQ, используемого адаптером, номер канала прямого доступа к памяти DMA (если адаптер поддерживает режим DMA) и базовый адрес портов ввода/вывода.

Если сетевой адаптер, аппаратура компьютера и операционная система поддерживают стандарт Plug-and-Play, то конфигурирование адаптера и его драйвера осуществляется автоматически. В противном случае нужно сначала сконфигурировать сетевой адаптер, а затем повторить параметры его конфигурации для драйвера. В общем случае, детали процедуры конфигурирования сетевого адаптера и его драйвера во многом зависят от производителя адаптера, а также от возможностей шины, для которой разработан адаптер.

Классификация сетевых адаптеров

В качестве примера классификации адаптеров используем подход фирмы 3Com, имеющей репутацию лидера в области адаптеров Ethernet. Фирма 3Com считает, что сетевые адаптеры Ethernet прошли в своем развитии три поколения.

Адаптеры первого поколения были выполнены на дискретных логических микросхемах, в результате чего обладали низкой надежностью. Они имели буферную память только на один кадр, что приводило к низкой производительности адаптера, так как все кадры передавались из компьютера в сеть или из сети в компьютер последовательно. Кроме этого, задание конфигурации адаптера первого поколения происходило вручную, с помощью перемычек. Для каждого типа адаптеров использовался свой драйвер, причем интерфейс между драйвером и сетевой операционной системой не был стандартизирован.

В сетевых адаптерах второго поколения для повышения производительности стали применять метод многокадровой буферизации. При этом следующий кадр загружается из памяти компьютера в буфер адаптера одновременно с передачей предыдущего кадра в сеть. В режиме приема, после того как адаптер полностью принял один кадр, он может начать передавать этот кадр из буфера в память компьютера одновременно с приемом другого кадра из сети.

В сетевых адаптерах второго поколения широко используются микросхемы с высокой степенью интеграции, что повышает надежность адаптеров. Кроме того, драйверы этих адаптеров основаны на стандартных спецификациях. Адаптеры второго поколения обычно поставляются с драйверами, работающими как в стандарте NDIS (спецификация интерфейса сетевого драйвера), разработанном фирмами 3Com и Microsoft и одобренном IBM, так и в стандарте ODI (интерфейс открытого драйвера), разработанном фирмой Novell.

В сетевых адаптерах третьего поколения (к ним фирма 3Com относит свои адаптеры семейства EtherLink III) осуществляется конвейерная схема обработки кадров. Она заключается в том, что процессы приема кадра из оперативной памяти компьютера и передачи его в сеть совмещаются во времени. Таким образом, после приема нескольких первых байт кадра начинается их передача. Это существенно (на 25-55 %) повышает производительность цепочки *оперативная память – адаптер – физический канал – адаптер – оперативная память*. Такая схема очень чувствительна к порогу начала передачи, то есть к количеству байт кадра, которое загружается в буфер адаптера перед началом передачи в сеть. Сетевой адаптер третьего поколения осуществляет самонастройку этого параметра путем анализа рабочей среды, а также методом расчета, без участия администратора сети. Самонастройка обеспечивает максимально возможную производительность для конкретного сочетания производительности внутренней шины компьютера, его системы прерываний и системы прямого доступа к памяти.

Адаптеры третьего поколения базируются на специализированных интегральных схемах (ASIC), что повышает производительность и надежность адаптера при одновременном снижении его стоимости. Компания 3Com назвала свою технологию конвейерной обработки кадров Parallel Tasking, другие компании также реализовали похожие схемы в своих адаптерах. Повышение производительности канала «адаптер-память» очень важно для повышения производительности сети в целом, так как производительность сложного маршрута обработки кадров, включающего, например, концентраторы, коммутаторы, маршрутизаторы, глобальные каналы связи и т. п., всегда определяется производительностью самого медленного элемента этого маршрута. Следовательно, если сетевой адаптер сервера или клиентского компьютера работает медленно, никакие быстрые коммутаторы не смогут повысить скорость работы сети.

Выпускаемые сегодня сетевые адаптеры можно отнести к четвертому поколению. В эти адаптеры обязательно входит ASIC, выполняющая функции MAC-уровня, а также большое количество высокоуровневых функций. В набор таких функций может входить поддержка агента удаленного мониторинга RMON, схема приоритезации кадров, функции дистанционного управления компьютером и т. п. В серверных вариантах адаптеров почти обязательно наличие мощного процессора, разгружающего центральный процессор. Примером сетевого адаптера четвертого поколения может служить адаптер компании 3Com Fast EtherLink XL 10/100.

4.2.2. Концентраторы

Основные и дополнительные функции концентраторов

Практически во всех современных технологиях локальных сетей определено устройство, которое имеет несколько равноправных названий – концентратор (concentrator), хаб (hub), повторитель (repeater). В зависимости от области применения этого устройства в значительной степени изменяется состав его функций и конструктивное исполнение. Неизменной остается только основная функция – это *повторение кадра* либо на всех портах (как определено в стандарте Ethernet), либо только на некоторых портах, в соответствии с алгоритмом, определенным соответствующим стандартом.

Концентратор обычно имеет несколько портов, к которым с помощью отдельных физических сегментов кабеля подключаются конечные узлы сети – компьютеры. Концентратор объединяет отдельные физические сегменты сети в единую разделяемую среду, доступ к которой осуществляется в соответствии с одним из рассмотренных протоколов локальных сетей – Ethernet, Token Ring и т. п. Так как логика доступа к разделяемой среде существенно зависит от технологии, то для каждого типа технологии выпускаются свои концентраторы – Ethernet; Token Ring; FDDI и lOOVG-AnyLAN. Для конкретного протокола иногда используется свое, узкоспециализированное название этого устройства, более точно отражающее его функции или же использующееся в силу традиций, например, для концентраторов Token Ring характерно название MSAU.

Каждый концентратор выполняет некоторую основную функцию, определенную в соответствующем протоколе той технологии, которую он поддерживает. Хотя эта функция достаточно детально определена в стандарте технологии, при ее реализации концентраторы разных производителей могут отличаться такими деталями, как количество портов, поддержка нескольких типов кабелей и т. п.

Кроме основной функции концентратор может выполнять некоторое количество дополнительных функций, которые либо в стандарте вообще не определены, либо являются факультативными. Например, концентратор Token Ring может выполнять функцию отключения некорректно работающих портов и перехода на резервное кольцо, хотя в стандарте такие его возможности не описаны. Концентратор оказался удобным устройством для выполнения дополнительных функций, облегчающих контроль и эксплуатацию сети.

Рассмотрим особенности реализации основной функции концентратора на примере концентраторов Ethernet.

В технологии Ethernet устройства, объединяющие несколько физических сегментов коаксиального кабеля в единую разделяемую среду, использовались давно и получили название «повторителей» по своей основной функции – повторению на всех своих портах сигналов, полученных на входе одного из портов. В сетях на основе коаксиального кабеля обычными являлись двухпортовые повторители, соединяющие только два сегмента кабеля, поэтому термин концентратор к ним обычно не применялся.

С появлением спецификации 10Base-T для витой пары повторитель стал неотъемлемой частью сети Ethernet, так как без него связь можно было организовать только между двумя узлами сети. Многопортовые повторители Ethernet на витой паре стали называть концентраторами или хабами, так как в одном устройстве действительно концентрировались связи между большим количеством узлов сети. Концентратор Ethernet обычно имеет от 8 до 72 портов, причем основная часть портов предназначена для подключения кабелей на витой паре. На рис. 4.5 показан типичный концентратор Ethernet, рассчитанный на образование небольших сегментов разделяемой среды. Он имеет 16 портов стандарта 10Base-T с разъемами RJ-45, а также один порт AUI для подключения внешнего трансивера. Обычно к этому порту подключается трансивер, работающий на коаксиал или оптоволокно. С помощью этого трансивера концентратор подключается к магистральному кабелю, соединяющему несколько концентраторов между собой, либо таким образом обеспечивается подключение станции, удаленной от концентратора более чем на 100 м.

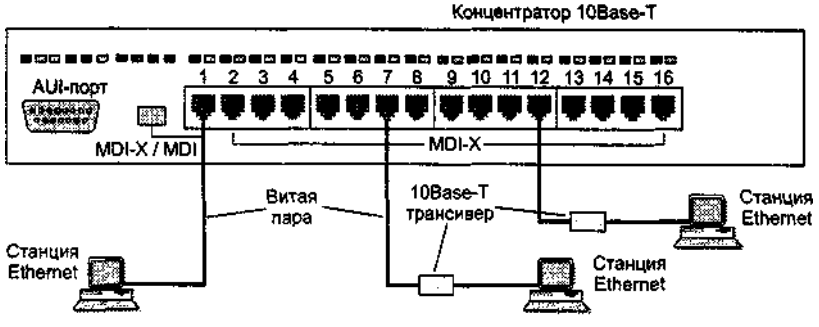


Рис. 4.5. Концентратор Ethernet

Для соединения концентраторов технологии 10Base-T между собой в иерархическую систему коаксиальный или оптоволоконный кабель не обязателен, можно применять те же порты, что и для подключения конечных станций, с учетом одного обстоятельства. Дело в том, что обычный порт RJ-45, предназначенный для подключения сетевого адаптера и называемый MDI-X (кроссированный MDI), имеет инвертированную разводку контактов разъема, чтобы сетевой адаптер можно было подключить к концентратору с помощью стандартного соединительного кабеля, не кроссирующего контакты (рис. 4.6). В случае соединения концентраторов через стандартный порт MDI-X приходится использовать нестандартный кабель с перекрестным соединением пар. Поэтому некоторые изготовители снабжают концентратор выделенным портом MDI, в котором нет кроссирования пар. Таким образом, два концентратора можно соединить обычным некроссированным кабелем, если это делать через порт MDI-X одного концентратора и порт MDI второго. Чаще один порт концентратора может работать и как порт MDI-X, и как порт MDI, в зависимости от положения кнопочного переключателя, как это показано в нижней части рис. 4.6.

Многопортовый повторитель-концентратор Ethernet может по-разному рассматриваться при использовании правила 4-х хабов. В большинстве моделей все порты связаны с единственным блоком повторения, и при прохождении сигнала между двумя портами повторителя блок повторения вносит задержку всего один раз. Поэтому такой концентратор нужно считать одним повторителем с ограничениями, накладываемыми правилом 4-х хабов. Но существуют и другие модели повторителей, в которых на несколько портов имеется свой блок повторения. В таком случае каждый блок повторения нужно считать отдельным повторителем и учитывать его отдельно в правиле 4-х хабов.

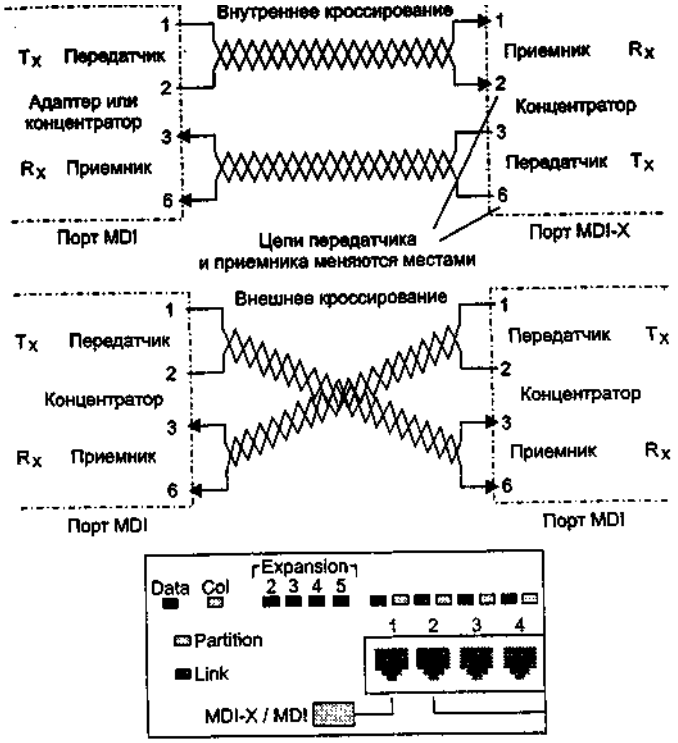


Рис. 4.6. Соединения типа «станция—концентратор» и «концентратор—концентратор» на витой паре

Некоторые отличия могут демонстрировать модели концентраторов, работающие на одномодовый волоконно-оптический кабель. Дальность сегмента кабеля, поддерживаемого концентратором FDDI, на таком кабеле может значительно отличаться в зависимости от мощности лазерного излучателя - от 10 до 40 км.

Однако если существующие различия при выполнении основной функции концентраторов не столь велики, то их намного превосходит разброс в возможностях реализации концентраторами дополнительных функций.

Отключение портов

Очень полезной при эксплуатации сети является способность концентратора отключать некорректно

работающие порты, изолируя тем самым остальную часть сети от возникших в узле проблем. Эту функцию называют *автосегментацией* (*autopartitioning*). Для концентратора FDDI эта функция для многих ошибочных ситуаций является основной, так как определена в протоколе. В то же время для концентратора Ethernet или Token Ring функция автосегментации для многих ситуаций является дополнительной, так как стандарт не описывает реакцию концентратора на эту ситуацию. Основной причиной отключения порта в стандартах Ethernet и Fast Ethernet является отсутствие ответа на последовательность импульсов link test, посылаемых во все порты каждые 16 мс. В этом случае неисправный порт переводится в состояние «отключен», но импульсы link test будут продолжать посылаться в порт с тем, чтобы при восстановлении устройства работа с ним была продолжена автоматически.

Рассмотрим ситуации, в которых концентраторы Ethernet и Fast Ethernet выполняют отключение порта.

- *Ошибки на уровне кадра*. Если интенсивность прохождения через порт кадров, имеющих ошибки, превышает заданный порог, то порт отключается, а затем, при отсутствии ошибок в течение заданного времени, включается снова. Такими ошибками могут быть: неверная контрольная сумм, неверная длина кадра (больше 1518 байт или меньше 64 байт), неоформленный заголовок кадра.
- *Множественные коллизии*. Если концентратор фиксирует, что источником коллизии был один и тот же порт 60 раз подряд, то порт отключается. Через некоторое время порт снова будет включен.
- *Затянувшаяся передача* (*jabber*). Как и сетевой адаптер, концентратор контролирует время прохождения одного кадра через порт. Если это время превышает время передачи кадра максимальной длины в 3 раза, то порт отключается.

Поддержка резервных связей

Так как использование резервных связей в концентраторах определено только в стандарте FDDI, то для остальных стандартов разработчики концентраторов поддерживают такую функцию с помощью своих частных решений. Например, концентраторы Ethernet/Fast Ethernet могут образовывать только иерархические связи без петель. Поэтому резервные связи всегда должны соединять отключенные порты, чтобы не нарушать логику работы сети. Обычно при конфигурировании концентратора администратор должен определить, какие порты являются основными, а какие по отношению к ним - резервными (рис. 4.7). Если по какой-либо причине порт отключается (срабатывает механизм автосегментации), концентратор делает активным его резервный порт.

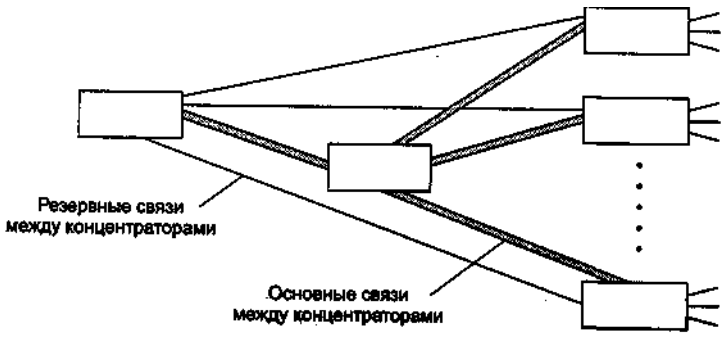


Рис. 4.7. Резервные связи между концентраторами Ethernet

В некоторых моделях концентраторов разрешается использовать механизм назначения резервных портов только для оптоволоконных портов, считая, что нужно резервировать только наиболее важные связи, которые обычно выполняются на оптическом кабеле. В других же моделях резервным можно сделать любой порт.

Защита от несанкционированного доступа

Разделяемая среда предоставляет очень удобную возможность для несанкционированного прослушивания сети и получения доступа к передаваемым данным. Для этого достаточно подключить

компьютер с программным анализатором протоколов к свободному разъему концентратора, записать на диск весь проходящий по сети трафик, а затем выделить из него нужную информацию.

Разработчики концентраторов предоставляют некоторый способ защиты данных в разделяемых средах.

Наиболее простой способ – назначение разрешенных MAC-адресов портам концентратора. В стандартном концентраторе Ethernet порты MAC-адресов не имеют. Защита заключается в том, что администратор вручную связывает с каждым портом концентратора некоторый MAC-адрес. Этот MAC-адрес является адресом станции, которой разрешается подключаться к данному порту. Например, на рис. 4.8 первому порту концентратора назначен MAC-адрес 123 (условная запись). Компьютер с MAC-адресом 123 нормально работает с сетью через данный порт. Если злоумышленник отсоединяет этот компьютер и присоединяет вместо него свой, концентратор заметит, что при старте нового компьютера в сеть начали поступать кадры с адресом источника 789. Так как этот адрес является недопустимым для первого порта, то эти кадры фильтруются, порт отключается, а факт нарушения прав доступа может быть зафиксирован.

Заметим, что для реализации описанного метода защиты данных концентратор нужно предварительно сконфигурировать. Для этого концентратор должен иметь блок управления. Такие концентраторы обычно называют интеллектуальными. Блок управления представляет собой компактный вычислительный блок со встроенным программным обеспечением. Для взаимодействия администратора с блоком управления концентратор имеет консольный порт (чаще всего RS-232), к которому подключается терминал или персональный компьютер с программой эмуляции терминала. При присоединении терминала блок управления организует на его экране диалог, с помощью которого администратор вводит значения MAC-адресов. Блок управления может поддерживать и другие операции конфигурирования, например ручное отключение или включение портов и т. д. Для этого при подключении терминала блок управления выдает на экран некоторое меню, с помощью которого администратор выбирает нужное действие.

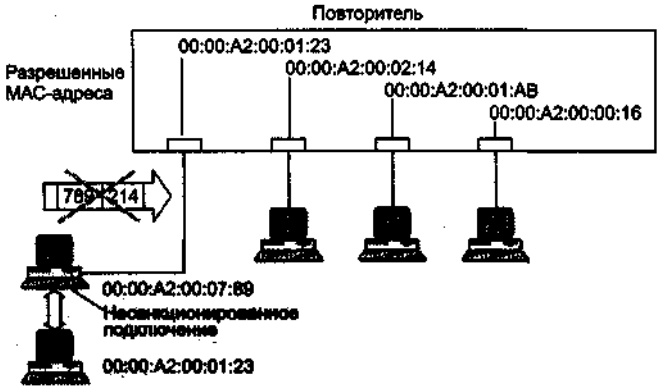


Рис. 4.8. Изоляция портов: передача кадров только от станций с фиксированными адресами

Другим способом защиты данных от несанкционированного доступа является их шифрация. Однако процесс истинной шифрации требует большой вычислительной мощности, и для повторителя, не буферизующего кадр, выполнить шифрацию «на лету» весьма сложно. Вместо этого в концентраторах применяется метод случайного искажения поля данных в пакетах, передаваемых портам с адресом, отличным от адреса назначения пакета. Этот метод сохраняет логику случайного доступа к среде, так как все станции видят занятость среды кадром информации, но только станция, которой послан этот кадр, может понять содержание поля данных кадра (рис. 4.9). Для реализации этого метода концентратор также нужно снабдить информацией о том, какие MAC-адреса имеют станции, подключенные к его портам. Обычно поле данных в кадрах, направляемых станциям, отличным от адресата, заполняется нулями.

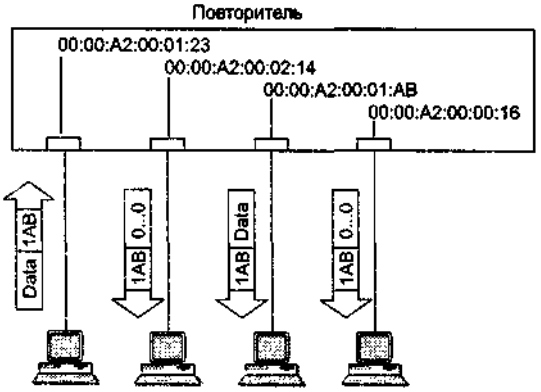


Рис. 4.9. Искажение поля данных в кадрах, не предназначенных для приема станциями

Многосегментные концентраторы

При рассмотрении некоторых моделей концентраторов возникает вопрос – зачем в этой модели имеется такое большое количество портов, например 192 или 240? Имеет ли смысл разделять среду в 10 или 16 Мбит/с между таким большим количеством станций? Возможно, десять – пятнадцать лет назад ответ в некоторых случаях мог бы быть и положительным, например, для тех сетей, в которых компьютеры пользовались сетью только для отправки небольших почтовых сообщений или для переписывания небольшого текстового файла. Сегодня таких сетей осталось крайне мало, и даже 5 компьютеров могут полностью загрузить сегмент Ethernet или Token Ring, а в некоторых случаях – и сегмент Fast Ethernet. Для чего же тогда нужен концентратор с большим количеством портов, если ими практически нельзя воспользоваться из-за ограничений по пропускной способности, приходящейся на одну станцию? Ответ состоит в том, что в таких концентраторах имеется несколько несвязанных внутренних шин, которые предназначены для создания нескольких разделяемых сред. Например, концентратор, изображенный на рис. 4.10, имеет три внутренние шины Ethernet. Если, например, в таком концентраторе 72 порта, то каждый из этих портов может быть связан с любой из трех внутренних шин. На рисунке первые два компьютера связаны с шиной Ethernet 3, а третий и четвертый компьютеры – с шиной Ethernet 1. Первые два компьютера образуют один разделяемый сегмент, а третий и четвертый – другой разделяемый сегмент.

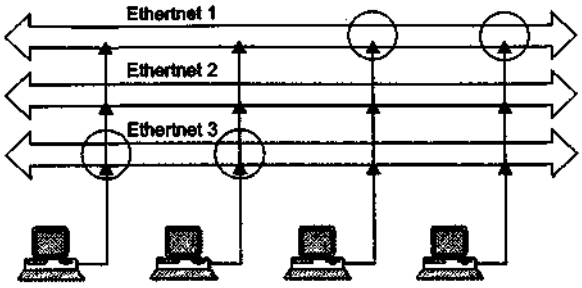


Рис. 4.10. Многосегментный концентратор

Между собой компьютеры, подключенные к разным сегментам, общаться через концентратор не могут, так как шины внутри концентратора никак не связаны.

Многосегментные концентраторы нужны для создания разделяемых сегментов, состав которых может легко изменяться. Большинство многосегментных концентраторов, например System 5000 компании Nortel Networks или PortSwitch Hub компании 3Com, позволяют выполнять операцию соединения порта с одной из внутренних шин чисто программным способом, например с помощью локального конфигурирования через консольный порт. В результате администратор сети может присоединять компьютеры пользователей к любым портам концентратора, а затем с помощью программы конфигурирования концентратора управлять составом каждого сегмента. Если завтра сегмент 1 станет перегруженным, то его компьютеры можно распределить между оставшимися сегментами концентратора.

Возможность многосегментного концентратора программно изменять связи портов с внутренними

шинами называется *конфигурационной коммутацией* (*configuration switching*).

ВНИМАНИЕ	Конфигурационная коммутация не имеет ничего общего с коммутацией кадров, которую выполняют мосты и коммутаторы

Многосегментные концентраторы – это программируемая основа больших сетей. Для соединения сегментов между собой нужны устройства другого типа – мосты/коммутаторы или маршрутизаторы. Такое межсетевое устройство должно подключаться к нескольким портам многосегментного концентратора, подсоединенным к разным внутренним шинам, и выполнять передачу кадров или пакетов между сегментами точно так же, как если бы они были образованы отдельными устройствами-концентраторами.

Для крупных сетей многосегментный концентратор играет роль интеллектуального кроссового шкафа, который выполняет новое соединение не за счет механического перемещения вилки кабеля в новый порт, а за счет программного изменения внутренней конфигурации устройства.

Управление концентратором по протоколу SNMP

Как видно из описания дополнительных функций, многие из них требуют конфигурирования концентратора. Это конфигурирование может производиться локально, через интерфейс RS-232C, который имеется у любого концентратора, имеющего блок управления. Кроме конфигурирования в большой сети очень полезна функция наблюдения за состоянием концентратора: работоспособен ли он, в каком состоянии находятся его порты.

При большом количестве концентраторов и других коммуникационных устройств в сети постоянное наблюдение за состоянием многочисленных портов и изменением их параметров становится очень обременительным занятием, если оно должно выполняться с помощью локального подключения терминала. Поэтому большинство концентраторов, поддерживающих интеллектуальные дополнительные функции, могут управляться централизованно по сети с помощью популярного протокола управления SNMP (Simple Network Management Protocol) из стека TCP/IP.

Упрощенная структура системы управления изображена на рис. 4.11.

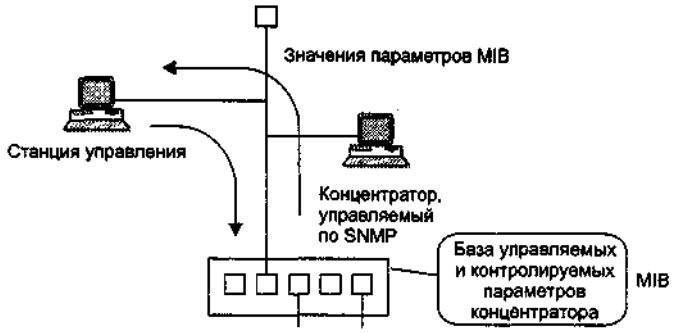


Рис. 4.11. Структура системы управления на основе протокола SNMP

В блок управления концентратором встраивается так называемый SNMP-агент. Этот агент собирает информацию о состоянии контролируемого устройства и хранит ее в так называемой базе данных управляющей информации – *Management Information Base, MIB*. Эта база данных имеет стандартную структуру, что позволяет одному из компьютеров сети, выполняющему роль центральной станции управления, запрашивать у агента значения стандартных переменных базы MIB. В базе MIB хранятся не только данные о состоянии устройства, но и управляющая информация, воздействующая на это устройство. Например, в MIB есть переменная, управляющая состоянием порта, имеющая значения «включить» и «выключить». Если станция управления меняет значение управляющей переменной, то агент должен выполнить это указание и воздействовать на устройство соответствующим образом, например выключить порт или изменить связь порта с внутренними шинами концентратора.

Взаимодействие между станцией управления (по-другому – менеджером системы управления) и встроенными в коммуникационные устройства агентами происходит по протоколу SNMP. Концентратор, который управляется по протоколу SNMP, должен поддерживать основные протоколы

стека TCP/IP и иметь IP- и MAC-адреса. Точнее, эти адреса относятся к агенту концентратора. Поэтому администратор, который хочет воспользоваться преимуществами централизованного управления концентраторами по сети, должен знать стек протоколов TCP/IP и сконфигурировать IP-адреса их агентов.

Конструктивное исполнение концентраторов

На конструктивное устройство концентраторов большое влияние оказывает их область применения. Концентраторы рабочих групп чаще всего выпускаются как. устройства с фиксированным количеством портов, корпоративные концентраторы – как модульные устройства на основе шасси, а концентраторы отделов могут иметь стековую конструкцию. Такое деление не является жестким, и в качестве корпоративного концентратора может использоваться, например, модульный концентратор.

Концентратор с фиксированным количеством портов – это наиболее простое конструктивное исполнение, когда устройство представляет собой отдельный корпус со всеми необходимыми элементами (портами, органами индикации и управления, блоком питания), и эти элементы заменять нельзя. Обычно все порты такого концентратора поддерживают одну среду передачи, общее количество портов изменяется от 4-8 до 24. Один порт может быть специально выделен для подключения концентратора к магистрали сети или же для объединения концентраторов (в качестве такого порта часто используется порт с интерфейсом AUI, в этом случае применение соответствующего трансивера позволяет подключить концентратор к практически любой физической среде передачи данных).

Модульный концентратор выполняется в виде отдельных модулей с фиксированным количеством портов, устанавливаемых на общее шасси. Шасси имеет внутреннюю шину для объединения отдельных модулей в единый повторитель. Часто такие концентраторы являются многосегментными, тогда в пределах одного модульного концентратора работает несколько несвязанных между собой повторителей. Для модульного концентратора могут существовать различные типы модулей, отличающиеся количеством портов и типом поддерживаемой физической среды. Часто агент протокола SNMP выполняется в виде отдельного модуля, при установке которого концентратор превращается в интеллектуальное устройство. Модульные концентраторы позволяют более точно подобрать необходимую для конкретного применения конфигурацию концентратора, а также гибко и с минимальными затратами реагировать на изменения конфигурации сети.

Ввиду ответственной работы, которую выполняют корпоративные модульные концентраторы, они снабжаются модулем управления, системой терморегулирования, избыточными источниками питания и возможностью замены модулей «на ходу».

Недостатком концентратора на основе шасси является высокая начальная стоимость такого устройства для случая, когда предприятию на первом этапе создания сети нужно установить всего 1-2 модуля. Высокая стоимость шасси вызвана тем, что оно поставляется вместе со всеми общими устройствами, такими как избыточные источники питания и т. п. Поэтому для сетей средних размеров большую популярность завоевали стековые концентраторы.

Стековый концентратор, как и концентратор с фиксированным числом портов, выполнен в виде отдельного корпуса без возможности замены отдельных его модулей. Типичный вид нескольких стековых концентраторов Ethernet показан на рис. 4.12. Однако стековыми эти концентраторы называются не потому, что они устанавливаются один на другой. Такая чисто конструктивная деталь вряд ли удостоилась бы особого внимания, так как установка нескольких устройств одинаковых габаритных размеров в общую стойку практикуется очень давно. Стековые концентраторы имеют специальные порты и кабели для объединения нескольких таких корпусов в единый повторитель (рис. 4.13), который имеет общий блок повторения, обеспечивает общую ресинхронизацию сигналов для всех своих портов и поэтому с точки зрения правила 4-х хабов считается одним повторителем.

Если стековые концентраторы имеют несколько внутренних шин, то при соединении в стек эти шины объединяются и становятся общими для всех устройств стека. Число объединяемых в стек корпусов может быть достаточно большим (обычно до 8, но бывает и больше). Стековые концентраторы могут поддерживать различные физические среды передачи, что делает их почти такими же гибкими, как и модульные концентраторы, но при этом стоимость этих устройств в расчете на один порт получается обычно ниже, так как сначала предприятие может купить одно устройство без избыточного шасси, а потом нарастить стек еще несколькими аналогичными устройствами.

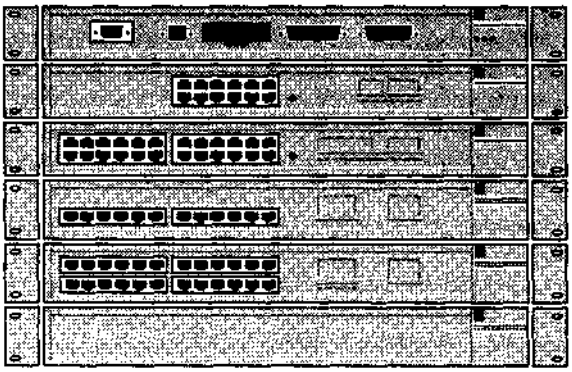


Рис. 4.12. Стековые концентраторы Ethernet

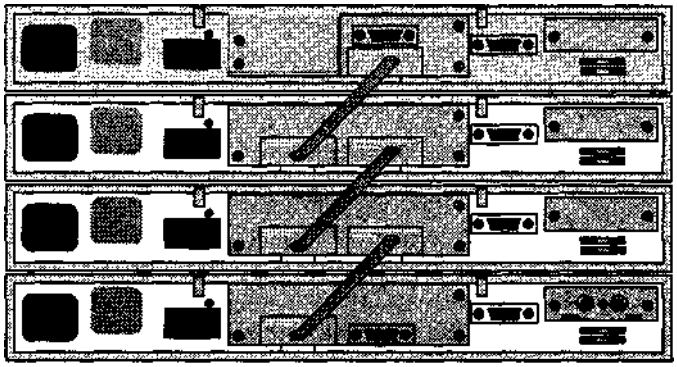


Рис. 4.13. Объединение стековых концентраторов в единое устройство с помощью специальных разъемов на задней панели

Стековые концентраторы, выпускаемые одним производителем, выполняются в едином конструктивном стандарте, что позволяет легко устанавливать их друг на друга, образуя единое настольное устройство, или помещать их в общую стойку. Экономия при организации стека происходит еще и за счет единого для всех устройств стека модуля SNMP-управления (который вставляется в один из корпусов стека как дополнительный модуль), а также общего избыточного источника питания.

Модульно-стековые концентраторы представляют собой модульные концентраторы, объединенные специальными связями в стек. Как правило, корпуса таких концентраторов рассчитаны на небольшое количество модулей (1-3). Эти концентраторы сочетают достоинства концентраторов обоих типов.

Выводы

- От производительности сетевых адаптеров зависит производительность любой сложной сети, так как данные всегда проходят не только через коммутаторы и маршрутизаторы сети, но и через адаптеры компьютеров, а результирующая производительность последовательно соединенных устройств определяется производительностью самого медленного устройства.
- Сетевые адаптеры характеризуются типом поддерживаемого протокола, производительностью, шиной компьютера, к которой они могут присоединяться, типом приемопередатчика, а также наличием собственного процессора, разгружающего центральный процессор компьютера от рутинной работы.
- Сетевые адаптеры для серверов обычно имеют собственный процессор, а клиентские сетевые адаптеры – нет.
- Современные адаптеры умеют адаптироваться к временным параметрам шины и оперативной памяти компьютера для повышения производительности обмена «сеть–компьютер».
- Концентраторы, кроме основной функции протокола (побитного повторения кадра на всех или последующем порту), всегда выполняют ряд полезных дополнительных функций, определяемых производителем концентратора.
- Автосегментация – одна из важнейших дополнительных функций, с помощью которой концентратор отключает порт при обнаружении разнообразных проблем с кабелем и конечным узлом, подключенным к данному порту.

- В число дополнительных функций входят функции защиты сети от несанкционированного доступа, запрещающие подключение к концентратору компьютеров с неизвестными MAC-адресами, а также заполняющие нулями поля данных кадров, поступающих не к станции назначения.
- Стековые концентраторы сочетают преимущества модульных концентраторов и концентраторов с фиксированным количеством портов.
- Многосегментные концентраторы позволяют делить сеть на сегменты программным способом, без физической перекоммутации устройств.
- Сложные концентраторы, выполняющие дополнительные функции, обычно могут управляться централизованно по сети по протоколу SNMP.

4.3. Логическая структуризация сети с помощью мостов и коммутаторов

Под логической структуризацией сети понимается разбиение общей разделяемой среды на логические сегменты, которые представляют самостоятельные разделяемые среды с меньшим количеством узлов. Сеть, разделенная на логические сегменты, обладает более высокой производительностью и надежностью. Взаимодействие между логическими сегментами организуется с помощью мостов и коммутаторов.

4.3.1. Причины логической структуризации локальных сетей

Ограничения сети, построенной на общей разделяемой среде

При построении небольших сетей, состоящих из 10-30 узлов, использование стандартных технологий на разделяемых средах передачи данных приводит к экономичным и эффективным решениям. Во всяком случае, это утверждение справедливо для очень большого числа сегодняшних сетей, даже тех, в которых передаются большие объемы мультимедийной информации, – появление высокоскоростных технологий со скоростями обмена 100 и 1000 Мбит/с решает проблему качества транспортного обслуживания таких сетей.

Эффективность разделяемой среды для небольшой сети проявляется в первую очередь в следующих свойствах:

- простой топологии сети, допускающей легкое наращивание числа узлов (в небольших пределах);
- отсутствии потерь кадров из-за переполнения буферов коммуникационных устройств, так как новый кадр не передается в сеть, пока не принят предыдущий – сама логика разделения среды регулирует поток кадров и приостанавливает станции, слишком часто генерирующие кадры, заставляя их ждать доступа;
- простоте протоколов, обеспечившей низкую стоимость сетевых адаптеров, повторителей и концентраторов.

Однако справедливым является и другое утверждение – крупные сети, насчитывающие сотни и тысячи узлов, не могут быть построены на основе одной разделяемой среды даже такой скоростной технологии, как Gigabit Ethernet. И не только потому, что практически все технологии ограничивают количество узлов в разделяемой среде: все виды семейства Ethernet – 1024 узлами, Token Ring – 260 узлами, а FDDI – 500 узлами. Даже сеть средних размеров, состоящая из 50-100 компьютеров и укладывающаяся в разрешенный максимум количества узлов, чаще всего будет плохо работать на одной разделяемой среде.

Основные недостатки сети на одной разделяемой среде начинают проявляться при превышении некоторого порога количества узлов, подключенных к разделяемой среде, и состоят в следующем. Даже та доля пропускной способности разделяемого сегмента, которая должна в среднем доставаться одному узлу (то есть, например, 10/N Мбит/с для сегмента Ethernet с N компьютерами), очень часто узлу не достается. Причина заключается в случайном характере метода доступа к среде, используемом во всех технологиях локальных сетей. Наиболее тяжелые условия для узлов сети создает метод доступа CSMA/CD технологии Ethernet, но и в других технологиях, таких как Token Ring или FDDI, где метод

доступа носит менее случайный характер и даже часто называется детерминированным, случайный фактор доступа к среде все равно присутствует и оказывает свое негативное влияние на пропускную способность, достающуюся отдельному узлу.

На рис. 4.14 показана зависимость задержек доступа к среде передачи данных в сетях Ethernet, Token Ring и FDDI от коэффициента использования сети *p*, который также часто называют коэффициентом нагрузки сети. Напомним, что коэффициент использования сети равен отношению трафика, который должна передать сеть, к ее максимальной пропускной способности. Для сети Ethernet максимальная пропускная способность равна 10 Мбит/с, а трафик, который она должна передать, равен сумме интенсивностей трафика; генерируемого каждым узлом сети. Коэффициент использования обычно измеряют в относительных единицах или процентах.

Как видно из рисунка, всем технологиям присущ экспоненциальный рост величины задержек доступа при увеличении коэффициента использования сети, отличается только порог, при котором наступает резкий перелом в поведении сети, когда почти прямолинейная зависимость переходит в крутую экспоненту. Для всего семейства технологий Ethernet это 40-50 %, для технологии Token Ring – 60 %, а технологии FDDI - 70 %.

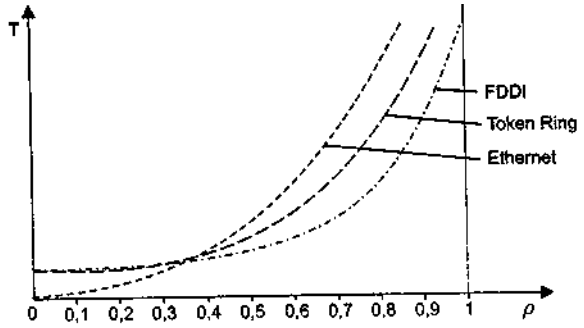


Рис. 4.14. Задержки доступа к среде передачи данных для технологий Ethernet, Token Ring и FDDI

Количество узлов, при которых коэффициент использования сети начинает приближаться к опасной границе, зависит от типа функционирующих в узлах приложений. Если раньше для сетей Ethernet считалось, что 30 узлов – это вполне приемлемое число для одного разделяемого сегмента, то сегодня для мультимедийных приложений, перекачивающих большие файлы данных, эту цифру нужно уточнять с помощью натурных или имитационных экспериментов.

Влияние задержек и коллизий на полезную пропускную способность сети Ethernet хорошо отражает график, представленный на рис. 4.15.



Рис. 4.15. Зависимость полезной пропускной способности сети Ethernet от коэффициента использования

При загрузке сети до 50 % технология Ethernet на разделяемом сегменте хорошо справляется с передачей трафика, генерируемого конечными узлами. Однако при повышении интенсивности генерируемого узлами трафика сеть все больше времени начинает проводить неэффективно, повторно передавая кадры, которые вызвали коллизию. При возрастании интенсивности генерируемого трафика до такой величины, когда коэффициент использования сети приближается к 1, вероятность

столкновения кадров настолько увеличивается, что практически любой кадр, который какая-либо станция пытается передать, сталкивается с другими кадрами, вызывая коллизию. Сеть перестает передавать полезную пользовательскую информацию и работает «на себя», обрабатывая коллизии.

Этот эффект хорошо известен на практике и исследован путем имитационного моделирования, поэтому сегменты Ethernet не рекомендуется загружать так, чтобы среднее значение коэффициента использования превосходило 30 %. Именно поэтому во многих системах управления сетями пороговая граница для индикатора коэффициента загрузки сети Ethernet по умолчанию устанавливается на величину 30 %.

Технология Ethernet наиболее чувствительна к перегрузкам разделяемого сегмента, но и другие технологии также весьма страдают от этого эффекта, поэтому ограничения, связанные с возникающими коллизиями и большим временем ожидания доступа при значительной загрузке разделяемого сегмента, чаще всего оказываются более серьезными, чем ограничение на максимальное количество узлов, определенное в стандарте из соображений устойчивой передачи электрических сигналов в кабелях.

В результате даже сеть средних размеров трудно построить на одном разделяемом сегменте так, чтобы она работала эффективно при изменении интенсивности генерируемого станциями трафика. Кроме того, при использовании разделяемой среды проектировщик сети сталкивается с жесткими ограничениями максимальной длины сети, которые для всех технологий лежат в пределах нескольких километров, и только технология FDDI позволяет строить локальные сети, длина которых измеряется десятками километров.

Преимущества логической структуризации сети

Ограничения, возникающие из-за использования общей разделяемой среды, можно преодолеть, разделив сеть на несколько разделяемых сред и соединив отдельные сегменты сети такими устройствами, как мосты, коммутаторы или маршрутизаторы (рис. 4.16).

Перечисленные устройства передают кадры с одного своего порта на другой, анализируя адрес назначения, помещенный в этих кадрах. (В отличие от концентраторов, которые повторяют кадры на всех своих портах, передавая их во все подсоединенные к ним сегменты, независимо от того, в каком из них находится станция назначения.) Мосты и коммутаторы выполняют операцию передачи кадров на основе плоских адресов канального уровня, то есть MAC-адресов, а маршрутизаторы – на основе номера сети. При этом единая разделяемая среда, созданная концентраторами (или в предельном случае – одним сегментом кабеля), делится на несколько частей, каждая из которых присоединена к порту моста, коммутатора или маршрутизатора.

Говорят, что при этом сеть делится на логические сегменты или сеть подвергается *логической структуризации*. Логический сегмент представляет собой единую разделяемую среду. Деление сети на логические сегменты приводит к тому, что нагрузка, приходящаяся на каждый из вновь образованных сегментов, почти всегда оказывается меньше, чем нагрузка, которую испытывала исходная сеть. Следовательно, уменьшаются вредные эффекты от разделения среды: снижается время ожидания доступа, а в сетях Ethernet – и интенсивность коллизий.

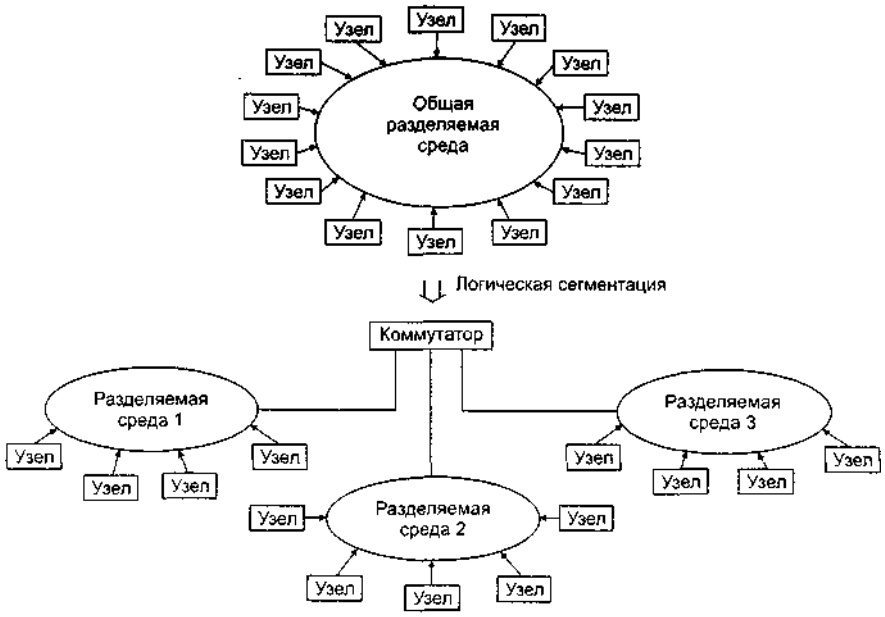


Рис. 4.16. Логическая структуризация сети

Для иллюстрации этого эффекта рассмотрим рис. 4.17. На нем изображены два сегмента, соединенные мостом. Внутри сегментов имеются повторители. До деления сети на сегменты весь трафик, генерируемый узлами сети, был общим (представим, что место межсетевого устройства также занимал повторитель) и учитывался при определении коэффициента использования сети. Если обозначить среднюю интенсивность трафика, идущего от узла i к узлу j через C_{ij} то суммарный трафик, который должна была передавать сеть до деления на сегменты, равен $C_{\sum} = \sum C_{ij}$ (считаем, что суммирование проводится по всем узлам).

После разделения сети на сегменты нагрузка каждого сегмента изменилась. При ее вычислении теперь нужно учитывать только внутрисегментный трафик, то есть трафик кадров, которые циркулируют между узлами одного сегмента, а также межсегментный трафик, который либо направляется от узла данного сегмента узлу другого сегмента, либо приходит от узла другого сегмента в узел данного сегмента. Внутренний трафик другого сегмента теперь нагрузку на данный сегмент не создает.

Поэтому нагрузка, например, сегмента S1 стала равна $C_{s1} + C_{s1\text{-}s2}$, где C_{s1} – внутренний трафик сегмента S1, а $C_{s1\text{-}s2}$– межсегментный трафик. Чтобы показать, что нагрузка сегмента S1 уменьшилась, заметим, что общую нагрузку сети до разделения на сегменты можно записать в такой форме: $C_{\sum} = C_{s1} + C_{s1\text{-}s2} + C_{s2}$, а значит, нагрузка сегмента S1 после разделения стала равной $C_{\sum} - C_{s2}$, то есть уменьшилась на величину внутреннего трафика сегмента S2. А раз нагрузка на сегмент уменьшилась, то в соответствии с графиками, приведенными на рис. 4.14 и 4.15, задержки в сегментах также уменьшились, а полезная пропускная способность сегмента в целом и полезная пропускная способность, приходящаяся на один узел, увеличились.

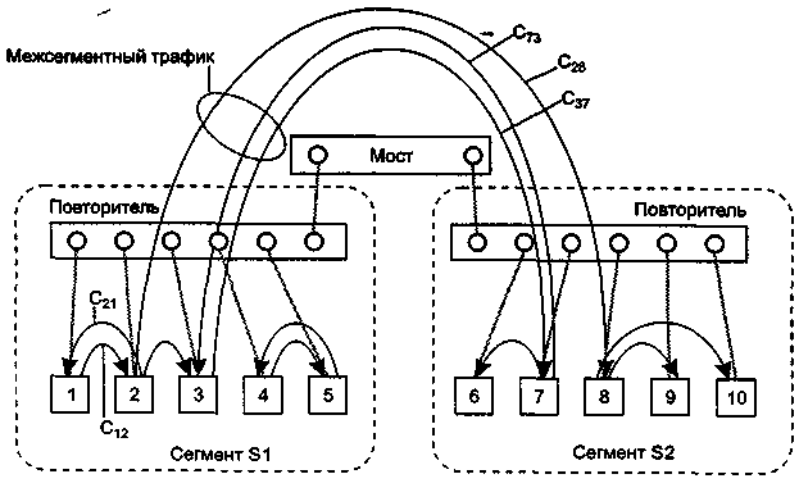


Рис. 4.17. Изменение нагрузки при делении сети на сегменты

Выше было сказано, что деление сети на логические сегменты *почти* всегда уменьшает нагрузку в новых сегментах. Слово «почти» учитывает очень редкий случай, когда сеть разбита на сегменты так, что внутренний трафик каждого сегмента равен нулю, то есть весь трафик является межсегментным. Для примера из рис. 4.17 это означало бы, что все компьютеры сегмента S1 обмениваются данными только с компьютерами сегмента S2, и наоборот.

Такой случай является, естественно, экзотическим. На практике на предприятии всегда можно выделить группу компьютеров, которые принадлежат сотрудникам, выполняющим общую задачу. Это могут быть сотрудники одной рабочей группы, отдела, другого структурного подразделения предприятия. В большинстве случаев им нужен доступ к ресурсам сети их отдела и только изредка – доступ к удаленным ресурсам. И хотя уже упомянутое эмпирическое правило, говорящее о том, что можно разделить сеть на сегменты так, что 80 % трафика составляет обращение к локальным ресурсам и только 20 % – к удаленным, сегодня трансформируется в правило 50 на 50 % и даже 20 на 80 %, все равно внутрисегментный трафик существует. Если его нет, значит, сеть разбита на логические подсети неверно.

Большинство крупных сетей разрабатывается на основе структуры с общей магистралью, к которой через мосты и маршрутизаторы присоединяются подсети. Эти подсети обслуживают различные отделы. Подсети могут делиться и далее на сегменты, предназначенные для обслуживания рабочих групп.

В общем случае деление сети на логические сегменты повышает производительность сети (за счет разгрузки сегментов), а также гибкость построения сети, увеличивая степень защиты данных, и облегчает управление сетью.

Сегментация увеличивает гибкость сети. При построении сети как совокупности подсетей каждая подсеть может быть адаптирована к специфическим потребностям рабочей группы или отдела. Например в одной подсети может использоваться технология Ethernet и ОС NetWare, а в другой Token Ring и OS-400, в соответствии с традициями того или иного отдела или потребностями имеющихся приложений. Вместе с тем, у пользователей обеих подсетей есть возможность обмениваться данными через межсетевые устройства, такие как мосты, коммутаторы, маршрутизаторы. Процесс разбиения сети на логические сегменты можно рассматривать и в обратном направлении, как процесс создания большой сети из модулей – уже имеющихся подсетей.

Подсети повышают безопасность данных. При подключении пользователей к различным физическим сегментам сети можно запретить доступ определенных пользователей к ресурсам других сегментов. Устанавливая различные логические фильтры на мостах, коммутаторах и маршрутизаторах, можно контролировать доступ к ресурсам, чего не позволяют сделать повторители.

Подсети упрощают управление сетью. Побочным эффектом уменьшения трафика и повышения безопасности данных является упрощение управления сетью. Проблемы очень часто локализуются внутри сегмента. Как и в случае структурированной кабельной системы, проблемы одной подсети не оказывают влияния на другие подсети. Подсети образуют логические домены управления сетью.

Сети должны проектироваться на двух уровнях: физическом и логическом. Логическое проектирование определяет места расположения ресурсов, приложений и способы группировки этих ресурсов в логические сегменты.

Структуризация с помощью мостов и коммутаторов

В данной главе рассматриваются устройства логической структуризации сетей, работающие на канальном уровне стека протоколов, а именно – мосты и коммутаторы. Структуризация сети возможна также на основе маршрутизаторов, которые для выполнения этой задачи привлекают протоколы сетевого уровня. Каждый способ структуризации – с помощью канального протокола и с помощью сетевого протокола – имеет свои преимущества и недостатки. В современных сетях часто используют комбинированный способ логической структуризации – небольшие сегменты объединяются устройствами канального уровня в более крупные подсети, которые, в свою очередь, соединяются маршрутизаторами.

Итак, сеть можно разделить на логические сегменты с помощью устройств двух типов – мостов (bridge) и/или коммутаторов (switch, switching hub). Сразу после появления коммутаторов в начале 90-х годов сложилось мнение, что мост и коммутатор – это принципиально различные устройства. И хотя постепенно представление о коммутаторах изменилось, это мнение можно услышать и сегодня.

Тем не менее мост и коммутатор – это функциональные близнецы. Оба эти устройства продвигают кадры на основании одних и тех же алгоритмов. Мосты и коммутаторы используют два типа алгоритмов: *алгоритм прозрачного моста* (*transparent bridge*), описанного в стандарте IEEE 802.1D, либо алгоритм *моста с маршрутизацией от источника* (*source routing bridge*) компании IBM для сетей Token Ring. Эти стандарты были разработаны задолго до появления первого коммутатора, поэтому в них используется термин «мост». Когда же на свет появилась первая промышленная модель коммутатора для технологии Ethernet, то она выполняла тот же алгоритм продвижения кадров IEEE 802.1D, который был с десяток лет отработан мостами локальных и глобальных сетей. Точно так же поступают и все современные коммутаторы. Коммутаторы, которые продвигают кадры протокола Token Ring, работают по алгоритму Source Routing, характерному для мостов IBM;

Основное отличие коммутатора от моста заключается в том, что мост обрабатывает кадры последовательно, а коммутатор – параллельно. Это обстоятельство связано с тем, что мосты появились в те времена, когда сеть делили на небольшое количество сегментов, а межсегментный трафик был небольшим (он подчинялся правилу 80 на 20 %). Сеть чаще всего делили на два сегмента, поэтому и термин был выбран соответствующий – мост. Для обработки потока данных со средней интенсивностью 1 Мбит/с мосту вполне хватало производительности одного процессорного блока.

При изменении ситуации в конце 80-х – начале 90-х годов – появлении быстрых протоколов, производительных персональных компьютеров, мультимедийной информации, разделении сети на большое количество сегментов – классические мосты перестали справляться с работой. Обслуживание потоков кадров между теперь уже несколькими портами с помощью одного процессорного блока требовало значительного повышения быстродействия процессора, а это довольно дорогостоящее решение.

Более эффективным оказалось решение, которое и «породило» коммутаторы: для обслуживания потока, поступающего на каждый порт, в устройство ставился отдельный специализированный процессор, который реализовывал алгоритм моста. По сути, коммутатор – это мультипроцессорный мост, способный параллельно продвигать кадры сразу между всеми парами своих портов. Но если при добавлении процессорных блоков компьютер не перестали называть компьютером, а добавили только прилагательное «мультипроцессорный», то с мультипроцессорными мостами произошла метаморфоза – они превратились в коммутаторы. Этому способствовал способ связи между отдельными процессорами коммутатора – они связывались коммутационной матрицей, похожей на матрицы мультипроцессорных компьютеров, связывающие процессоры с блоками памяти.

Постепенно коммутаторы вытеснили из локальных сетей классические однопроцессорные мосты. Основная причина этого – очень высокая производительность, с которой коммутаторы передают кадры между сегментами сети. Если мосты могли даже замедлять работу сети, когда их производительность оказывалась меньше интенсивности межсегментного потока кадров, то коммутаторы всегда выпускаются с процессорами портов, которые могут передавать кадры с той максимальной скоростью, на которую рассчитан протокол. Добавление к этому параллельной передачи кадров между портами сделало производительность коммутаторов на несколько порядков выше, чем мостов – коммутаторы могут передавать до нескольких миллионов кадров в секунду, в то время как мосты обычно обрабатывали 3-5 тысяч кадров в секунду. Это и предопределило судьбу мостов и коммутаторов.

Процесс вытеснения мостов начал протекать достаточно быстро с 1994 года, и сегодня локальные мосты практически не производятся сетевой индустрией. За время своего существования уже без конкурентов-мостов коммутаторы вобрали в себя многие дополнительные функции, которые появлялись в результате естественного развития сетевых технологий. К этим функциям относятся, например, поддержка виртуальных сетей (VLAN), приоритезация трафика, использование магистрального порта по умолчанию и т. п.

Сегодня мосты по-прежнему работают в сетях, но только на достаточно медленных глобальных связях между двумя удаленными локальными сетями. Такие мосты называются удаленными мостами (remote bridge), и алгоритм их работы ничем не отличается от стандарта 802.1D или Source Routing.

Прозрачные мосты умеют, кроме передачи кадров в рамках одной технологии, транслировать протоколы локальных сетей, например Ethernet в Token Ring, FDDI в Ethernet и т. п. Это свойство прозрачных мостов описано в стандарте IEEE 802.1H.

В дальнейшем будем называть устройство, которое продвигает кадры по алгоритму моста и работает в локальной сети, современным термином «коммутатор». При описании же самих алгоритмов 802.1D и Source Routing в следующем разделе будем по традиции называть устройство мостом, как собственно

оно в этих стандартах и называется.

4.3.2. Принципы работы мостов

Алгоритм работы прозрачного моста

Прозрачные мосты незаметны для сетевых адаптеров конечных узлов, так как они самостоятельно строят специальную адресную таблицу, на основании которой можно решить, нужно передавать пришедший кадр в какой-либо другой сегмент или нет. Сетевые адаптеры при использовании прозрачных мостов работают точно так же, как и в случае их отсутствия, то есть не предпринимают никаких дополнительных действий, чтобы кадр прошел через мост. Алгоритм прозрачного моста не зависит от технологии локальной сети, в которой устанавливается мост, поэтому прозрачные мосты Ethernet работают точно так же, как прозрачные мосты FDDI.

Прозрачный мост строит свою адресную таблицу на основании пассивного наблюдения за трафиком, циркулирующим в подключенных к его портам сегментах. При этом мост учитывает адреса источников кадров данных, поступающих на порты моста. По адресу источника кадра мост делает вывод о принадлежности этого узла тому или иному сегменту сети.

Рассмотрим процесс автоматического создания адресной таблицы моста и ее использования на примере простой сети, представленной на рис. 4.18.

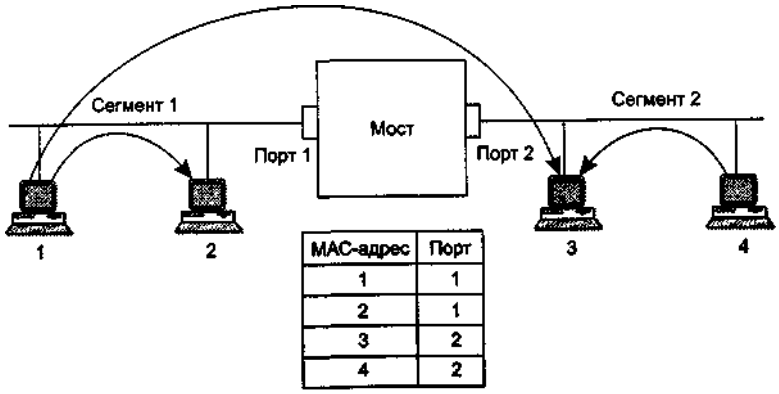


Рис. 4.18. Принцип работы прозрачного моста

Мост соединяет два логических сегмента. Сегмент 1 составляют компьютеры, подключенные с помощью одного отрезка коаксиального кабеля к порту 1 моста, а сегмент 2 – компьютеры, подключенные с помощью другого отрезка коаксиального кабеля к порту 2 моста.

Каждый порт моста работает как конечный узел своего сегмента за одним исключением – порт моста не имеет собственного MAC-адреса. Порт моста работает в так называемом *неразборчивом* (*promisquous*) режиме захвата пакетов, когда все поступающие на порт пакеты запоминаются в буферной памяти. С помощью такого режима мост следит за всем трафиком, передаваемым в присоединенных к нему сегментах, и использует проходящие через него пакеты для изучения состава сети. Так как в буфер записываются все пакеты, то адрес порта мосту не нужен.

В исходном состоянии мост ничего не знает о том, компьютеры с какими MAC-адресами подключены к каждому из его портов. Поэтому в этом случае мост просто передает любой захваченный и буферизованный кадр на все свои порты за исключением того, от которого этот кадр получен. В нашем примере у моста только два порта, поэтому он передает кадры с порта 1 на порт 2, и наоборот. Отличие работы моста в этом режиме от повторителя в том, что он передает кадр не побитно, а с буферизацией. Буферизация разрывает логику работы всех сегментов как единой разделяемой среды. Когда мост собирается передать кадр с сегмента на сегмент, например с сегмента 1 на сегмент 2, он заново пытается получить доступ к сегменту 2 как конечный узел по правилам алгоритма доступа, в данном примере – по правилам алгоритма CSMA/CD.

Одновременно с передачей кадра на все порты мост изучает адрес источника кадра и делает новую запись о его принадлежности в своей адресной таблице, которую также называют таблицей фильтрации или маршрутизации. Например, получив на свой порт 1 кадр от компьютера 1, мост делает первую запись в своей адресной таблице: MAC-адрес 1 – порт 1. Если все четыре компьютера данной сети

проявляют активность и посылают друг другу кадры, то скоро мост построит полную адресную таблицу сети, состоящую из 4 записей – по одной записи на узел.

После того как мост прошел этап обучения, он может работать более рационально. При получении кадра, направленного, например, от компьютера 1 компьютеру 3, он просматривает адресную таблицу на предмет совпадения ее адресов с адресом назначения 3. Поскольку такая запись есть, то мост выполняет второй этап анализа таблицы – проверяет, находятся ли компьютеры с адресами источника (в нашем случае – это адрес 1) и адресом назначения (адрес 3) в одном сегменте. Так как в нашем примере они находятся в разных сегментах, то мост выполняет операцию *продвижения* (*forwarding*) кадра – передает кадр на другой порт, предварительно получив доступ к другому сегменту.

Если бы оказалось, что компьютеры принадлежат одному сегменту, то кадр просто был бы удален из буфера и работа с ним на этом бы закончилась. Такая операция называется *фильтрацией* (*filtering*).

Если же адрес назначения неизвестен, то мост передает кадр на все свои порты, кроме порта – источника кадра, как и на начальной стадии процесса обучения.

На самом деле мы несколько упростили алгоритм работы моста. Его процесс обучения никогда не заканчивается. Мост постоянно следит за адресами источника буферизуемых кадров, чтобы быть в состоянии автоматически приспосабливаться к изменениям, происходящим в сети, – перемещениям компьютеров из одного сегмента сети в другой, появлению новых компьютеров. С другой стороны, мост не ждет, когда адресная таблица заполнится полностью (да это и невозможно, поскольку заранее не известно, сколько компьютеров и адресов будут находиться в сегментах моста). Как только в таблице появляется первый адрес, мост пытается его использовать, проверяя совпадение с ним адресов назначения всех поступающих пакетов.

Входы адресной таблицы могут быть динамическими, создаваемыми в процессе самообучения моста, и статическими, создаваемыми вручную администратором сети. Динамические входы имеют срок жизни – при создании или обновлении записи в адресной таблице с ней связывается отметка времени. По истечении определенного тайм-аута запись помечается как недействительная, если за это время мост не принял ни одного кадра с данным адресом в поле адреса источника. Это дает возможность автоматически реагировать на перемещения компьютера из сегмента в сегмент – при его отключении от старого сегмента запись о его принадлежности к нему со временем вычеркивается из адресной таблицы. После включения этого компьютера в работу в другом сегменте его кадры начнут попадать в буфер моста через другой порт, и в адресной таблице появится новая запись, соответствующая текущему состоянию сети.

Статические записи не имеют срока жизни, что дает администратору возможность подправлять работу моста, если это необходимо.

Кадры с широковещательными MAC-адресами передаются мостом на все его порты, как и кадры с неизвестным адресом назначения. Такой режим распространения кадров называется *затоплением сети* (*flood*). Наличие мостов в сети не препятствует распространению широковещательных кадров по всем сегментам сети, сохраняя ее прозрачность. Однако это является достоинством только в том случае, когда широковещательный адрес выработан корректно работающим узлом. Однако часто случается так, что в результате каких-либо программных или аппаратных сбоев протокол верхнего уровня или сам сетевой адаптер начинают работать некорректно и постоянно с высокой интенсивностью генерировать кадры с широковещательным адресом в течение длительного промежутка времени. Мост в этом случае передает эти кадры во все сегменты, затапливая сеть ошибочным трафиком. Такая ситуация называется *широковещательным штормом* (*broadcast storm*).

К сожалению, мосты не защищают сети от широковещательного шторма, во всяком случае, по умолчанию, как это делают маршрутизаторы. Максимум, что может сделать администратор с помощью моста для борьбы с широковещательным штормом – установить для каждого узла предельно допустимую интенсивность генерации кадров с широковещательным адресом. Но при этом нужно точно знать, какая интенсивность является нормальной, а какая – ошибочной. При смене протоколов ситуация в сети может измениться, и то, что вчера считалось ошибочным, сегодня может оказаться нормой. Таким образом, мосты располагают весьма грубыми средствами борьбы с широковещательным штормом.

На рис. 4.19 показана типичная структура моста. Функции доступа к среде при приеме и передаче кадров выполняют микросхемы MAC, которые идентичны микросхемам сетевого адаптера.

На рис. 4.20 показана копия экрана терминала с адресной таблицей модуля локального моста концентратора System 3000 компании SynOptics (сам концентратор уже не выпускается, но в свое время

он сыграл роль пионера в становлении многосегментных концентраторов Ethernet на витой паре, причем концентратор имел модуль моста, который мог соединять внутренние сегменты без привлечения внешнего моста). Терминал подключен к консольному порту, и информация на его экране высвечена модулем управления моста.

Из помещенной на экране адресной таблицы (Forwarding Table) видно, что сеть состоит из двух сегментов – LAN A и LAN B. В сегменте LAN A имеются, по крайней мере, 3 станция, а в сегменте LAN B – 2 станции. Четыре адреса, помеченные звездочками, являются статическими, то есть назначенными администратором вручную. Адрес, помеченный знаком «+», является динамическим адресом с истекшим сроком жизни.

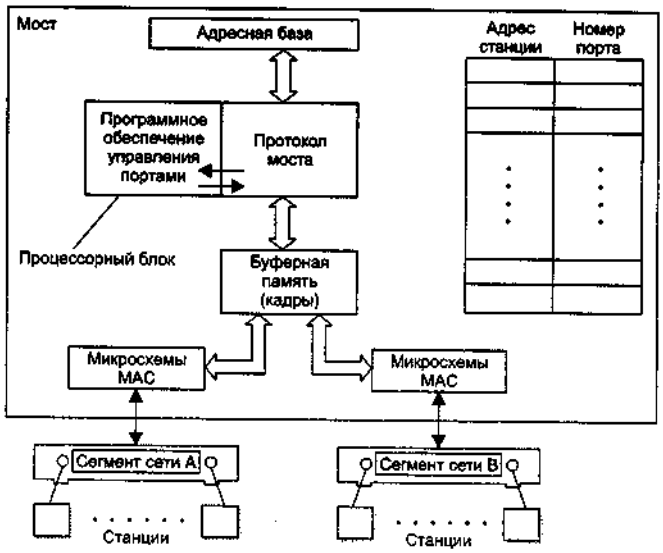


Рис. 4.19. Структура моста

Таблица имеет столбец «Dispn» – «Распоряжение», которое говорит мосту, какую операцию нужно проделать с кадром, имеющим данный адрес назначения. Обычно при автоматическом составлении таблицы в этом поле ставится условное обозначение порта назначения, но при ручном задании адреса в это поле можно внести нестандартную операцию обработки кадра. Например, операция «Flood» – «Затопление» заставляет мост распространять кадр в широковещательном режиме, несмотря на то что его адрес назначения не является широковещательным. Операция «Discard» – «Отбросить» говорит мосту, что кадр с таким адресом не нужно передавать на порт назначения.

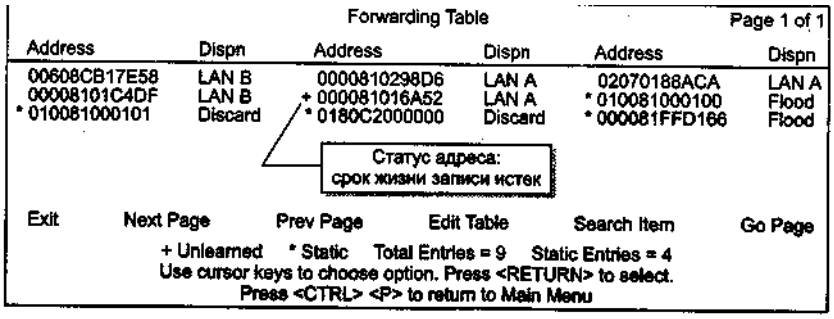


Рис. 4.20. Адресная таблица моста System 3000 Local Bridge

Собственно операции, задаваемые в поле «Dispn», являются особыми условиями фильтрации кадров, дополняющими стандартные условия распространения кадров. Такие условия обычно называют *пользовательскими фильтрами.*

Мосты с маршрутизацией от источника

Мосты с маршрутизацией от источника применяются для соединения колец Token Ring и FDDI, хотя для этих же целей могут использоваться и прозрачные мосты. Маршрутизация от источника (Source Routing, SR) основана на том, что станция-отправитель помещает в посылаемый в другое кольцо кадр

всю адресную информацию о промежуточных мостах и кольцах, которые должен пройти кадр перед тем, как попасть в кольцо, к которому подключена станция-получатель. Хотя в название этого способа входит термин «маршрутизация», настоящей маршрутизации в строгом понимании этого термина здесь нет, так как мосты и станции по-прежнему используют для передачи кадров данных только информацию MAC-уровня, а заголовки сетевого уровня для мостов данного типа по-прежнему остаются неразличимой частью поля данных кадра.

Рассмотрим принципы работы мостов Source Routing (в дальнейшем, SR-мосты) на примере сети, изображенной на рис. 4.21. Сеть состоит из трех колец, соединенных тремя мостами. Для задания маршрута кольца и мосты имеют идентификаторы. SR-мосты не строят адресную таблицу, а при продвижении кадров пользуются информацией, имеющейся в соответствующих полях кадра данных.

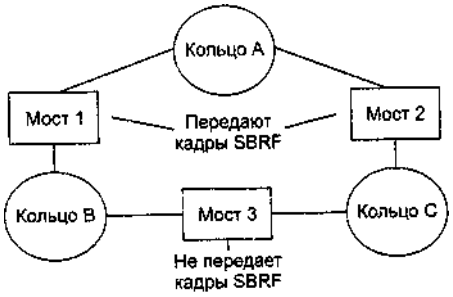


Рис. 4.21. Мосты типа Source Routing

При получении каждого пакета SR-мосту нужно только просмотреть поле маршрутной информации (поле Routing Information Field, RIF, в кадре Token Ring или FDDI) на предмет наличия в нем своего идентификатора. И если он там присутствует и сопровождается идентификатором кольца, которое подключено к данному мосту, то в этом случае мост копирует поступивший кадр в указанное кольцо. В противном случае кадр в другое кольцо не копируется. В любом случае исходная копия кадра возвращается по исходному кольцу станции-отправителю, и если он был передан в другое кольцо, то бит A (адрес распознан) и бит C (кадр скопирован) поля статуса кадра устанавливаются в 1, чтобы сообщить станции-отправителю, что кадр был получен станцией назначения (в данном случае передан мостом в другое кольцо).

Так как маршрутная информация в кадре нужна не всегда, а только для передачи кадра между станциями, подключенными к разным кольцам, то наличие в кадре поля RIF обозначается установкой в 1 бит индивидуального/группового адреса (I/G) (при этом данный бит используется не по назначению, так как адрес источника всегда индивидуальный).

Поле RIF имеет управляющее подполе, состоящее из трех частей.

- *Тип кадра* определяет тип поля RIF. Существуют различные типы полей RIF, использующиеся для нахождения маршрута и для отправки кадра по известному маршруту.
- *Поле максимальной длины кадра* используется мостом для связи колец, в которых установлено различное значение MTU. С помощью этого поля мост уведомляет станцию о максимально возможной длине кадра (то есть минимальном значении MTU на протяжении всего составного маршрута).
- *Длина поля RIF* необходима, так как заранее неизвестно количество описателей маршрута, задающих идентификаторы пересекаемых колец и мостов.

Для работы алгоритма маршрутизации от источника используются два дополнительных типа кадра – одномаршрутный широковещательный кадр-исследователь SRBF (single-route broadcast frame) и многомаршрутный широковещательный кадр-исследователь ARBF (all-route broadcast frame).

Все SR-мосты должны быть сконфигурированы администратором вручную, чтобы передавать кадры ARBF на все порты, кроме порта-источника кадра, а для кадров SRBF некоторые порты мостов нужно заблокировать, чтобы в сети не было петель. В примере сети на рис. 4.21 для исключения петли администратор заблокировал оба порта моста 3 для передачи кадров SRBF.

Кадр первого типа отправляется станцией, когда она, во-первых, определяет, что станция назначения находится в другом кольце, а во-вторых, ей неизвестно, через какие мосты и кольца пролегает путь к этой станции назначения, то есть неизвестен маршрут до этой станции. Первое обстоятельство выясняется, если кадр, отправленный по кольцу, возвращается в станцию-источник с

неустановленными признаками распознавания адреса и копирования. Значит, ни одна из станций исходного кольца не является станцией назначения, и кадр надо передавать по некоторому составному маршруту. Отсутствие маршрута к станции назначения в таблице моста является вторым обстоятельством, которое и вызывает отправку одномаршрутного кадра-исследователя SRBF.

В кадре SRBF станция задает длину поля RIF, равную нулю. Как и прозрачные мосты, SR-мосты работают в режиме «неразборчивого» захвата, буферизуя и анализируя все кадры. При получении кадра SRBF SR-мост передает его в исходном виде на все незаблокированные для этого типа кадров порты. Необходимость в конфигурировании топологии без петель для кадров-исследователей SRBF вызвана тем, что таким способом предотвращается возможность бесконечного зацикливания этих кадров.

В конце концов кадр-исследователь SRBF, распространяясь по всем кольцам сети, доходит до станции назначения. В ответ станция назначения отправляет многомаршрутный широковещательный кадр-исследователь ARBF станции-отправителю. В отличие от кадра SRBF этот кадр передается мостами через все порты. При приеме такого кадра каждый промежуточный мост добавляет в поле маршрутной информации RIF новый описатель маршрута (свой идентификатор и идентификатор сегмента, с которого получен кадр), наращивает длину поля маршрутной информации и широковещательно его распространяет.

Для предотвращения зацикливания кадров ARBF мосты обрабатывают их следующим образом. Перед передачей кадра на какой-либо сегмент мост проверяет, нет ли идентификатора этого сегмента в списке маршрутов кадра. Если такой сегмент уже был пройден кадром, то кадр в данный сегмент не направляется.

Станция-источник получает в общем случае несколько кадров-ответов, прошедших по всем возможным маршрутам составной сети, и выбирает наилучший маршрут (обычно по количеству пересечений промежуточных мостов). Именно для получения информации о всех возможных маршрутах кадр ARBF передается по всем возможным направлениям.

Затем маршрутная информация помещается в таблицу маршрутизации станции и используется для отправки кадров данных станции назначения по наилучшему маршруту за счет помещения последовательности номеров сетей и мостов в заголовке каждого такого кадра.

Мосты с маршрутизацией от источника имеют по сравнению с прозрачными мостам как преимущества, так и недостатки, отраженные в табл. 4.1.

Наличие двух возможных алгоритмов работы мостов – от источника и в прозрачном режиме – создает трудности для построения сложных сетей Token Ring. Мосты, работающие от источника, не могут поддерживать сегменты, рассчитанные на работу в прозрачном режиме, и наоборот.

До некоторого времени эта проблема решалась двумя способами. Один способ заключался в использовании во всех сегментах либо только маршрутизации от источника, либо только прозрачных мостов. Другим способом была установка маршрутизаторов. Сегодня имеется третье решение. Оно основано на стандарте, который позволяет объединить обе технологии работы моста в одном устройстве. Этот стандарт, называемый SRT (Source Route Transparent), позволяет мосту работать в любом режиме. Мост просматривает специальные флаги в заголовке кадров Token Ring и автоматически определяет, какой из алгоритмов нужно применить.

Таблица 4.1. Преимущества и недостатки мостов с маршрутизацией от источника

Преимущества	Недостатки
Более рациональные маршруты	Более дорогие сетевые адаптеры, принимающие участие в маршрутизации
Проще и дешевле – не нужно строить таблицы фильтрации	Сеть непрозрачна – кольца имеют номера
Более высокая скорость – не нужно просматривать таблицы фильтрации	Увеличивается трафик за счет широковещательных пакетов

Ограничения топологии сети, построенной на мостах

Слабая защита от широковещательного шторма – одно из главных ограничений моста, но не единственное. Другим серьезным ограничением их функциональных возможностей является невозможность поддержки петлеобразных конфигураций сети. Рассмотрим это ограничение на примере сети, изображенной на рис. 4.22.

Два сегмента параллельно соединены двумя мостами, так что образовалась активная петля. Пусть новая станция с адресом 10 впервые начинает работу в данной сети. Обычно начало работы любой операционной системы сопровождается рассылкой широковещательных кадров, в которых станция заявляет о своем существовании и одновременно ищет серверы сети.

На этапе 1 станция посылает первый кадр с широковещательным адресом назначения и адресом источника 10 в свой сегмент. Кадр попадает как в моет 1, так и в мост 2. В обоих мостах новый адрес источника 10 заносится в адресную таблицу с пометкой о его принадлежности сегменту 1, то есть создается новая запись вида:

MAC-адрес	Порт
10	1

Так как адрес назначения широковещательный, то каждый мост должен передать кадр на сегмент 2. Эта передача происходит поочередно, в соответствии с методом случайного доступа технологии Ethernet. Пусть первым доступ к сегменту 2 получил мост 1 (этап 2 на рис. 4.22). При появлении пакета на сегменте 2 мост 2 принимает его в свой буфер и обрабатывает. Он видит, что адрес 10 уже есть в его адресной таблице, но пришедший кадр является более свежим, и он утверждает, что адрес 10 принадлежит сегменту 2, а не 1. Поэтому мост 2 корректирует содержимое базы и делает запись о том, что адрес 10 принадлежит сегменту 2.

Теперь адресная таблица моста 2 будет иметь уже другую запись о станции с адресом 10:

MAC-адрес	Порт
10	1
10	2

Аналогично поступает мост 1, когда мост 2 передает свою копию кадра на сегмент 2.

Результаты наличия петли перечислены ниже.

- «Размножение» кадра, то есть появление нескольких его копий (в данном случае – двух, но если бы сегменты были соединены тремя мостами – то трех и т. д.).
- Бесконечная циркуляция обеих копий кадра по петле в противоположных направлениях, а значит, засорение сети ненужным трафиком.
- Постоянная перестройка мостами своих адресных таблиц, так как кадр с адресом источника 10 будет появляться то на одном порту, то на другом.

Чтобы исключить все эти нежелательные эффекты, мосты нужно применять так, чтобы между логическими сегментами не было петель, то есть строить с помощью мостов только древовидные структуры, гарантирующие наличие только одного пути между любыми двумя сегментами. Тогда кадры от каждой станции будут поступать в мост всегда с одного и того же порта, и мост сможет правильно решать задачу выбора рационального маршрута в сети.

Ограничение топологии структурированной сети древовидной структурой вытекает из самого принципа построения адресной таблицы мостом, а поэтому точно так же это ограничение действует и на коммутаторы.

В простых сетях сравнительно легко гарантировать существование одного и только одного пути между двумя сегментами. Но когда количество соединений возрастает и сеть становится сложной, то вероятность непреднамеренного образования петли оказывается высокой. Кроме того, желательно для повышения надежности иметь между мостами резервные связи, которые не участвуют при нормальной работе основных связей в передаче информационных пакетов станций, но при отказе какой-либо

основной связи образуют новую связную рабочую конфигурацию без петель.

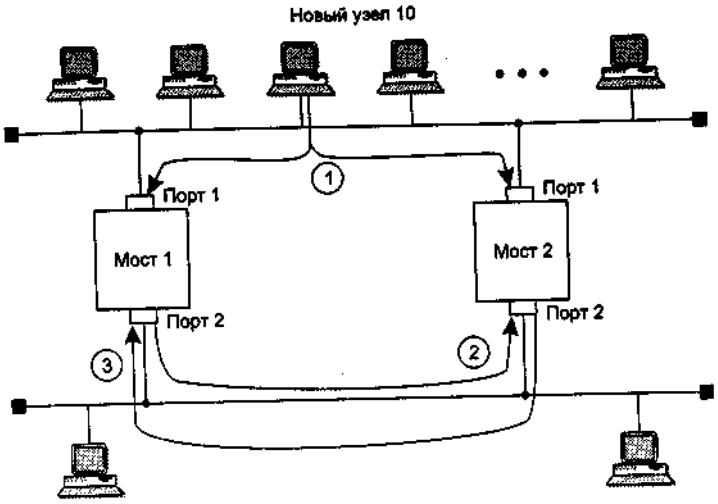


Рис. 4.22. Влияние замкнутых маршрутов на работу мостов

Поэтому в сложных сетях между логическими сегментами прокладывают избыточные связи, которые образуют петли, но для исключения активных петель блокируют некоторые порты мостов. Наиболее просто эта задача решается вручную, но существуют и алгоритмы, которые позволяют решать ее автоматически. Наиболее известным является стандартный алгоритм покрывающего дерева (Spanning Tree Algorithm, STA), который будет детально рассмотрен ниже. Кроме того, имеются фирменные алгоритмы, решающие ту же задачу, но с некоторыми улучшениями для конкретных моделей коммутаторов.

4.3.3. Коммутаторы локальных сетей

Технология коммутации сегментов Ethernet была предложена фирмой Kalpana в 1990 году в ответ на растущие потребности в повышении пропускной способности связей высокопроизводительных серверов с сегментами рабочих станций.

Структурная схема коммутатора EtherSwitch, предложенного фирмой Kalpana, представлена на рис. 4.23.

Каждый из 8 портов 10Base-T обслуживается одним процессором пакетов Ethernet – EPP (Ethernet Packet Processor). Кроме того, коммутатор имеет системный модуль, который координирует работу всех процессоров EPP. Системный модуль ведет общую адресную таблицу коммутатора и обеспечивает управление коммутатором по протоколу SNMP. Для передачи кадров между портами используется коммутационная матрица, подобная тем, которые работают в телефонных коммутаторах или мультипроцессорных компьютерах, соединяя несколько процессоров с несколькими модулями памяти.

Коммутационная матрица работает по принципу коммутации каналов. Для 8 портов матрица может обеспечить 8 одновременных внутренних каналов при полудуплексном режиме работы портов и 16 – при полнодуплексном, когда передатчик и приемник каждого порта работают независимо друг от друга.

При поступлении кадра в какой-либо порт процессор EPP буферизует несколько первых байт кадра, чтобы прочитать адрес назначения. После получения адреса назначения процессор сразу же принимает решение о передаче пакета, не дожидаясь прихода остальных байт кадра. Для этого он просматривает свой собственный кэш адресной таблицы, а если не находит там нужного адреса, обращается к системному модулю, который работает в многозадачном режиме, параллельно обслуживая запросы всех процессоров EPP. Системный модуль производит просмотр общей адресной таблицы и возвращает процессору найденную строку, которую тот буферизует в своем кэше для последующего использования.

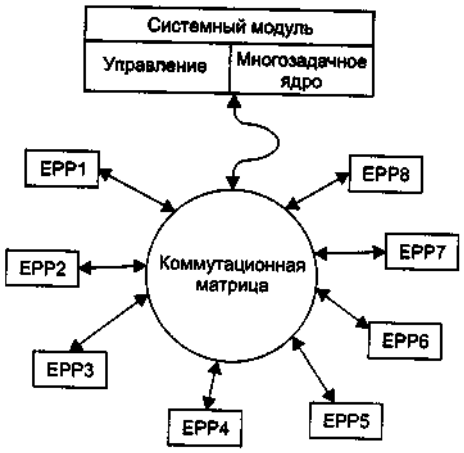


Рис. 4.23. Структура коммутатора EtherSwitch компании Kalpana

После нахождения адреса назначения процессор EPP знает, что нужно дальше делать с поступающим кадром (во время просмотра адресной таблицы процессор продолжал буферизацию поступающих в порт байтов кадра). Если кадр нужно отфильтровать, процессор просто прекращает записывать в буфер байты кадра, очищает буфер и ждет поступления нового кадра.

Если же кадр нужно передать на другой порт, то процессор обращается к коммутационной матрице и пытается установить в ней путь, связывающий его порт с портом, через который идет маршрут к адресу назначения. Коммутационная матрица может это сделать только в том случае, когда порт адреса назначения в этот момент свободен, то есть не соединен с другим портом.

Если же порт занят, то, как и в любом устройстве с коммутацией каналов, матрица в соединении отказывает. В этом случае кадр полностью буферизуется процессором входного порта, после чего процессор ожидает освобождения выходного порта и образования коммутационной матрицей нужного пути.

После того как нужный путь установлен, в него направляются буферизованные байты кадра, которые принимаются процессором выходного порта. Как только процессор выходного порта получает доступ к подключенному к нему сегменту Ethernet по алгоритму CSMA/CD, байты кадра сразу же начинают передаваться в сеть. Процессор входного порта постоянно хранит несколько байт принимаемого кадра в своем буфере, что позволяет ему независимо и асинхронно принимать и передавать байты кадра (рис. 4.24).

При свободном в момент приема кадра состоянии выходного порта задержка между приемом первого байта кадра коммутатором и появлением этого же байта на выходе порта адреса назначения составляла у коммутатора компании Kalpana всего 40 мкс, что было гораздо меньше задержки кадра при его передаче мостом.

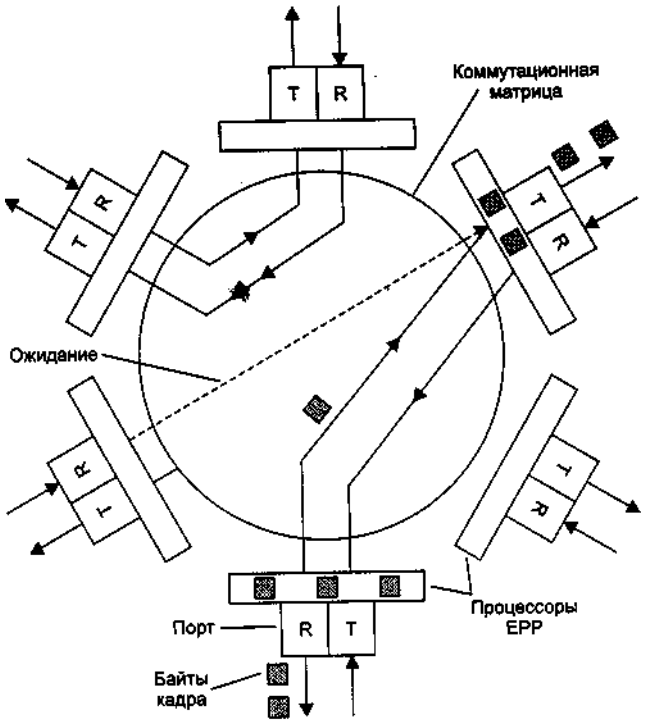


Рис. 4.24. Передача кадра через коммутационную матрицу

Описанный способ передачи кадра без его полной буферизации получил название коммутации «на лету» («on-the-fly») или «напролет» («cut-through»). Этот способ представляет, по сути, конвейерную обработку кадра, когда частично совмещаются во времени несколько этапов его передачи (рис. 4.25).

1. Прием первых байт кадра процессором входного порта, включая прием байт адреса назначения.
2. Поиск адреса назначения в адресной таблице коммутатора (в кэше процессора или в общей таблице системного модуля).
3. Коммутация матрицы.
4. Прием остальных байт кадра процессором входного порта.
5. Прием байт кадра (включая первые) процессором выходного порта через коммутационную матрицу.
6. Получение доступа к среде процессором выходного порта.
7. Передача байт кадра процессором выходного порта в сеть.

Этапы 2 и 3 совместить во времени нельзя, так как без знания номера выходного порта операция коммутации матрицы не имеет смысла.

По сравнению с режимом полной буферизации кадра, также приведенном на рис. 4,25, экономия от конвейеризации получается ощутимой.

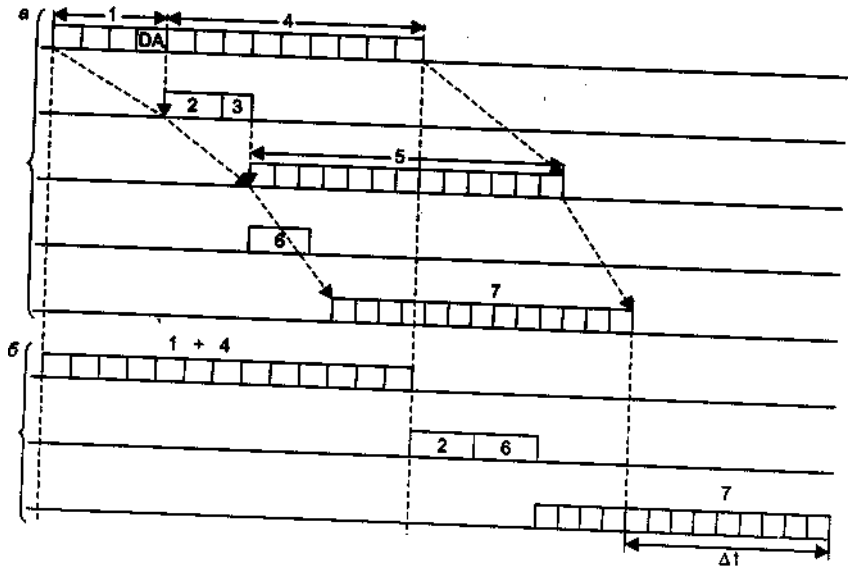


Рис. 4.25. Экономия времени при конвейерной обработке кадра: *а* – конвейерная обработка; *б* – обычная обработка с полной буферизацией

Однако главной причиной повышения производительности сети при использовании коммутатора является *параллельная* обработка нескольких кадров.

Этот эффект иллюстрирует рис. 4.26. На рисунке изображена идеальная в отношении повышения производительности ситуация, когда четыре порта из восьми передают данные с максимальной для протокола Ethernet скоростью 10 Мб/с, причем они передают эти данные на остальные четыре порта коммутатора не конфликтуя – потоки данных между узлами сети распределились так, что для каждого принимающего кадры порта есть свой выходной порт. Если коммутатор успевает обрабатывать входной трафик даже при максимальной интенсивности поступления кадров на входные порты, то общая производительность коммутатора в приведенном примере составит 4x10 = 40 Мбит/с, а при обобщении примера для N портов – (N/2)xlO Мбит/с. Говорят, что коммутатор предоставляет каждой станции или сегменту, подключенным к его портам, выделенную пропускную способность протокола.

Естественно, что в сети не всегда складывается такая ситуация, которая изображена на рис. 4.26. Если двум станциям, например станциям, подключенным к портам 3 и 4, одновременно нужно записывать данные на один и тот же сервер, подключенный к порту 8, то коммутатор не сможет выделить каждой станции поток данных по 10 Мбит/с, так как порт 8 не может передавать данные со скоростью 20 Мбит/с. Кадры станций будут ожидать во внутренних очередях входных портов 3 и 4, когда освободится порт 8 для передачи очередного кадра Очевидно, хорошим решением для такого распределения потоков данных было бы подключение сервера к более высокоскоростному порту, например Fast Ethernet.

Так как главное достоинство коммутатора, благодаря которому он завоевал очень хорошие позиции в локальных сетях, это его высокая производительность, то разработчики коммутаторов стараются выпускать так называемые *неблокирующие* (*non-blocking*) модели коммутаторов.

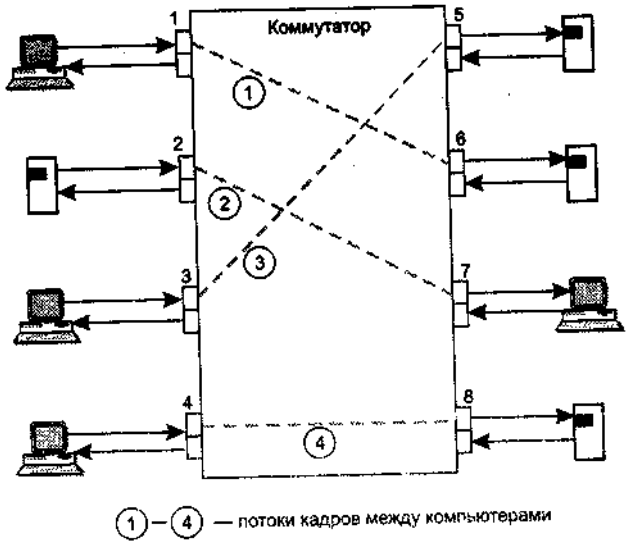


Рис. 4.26. Параллельная передача кадров коммутатором

Неблокирующий коммутатор – это такой коммутатор, который может передавать кадры через свои порты с той же скоростью, с которой они на них поступают. Естественно, что даже неблокирующий коммутатор не может разрешить в течение долгого промежутка времени ситуации, подобные описанной выше, когда блокировка кадров происходит из-за ограниченной скорости выходного порта,

Обычно имеют в виду устойчивый неблокирующий режим работы коммутатора, когда коммутатор передает кадры со скоростью их поступления в течение произвольного промежутка времени. Для обеспечения такого режима нужно, естественно, такое распределение потоков кадров по выходным портам, чтобы они справлялись с нагрузкой и коммутатор мог всегда в среднем передать на выходы столько кадров, сколько их поступило на входы. Если же входной поток кадров (просуммированный по всем портам) в среднем будет превышать выходной поток кадров (также просуммированный по всем портам), то кадры будут накапливаться в буферной памяти коммутатора, а при превышении ее объема – просто отбрасываться. Для обеспечения неблокирующего режима коммутатора необходимо выполнение достаточно простого условия:

$C_k = (\sum C_{pi})/2$,

где C_k – производительность коммутатора, C_{pi} – максимальная производительность протокола, поддерживаемого i-м портом коммутатора. Суммарная производительность портов учитывает каждый проходящий кадр дважды – как входящий кадр и как выходящий, а так как в устойчивом режиме входной трафик равен выходному, то минимально достаточная производительность коммутатора для поддержки неблокирующего режима равна половине суммарной производительности портов. Если порт работает в полудуплексном режиме, например Ethernet 10 Мбит/с, то производительность порта C_{pi} равна 10 Мбит/с, а если в полнодуплексном, то его Cpi будет составлять 20 Мбит/с.

Иногда говорят, что коммутатор поддерживает мгновенный неблокирующий режим. Это означает, что он может принимать и обрабатывать кадры от всех своих портов на максимальной скорости протоколов, независимо от того, обеспечиваются ли условия устойчивого равновесия между входным и выходным трафиком. Правда, обработка некоторых кадров при этом может быть неполной – при занятости выходного порта кадр помещается в буфер коммутатора. Для поддержки неблокирующего мгновенного режима коммутатор должен обладать большей собственной производительностью, а именно, она должна быть равна суммарной производительности его портов:

$C_k = \sum C_{pi}$.

Первый коммутатор для локальных сетей не случайно появился для технологии Ethernet. Кроме очевидной причины, связанной с наибольшей популярностью сетей Ethernet, существовала и другая, не менее важная причина – эта технология больше других страдает от повышения времени ожидания доступа к среде при повышении загрузки сегмента. Поэтому сегменты Ethernet в крупных сетях в первую очередь нуждались в средстве разгрузки узких мест сети, и этим средством стали коммутаторы фирмы Kalpana, а затем и других компаний.

Некоторые компании стали развивать технологию коммутации для повышения производительности других технологий локальных сетей, таких как Token Ring и FDDI. Эти коммутаторы поддерживали как алгоритм работы прозрачного моста, так и алгоритм моста с маршрутизацией от источника. Внутренняя

организация коммутаторов различных производителей иногда очень отличалась от структуры первого коммутатора EtherSwitch, однако принцип параллельной обработки кадров по каждому порту оставался неизменным.

Широкому применению коммутаторов, безусловно, способствовало то обстоятельство, что внедрение технологии коммутации не требовало замены установленного в сетях оборудования – сетевых адаптеров, концентраторов, кабельной системы. Порты коммутаторов работали в обычном полудуплексном режиме, поэтому к ним прозрачно можно было подключить как конечный узел, так и концентратор, организующий целый логический сегмент.

Так как коммутаторы и мосты прозрачны для протоколов сетевого уровня, то их появление в сети не оказало никакого влияния на маршрутизаторы сети, если они там имелись.

Удобство использования коммутатора состоит еще и в том, что это самообучающееся устройство и, если администратор не нагружает его дополнительными функциями, конфигурировать его не обязательно – нужно только правильно подключить разъемы кабелей к портам коммутатора, а дальше он будет работать самостоятельно и эффективно выполнять поставленную перед ним задачу повышения производительности сети.

4.3.4. Полнодуплексные протоколы локальных сетей

Изменения в работе MAC-уровня при полнодуплексной работе

Технология коммутации сама по себе не имеет непосредственного отношения к методу доступа к среде, который используется портами коммутатора. При подключении сегментов, представляющих собой разделяемую среду, порт коммутатора должен поддерживать полудуплексный режим, так как является одним из узлов этого сегмента.

Однако, когда к каждому порту коммутатора подключен не сегмент, а только один компьютер, причем по двум раздельным каналам, как это происходит почти во всех стандартах физического уровня, кроме коаксиальных версий Ethernet, ситуация становится не такой однозначной. Порт может работать как в обычном полудуплексном режиме, так и в полнодуплексном. Подключение к портам коммутатора не сегментов, а отдельных компьютеров называется *микросегментацией*.

В обычном режиме работы порт коммутатора по-прежнему распознает коллизии. Доменом коллизий в этом случае будет участок сети, включающий передатчик коммутатора, приемник коммутатора, передатчик сетевого адаптера компьютера, приемник сетевого адаптера компьютера и две витые пары, соединяющие передатчики с приемниками (рис. 4.27).

Коллизия возникает, когда передатчики порта коммутатора и сетевого адаптера одновременно или почти одновременно начинают передачу своих кадров, считая, что изображенный на рисунке сегмент свободен. Правда, вероятность коллизии в таком сегменте гораздо меньше, чем в сегменте, состоящем из 20-30 узлов, но она не нулевая. При этом максимальная производительность сегмента Ethernet в 14880 кадров в секунду при минимальной длине кадра делится между передатчиком порта коммутатора и передатчиком сетевого адаптера. Если считать, что она делится пополам, то каждому предоставляется возможность передавать примерно по 7440 кадров в секунду.

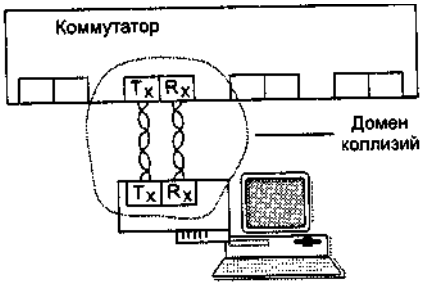


Рис. 4.27. Домен коллизий, образуемый компьютером и портом коммутатора

В полнодуплексном режиме одновременная передача данных передатчиком порта коммутатора и сетевого адаптера коллизией не считается. В принципе, это достаточно естественный режим работы для индивидуальных полнодуплексных каналов связи, и он часто используется в протоколах

территориальных сетей. При полнодуплексной связи порты Ethernet могут передавать данные со скоростью 20 Мбит/с – по 10 Мбит/с в каждом направлении.

Естественно, необходимо, чтобы MAC-узлы взаимодействующих устройств поддерживали этот специальный режим. В случае когда только один узел будет поддерживать полнодуплексный режим, второй узел будет постоянно фиксировать коллизии и приостанавливать свою работу, в то время как другой узел будет продолжать передавать данные, которые никто в этот момент не принимает. Изменения, которые нужно сделать в логике MAC-узла, чтобы он мог работать в полнодуплексном режиме, минимальны – нужно просто отменить фиксацию и отработку коллизий в сетях Ethernet, а в сетях Token Ring и FDDI – посылать кадры в коммутатор, не дожидаясь прихода токена доступа, а тогда, когда это нужно конечному узлу. Фактически, при работе в полнодуплексном режиме MAC-узел не использует метод доступа к среде, разработанный для данной технологии.

Так как переход на полнодуплексный режим работы требует изменения логики работы MAC-узлов и драйверов сетевых адаптеров, то он сначала был опробован при соединении двух коммутаторов. Уже первые модели коммутатора EtherSwitch компании Kalpana поддерживали полнодуплексный режим при взаимном соединении, обеспечивая скорость взаимного обмена 20 Мбит/с.

Позже появились версии полнодуплексного соединения FDDI-коммутаторов, которые при одновременном использовании двух колец FDDI обеспечивали скорость обмена в 200 Мбит/с.

Сейчас для каждой технологии можно найти модели коммутаторов, которые поддерживают полнодуплексный обмен при соединении коммутатор–коммутатор.

После опробования полнодуплексной технологии на соединениях коммутатор-коммутатор разработчики реализовали ее и в сетевых адаптерах, в основном адаптерах Ethernet и Fast Ethernet. При разработке технологий Fast Ethernet и Gigabit Ethernet полнодуплексный режим стал одним из двух полноправных стандартных режимов работы узлов сети. Многие сетевые адаптеры сейчас могут поддерживать оба режима работы, отрабатывая логику алгоритма доступа CSMA/CD при подключении к порту концентратора и работая в полнодуплексном режиме при подключении к порту коммутатора.

При использовании полнодуплексных версий протоколов происходит некоторое сближение различных технологий, так как метод доступа во многом определял лицо каждой технологии. Различие технологий остается в различных форматах кадров, а также в процедурах контроля корректности работы сети на физическом и канальном уровнях.

Полнодуплексные версии протоколов могли бы быть реализованы и в мостах. Принципиальных препятствий для этого не было, просто в период применения локальных мостов потребности в высокоскоростной передаче межсегментного трафика не возникало.

Проблема управления потоком данных при полнодуплексной работе

Простой отказ от поддержки алгоритма доступа к разделяемой среде без какой-либо модификации протокола ведет к повышению вероятности потерь кадров коммутаторами, так как при этом теряется контроль за потоками кадров, направляемых конечными узлами в сеть. Раньше поток кадров регулировался методом доступа к разделяемой среде, так что слишком часто генерирующий кадры узел вынужден был ждать своей очереди к среде и фактическая интенсивность потока данных, который направлял в сеть этот узел, была заметно меньше той интенсивности, которую узел хотел бы отправить в сеть. При переходе на полнодуплексный режим узлу разрешается отправлять кадры в коммутатор всегда, когда это ему нужно, поэтому коммутаторы сети могут в этом режиме сталкиваться с перегрузками, не имея при этом никаких средств регулирования («притормаживания») потока кадров.

Причина перегрузок обычно кроется не в том, что коммутатор является блокирующим, то есть ему не хватает производительности процессоров для обслуживания потоков кадров, а в ограниченной пропускной способности отдельного порта, которая определяется временными параметрами протокола. Например, порт Ethernet не может передавать больше 14 880 кадров в секунду, если он не нарушает временных соотношений, установленных стандартом.

Поэтому, если входной трафик неравномерно распределяется между выходными портами, легко представить ситуацию, когда в какой-либо выходной порт коммутатора будет направляться трафик с суммарной средней интенсивностью большей, чем протокольный максимум. На рис. 4.28 изображена как раз такая ситуация, когда в порт 3 коммутатора направляется трафик от портов 1,2,4 и 6, с суммарной интенсивностью в 22 100 кадров в секунду. Порт 3 оказывается загружен на 150 %. Естественно, что когда кадры поступают в буфер порта со скоростью 20 100 кадров в секунду, а уходят

со скоростью 14 880 кадров в секунду, то внутренний буфер выходного порта начинает неуклонно заполняться необработанными кадрами.

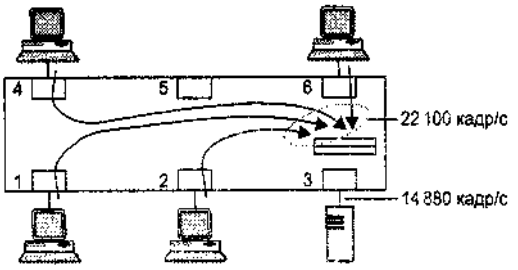


Рис. 4.28. Переполнение буфера порта из-за несбалансированности трафика

Какой бы ни был объем буфера порта, он в какой-то момент времени обязательно переполнится. Нетрудно подсчитать, что при размере буфера в 100 Кбайт в приведенном примере полное заполнение буфера произойдем через 0,22 секунды после начала его работы (буфер такого размера может хранить до 1600 кадров размером в 64 байт). Увеличение буфера до1 Мбайт даст увеличение времени заполнения буфера до 2,2 секунд, что также неприемлемо. А потери кадров всегда очень нежелательны, так как снижают полезную производительность сети, и коммутатор, теряющий кадры, может значительно ухудшить производительность сети вместо ее улучшения.

Коммутаторы локальных сетей – не первые устройства, которые сталкиваются с такой проблемой. Мосты также могут испытывать перегрузки, однако такие ситуации при использовании мостов встречались редко из-за небольшой интенсивности межсегментного трафика, поэтому разработчики мостов не стали встраивать в протоколы локальных сетей или в сами мосты механизмы регулирования потока В глобальных сетях коммутаторы технологии X.25 поддерживают протокол канального уровня LAP-B, который имеет специальные кадры управления потоком «Приемник готов» (RR) и «Приемник не готов» (RNR), аналогичные по назначению кадрам протокола LLC2 (это не удивительно, так как оба протокола принадлежат семейству протоколов HDLC. Протокол LAP-B работает между соседними коммутаторами сети X.25 и в том случае, когда очередь коммутатора доходит до опасной границы, запрещает своим ближайшим соседям с помощью кадра «Приемник не готов» передавать ему кадры, пока очередь не уменьшится до нормального уровня. В сетях X.25 такой протокол необходим, так как эти сети никогда не использовали разделяемые среды передачи данных, а работали по индивидуальным каналам связи в полнодуплексном режиме.

При разработке коммутаторов локальных сетей ситуация коренным образом отличалась от ситуации, при которой создавались коммутаторы территориальных сетей. Основной задачей было сохранение конечных узлов в неизменном виде, что исключало корректировку протоколов локальных сетей. А в этих протоколах процедур управления потоком не было – общая среда передачи данных в режиме разделения времени исключала возникновение ситуаций, когда сеть переполнялась бы необработанными кадрами. Сеть не накапливала данных в каких-либо промежуточных буферах при использовании только повторителей или концентраторов.

ПРИМЕЧАНИЕ Здесь речь идет о протоколах MAC-уровня (Ethernet, Token Ring и т. п), так как мосты и коммутаторы имеют дело только с ними. Протокол LLC2, который умеет управлять потоком данных, для целей управления потоком кадров в коммутаторах использовать нельзя. Для коммутаторов протокол LLC (все его процедуры: 1,2 и 3) прозрачен, как и все остальные протоколы верхних уровней, - коммутатор не анализирует заголовок LLC, считая его просто полем данных кадра MAC-уровня.

Применение коммутаторов без изменения протокола работы оборудования всегда порождает опасность потери кадров. Если порты коммутатора работают в обычном, то есть в полудуплексном режиме, то у коммутатора имеется возможность оказать некоторое воздействие на конечный узел и заставить его приостановить передачу кадров, пока у коммутатора не разгрузятся внутренние буферы. Нестандартные методы управления потоком в коммутаторах при сохранении протокола доступа в неизменном виде будут рассмотрены ниже.

Если же коммутатор работает в полнодуплексном режиме, то протокол работы конечных узлов, да и

его портов все равно меняется. Поэтому имело смысл для поддержки полнодуплексного режима работы коммутаторов несколько модифицировать протокол взаимодействия узлов, встроив в него явный механизм управления потоком кадров.

Работа над выработкой стандарта для управления потоком кадров в полнодуплексных версиях Ethernet и Fast Ethernet продолжалась несколько лет. Такой длительный период объясняется разногласиями членов соответствующих комитетов по стандартизации, отстаивающих подходы фирм, которые реализовали в своих коммутаторах собственные методы управления потоком.

В марте 1997 года принят стандарт IEEE 802.3x на управление потоком в полнодуплексных версиях протокола Ethernet. Он определяет весьма простую процедуру управления потоком, подобную той, которая используется в протоколах LLC2 и LAP-B. Эта процедура подразумевает две команды – «Приостановить передачу» и «Возобновить передачу», которые направляются соседнему узлу. Отличие от протоколов типа LLC2 в том, что эти команды реализуются на уровне символов кодов физического уровня, таких как 4B/5B, а не на уровне команд, оформленных в специальные управляющие кадры. Сетевой адаптер или порт коммутатора, поддерживающий стандарт 802.3x и получивший команду «Приостановить передачу», должен прекратить передавать кадры впредь до получения команды «Возобновить передачу».

Некоторые специалисты высказывают опасение, что такая простая процедура управления потоком окажется непригодной в сетях Gigabit Ethernet. Полная приостановка приема кадров от соседа при такой большой скорости передачи кадров (1 488 090 кадр/с) может быстро вызвать переполнение внутреннего буфера теперь у этого соседа, который в свою очередь полностью заблокирует прием кадров у своих ближайших соседей. Таким образом, перегрузка просто распространится по сети, вместо того чтобы постепенно исчезнуть. Для работы с такими скоростными протоколами необходим более тонкий механизм регулирования потока, который бы указывал, на какую величину нужно уменьшить интенсивность потока входящих кадров в перегруженный коммутатор, а не приостанавливал этот поток до нуля. Подобный плавный механизм регулирования потока появился у коммутаторов ATM через несколько лет после их появления. Поэтому существует мнение, что стандарт 802.3x – это временное решение, которое просто закрепило существующие фирменные простые механизмы управления потоком ведущих производителей коммутаторов. Пройдет некоторое время, и этот стандарт сменит другой стандарт – более сложный и более приспособленный для высокоскоростных технологий, таких как Gigabit Ethernet.

4.3.5. Управления потоком кадров при полудуплексной работе

При работе порта в полудуплексном режиме коммутатор не может изменять протокол и пользоваться для управления потоком новыми командами, такими как «Приостановить передачу» и «Возобновить передачу».

Зато у коммутатора появляется возможность воздействовать на конечный узел с помощью механизмов алгоритма доступа к среде, который конечный узел обязан отрабатывать. Эти приемы основаны на том, что конечные узлы строго соблюдают все параметры алгоритма доступа к среде, а порты коммутатора – нет. Обычно применяются два основных способа управления потоком кадров – обратное давление на конечный узел и агрессивный захват среды.

Метод обратного давления (*backpressure*) состоит в создании искусственных коллизий в сегменте, который чересчур интенсивно посылает кадры в коммутатор. Для этого коммутатор обычно использует jam-последовательность, отправляемую на выход порта, к которому подключен сегмент (или узел), чтобы приостановить его активность. Кроме того, метод обратного давления может применяться в тех случаях, когда процессор порта не рассчитан на поддержку максимально возможного для данного протокола трафика. Один из первых примеров применения метода обратного давления как раз связан с таким случаем – метод был применен компанией LANNET в модулях LSE-1 и LSE-2, рассчитанных на коммутацию трафика Ethernet с максимальной интенсивностью соответственно 1 Мбит/с и 2 Мбит/с.

Второй метод «торможения» конечного узла в условиях перегрузки внутренних буферов коммутатора основан на так называемом *агрессивном поведении порта коммутатора* при захвате среды либо после окончания передачи очередного пакета, либо после коллизии. Эти два случая иллюстрируются рис. 4.29, *а* и *б*.

В первом случае коммутатор окончил передачу очередного кадра и вместо технологической паузы в 9,6 мкс сделал паузу в 9,1 мкс и начал передачу нового кадра Компьютер не смог захватить среду, так

как он выдержал стандартную паузу в 9,6 мкс и обнаружил после этого, что среда уже занята

Во втором случае кадры коммутатора и компьютера столкнулись и была зафиксирована коллизия. Так как компьютер сделал паузу после коллизии в 51,2 мкс, как это положено по стандарту (интервал отсрочки равен 512 битовых интервалов), а коммутатор – 50 мкс, то и в этом случае компьютеру не удалось передать свой кадр.

Коммутатор может пользоваться этим механизмом адаптивно, увеличивая степень своей агрессивности по мере необходимости.

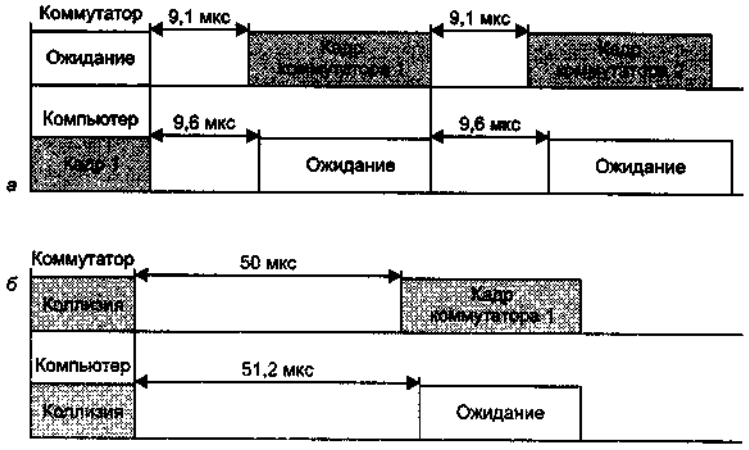


Рис. 4.29. Агрессивное поведение коммутатора при перегрузках буферов

Многие производители реализуют с помощью сочетания описанных двух методов достаточно тонкие механизмы управления потоком кадров при перегрузках. Эти методы используют алгоритмы чередования передаваемых и принимаемых кадров (frame interleave). Алгоритм чередования должен быть гибким и позволять компьютеру в критических ситуациях на каждый принимаемый кадр передавать несколько своих, разгружая внутренний буфер кадров, причем не обязательно снижая при этом интенсивность приема кадров до нуля, а просто уменьшая ее до необходимого уровня.

Практически во всех моделях коммутаторов, кроме самых простых моделей для рабочих групп, реализуют тот или иной алгоритм управления потоком кадров при полудуплексном режиме работы портов. Этот алгоритм, как правило, реализует более тонкое управление потоком, чем стандарт 802.3x, не приостанавливая до нуля прием кадров от соседнего узла и тем самым не способствуя переносу перегрузки в соседний коммутатор, если к порту подключен не конечный узел, а другой коммутатор.

Выводы

- Логическая структуризация сети необходима при построении сетей средних и крупных размеров. Использование общей разделяемой среды приемлемо только для сети, состоящей из 5-10 компьютеров.
- Деление сети на логические сегменты повышает производительность, надежность, гибкость построения и управляемость сети.
- Для логической структуризации сети применяются мосты и их современные преемники – коммутаторы и маршрутизаторы. Первые два типа устройств позволяют разделить сеть на логические сегменты с помощью минимума средств – только на основе протоколов канального уровня. Кроме того, эти устройства не требуют конфигурирования.
- Логические сегменты, построенные на основе коммутаторов, являются строительными элементами более крупных сетей, объединяемых маршрутизаторами.
- Коммутаторы – наиболее быстродействующие современные коммуникационные устройства, они позволяют соединять высокоскоростные сегменты без блокирования (уменьшения пропускной способности) межсегментного трафика.
- Пассивный способ построения адресной таблицы коммутаторами – с помощью слежения за проходящим трафиком – приводит к невозможности работы в сетях с петлевидными связями. Другим недостатком сетей, построенных на коммутаторах, является отсутствие защиты от широковещательного шторма, который эти устройства обязаны передавать в соответствии с

алгоритмом работы.

- Применение коммутаторов позволяет сетевым адаптерам использовать полнодуплексный режим работы протоколов локальных сетей (Ethernet, Fast Ethernet, Gigabit Ethernet, Token Ring, FDDI). В этом режиме отсутствует этап доступа к разделяемой среде, а общая скорость передачи данных удваивается.
- В полнодуплексном режиме для борьбы с перегрузками коммутаторов используется метод управления потоком, описанный в стандарте 802.3x. Он повторяет алгоритмы полной приостановки трафика по специальной команде, известной из технологий глобальных сетей.
- При полудуплексном режиме работы коммутаторы используют для управления потоком при перегрузках два метода: агрессивный захват среды и обратное давление на конечный узел. Применение этих методов позволяет достаточно гибко управлять потоком, чередуя несколько передаваемых кадров с одним принимаемым.

4.4. Техническая реализация и дополнительные функции коммутаторов

Несмотря на то что в коммутаторах работают известные и хорошо отработанные алгоритмы прозрачных мостов и мостов с маршрутизацией от источника, существует большое разнообразие моделей коммутаторов. Они отличаются как внутренней организацией, так и набором выполняемых дополнительных функций, таких как трансляция протоколов, поддержка алгоритма покрывающего дерева, образование виртуальных логических сетей и ряда других.

4.4.1. Особенности технической реализации коммутаторов

После того как технология коммутации привлекла общее внимание и получила высокие оценки специалистов, многие компании занялись реализацией этой технологии в своих устройствах, применяя для этого различные технические решения. Многие коммутаторы первого поколения были похожи на маршрутизаторы, то есть основывались на центральном процессоре общего назначения, связанном с интерфейсными портами по внутренней скоростной шине (рис. 4.30). Однако это были скорее пробные устройства, предназначенные для освоения самой компанией технологии коммутации, а не для завоевания рынка.

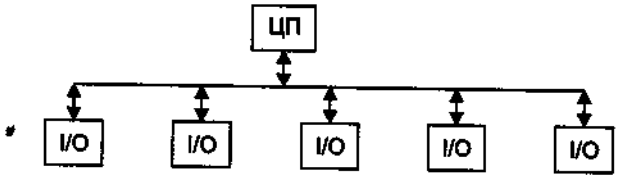


Рис. 4.30. Коммутатор на процессоре общего назначения

Основным недостатком таких коммутаторов была их низкая скорость. Универсальный процессор никак не мог справиться с большим объемом специализированных операций по пересылке кадров между интерфейсными модулями.

Для ускорения операций коммутации нужны были специализированные процессоры со специализированными средствами обмена данными, как в первом коммутаторе Kalpana, и они вскоре появились. Сегодня все коммутаторы используют заказные специализированные БИС – ASIC, которые оптимизированы для выполнения основных операций коммутации. Часто в одном коммутаторе используется несколько специализированных БИС, каждая из которых выполняет функционально законченную часть операций. Сравнительно низкая стоимость современных коммутаторов по сравнению с их предшественниками 3-5-летней давности объясняется массовым характером производства основных БИС, на которых каждая компания строит свои коммутаторы.

Кроме процессорных микросхем для успешной неблокирующей работы коммутатору нужно также иметь быстродействующий узел для передачи кадров между процессорными микросхемами портов.

В настоящее время коммутаторы используют в качестве базовой одну из трех схем, на которой строится такой узел обмена:

- коммутационная матрица;
- разделяемая многовходовая память;

- общая шина.

Часто эти три способа взаимодействия комбинируются в одном коммутаторе.

Коммутаторы на основе коммутационной матрицы

Коммутационная матрица обеспечивает основной и самый быстрый способ взаимодействия процессоров портов, именно он был реализован в первом промышленном коммутаторе локальных сетей. Однако реализация матрицы возможна только для определенного числа портов, причем сложность схемы возрастает пропорционально квадрату количества портов коммутатора (рис. 4.31).

Более детальное представление одного из возможных вариантов реализации коммутационной матрицы для 8 портов дано на рис. 4.32. Входные блоки процессоров портов на основании просмотра адресной таблицы коммутатора определяют по адресу назначения номер выходного порта. Эту информацию они добавляют к байтам исходного кадра в виде специального ярлыка – тэга (tag). Для данного примера тэг представляет собой просто 3-разрядное двоичное число, соответствующее номеру выходного порта.

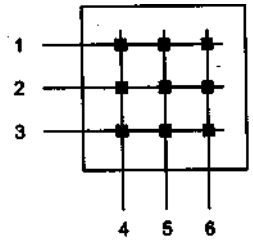


Рис. 4.31. Коммутационная матрица

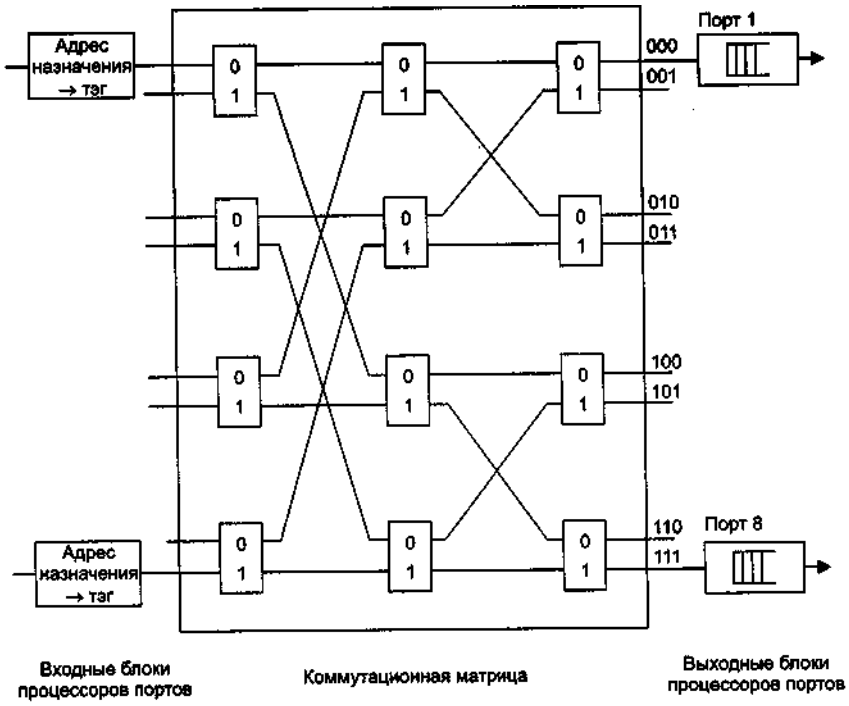


Рис. 4.32. Реализация коммутационной матрицы 8×8 с помощью двоичных переключателей

Матрица состоит из трех уровней двоичных переключателей, которые соединяют свой вход с одним из двух выходов в зависимости от значения бита тэга. Переключатели первого уровня управляются первым битом тэга, второго – вторым, а третьего – третьим.

Матрица может быть реализована и по-другому, на основании комбинационных схем другого типа, но ее особенностью все равно остается технология коммутации физических каналов. Известным недостатком этой технологии является отсутствие буферизации данных внутри коммутационной матрицы – если составной канал невозможно построить из-за занятости выходного порта или промежуточного коммутационного элемента, то данные должны накапливаться в их источнике, в данном случае – во входном блоке порта, принявшего кадр. Основные достоинства таких матриц –

высокая скорость коммутации и регулярная структура, которую удобно реализовывать в интегральных микросхемах. Зато после реализации матрицы NxN в составе БИС проявляется еще один ее недостаток – сложность наращивания числа коммутируемых портов.

Коммутаторы с общей шиной

В коммутаторах с общей шиной процессоры портов связывают высокоскоростной шиной, используемой в режиме разделения времени.

Пример такой архитектуры приведен на рис. 4.33. Чтобы шина не блокировала работу коммутатора, ее производительность должна равняться, по крайней мере, сумме производительности всех портов коммутатора. Для модульных коммутаторов некоторые сочетания модулей с низкоскоростными портами могут приводить к неблокирующей работе, а установка модулей с высокоскоростными портами может приводить к тому, что блокирующим элементом станет, например, общая шина.

Кадр должен передаваться по шине небольшими частями, по нескольку байт, чтобы передача кадров между несколькими портами происходила в псевдопараллельном режиме, не внося задержек в передачу кадра в целом. Размер такой ячейки данных определяется производителем коммутатора. Некоторые производители, например LANNET или Centillion, выбрали в качестве порции данных, переносимых за одну операцию по шине, ячейку ATM с ее полем данных в 48 байт. Такой подход облегчает трансляцию протоколов локальных сетей в протокол ATM, если коммутатор поддерживает эти технологии.

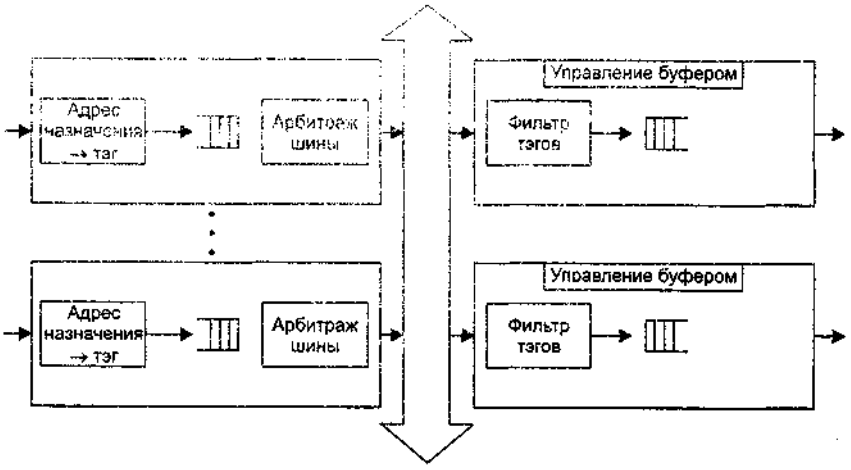


Рис. 4.33. Архитектура коммутатора с общей шиной

Входной блок процессора помещает в ячейку, переносимую по шине, тэг, в котором указывает номер порта назначения. Каждый выходной блок процессора порта содержит фильтр тэгов, который выбирает тэги, предназначенные данному порту.

Шина, так же как и коммутационная матрица, не может осуществлять промежуточную буферизацию, но так как данные кадра разбиваются на небольшие ячейки, то задержек с начальным ожиданием доступности выходного порта в такой схеме нет – здесь работает принцип коммутации пакетов, а не каналов.

Коммутаторы с разделяемой памятью

Третья базовая архитектура взаимодействия портов – двухвходовая разделяемая память. Пример такой архитектуры приведен на рис. 4.34.

Входные блоки процессоров портов соединяются с переключаемым входом разделяемой памяти, а выходные блоки этих же процессоров соединяются с переключаемым выходом этой памяти. Переключением входа и выхода разделяемой памяти управляет менеджер очередей выходных портов. В разделяемой памяти менеджер организует несколько очередей данных, по одной для каждого выходного порта. Входные блоки процессоров передают менеджеру портов запросы на запись данных в очередь того порта, который соответствует адресу назначения пакета. Менеджер по очереди подключает вход памяти к одному из входных блоков процессоров и тот переписывает часть данных кадра в очередь определенного выходного порта. По мере заполнения очередей менеджер производит

также поочередное подключение выхода разделяемой памяти к выходным блокам процессоров портов, и данные из очереди переписываются в выходной буфер процессора.

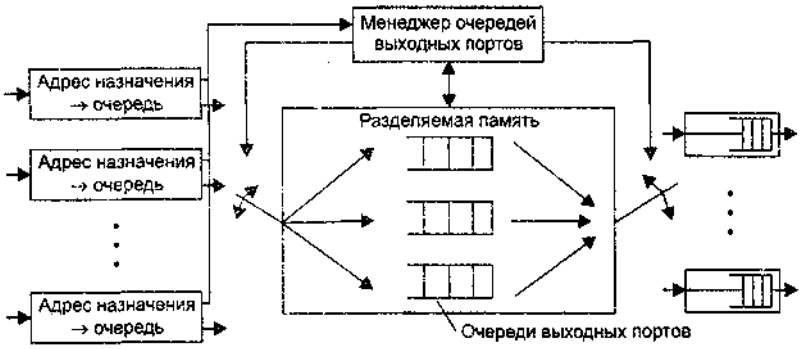


Рис. 4.34. Архитектура разделяемой памяти

Память должна быть достаточно быстродействующей для поддержания скорости переписи данных между N портами коммутатора. Применение общей буферной памяти, гибко распределяемой менеджером между отдельными портами, снижает требования к размеру буферной памяти процессора порта.

Комбинированные коммутаторы

У каждой из описанных архитектур есть свои преимущества и недостатки, поэтому часто в сложных коммутаторах эти архитектуры применяются в комбинации друг с другом. Пример такого комбинирования приведен на рис. 4.35.

Коммутатор состоит из модулей с фиксированным количеством портов (2-12), выполненных на основе специализированной БИС, реализующей архитектуру коммутационной матрицы. Если порты, между которыми нужно передать кадр данных, принадлежат одному модулю, то передача кадра осуществляется процессорами модуля на основе имеющейся в модуле коммутационной матрицы. Если же порты принадлежат разным модулям, то процессоры общаются по общей шине. При такой архитектуре передача кадров внутри модуля будет происходить быстрее, чем при межмодульной передаче, так как коммутационная матрица – наиболее быстрый, хотя и наименее масштабируемый способ взаимодействия портов. Скорость внутренней шины коммутаторов может достигать нескольких Гбит/с, а у наиболее мощных моделей – до 20-30 Гбит/с.

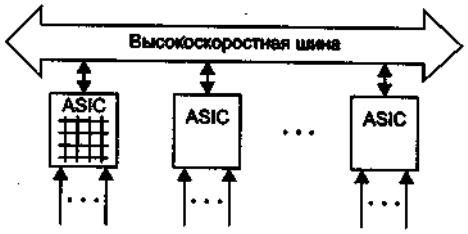


Рис. 4.35. Комбинирование архитектур коммутационной матрицы и общей шины

Можно представить и другие способы комбинирования архитектур, например использование разделяемой памяти для взаимодействия модулей.

Конструктивное исполнение коммутаторов

В конструктивном отношении коммутаторы делятся на следующие типы:

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

Первый тип коммутаторов обычно предназначен для организации небольших рабочих групп.

Модульные коммутаторы на основе шасси чаще всего предназначены для применения на магистрали сети. Поэтому они выполняются на основе какой-либо комбинированной схемы, в которой

взаимодействие модулей организуется по быстродействующей шине или же на основе быстрой разделяемой памяти большого объема. Модули такого коммутатора выполняются на основе технологии «hot swap», то есть допускают замену на ходу, без выключения коммутатора, так как центральное коммуникационное устройство сети не должно иметь перерывов в работе. Шасси обычно снабжается резервированными источниками питания и резервированными вентиляторами в тех же целях.

С технической точки зрения определенный интерес представляют стековые коммутаторы. Эти устройства представляют собой коммутаторы, которые могут работать автономно, так как выполнены в отдельном корпусе, но имеют специальные интерфейсы, которые позволяют их объединять в общую систему, работающую как единый коммутатор. Говорят, что в этом случае отдельные коммутаторы образуют стек.

Обычно такой специальный интерфейс представляет собой высокоскоростную шину, которая позволяет объединить отдельные корпуса подобно модулям в коммутаторе на основе шасси. Так как расстояния между корпусами больше, чем между модулями на шасси, скорость обмена по шине обычно ниже, чем у модульных коммутаторов: 200-400 Мбит/с. Не очень высокие скорости обмена между коммутаторами стека обусловлены также тем, что стековые коммутаторы обычно занимают промежуточное положение между коммутаторами с фиксированным количеством портов и коммутаторами на основе шасси. Стековые коммутаторы применяются для создания сетей рабочих групп и отделов, поэтому сверхвысокие скорости шин обмена им не очень нужны и не соответствуют их ценовому диапазону.

Структура стека коммутаторов, соединяемых по скоростным специальным портам, показана на рис. 4.36.

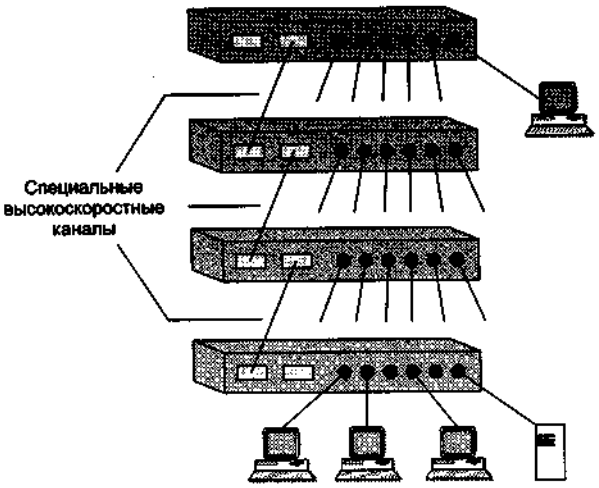


Рис. 4.36. Стек коммутаторов, объединяемых по высокоскоростным каналам

Компания Cisco предложила другой подход к организации стека. Ее коммутатор Catalyst 3000 также имеет специальный скоростной интерфейс 280 Мбит/с для организации стека, но с его помощью коммутаторы соединяются не друг с другом, а с отдельным устройством, содержащим коммутационную матрицу 8x8, организующую более высокопроизводительный обмен между любыми парами коммутаторов.

Существуют коммутаторы, которые позволяют объединить два коммутатора полнодуплексным каналом более чем по одной паре портов. Например, коммутаторы модели 28115 компании Nortel Networks имеют по два порта Fast Ethernet, с помощью которых можно соединять коммутаторы, образуя полнодуплексный канал с производительностью 400 Мбит/с (рис. 4.37).

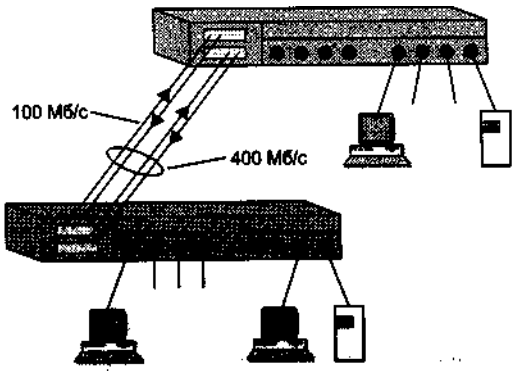


Рис. 4.37. Транковое полнодуплексное соединение коммутаторов 2815 компании Nortel Networks

Такие соединения называются *транковыми* и являются частной разработкой каждой компании, выпускающей коммуникационное оборудование, так как нарушают не только логику доступа к разделяемым средам, но и топологию соединения мостов, запрещающую петлевидные контуры (а такой контур всегда образуется при соединении коммутаторов более чем одной парой портов). При соединении коммутаторов разных производителей транк работать не будет, так как каждый производитель добавляет к логике изучения адресов сети коммутатором по транковой связи что-то свое, чтобы добиться от него правильной работы.

4.4.2. Характеристики, влияющие на производительность коммутаторов

Производительность коммутатора - то свойство, которое сетевые интеграторы и администраторы ждут от этого устройства в первую очередь.

Основными показателями коммутатора, характеризующими его производительность, являются:

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

Кроме того, существует несколько характеристик коммутатора, которые в наибольшей степени влияют на указанные характеристики производительности. К ним относятся.

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

Скорость фильтрации и скорость продвижения

Скорость фильтрации и продвижения кадров - это две основные характеристики производительности коммутатора. Эти характеристики являются интегральными показателями, они не зависят от того, каким образом технически реализован коммутатор.

Скорость фильтрации (*filtering*) определяет скорость, с которой коммутатор выполняет следующие этапы обработки кадров:

- прием кадра в свой буфер;
- просмотр адресной таблицы с целью нахождения порта для адреса назначения кадра;
- уничтожение кадра, так как его порт назначения и порт источника принадлежат одному логическому сегменту.

Скорость фильтрации практически у всех коммутаторов является неблокирующей – коммутатор успевает отбрасывать кадры в темпе их поступления.

Скорость продвижения (*forwarding*) определяет скорость, с которой коммутатор выполняет следующие этапы обработки кадров.

- прием кадра в свой буфер;

- просмотр адресной таблицы с целью нахождения порта для адреса назначения кадра;
- передача кадра в сеть через найденный по адресной таблице порт назначения.

Как скорость фильтрации, так и скорость продвижения измеряются обычно в кадрах в секунду. Если в характеристиках коммутатора не уточняется, для какого протокола и для какого размера кадра приведены значения скоростей фильтрации и продвижения, то по умолчанию считается, что эти показатели даются для протокола Ethernet и кадров минимального размера, то есть кадров длиной 64 байт (без преамбулы) с полем данных в 46 байт. Если скорости указаны для какого-либо определенного протокола, например Token Ring или FDDI, то они также даны для кадров минимальной длины этого протокола (например, кадров длины 29 байт для протокола FDDI). Применение в качестве основного показателя скорости работы коммутатора кадров минимальной длины объясняется тем, что такие кадры всегда создают для коммутатора наиболее тяжелый режим работы по сравнению с кадрами другого формата при равной пропускной способности переносимых пользовательских данных. Поэтому при проведении тестирования коммутатора режим передачи кадров минимальной длины используется как наиболее сложный тест, который должен проверить способность коммутатора работать при наихудшем сочетании параметров трафика. Кроме того, для пакетов минимальной длины скорость фильтрации и продвижения максимальна, что имеет немаловажное значение при рекламе коммутатора.

Пропускная способность коммутатора измеряется количеством пользовательских данных (в мегабитах в секунду), переданных в единицу времени через его порты. Так как коммутатор работает на канальном уровне, для него пользовательскими данными являются те данные, которые переносятся в поле данных кадров протоколов канального уровня – Ethernet, Token Ring, FDDI и т. п. Максимальное значение пропускной способности коммутатора всегда достигается на кадрах максимальной длины, так как при этом доля накладных расходов на служебную информацию кадра гораздо ниже, чем для кадров минимальной длины, а время выполнения коммутатором операций по обработке кадра, приходящееся на один байт пользовательской информации, существенно меньше. Поэтому коммутатор может быть блокирующим для кадров минимальной длины, но при этом иметь очень хорошие показатели пропускной способности.

Задержка передачи кадра измеряется как время, прошедшее с момента прихода первого байта кадра на входной порт коммутатора до момента появления этого байта на его выходном порту. Задержка складывается из времени, затрачиваемого на буферизацию байт кадра, а также времени, затрачиваемого на обработку кадра коммутатором, – просмотра адресной таблицы, принятия решения о фильтрации или продвижении и получения доступа к среде выходного порта.

Величина вносимой коммутатором задержки зависит от режима его работы. Если коммутация осуществляется «на лету», то задержки обычно невелики и составляют от 5 до 40 мкс, а при полной буферизации кадров – от 50 до 200 мкс (для кадров минимальной длины).

Коммутатор – это многопортовое устройство, поэтому для него принято все приведенные выше характеристики (кроме задержки передачи кадра) давать в двух вариантах. Первый вариант - суммарная производительность коммутатор при одновременной передаче трафика по всем его портам, второй вариант – производительность, приведенная в расчете на один порт. Обычно производители коммутаторов указывают общую максимальную пропускную способность устройства.

Коммутация «на лету» или с буферизацией

На производительности коммутатора сказывается способ передачи пакетов – «на лету» или с буферизацией. Коммутаторы, передающие пакеты «на лету», вносят меньшие задержки передачи кадров на каждом промежуточном коммутаторе, поэтому общее уменьшение задержки доставки данных может быть значительным, что важно для мультимедийного трафика. Кроме того, выбранный способ коммутации оказывает влияние на возможности реализации некоторых полезных дополнительных функций, например трансляцию протоколов канального уровня.

В табл. 4.2 дается сравнение возможностей двух способов коммутации.

Таблица 4.2. Возможности коммутаторов при коммутации «на лету» и с полной буферизацией

Функция	На лету	С буферизацией

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
Функция анализа трафика	Нет	Да

Средняя величина задержки коммутаторов, работающих «на лету», при высокой нагрузке объясняется тем, что в этом случае выходной порт часто бывает занят приемом другого пакета, поэтому вновь поступивший пакет для данного порта все равно приходится буферизовать.

Коммутатор, работающий «на лету», может выполнять проверку некорректности передаваемых кадров, но не может изъять плохой кадр из сети, так как часть его байт (и, как правило, большая часть) уже переданы в сеть.

Так как каждый способ имеет свои достоинства и недостатки, в тех моделях коммутаторов, которым нужно транслировать протоколы, иногда применяется механизм адаптивной смены режима работы коммутатора. Основной режим такого коммутатора – коммутация «на лету», но коммутатор постоянно контролирует трафик и при превышении интенсивности появления плохих кадров некоторого порога переходит на режим полной буферизации. Затем коммутатор может вернуться к коммутации «на лету».

Размер адресной таблицы

Максимальная емкость адресной таблицы определяет предельное количество MAC-адресов, с которыми может одновременно оперировать коммутатор, Так как коммутаторы чаще всего используют для выполнения операций каждого порта выделенный процессорный блок со своей памятью для хранения экземпляра адресной таблицы, то размер адресной таблицы для коммутаторов обычно приводится в расчете на один порт. Экземпляры адресной таблицы разных процессорных модулей не обязательно содержат одну и ту же адресную информацию – скорее всего, повторяющихся адресов будет не так много, если только распределение трафика каждого порта между остальными портами не полностью равновероятно. Каждый порт хранит только те наборы адресов, с которыми он работал в последнее время.

Значение максимального числа MAC-адресов, которое может запомнить процессор порта, зависит от области применения коммутатора. Коммутаторы рабочих групп обычно поддерживают всего несколько адресов на порт, так как они предназначены для образования микросегментов. Коммутаторы отделов должны поддерживать несколько сотен адресов, а коммутаторы магистралей сетей – до нескольких тысяч, обычно 4000-8000 адресов.

Недостаточная емкость адресной таблицы может служить причиной замедления работы коммутатора и засорения сети избыточным трафиком. Если адресная таблица процессора порта полностью заполнена, а он встречает новый адрес источника в поступившем пакете, процессор должен вытеснить из таблицы какой-либо старый адрес и поместить на его место новый. Эта операция сама по себе отнимет у процессора часть времени, но главные потери производительности будут наблюдаться при поступлении кадра с адресом назначения, который пришлось удалить из адресной таблицы. Так как адрес назначения кадра неизвестен, то коммутатор должен передать этот кадр на все остальные порты. Эта операция будет создавать лишнюю работу для многих процессоров портов, кроме того, копии этого кадра будут попадать и на те сегменты сети, где они совсем не обязательны.

Некоторые производители коммутаторов решают эту проблему за счет изменения алгоритма обработки кадров с неизвестным адресом назначения. Один из портов коммутатора конфигурируется как магистральный порт, на который по умолчанию передаются все кадры с неизвестным адресом. В маршрутизаторах такой прием применяется давно, позволяя сократить размеры адресных таблиц в

сетях, организованных по иерархическому принципу.

Передача кадра на магистральный порт производится в расчете на то, что этот порт подключен к вышестоящему коммутатору при иерархическом соединении коммутаторов в крупной сети, который имеет достаточную емкость адресной таблицы и знает, куда нужно передать любой кадр.

Объем буфера кадров

Внутренняя буферная память коммутатора нужна для временного хранения кадров данных в тех случаях, когда их невозможно немедленно передать на выходной порт. Буфер предназначен для сглаживания кратковременных пульсаций трафика. Ведь даже если трафик хорошо сбалансирован и производительность процессоров портов, а также других обрабатывающих элементов коммутатора достаточна для передачи средних значений трафика, это не гарантирует, что их производительности хватит при пиковых значениях нагрузок. Например, трафик может в течение нескольких десятков миллисекунд поступать одновременно на все входы коммутатора, не давая ему возможности передавать принимаемые кадры на выходные порты

Для предотвращения потерь кадров при кратковременном многократном превышении среднего значения интенсивности графика (а для локальных сетей часто встречаются значения коэффициента пульсации трафика в диапазоне 50-100) единственным средством служит буфер большого объема. Как и в случае адресных таблиц, каждый процессорный модуль порта обычно имеет свою буферную память для хранения кадров. Чем больше объем этой памяти, тем менее вероятны потери кадров при перегрузках, хотя при несбалансированности средних значений трафика буфер все равно рано или поздно переполнится.

Обычно коммутаторы, предназначенные для работы в ответственных частях сети, имеют буферную память в несколько десятков или сотен килобайт на порт. Хорошо, когда эту буферную память можно перераспределять между несколькими портами, так как одновременные перегрузки по нескольким портам маловероятны. Дополнительным средством защиты может служить общий для всех портов буфер в модуле управления коммутатором. Такой буфер обычно имеет объем в несколько мегабайт.

4.4.3. Дополнительные функции коммутаторов

Так как коммутатор представляет собой сложное вычислительное устройство, имеющее несколько процессорных модулей, то естественно нагрузить его помимо выполнения основной функции передачи кадров с порта на порт по алгоритму моста и некоторыми дополнительными функциями, полезными при построении надежных и гибких сетей. Ниже описываются наиболее распространенные дополнительные функции коммутаторов, которые поддерживаются большинством производителей коммуникационного оборудования.

Поддержка алгоритма Spanning Tree

Алгоритм покрывающего дерева – Spanning Tree Algorithm (*STA*) позволяет коммутаторам автоматически определять древовидную конфигурацию связей в сети при произвольном соединения портов между собой. Как уже отмечалось, для нормальной работы коммутатора требуется отсутствие замкнутых маршрутов в сети. Эти маршруты могут создаваться администратором специально для образования резервных связей или же возникать случайным образом, что вполне возможно, если сеть имеет многочисленные связи, а кабельная система плохо структурирована или документирована.

Поддерживающие алгоритм STA коммутаторы автоматически создают активную древовидную конфигурацию связей (то есть связную конфигурацию без петель) на множестве всех связей сети. Такая конфигурация называется покрывающим деревом – Spanning Tree (иногда ее называют основным деревом), и ее название дало имя всему алгоритму. Алгоритм Spanning Tree описан в стандарте IEEE 802.1D, том же стандарте, который определяет принципы работы прозрачных мостов.

Коммутаторы находят покрывающее дерево адаптивно, с помощью обмена служебными пакетами. Реализация в коммутаторе алгоритма STA очень важна для работы в больших сетях – если коммутатор не поддерживает этот алгоритм, то администратор должен самостоятельно определить, какие порты нужно перевести в заблокированное состояние, чтобы исключите петли. К тому же при отказе какого-либо кабеля, порта или коммутатора администратор должен, во-первых, обнаружить факт отказа, а во-

вторых, ликвидировать последствия отказа, переведя резервную связь в рабочий режим путем активизации некоторых портов. При поддержке коммутаторами сети протокола Spanning Tree отказы обнаруживаются автоматически, за счет постоянного тестирования связности сети служебными пакетами. После обнаружения потери связности протокол строит новое покрывающее дерево, если это возможно, и сеть автоматически восстанавливает работоспособность.

Алгоритм Spanning Tree определяет активную конфигурацию сети за три этапа.

- Сначала в сети определяется корневой коммутатор (root switch), от которого строится дерево. Корневой коммутатор может быть выбран автоматически или назначен администратором. При автоматическом выборе корневым становится коммутатор с меньшим значением MAC-адреса его блока управления.
- Затем, на втором этапе, для каждого коммутатора определяется корневой порт (root port) – это порт, который имеет по сети кратчайшее расстояние до корневого коммутатора (точнее, до любого из портов корневого коммутатора).
- И наконец, на третьем этапе для каждого сегмента сети выбирается так называемый назначенный порт (designated port) – это порт, который имеет кратчайшее расстояние от данного сегмента до корневого коммутатора. После определения корневых и назначенных портов каждый коммутатор блокирует остальные порты, которые не попали в эти два класса портов. Можно математически доказать, что при таком выборе активных портов в сети исключаются петли и оставшиеся связи образуют покрывающее дерево (если оно может быть построено при существующих связях в сети).

Понятие расстояния играет важную роль в построении покрывающего дерева. Именно по этому критерию выбирается единственный порт, соединяющий каждый коммутатор с корневым коммутатором, и единственный порт, соединяющий каждый сегмент сети с корневым коммутатором.

На рис. 4.38 показан пример построения конфигурации покрывающего дерева для сети, состоящей из 5 сегментов и 5 коммутаторов. Корневые порты закрашены темным цветом, назначенные порты не закрашены, а заблокированные порты перечеркнуты. В активной конфигурации коммутаторы 2 и 4 не имеют портов, передающих кадры данных, поэтому они закрашены как резервные.

Расстояние до корня определяется как суммарное условное время на передачу одного бита данных от порта данного коммутатора до порта корневого коммутатора. При этом считается, что время внутренних передач данных (с порта на порт) коммутатором пренебрежимо мало, а учитывается только время на передачу данных по сегментам сети, соединяющим коммутаторы. Условное время сегмента рассчитывается как время, затрачиваемое на передачу одного бита информации в 10 наносекундных единицах между непосредственно связанными по сегменту сети портами. Так, для сегмента Ethernet это время равно 10 условным единицам, а для сегмента Token Ring 16 Мбит/с – 6,25. (Алгоритм STA не связан с каким-либо определенным стандартом канального уровня, он может применяться к коммутаторам, соединяющим сети различных технологий.)

В приведенном примере предполагается, что все сегменты работают на одной скорости, поэтому они имеют одинаковые условные расстояния, которые поэтому не показаны на рисунке.

Для автоматического определения начальной активной конфигурации дерева все коммутаторы сети после их инициализации начинают периодически обмениваться специальными пакетами, называемыми *протокольными блоками данных моста – BPDU* (*Bridge Protocol Data Unit*), что отражает факт первоначальной разработки алгоритма STA для мостов.

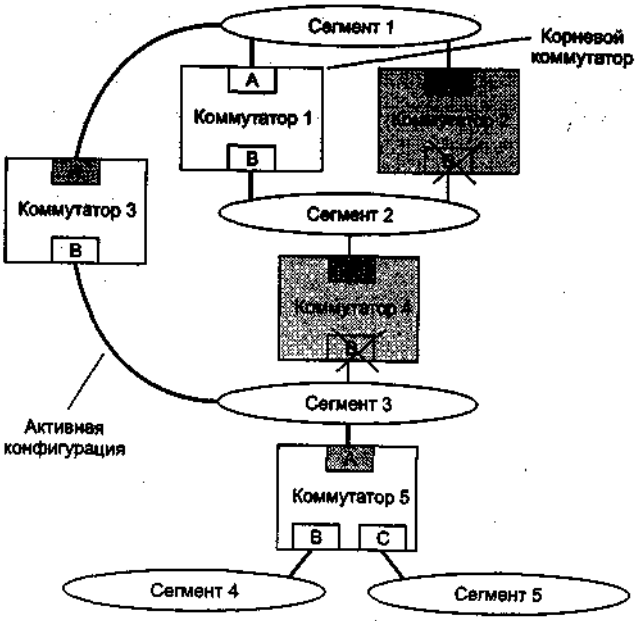


Рис. 4.38. Построение покрывающего дерева сети по алгоритму STA

Пакеты BPDU помещаются в поле данных кадров канального уровня, например кадров Ethernet или FDDI. Желательно, чтобы все коммутаторы поддерживали общий групповой адрес, с помощью которого кадры, содержащие пакеты BPDU, могли бы одновременно передаваться всем коммутаторам сети. Иначе пакеты BPDU рассылаются широковещательно.

Поля пакета BPDU перечислены ниже.

- Идентификатор версии протокола STA - 2 байта. Коммутаторы должны поддерживать одну и ту же версию протокола STA, иначе может установиться активная конфигурация с петлями.
- Тип BPDU - 1 байт. Существуют два типа BPDU - конфигурационный BPDU, то есть заявка на возможность стать корневым коммутатором, на основании которой происходит определение активной конфигурации, и BPDU уведомления о реконфигурации, которое посылается коммутатором, обнаружившим событие, требующее проведения реконфигурации - отказ линии связи, отказ порта, изменение приоритетов коммутатора или портов.
- Флаги – 1 байт. Один бит содержит флаг изменения конфигурации, второй – флаг подтверждения изменения конфигурации.
- Идентификатор корневого коммутатора - 8 байт.
- Расстояние до корня – 2 байта.
- Идентификатор коммутатора – 8 байт.
- Идентификатор порта – 2 байта.
- Время жизни сообщения - 2 байта. Измеряется в единицах по 0,5 с, служит для выявления устаревших сообщений. Когда пакет BPDU проходит через коммутатор, тот добавляет ко времени жизни пакета время его задержки данным коммутатором.
- Максимальное время жизни сообщения – 2 байта. Если пакет BPDU имеет время жизни, превышающее максимальное, то он игнорируется коммутаторами.
- Интервал hello, через который посылаются пакеты BPDU.
- Задержка смены состояний – 2 байта. Задержка определяет минимальное время перехода портов коммутатора в активное состояние. Такая задержка необходима, чтобы исключить возможность временного возникновения петель при неодновременной смене состояний портов во время реконфигурации.

У пакета BPDU уведомления о реконфигурации отсутствуют все поля, кроме двух первых.

Идентификаторы коммутаторов состоят из 8 байт, причем младшие 6 являются MAC-адресом блока управления коммутатора. Старшие 2 байта в исходном состоянии заполнены нулями, но администратор может изменить значение этих байтов, тем самым назначив определенный коммутатор корневым.

После инициализации каждый коммутатор сначала считает себя корневым. Поэтому он начинает через интервал hello генерировать через все свои порты сообщения BPDU конфигурационного типа. В

них он указывает свой идентификатор в качестве идентификатора корневого коммутатора (и в качестве идентификатора данного коммутатора также), расстояние до корня устанавливается в 0, а в качестве идентификатора порта указывается идентификатор того порта, через который передается BPDU. Как только коммутатор получает BPDU, в котором имеется идентификатор корневого коммутатора, со значением, меньшим его собственного, он перестает генерировать свои собственные кадры BPDU, а начинает ретранслировать только кадры нового претендента на звание корневого коммутатора. На рис. 4.38 у коммутатора 1 идентификатор имеет наименьшее значение, раз он стал в результате обмена кадрами корневым.

При ретрансляции кадров каждый коммутатор наращивает расстояние до корня, указанное в пришедшем BPDU, на условное время сегмента, по которому принят данный кадр. Тем самым в кадре BPDU, по мере прохождения через коммутаторы, накапливается расстояние до корневого коммутатора. Если считать, что все сегменты рассматриваемого примера являются сегментами Ethernet, то коммутатор 2, приняв от коммутатора BPDU по сегменту 1 с расстоянием, равным 0, наращивает его на 10 единиц.

Ретранслируя кадры, каждый коммутатор для каждого своего порта запоминает минимальное расстояние до корня, встретившееся во всех принятых этим портом кадрах BPDU. При завершении процедуры установления конфигурации покрывающего дерева (по времени) каждый коммутатор находит свой корневой порт – это порт, для которого минимальное расстояние до корня оказалось меньше, чем у других портов. Так, коммутатор 3 выбирает порт A в качестве корневого, поскольку по порту A минимальное расстояние до корня равно 10 (BPDU с таким расстоянием принят от корневого коммутатора через сегмент 1). Порт B коммутатора 3 обнаружил в принимаемых кадрах минимальное расстояние в 20 единиц – это соответствовало случаю прохождения кадра от порта B корневого моста через сегмент 2, затем через мост 4 и сегмент 3.

Кроме корневого порта коммутаторы распределенным образом выбирают для каждого сегмента сети назначенный порт. Для этого они исключают из рассмотрения свой корневой порт (для сегмента, к которому он подключен, всегда существует другой коммутатор, который ближе расположен к корню), а для всех своих оставшихся портов сравнивают принятые по ним минимальные расстояния до корня с расстоянием до корня своего корневого порта. Если у какого-либо своего порта принятые им расстояния до корня больше, чем расстояние маршрута, пролегающего через свой корневой порт, то это значит, что для сегмента, к которому подключен данный порт, кратчайшее расстояние к корневому коммутатору ведет именно через данный порт. Коммутатор делает все свои порты, у которых такое условие выполняется, назначенными.

Если в процессе выбора корневого порта или назначенного порта несколько портов оказываются равными по критерию кратчайшего расстояния до корневого коммутатора, то выбирается порт с наименьшим идентификатором.

В качестве примера рассмотрим выбор корневого порта для коммутатора 2 и назначенного порта для сегмента 2. Мост 2 при выборе корневого порта столкнулся с ситуацией, когда порт A и порт B имеют равное расстояние до корня – по 10 единиц (порт A принимает кадры от порта B корневого коммутатора через один промежуточный сегмент – сегмент 1, а порт B принимает кадры от порта A корневого коммутатора также через один промежуточный сегмент – через сегмент 2). Идентификатор A имеет меньшее числовое значение, чем B (в силу упорядоченности кодов символов), поэтому порт A стал корневым портом коммутатора 2.

При проверке порта B на случай, не является ли он назначенным для сегмента 2, коммутатор 2 обнаружил, что через этот порт он принимал кадры с указанным в них минимальным расстоянием 0 (это были кадры от порта B корневого коммутатора 1). Так как собственный корневой порт у коммутатора 2 имеет расстояние до корня 10, то порт B не является назначенным для сегмента 2.

Затем все порты, кроме корневого и назначенных, переводятся каждым коммутатором в заблокированное состояние. На этом построение покрывающего дерева заканчивается.

В процессе нормальной работы корневой коммутатор продолжает генерировать служебные кадры BPDU, а остальные коммутаторы продолжают их принимать своими корневыми портами и ретранслировать назначенными. Если у коммутатора нет назначенных портов, как у коммутаторов 2 и 4, то они все равно продолжают принимать участие в работе протокола Spanning Tree, принимая служебные кадры корневым портом. Если по истечении тайм-аута корневой порт любого коммутатора сети не получает служебный кадр BPDU, то он инициализирует новую процедуру построения покрывающего дерева, оповещая об этом другие коммутаторы BPDU уведомления о реконфигурации.

Получив такой кадр, все коммутаторы начинают снова генерировать BDPU конфигурационного типа, в результате чего устанавливается новая активная конфигурация.

Трансляция протоколов канального уровня

Коммутаторы могут выполнять трансляцию одного протокола канального уровня в другой, например Ethernet в FDDI, Fast Ethernet в Token Ring и т. п. При этом они работают по тем же алгоритмам, что и транслирующие мосты, то есть в соответствии со спецификациями IEEE 802.1H и RFC 1042, определяющими .правила преобразования полей кадров разных протоколов.

Трансляцию протоколов локальных сетей облегчает тот факт, что наиболее сложную работу, которую при объединении гетерогенных сетей часто выполняют маршрутизаторы и шлюзы, а именно работу по трансляции адресной информации, в данном случае выполнять не нужно. Все конечные узлы локальных сетей имеют уникальные адреса одного и того же формата независимо от поддерживаемого протокола. Поэтому адрес сетевого адаптера Ethernet понятен сетевому адаптеру FDDI, и они могут использовать эти адреса в полях своих кадров не задумываясь о том, что узел, с которым они взаимодействуют, принадлежит сети, работающей по другой технологии.

Поэтому при согласовании протоколов локальных сетей коммутаторы не строят таблиц соответствия адресов узлов, а переносят адреса назначения и источника из кадра одного протокола в кадр другого.

Кроме изменения порядка бит при передаче байт адреса трансляция протокола Ethernet (и Fast Ethernet, который использует формат кадров Ethernet) в протоколы FDDI и Token Ring включает выполнение перечисленных ниже (возможно, не всех) операций.

- Вычисление длины поля данных кадра и помещение этого значения в поле Length при передаче кадра из сети FDDI или Token Ring в сеть Ethernet 802.3 (в кадрах FDDI и Token Ring поле длины отсутствует).
- Заполнение полей статуса кадра при передаче кадров из сети FDDI или Token Ring в сеть Ethernet. Кадры FDDI и Token Ring имеют два бита, устанавливаемые станцией, которой предназначался кадр, – бит распознавания адреса A и бит копирования кадра C. При получении кадра станция должна установить эти два бита, чтобы кадр, вернувшийся по кольцу к сгенерировавшей его станции, принес данные обратной связи. При передаче коммутатором кадра в другую сеть нет стандартных правил для установки бит A и C в кадре, который возвращается по кольцу к станции-источнику. Поэтому производители коммутаторов решают эту проблему по своему усмотрению.
- Отбрасывание кадров, передаваемых из сетей FDDI или Token Ring в сеть Ethernet с размером поля данных большим, чем 1500 байт, так как это максимально возможное значение поля данных для сетей Ethernet. В дальнейшем, не дождавшись ответа от станции назначения из сети Ethernet, протокол верхнего уровня станции из сети FDDI, возможно, уменьшит размер передаваемых в одном кадре данных, и тогда коммутатор сможет передавать кадры между этими станциями. Другим вариантом решения проблемы является поддержка коммутатором IP-фрагментации, но это требует, во-первых, реализации в коммутаторе протокола сетевого уровня, а во-вторых, поддержки протокола IP взаимодействующими узлами транслируемых сетей.
- Заполнение поля Type (тип протокола в поле данных) кадра Ethernet II при приходе кадров из сетей, поддерживающих кадры FDDI или Token Ring, в которых это поле отсутствует, зато имеются поля DSAP и SSAP, выполняющие то же назначение, но с другими кодами для обозначения протоколов. Для упрощения трансляции спецификация RFC 1042 предлагает всегда использовать в сетях FDDI и Token Ring кадры с заголовками LLC/SNAP, которые имеют то же поле Type и с теми же значениями, что я кадры Ethernet II. При преобразовании кадров значение из шля Type заголовка LLC/SNAP переносится в поде Type кадра Ethernet II, и наоборот. Если в сети Ethernet имеются форматы кадров, отличные от Ethernet II, то они также должны иметь заголовок LLC/SNAP.
- Пересчет контрольной суммы кадра в соответствии со сформированными значениями служебных полей кадра.

Возможности коммутаторов по фильтрации трафика

Многие коммутаторы позволяют администраторам задавать дополнительные условия фильтрации кадров наряду со стандартными условиями их фильтрации в соответствии с информацией адресной таблицы. Пользовательские фильтры предназначены для создания дополнительных барьеров на пути кадров, которые ограничивают доступ определенных групп пользователей к определенным службам сети.

Наиболее простыми являются пользовательские фильтры на основе MAC-адресов станций. Так как MAC-адреса – это та информация, с которой работает коммутатор, то он позволяет задавать такие фильтры в удобной для администратора форме, возможно, проставляя некоторые условия в дополнительном поле адресной таблицы, подобно тем, которые были указаны в адресной таблице моста System 3000 на рис. 4.20 – например, отбрасывать кадры с определенным адресом. При этом пользователю, работающему на компьютере с данным MAC-адресом, полностью запрещается доступ к ресурсам другого сегмента сети.

Часто администратору требуется задать более тонкие условия фильтрации, например запретить некоторому пользователю печатать свои документы на определенном сервере печати NetWare чужого сегмента, а остальные ресурсы этого сегмента сделать доступными. Для реализации такого фильтра нужно запретить передачу кадров с определенным MAC-адресом, в которых вложены пакеты IPX, в поле «номер сокета» которых будет указано значение, соответствующее службе печати NetWare. Коммутаторы не анализируют протоколы верхних уровней, такие как IPX, поэтому администратору приходится для задания условий такой фильтрации вручную определять поле, по значению которого нужно осуществлять фильтрацию, в виде пары «смещение - размер» относительно начала поля данных кадра канального уровня, а затем еще указать в шестнадцатеричном формате значение этого поля для службы печати.

Обычно условия фильтрации записываются в виде булевых выражений, формируемых с помощью логических операторов AND и OR.

Наложение дополнительных условий фильтрации может снизить производительность коммутатора, так как вычисление булевых выражений требует проведения дополнительных вычислений процессорами портов.

Приоритетная обработка кадров

Построение сетей на основе коммутаторов позволяет использовать приоритезацию трафика, причем делать это независимо от технологии сети. Эта новая возможность (по сравнению с сетями, построенными целиком на концентраторах) является следствием того, что коммутаторы буферизуют кадры перед их отправкой на другой порт. Коммутатор обычно ведет для каждого входного и выходного порта не одну, а несколько очередей, причем каждая очередь имеет свой приоритет обработки. При это коммутатор может быть сконфигурирован, например, так, чтобы передавать один низкоприоритетный пакет на каждые 10 высокоприоритетных пакетов.

Поддержка приоритетной обработки может особенно пригодиться для приложений, предъявляющих различные требования к допустимым задержкам кадров и к пропускной способности сети для потока кадров.

Приоритезация трафика коммутаторами сегодня является одним из основных механизмов обеспечения качества транспортного обслуживания в локальных сетях. Это, естественно, не гарантированное качество обслуживания, а только механизм best effort – «с максимальными усилиями». К каким уровням задержек приводит приписывание того или иного уровня приоритета кадру, какую пропускную способность обеспечивает приоритет потоку кадров – схема приоритезации не говорит. Выяснить последствия ее применения можно только путем проведения натурных экспериментов или же с помощью имитационного моделирования. Ясно только одно – более приоритетные кадры будут обрабатываться раньше менее приоритетных, поэтому все показатели качества обслуживания у них будут выше, чем у менее приоритетных. Остается вопрос – насколько? Гарантии качества обслуживания дают другие схемы, которые основаны на предварительном резервировании качества обслуживания. Например, такие схемы используются в технологиях глобальных сетей frame relay и ATM или в протоколе RSVP для сетей TCP/IP. Однако для коммутаторов такого рода протоколов нет, так что гарантий качества обслуживания они пока дать не могут.

Основным вопросом при приоритетной обработке кадров коммутаторами является вопрос назначения кадру приоритета. Так как не все протоколы канального уровня поддерживают поле

приоритета кадра, например у кадров Ethernet оно отсутствует, то коммутатор должен использовать какой-либо дополнительный механизм для связывания кадра с его приоритетом. Наиболее распространенный способ – приписывание приоритета портам коммутатора. При этом способе коммутатор помещает кадр в очередь кадров соответствующего приоритета в зависимости от того, через какой порт поступил кадр в коммутатор. Способ несложный, но недостаточно гибкий – если к порту коммутатора подключен не отдельный узел, а сегмент, то все узлы сегмента получают одинаковый приоритет.

Многие компании, выпускающие коммутаторы, реализовали в них ту или иную схему приоритетной обработки кадров. Примером фирменного подхода к назначению приоритетов на основе портов является технология PACE компании 3Com.

Более гибким является назначение приоритетов кадрам в соответствии с достаточно новым стандартом IEEE 802.1p. Этот стандарт разрабатывался совместно со стандартом 802.1Q, который рассматривается в следующем разделе, посвященном виртуальным локальным сетям. В обоих стандартах предусмотрен общий дополнительный заголовок для кадров Ethernet, состоящий из двух байт. В этом дополнительном заголовке, который вставляется перед полем данных кадра, 3 бита используются для указания приоритета кадра. Существует протокол, по которому конечный узел может запросить у коммутатора один из восьми уровней приоритета кадра. Если сетевой адаптер не поддерживает стандарт 802.1p, то коммутатор может назначать приоритеты кадрам на основе порта поступления кадра. Такие помеченные кадры будут обслуживаться в соответствии с их приоритетом всеми коммутаторами сети, а не только тем коммутатором, который непосредственно принял кадр от конечного узла. При передаче кадра сетевому адаптеру, не поддерживающему стандарт 802.1p, дополнительный заголовок должен быть удален.

4.4.4. Виртуальные локальные сети

Кроме своего основного назначения – повышения пропускной способности связей в сети – коммутатор позволяет локализовывать потоки информации в сети, а также контролировать эти потоки и управлять ими, опираясь на механизм пользовательских фильтров. Однако пользовательский фильтр может запретить передачи кадров только по конкретным адресам, а широковещательный трафик он передает всем сегментам сети. Так требует алгоритм работы моста, который реализован в коммутаторе, поэтому сети, созданные на основе мостов и коммутаторов, иногда называют плоскими – из-за отсутствия барьеров на пути широковещательного трафика.

Технология *виртуальных локальных сетей* (*Virtual LAN, VLAN*), которая появилась несколько лет тому назад в коммутаторах, позволяет преодолеть указанное ограничение. Виртуальной сетью называется группа узлов сети, трафик которой, в том числе и широковещательный, на канальном уровне полностью изолирован от других узлов сети (рис. 4.39). Это означает, что передача кадров между разными виртуальными сетями на основании адреса канального уровня невозможна, независимо от типа адреса – уникального, группового или широковещательного. В то же время внутри виртуальной сети кадры передаются по технологии коммутации, то есть только на тот порт, который связан с адресом назначения кадра. Виртуальные сети могут пересекаться, если один или несколько компьютеров входят в состав более чем одной виртуальной сети. На рис. 4.39 сервер электронной почты входит в состав 3 и 4 виртуальных сетей. Это значит, что его кадры передаются коммутаторами всем компьютерам, входящим в эти сети. Если же какой-то компьютер входит в состав только виртуальной сети 3, то его кадры до сети 4 доходить не будут, но он может взаимодействовать с компьютерами сети 4 через общий почтовый сервер. Такая схема не полностью защищает виртуальные сети друг от друга – так, широковещательный шторм, возникший на сервере электронной почты, захлестнет сеть 3 и сеть 4.

Говорят, что виртуальная сеть образует *домен широковещательного трафика* (*broadcast domain*), по аналогии с доменом коллизий, который образуется повторителями сетей Ethernet.

Назначение технологии виртуальных сетей состоит в облегчении процесса создания изолированных сетей, которые затем должны связываться с помощью маршрутизаторов, реализующих какой-либо протокол сетевого уровня, например IP. Такое построение сети создает гораздо более мощные барьеры на пути ошибочного трафика из одной сети в другую. Сегодня считается, что любая крупная сеть должна включать маршрутизаторы, иначе потоки ошибочных кадров, например широковещательных, будут периодически затапливать всю сеть через прозрачные для них коммутаторы, приводя ее в неработоспособное состояние.

Технология виртуальных сетей создает гибкую основу для построения крупной сети, соединенной маршрутизаторами, так как коммутаторы позволяют создавать полностью изолированные сегменты программным путем, не прибегая к физической коммутации.

До появления технологии VLAN для создания отдельной сети использовались либо физически изолированные сегменты коаксиального кабеля, либо несвязанные между собой сегменты, построенные на повторителях и мостах. Затем эти сети связывались маршрутизаторами в единую составную сеть (рис. 4.40).

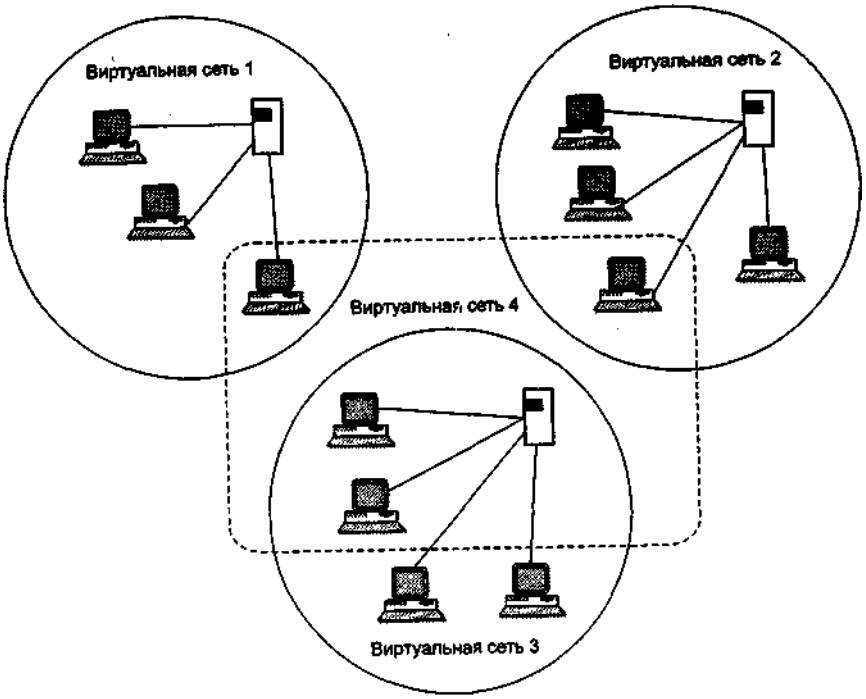


Рис. 4.39. Виртуальные сети

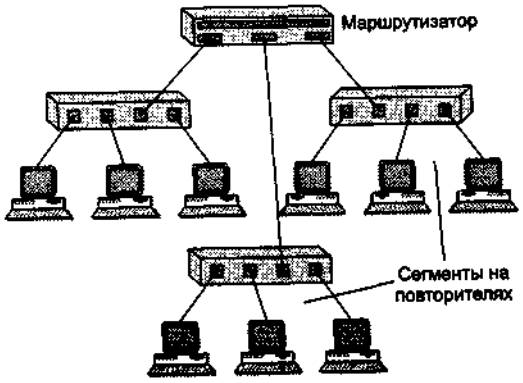


Рис. 4.40. Интерсеть, состоящая из сетей, построенных на основе повторителей

Изменение состава сегментов (переход пользователя в другую сеть, дробление крупных сегментов) при таком подходе подразумевает физическую перекоммутацию разъемов на передних панелях повторителей или в кроссовых панелях, что не очень удобно в больших сетях – много физической работы, к тому же высока вероятность ошибки.

Поэтому для устранения необходимости физической перекоммутации узлов стали применять многосегментные концентраторы, рассмотренные в разделе 4.2.2. Возникла возможность программировать состав разделяемого сегмента без физической перекоммутации.

Однако решение задачи изменения состава сегментов с помощью концентраторов накладывает большие ограничения на структуру сети – количество сегментов такого повторителя обычно невелико, поэтому выделить каждому узлу свой сегмент, как это можно сделать с помощью коммутатора, нереально. Кроме того, при таком подходе вся работа по передаче данных между сегментами ложится на маршрутизаторы, а коммутаторы со своей высокой производительностью остаются не у дел. Поэтому сети, построенные на основе повторителей с конфигурационной коммутацией, по-прежнему основаны

на разделении среды передачи данных между большим количеством узлов, и, следовательно, обладают гораздо меньшей производительностью по сравнению с сетями, построенными на основе коммутаторов.

При использовании технологии виртуальных сетей в коммутаторах одновременно решаются две задачи:

- повышение производительности в каждой из виртуальных сетей, так как коммутатор передает кадры в такой сети только узлу назначения;
- изоляция сетей друг от друга для управления правами доступа пользователей и создания защитных барьеров на пути широковещательных штормов.

Для связи виртуальных сетей в общую сеть требуется привлечение сетевого уровня. Он может быть реализован в отдельном маршрутизаторе, а может работать и в составе программного обеспечения коммутатора, который тогда становится комбинированным устройством – так называемым коммутатором 3-го уровня. Коммутаторы 3-го уровня рассматриваются в главе 5.

Технология образования и работы виртуальных сетей с помощью коммутаторов долгое время не стандартизировалась, хотя и была реализована в очень широком спектре моделей коммутаторов разных производителей. Такое положение изменилось после принятия в 1998 году стандарта IEEE 802.1Q, который определяет базовые правила построения виртуальных локальных сетей, не зависящие от протокола канального уровня, который поддерживает коммутатор.

В виду долгого отсутствия стандарта на VLAN каждый крупный производитель коммутаторов разработал свою технологию виртуальных сетей, которая, как правило, была несовместима с технологиями других производителей. Поэтому, несмотря на появление стандарта, можно не так уж редко встретиться с ситуацией, когда виртуальные сети, созданные на коммутаторах одного производителя, не распознаются и, соответственно, не поддерживаются коммутаторами другого производителя.

При создании виртуальных сетей на основе одного коммутатора обычно используется механизм группирования в сети портов коммутатора (рис. 4.41). При этом каждый порт приписывается той или иной виртуальной сети. Кадр, пришедший от порта, принадлежащего, например, виртуальной сети 1, никогда не будет передан порту, который не принадлежит этой виртуальной сети. Порт можно приписать нескольким виртуальным сетям, хотя на практике так делают редко – пропадает эффект полной изоляции сетей.

Группировка портов для одного коммутатора – наиболее логичный способ образования VLAN, так как виртуальных сетей, построенных на основе одного коммутатора, не может быть больше, чем портов. Если к одному порту подключен сегмент, построенный на основе повторителя, то узлы такого сегмента не имеет смысла включать в разные виртуальные сети – все равно трафик этих узлов будет общим.

Создание виртуальных сетей на основе группирования портов не требует от администратора большого объема ручной работы - достаточно каждый порт приписать к одной из нескольких заранее поименованных виртуальных сетей. Обычно такая операция выполняется с помощью специальной программы, прилагаемой к коммутатору. Администратор создает виртуальные сети путем перетаскивания мышью графических символов портов на графические символы сетей.

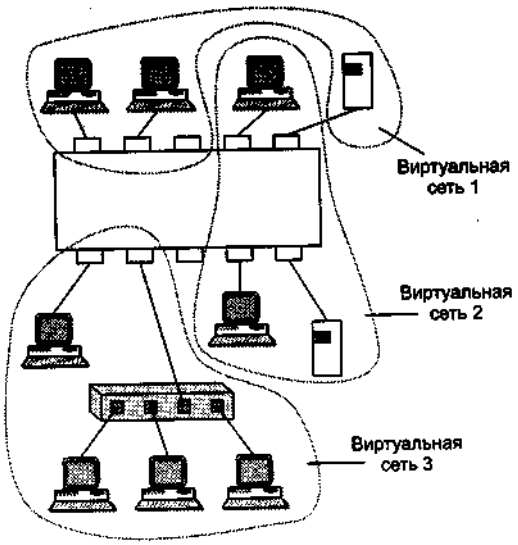


Рис. 4.41. Виртуальные сети, построенные на одном коммутаторе

Второй способ образования виртуальных сетей основан на группировании MAC-адресов. Каждый MAC-адрес, который изучен коммутатором, приписывается той или иной виртуальной сети. При существовании в сети множества узлов этот способ требует выполнения большого количества ручных операций от администратора. Однако он оказывается более гибким при построении виртуальных сетей на основе нескольких коммутаторов, чем способ группирования портов.

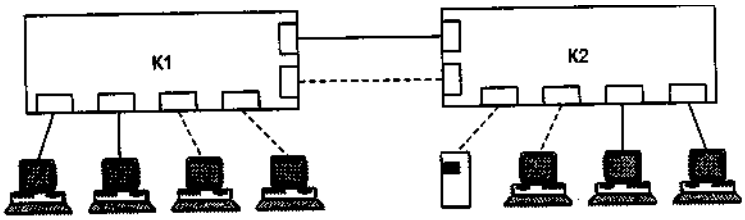


Рис. 4.42. Построение виртуальных сетей на нескольких коммутаторах с группировкой портов

Рисунок 4.42 иллюстрирует проблему, возникающую при создании виртуальных сетей на основе нескольких коммутаторов, поддерживающих технику группирования портов. Если узлы какой-либо виртуальной сети подключены к разным коммутаторам, то дли соединения коммутаторов каждой такой сети должна быть выделена своя пара портов. В противном случае, если коммутаторы будут связаны только одной парой портов, информация о принадлежности кадра той или иной виртуальной сети при передаче из коммутатора в коммутатор будет утеряна. Таким образом, коммутаторы с группировкой портов требуют для своею соединения столько портов, сколько виртуальных сетей они поддерживают. Порты и кабели используются при таком способе очень расточительно. Кроме того, при соединении виртуальных сетей через маршрутизатор для каждой виртуальной сети выделяется в этом случае отдельный кабель и отдельный порт маршрутизатора, что также приводит к большим накладным расходам.

Группирование MAC-адресов в виртуальную сеть на каждом коммутаторе избавляет от необходимости их связи несколькими портами, так как в этом случае MAC-адрес является меткой виртуальной сети. Однако этот способ требует выполнения большого количества ручных операций по маркировке MAC-адресов на каждом коммутаторе сети.

Описанные два подхода основаны только на добавлении дополнительной информации к адресным таблицам моста, и в них отсутствует возможность встраивания информации о принадлежности кадра к виртуальной сети в передаваемый кадр. Остальные подходы используют имеющиеся или дополнительные поля кадра для сохранения информации и принадлежности кадра при его перемещениях между коммутаторами сети. При этом нет необходимости запоминать в каждом коммутаторе принадлежность всех MAC-адресов интерсети виртуальным сетям.

Дополнительное поле с пометкой о номере виртуальной сети используется только тогда, когда кадр передается от коммутатора к коммутатору, а при передаче кадра конечному узлу оно удаляется. При этом модифицируется протокол взаимодействия «коммутатор - коммутатор», а программное и

аппаратное обеспечение конечных узлов остается неизменным. Примеров таких фирменных протоколов много, но общий недостаток у них один – они не поддерживаются другими производителями. Компания Cisco предложила в качестве стандартной добавки к кадрам любых протоколов локальных сетей заголовок протокола 802.10, предназначенного для поддержки функций безопасности вычислительных сетей. Сама компания использует этот метод в тех случаях, когда коммутаторы объединяются между собой по протоколу FDDI. Однако эта инициатива не была поддержана другими ведущими производителями коммутаторов.

Для хранения номера виртуальной сети в стандарте IEEE 802.1Q предусмотрен тот же дополнительный заголовок, что и стандарт 802.1p. Помимо 3-х бит для хранения приоритета кадра, описанных стандартом 802.1p, в этом заголовке 12 бит используются для хранения номера VLAN, к которой принадлежит кадр. Эта дополнительная информация позволяет коммутаторам разных производителей создавать до 4096 общих виртуальных сетей. Чтобы кадр Ethernet не увеличивался в объеме, при добавлении заголовка 802.1p/Q поле данных уменьшается на 2 байта.

Существуют два способа построения виртуальных сетей, которые используют уже имеющиеся поля для маркировки принадлежности кадра виртуальной сети, однако эти поля принадлежат не кадрам канальных протоколов, а пакетам сетевого уровня или ячейкам технологии ATM.

В первом случае виртуальные сети образуются на основе сетевых адресов, например адресов IP, то есть той же информации, которая используется при построении интерсетей традиционным способом. Этот эффективный способ работает тогда, когда коммутаторы поддерживают не только протоколы канального уровня, но и протоколы сетевого уровня, то есть являются комбинированными коммутаторами-маршрутизаторами, что бывает далеко не всегда.

Во втором случае виртуальные сети организуются с помощью виртуальных путей в ATM-сетях.

4.4.5. Типовые схемы применения коммутаторов в локальных сетях

Сочетание коммутаторов и концентраторов

При построении небольших сетей, составляющих нижний уровень иерархии корпоративной сети, вопрос о применении того или иного коммуникационного устройства сводится к вопросу о выборе между концентратором или коммутатором.

При ответе на этот вопрос нужно принимать во внимание несколько факторов. Безусловно, немаловажное значение имеет стоимость в пересчете за порт, которую нужно заплатить при выборе устройства. Из технических соображений в первую очередь нужно принять во внимание существующее распределение трафика между узлами сети. Кроме того, нужно учитывать перспективы развития сети: будут ли в скором времени применяться мультимедийные приложения, будет ли модернизироваться компьютерная база. Если да, то нужно уже сегодня обеспечить резервы по пропускной способности применяемого коммуникационного оборудования. Использование технологии intranet также ведет к увеличению объемов трафика, циркулирующего в сети, и это также необходимо учитывать при выборе устройства.

При выборе типа устройства – концентратор или коммутатор – нужно еще определить и тип протокола, который будут поддерживать его порты (или протоколов, если идет речь о коммутаторе, так как каждый порт может поддерживать отдельный протокол).

Сегодня выбор делается между протоколами трех скоростей – 10, 100 и 1000 Мбит/с. Поэтому, сравнивая применимость концентратора или коммутатора, необходимо рассмотреть варианты концентратора с портами на 10,100 и 1000 Мбит/с, а также несколько вариантов коммутаторов с различными комбинациями скоростей на портах.

Рассмотрим для примера вопрос о применимости коммутатора в сети с одним сервером и несколькими рабочими станциями, взаимодействующими только с сервером (рис. 4.43). Такая конфигурация сети часто встречается в сетях масштаба рабочей группы, особенно в сетях NetWare, где стандартные клиентские оболочки не могут взаимодействовать друг с другом.

Если коммутатор имеет все порты с одинаковой пропускной способностью, например 10 Мбит/с, в этом случае пропускная способность порта в 10 Мбит/с будет распределяться между всеми компьютерами сети. Возможности коммутатора по повышению общей пропускной способности сети оказываются для такой конфигурации невостребованными. Несмотря на микросегментацию сети, ее пропускная способность ограничивается пропускной способностью протокола одного порта, как и в

случае применения концентратора с портами 10 Мбит/с. Небольшой выигрыш при использовании коммутатора будет достигаться лишь за счет уменьшения количества коллизий – вместо коллизий кадры будут просто попадать в очередь к передатчику порта коммутатора, к которому подключен сервер.

Чтобы коммутатор работал в сетях с выделенным сервером более эффективно, производители коммутаторов выпускают модели с одним высокоскоростным портом на 100 Мбит/с для подключения сервера и несколькими низкоскоростными портами на 10 Мбит/с для подключения рабочих станций. В этом случае между рабочими станциями распределяется уже 100 Мбит/с, что позволяет обслуживать в неблокирующем режиме 10-30 станций в зависимости от интенсивности создаваемого ими трафика.

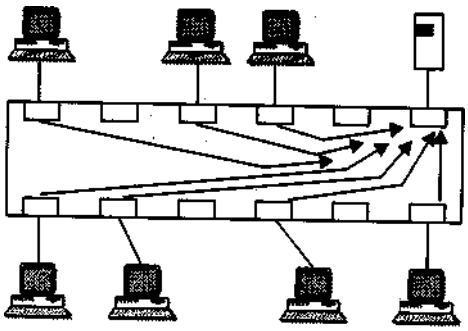


Рис. 4.43. Сеть с выделенным сервером

Однако с таким коммутатором может конкурировать концентратор, поддерживающий протокол с пропускной способностью 100 Мбит/с, например Fast Ethernet. Его стоимость в пересчете за порт будет несколько ниже стоимости за порт коммутатора с одним высокоскоростным портом, а производительность сети примерно та же.

Очевидно, что выбор коммуникационного устройства для сети с выделенным сервером достаточно сложен. Для принятия окончательного решения нужно принимать во внимание перспективы развития сети в отношении движения к сбалансированному трафику. Если в сети вскоре может появиться взаимодействие между рабочими станциями или же второй сервер, то выбор необходимо делать в пользу коммутатора, который сможет поддержать дополнительный трафик без ущерба по отношению к основному.

В пользу коммутатора может сыграть и фактор расстояний – применение коммутаторов не ограничивает максимальный диаметр сети величинами в 2500 м или 210м, которые определяют размеры домена коллизий при использовании концентраторов Ethernet и Fast Ethernet.

В целом существует тенденция постепенного вытеснения концентраторов коммутаторами, которая наблюдается примерно с 1996 года, который был назван весьма авторитетным журналом Data Communications «годом коммутаторов» в ежегодном прогнозе рынка сетевого оборудования.

Стянутая в точку магистраль на коммутаторе

При всем разнообразии структурных схем сетей, построенных на коммутаторах, все они используют две базовые структуры – стянутую в точку магистраль и распределенную магистраль. На основе этих базовых структур затем строятся разнообразные структуры конкретных сетей.

Стянутая в точку магистраль (*collapsed backbone*) – это структура, при которой объединение узлов, сегментов или сетей происходит на внутренней магистрали коммутатора. Пример сети рабочей группы такой структуры приведен на рис.4.44.

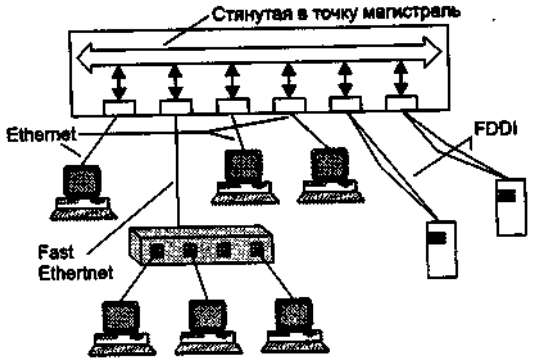


Рис. 4.44. Структура сети со стянутой в точку магистралью

Преимуществом такой структуры является высокая производительности магистрали, Так как для коммутатора производительность внутренней шины или схемы общей памяти, объединяющей модули портов, в несколько гигабит в секунду не является редкостью, то магистраль сети может быть весьма быстродействующей, причем ее скорость не зависит от применяемых в сети протоколов и может быть повышена с помощью замены одной модели коммутатора на другую

Положительной чертой такой схемы является не только высокая скорость магистрали, но и ее протокольная независимость. На внутренней магистрали коммутатора в независимом формате одновременно могут передаваться данные различных протоколов, например Ethernet, FDDI и Fast Ethernet, как это изображено на рис. 4.44 Подключение нового узла с новым протоколом часто требует не замены коммутатора, а просто добавления соответствующего интерфейсного модуля, поддерживающего этот протокол.

Если к каждому порту коммутатора в такой схеме подключен только один узел, то такая схема будет соответствовать микросегментированной сети.

Распределенная магистраль на коммутаторах

В сетях больших зданий или кампусов структура с коллапсированной магистралью не всегда рациональна или возможна. Такая структура приводит к протяженным кабельным системам, связывающим конечные узлы или коммутаторы сетей рабочих групп с центральным коммутатором, шина которого и является магистралью сети. Высокая плотность кабелей и их высокая стоимость ограничивают применение стянутой в точку магистрали в таких сетях. Иногда, особенно в сетях кампусов, просто невозможно стянуть все кабели в одно помещение из-за ограничений на длину связей, накладываемых технологией (например, все реализации технологий локальных сетей на витой паре ограничивают протяженность кабелей в 100 м).

Поэтому в локальных сетях, покрывающих большие территории, часто используется другой вариант построения сети – с распределенной магистралью. Пример такой сети приведен на рис. 4.45.

Распределенная магистраль – это разделяемый сегмент сети, поддерживающий определенный протокол, к которому присоединяются коммутаторы сетей рабочих групп и отделов. На примере распределенная магистраль построена на основе двойного кольца FDDI, к которому подключены коммутаторы этажей. Коммутаторы этажей имеют большое количество портов Ethernet, трафик которых транслируется в трафик протокола FDDI, когда он передается по магистрали с этажа на этаж.

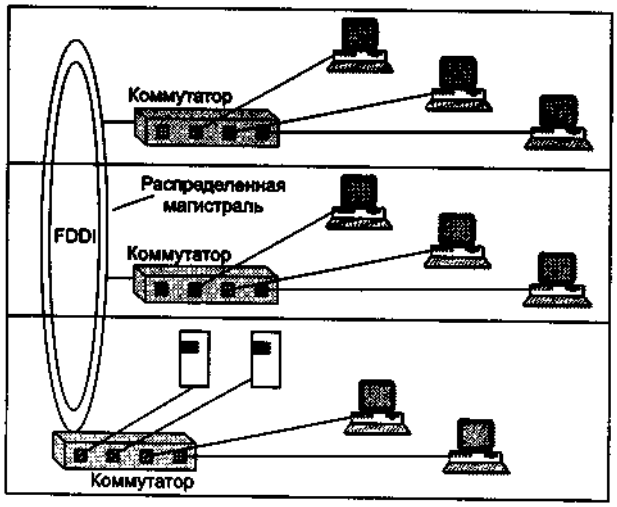


Рис. 4.45. Структура сети с распределенной магистралью

Распределенная магистраль упрощает связи между этажами, сокращает стоимость кабельной системы и преодолевает ограничения на расстояния.

Однако скорость магистрали в этом случае будет существенно ниже скорости магистрали на внутренней шине коммутатора. Причем скорость эта фиксированная и в настоящее время чаще всего не превышает 100 Мбит/с. Поэтому распределенная магистраль может применяться только при невысокой интенсивности трафика между этажами или зданиями. Широкое распространение в недалеком будущем технологии Gigabit Ethernet может снять это ограничение, что очень положительно скажется на структуре крупных сетей.

Пример на рис. 4.45 демонстрирует сочетание двух базовых структур, так как на каждом этаже сеть построена с использованием магистрали на внутренней шине коммутатора.

Выводы

- Коммутаторы связывают процессоры портов по трем основным схемам – коммутационная матрица, общая шина и разделяемая память. В коммутаторах с фиксированным количеством портов обычно используется коммутационная матрица, а в модульных коммутаторах – сочетание коммутационной матрицы в отдельных модулях с общей шиной и разделяемой памятью для связи модулей.
- Для поддержания неблокирующего режима работы коммутатора общая шина или разделяемая память должны обладать производительностью, превышающей сумму производительностей всех портов максимально высокоскоростного набора модулей, которые устанавливаются в шасси.
- Основными характеристиками производительности коммутатора являются: скорость фильтрации кадров, скорость продвижения кадров, общая пропускная способность по всем портам в мегабитах в секунду, задержка передачи кадра.
- На характеристики производительности коммутатора влияют: тип коммутации – «на лету» или с полной буферизацией, размер адресной таблицы, размер буфера кадров.
- Для автоматического поддержания резервных связей в сложных сетях в коммутаторах реализуется алгоритм покрывающего дерева – Spanning Tree Algorithm. Этот алгоритм основан на периодической генерации служебных кадров, с помощью которых выявляются и блокируются петлевидные связи в сети.
- Коммутаторы могут объединять сегменты разных технологий локальных сетей, транслируя протоколы канального уровня в соответствии со спецификацией IEEE 802.1H. Единственным ограничением трансляции является использование MTU одного размера в соединяемых сегментах.
- Коммутаторы поддерживают разнообразные пользовательские фильтры, основанные на MAC-адресах, а также на содержимом полей протоколов верхних уровней. В последнем случае администратор должен выполнить большой объем ручной работы по заданию положения поля

относительно начала кадра и его требуемому значению. Обычно фильтры допускают комбинацию нескольких условий с помощью логических операторов AND и OR.

- Коммутаторы обеспечивают поддержку качества обслуживания с помощью приоритетной обработки кадров. Стандарт 802.1p определяет дополнительное поле, состоящее из 3 бит, для хранения приоритета кадра независимо от технологии сети.
- Технология виртуальных локальных сетей (VLAN) позволяет в сети, построенной на коммутаторах, создать изолированные группы узлов, между которыми не передается любой тип трафика, в том числе и широковещательный. Виртуальные сети являются основой для создания крупных маршрутизируемых сетей и имеют преимущество перед физически изолированными сегментами гибкостью состава, изменяемого программным путем.
- В последнее время наблюдается отчетливая тенденция вытеснения коммутаторами концентраторов с нижних уровней крупных сетей.
- Существуют две основные схемы применения коммутаторов: со стянутой в точку магистралью и с распределенной магистралью. В больших сетях эти схемы применяют комбинированно.

Вопросы и упражнения

1. Что такое структурированная кабельная система?
2. Укажите в таблице применимость того или иного типа кабеля для разных подсистем.

Данные	Горизонтальная подсистема	Вертикальная подсистема	Подсистема кампуса
Неэкранированная витая пара			
Экранированная витая пара			
Толстый коаксиальный кабель			
Тонкий коаксиальный кабель			
Оптоволоконный кабель			
Беспроводная связь			

3. Что означает термин backbone?
4. Как влияет на производительность сети пропускная способность сетевого адаптера и пропускная способность порта концентратора?
5. Имеются ли отличия в работе сетевых адаптеров, соединяющих компьютер с коммутатором или с мостом, или с концентратором?
6. Как концентратор поддерживает резервные связи?
7. В соответствии с основной функцией концентратора – повторением сигнала -его относят к устройствам, работающим на физическом уровне модели OSI. Приведите примеры дополнительных функций концентратора, для выполнения которых концентратору требуется информация протоколов более высоких уровней?
8. Чем модульный концентратор отличается от стекового?
9. Почему для соединения концентраторов между собой используются специальные порты?
10. Каким образом мост/коммутатор строит свою внутреннюю таблицу?
11. Что случится, если во время работы моста/коммутатора произойдет реконфигурация сети, например будут подключены новые компьютеры?
12. О чем говорит размер внутренней адресной таблицы моста? Что произойдет, если таблица переполнится?
13. Можно ли утверждать, что у любого моста скорости продвижения не выше скорости фильтрации?
14. Что нужно сделать администратору сети, чтобы мосты, не поддерживающие алгоритм Spanning Tree, правильно работали в сети с петлями?
15. Что произойдет, если в сети, построенной на концентраторах, имеются замкнутые контуры (например, как на рис. 4.46)?

- (A) сеть будет работать нормально;
- (B) кадры не будут доходить до адресата;

- (C) в сети при передаче любого кадра будет возникать коллизия;
- (D) произойдет зацикливание кадров.

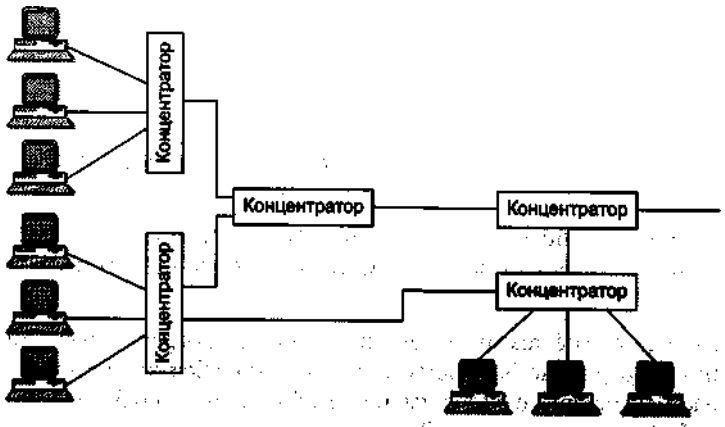


Рис. 4.46. Сеть с петлями, построенная на концентраторах

16. Какие дополнительные возможности имеют мосты, поддерживающие алгоритм Spanning Tree?
17. В чем отличие между резервированием связей маршрутизаторами, с одной стороны, и мостами, поддерживающими алгоритм Spanning Tree, с другой стороны?
18. Пусть на предприятии имеются две изолированные рабочие группы, в каждой из которых имеется свой сервер. В каких случаях лучше использовать:
    - два отдельных концентратора?
    - два концентратора, объединенные в стек?
    - один общий концентратор с большим количеством портов?
19. Пусть на предприятии в одном отделе установлена одноранговая сеть Windows 95, а в другом отделе – сеть NetWare с одним выделенным сервером. Каждая из сетей построена на основе одного концентратора. Как вы считаете, в каком отделе замена концентратора коммутатором может привести к существенному росту производительности? Рассмотрите следующие варианты замены концентратора на коммутатор:
    - (A) концентратор имеет порты 10 Мбит/с, коммутатор имеет все порты 10 Мбит/с;
    - (B) концентратор имеет порты 10 Мбит/с, коммутатор имеет порты 10 Мбит/с и 1 порт 100 Мбит/с;
    - (C) концентратор имеет порты 100 Мбит/с, коммутатор имеет все порты 100 Мбит/с.
20. В области сетевых технологий явно наметилась тенденция к использованию индивидуальных связей компьютеров с коммуникационными устройствами (в отличие от подключения к портам сегментов). С чем это связано?
21. Почему полнодуплексный Ethernet не поддерживается в концентраторах?
22. Каким образом коммутатор может управлять потоком пакетов, поступающих от сетевых адаптеров станций сети?
23. Существуют маршрутизаторы, работающие в режиме моста на некоторых портах. Как вы думаете, можно ли создать маршрутизатор или коммутатор, который способен работать в режиме концентратора на тех же портах, на которых выполняется маршрутизация?
24. Можно ли соединить транслирующим коммутатором сегменты, в которых установлено разное максимальное значение поля данных?
25. Имеется ли специфика в использовании мостов и коммутаторов? Приведите примеры, когда замена моста коммутатором не повышает производительности сети.
26. Почему недорогие коммутаторы, выполняющие ограниченное число функций, обычно работают по быстрому алгоритму обработки пакетов «на лету», а дорогие коммутаторы, с большим числом функций – по более медленному алгоритму буферизации пакетов?
27. Какая информация содержится в таблицах мостов/коммутаторов и маршрутизаторов?
28. Поясните определение: «Виртуальная локальная сеть – это домен распространения широковещательных сообщений».
29. В каких случаях появляется необходимость в создании виртуальных сегментов? Приведите примеры.

ГЛАВА 5. СЕТЕВОЙ УРОВЕНЬ КАК СРЕДСТВО ПОСТРОЕНИЯ БОЛЬШИХ СЕТЕЙ

5.1. Принципы объединения сетей на основе протоколов сетевого уровня

В стандартной модели взаимодействия открытых систем в функции сетевого уровня входит решение следующих задач:

- передача пакетов между конечными узлами в составных сетях;
- выбор маршрута передачи пакетов, наилучшего по некоторому критерию;
- согласование разных протоколов канального уровня, использующихся в отдельных подсетях одной составной сети.

Протоколы сетевого уровня реализуются, как правило, в виде программных модулей и выполняются на конечных узлах-компьютерах, называемых хостами, а также на промежуточных узлах-маршрутизаторах, называемых шлюзами. Функции маршрутизаторов могут выполнять как специализированные устройства, так и универсальные компьютеры с соответствующим программным обеспечением.

5.1.1. Ограничения мостов и коммутаторов

Создание сложной, структурированной сети, интегрирующей различные базовые технологии, может осуществляться и средствами канального уровня: для этого могут быть использованы некоторые типы мостов и коммутаторов. Мост или коммутатор разделяет сеть на сегменты, локализуя трафик внутри сегмента, что делает линии связи разделяемыми преимущественно между станциями данного сегмента. Тем самым сеть распадается на отдельные подсети, из которых могут быть построены составные сети достаточно крупных размеров.

Однако построение сложных сетей только на основе повторителей, мостов и коммутаторов имеет существенные ограничения и недостатки.

- Во-первых, в топологии получившейся сети должны *отсутствовать петли*. Действительно, мост/коммутатор может решать задачу доставки пакета адресату только тогда, когда между отправителем и получателем существует единственный путь. В то же время наличие избыточных связей, которые и образуют петли, часто необходимо для лучшей балансировки нагрузки, а также для повышения надежности сети за счет образования резервных путей.
- Во-вторых, логические сегменты сети, расположенные между мостами или коммутаторами, *слабо изолированы* друг от друга, а именно не защищены от так называемых широковещательных штормов. Если какая-либо станция посылает широковещательное сообщение, то это сообщение передается всем станциям всех логических сегментов сети. Защита от широковещательных штормов в сетях, построенных на основе мостов и коммутаторов, имеет количественный, а не качественный характер: администратор просто ограничивает количество широковещательных пакетов, которое разрешается генерировать некоторому узлу в единицу времени. Использование же механизма виртуальных сетей, реализованного во многих коммутаторах, хотя и позволяет достаточно гибко создавать изолированные по трафику группы станций, но при этом изолирует их полностью, так что узлы одной виртуальной сети не могут взаимодействовать с узлами другой виртуальной сети.
- В-третьих, в сетях, построенных на основе мостов и коммутаторов, достаточно сложно решается задача управления трафиком на основе значения данных, содержащихся в пакете. В таких сетях это возможно только с помощью пользовательских фильтров, для задания которых администратору приходится иметь дело с двоичным представлением содержимого пакетов.
- В-четвертых, реализация транспортной подсистемы только средствами физического и канального уровней, к которым относятся мосты и коммутаторы, приводит к недостаточно гибкой, одноуровневой системе адресации: в качестве адреса назначения используется MAC-адрес, жестко связанный с сетевым адаптером.
- Наконец, возможностью трансляции протоколов канального уровня обладают далеко не все типы мостов и коммутаторов, к тому же эти возможности ограничены. В частности, в

объединяемых сетях должны совпадать максимально допустимые размеры полей данных в кадрах, так как мостами и коммутаторами не поддерживается функция фрагментации кадров.

Наличие серьезных ограничений у протоколов канального уровня показывает, что построение на основе средств этого уровня больших неоднородных сетей является весьма проблематичным. Естественное решение в этих случаях – это привлечение средств более высокого, сетевого уровня.

5.1.2. Понятие internetworking

Основная идея введения сетевого уровня состоит в следующем. Сеть в общем случае рассматривается как совокупность нескольких сетей и называется составной сетью или интерсетью (*internetwork или internet*). Сети, входящие в составную сеть, называются подсетями (*subnet*), составляющими сетями или просто сетями (рис. 5.1).

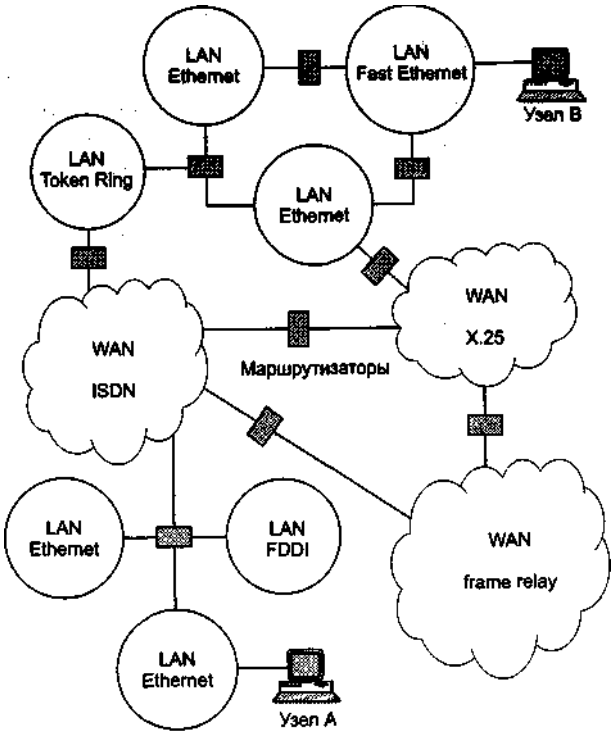


Рис. 5.1. Архитектура составной сети

Подсети соединяются между собой маршрутизаторами. Компонентами составной сети могут являться как локальные, так и глобальные сети. Внутренняя структура каждой сети на рисунке не показана, так как она не имеет значения при рассмотрении сетевого протокола. Все узлы в пределах одной подсети взаимодействуют, используя единую для них технологию. Так, в составную сеть, показанную на рисунке, входит несколько сетей разных технологий: локальные сети Ethernet, Fast Ethernet, Token Ring, FDDI и глобальные сети frame relay, X.25, ISDN. Каждая из этих технологий достаточна для того, чтобы организовать взаимодействие всех узлов в своей подсети, но не способна построить информационную связь между произвольно выбранными узлами, принадлежащими разным подсетям, например между узлом А и узлом В на рис. 5.1. Следовательно, для организации взаимодействия между любой произвольной парой узлов этой «большой» составной сети требуются дополнительные средства. Такие средства и предоставляет сетевой уровень.

Сетевой уровень выступает в качестве координатора, организующего работу всех подсетей, лежащих на пути продвижения пакета по составной сети. Для перемещения данных в пределах подсетей сетевой уровень обращается к используемым в этих подсетях технологиям. .

Хотя многие технологии локальных сетей (Ethernet, Token Ring, FDDI, Fast Ethernet и др.) используют одну и ту же систему адресации узлов на основе MAC-адресов, существует немало технологий (X.25, ATM, frame relay), в которых применяются другие схемы адресации. Адреса присвоенные узлам в соответствии с технологиями подсетей, называют локальными. Чтобы сетевой уровень мог выполнить свою задачу, ему необходима собственная система адресации, не зависящая от способов адресации узлов в отдельных подсетях, которая позволила бы на сетевом уровне

универсальным и однозначным способами идентифицировать любой узел составной сети.

Естественным способом формирования сетевого адреса является уникальная нумерация всех подсетей составной сети и нумерация всех узлов в пределах каждой подсети. Таким образом, сетевой адрес представляет собой пару: номер сети (подсети) и номер узла.

В качестве номера узла может выступать либо локальный адрес этого узла (такая схема принята в стеке IPX/SPX), либо некоторое число, никак не связанное с локальной технологией, которое однозначно идентифицирует узел в пределах данной подсети. В первом случае сетевой адрес становится зависимым от локальных технологий, что ограничивает его применение. Например, сетевые адреса IPX/SPX рассчитаны на работу в составных сетях, объединяющих сети, в которых используются только MAC-адреса или адреса аналогичного формата. Второй подход более универсален, он характерен для стека TCP/IP. И в том и другом случае каждый узел составной сети имеет наряду со своим локальным адресом еще один – универсальный сетевой адрес.

Данные, которые поступают на сетевой уровень и которые необходимо передать через составную сеть, снабжаются заголовком сетевого уровня. Данные вместе с заголовком образуют пакет. Заголовок пакета сетевого уровня имеет унифицированный формат, не зависящий от форматов кадров канального уровня тех сетей, которые могут входить в объединенную сеть, и несет наряду с другой служебной информацией данные о номере сети, которой предназначается этот пакет. Сетевой уровень определяет маршрут и перемещает пакет между подсетями.

При передаче пакета из одной подсети в другую пакет сетевого уровня, инкапсулированный в прибывший канальный кадр первой подсети, освобождается от заголовков этого кадра и окружается заголовками кадра канального уровня следующей подсети. Информацией, на основе которой делается эта замена, являются служебные поля пакета сетевого уровня. В поле адреса назначения нового кадра указывается локальный адрес следующего маршрутизатора.

ПРИМЕЧАНИЕ Если в подсети доставка данных осуществляется средствами канального и физического уровней (как, например, в стандартных локальных сетях), то пакеты сетевого уровня упаковываются в кадры канального уровня. Если же в какой-либо подсети для транспортировки сообщений используется технология, основанная на стеках с большим числом уровней, то пакеты сетевого уровня упаковываются в блоки передаваемых данных самого высокого уровня подсети.

Если проводить аналогию между взаимодействием разнородных сетей и перепиской людей из разных стран, то сетевая информация – это общепринятый индекс страны, добавленный к адресу письма, написанному на одном из сотни языков земного шара, например на санскрите. И даже если это письмо должно пройти через множество стран, почтовые работники которых не знают санскрита, понятный им индекс страны-адресата подскажет, через какие промежуточные страны лучше передать письмо, чтобы оно кратчайшим путем попало в Индию. А уже там работники местных почтовых отделений смогут прочитать точный адрес, указывающий город, улицу, дом и индивидуума, и доставить письмо адресату, так как адрес написан на языке и в форме, принятой в данной стране.

Основным полем заголовка сетевого уровня является номер сети-адресата. В рассмотренных нами ранее протоколах локальных сетей такого поля в кадрах предусмотрено не было – предполагалось, что все узлы принадлежат одной сети. Явная нумерация сетей позволяет протоколам сетевого уровня составлять точную карту межсетевых связей и выбирать рациональные маршруты при любой их топологии, в том числе альтернативные маршруты, если они имеются, что не умеют делать мосты и коммутаторы.

Кроме номера сети заголовок сетевого уровня должен содержать и другую информацию, необходимую для успешного перехода пакета из сети одного типа в сеть другого типа. К такой информации может относиться, например:

- номер фрагмента пакета, необходимый для успешного проведения операций сборки-разборки фрагментов при соединении сетей с разными максимальными размерами пакетов;
- время жизни пакета, указывающее, как долго он путешествует по интерсети, это время может использоваться для уничтожения «заблудившихся» пакетов;
- качество услуги – критерий выбора маршрута при межсетевых передачах – например, узел-отправитель может потребовать передать пакет с максимальной надежностью, возможно, в ущерб времени доставки.

Когда две или более сети организуют совместную транспортную службу, то такой режим взаимодействия обычно называют межсетевым взаимодействием (internetworking).

5.1.3. Принципы маршрутизации

Важнейшей задачей сетевого уровня является маршрутизация – передача пакетов между двумя конечными узлами в составной сети.

Рассмотрим принципы маршрутизации на примере составной сети, изображенной на рис. 5.2. В этой сети 20 маршрутизаторов объединяют 18 сетей в общую сеть; SI, S2.....S20 – это номера сетей. Маршрутизаторы имеют по нескольку портов (по крайней мере, по два), к которым присоединяются сети. Каждый порт маршрутизатора можно рассматривать как отдельный узел сети: он имеет собственный сетевой адрес и собственный локальный адрес в той подсети, которая к нему подключена. Например, маршрутизатор под номером 1 имеет три порта, к которым подключены сети SI, S2, S3. На рисунке сетевые адреса этих портов обозначены как M1(1), M1(2) и M1(3). Порт M1(1) имеет локальный адрес в сети с номером S1, порт M1(2) – в сети S2, а порт M1(3) – в сети S3. Таким образом, маршрутизатор можно рассматривать как совокупность нескольких узлов, каждый из которых входит в свою сеть. Как единое устройство маршрутизатор не имеет ни отдельного сетевого адреса, ни какого-либо локального адреса.

ПРИМЕЧАНИЕ Если маршрутизатор имеет блок управления (например, SNMP-управления), то этот блок имеет собственные локальный и сетевой адреса, по которым к нему обращается центральная станция управления, находящаяся где-то в составной сети.

В сложных составных сетях почти всегда существует несколько альтернативных маршрутов для передачи пакетов между двумя конечными узлами. Маршрут - это последовательность маршрутизаторов, которые должен пройти пакет от отправителя до пункта назначения. Так, пакет, отправленный из узла A в узел B, может пройти через маршрутизаторы 17,12,5,4 и 1 или маршрутизаторы 17,13,7, 6 и 3. Нетрудно найти еще несколько маршрутов между узлами A и B.

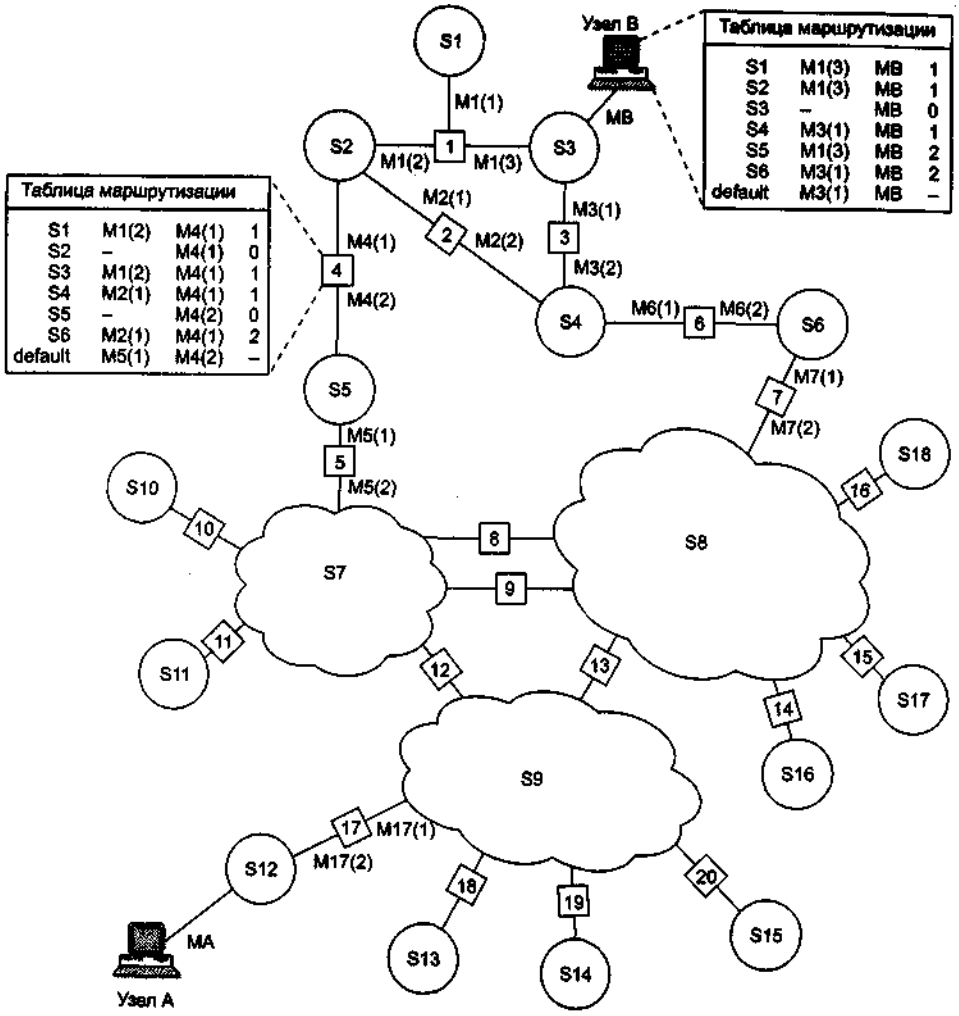


Рис. 5.2. Принципы маршрутизации в составной сети

Задачу выбора маршрута из нескольких возможных решают маршрутизаторы, а также конечные узлы. Маршрут выбирается на основании имеющейся у этих устройств информации о текущей конфигурации сети, а также на основании указанного критерия выбора маршрута. Обычно в качестве критерия выступает задержка прохождения маршрута отдельным пакетом или средняя пропускная способность маршрута для последовательности пакетов. Часто также используется весьма простой критерий, учитывающий только количество пройденных в маршруте промежуточных маршрутизаторов {хопов).

Чтобы по адресу сети назначения можно было бы выбрать рациональный маршрут дальнейшего следования пакета, каждый конечный узел и маршрутизатор анализируют специальную информационную структуру, которая называется таблицей маршрутизации. Используя условные обозначения для сетевых адресов маршрутизаторов и номеров сетей в том виде, как они приведены на рис. 5.2, посмотрим, как могла бы выглядеть таблица маршрутизации, например, в маршрутизаторе 4 (табл. 5.1).

Таблица 5.1. Таблица маршрутизации маршрутизатора 4

Номер сети назначения	Сетевой адрес следующего маршрутизарора	Сетевой адрес выходного порта	Расстояние до сети назначения
S1	M1(2)	M4(1)	1
S2	–	M4(1)	0 (подсоединена)
S3	M1(2)	M4(1)	1
S4	M2(1)	M4(1)	1
S5	–	M4(2)	0 (подсоединена)
S6	M2(1)	M4(1)	2

Default	M5(l)	M4(2)	–

ПРИМЕЧАНИЕ Таблица 5.1 значительно упрощена по сравнению с реальными таблицами, например, отсутствуют столбцы с масками, признаками состояния маршрута, временем, в течение которого действительны записи данной таблицы (их применение будет рассмотрено позже). Кроме того, как уже было сказано, здесь указаны адреса сетей условного формата, не соответствующие какому-либо определенному сетевому протоколу. Тем не менее эта таблица содержит основные поля, имеющиеся в реальных таблицах при использовании конкретных сетевых протоколов, таких как IP, IPX или X.25.

В первом столбце таблицы перечисляются номера сетей, входящих в интерсеть. В каждой строке таблицы следом за номером сети указывается сетевой адрес следующего маршрутизатора (более точно, сетевой адрес соответствующего порта следующего маршрутизатора), на который надо направить пакет, чтобы тот передвигался по направлению к сети с данным номером по рациональному маршруту.

Когда на маршрутизатор поступает новый пакет, номер сети назначения, извлеченный из поступившего кадра, последовательно сравнивается с номерами сетей из каждой строки таблицы. Строка с совпавшим номером сети указывает, на какой ближайший маршрутизатор следует направить пакет. Например, если на какой-либо порт маршрутизатора 4 поступает пакет, адресованный в сеть S6, то из таблицы маршрутизации следует, что адрес следующего маршрутизатора – M2(1), то есть очередным этапом движения данного пакета будет движение к порту 1 маршрутизатора 2.

Поскольку пакет может быть адресован в любую сеть составной сети, может показаться, что каждая таблица маршрутизации должна иметь записи обо всех сетях, входящих в составную часть. Но при таком подходе в случае крупной сети объем таблиц маршрутизации может оказаться очень большим, что повлияет на время ее просмотра, потребует много места для хранений и т. п. Поэтому на практике число записей в таблице маршрутизации стараются уменьшить за счет использования специальной записи – «*маршрутизатор по умолчанию*» (*default*). Действительно, если принять во внимание топологию составной сети, то в таблицах маршрутизаторов, находящихся на периферии составной сети, достаточно записать номера сетей, непосредственно подсоединенных к данному маршрутизатору или расположенных поблизости, на тупиковых маршрутах. Обо всех же остальных сетях можно сделать в таблице единственную запись, указывающую на маршрутизатор, через который пролегает путь ко всем этим сетям. Такой маршрутизатор называется маршрутизатором по умолчанию, а вместо номера сети в соответствующей строке помещается особая запись, например default. В нашем примере таким маршрутизатором по умолчанию для сети S5 является маршрутизатор 5, точнее его порт M5(1). Это означает, что путь из сети S5 почти ко всем сетям большой составной сети пролегает через этот порт маршрутизатора.

Перед тем как передать пакет следующему маршрутизатору, текущий маршрутизатор должен определить, на какой из нескольких собственных портов он должен поместить данный пакет. Для этого служит третий столбец таблицы маршрутизации. Еще раз подчеркнем, что каждый порт идентифицируется собственным сетевым адресом.

Некоторые реализации сетевых протоколов допускают наличие в таблице маршрутизации сразу нескольких строк, соответствующих одному и тому же адресу сети назначения. В этом случае при выборе маршрута принимается во внимание столбец «Расстояние до сети назначения». При этом под расстоянием понимается любая метрика, используемая в соответствии с заданным в сетевом пакете критерием (часто называемым классом сервиса). Расстояние может измеряться хопами, временем прохождения пакета по линиям связи, какой-либо характеристикой надежности линий связи на данном маршруте или другой величиной, отражающей качество данного маршрута по отношению к заданному критерию. Если маршрутизатор поддерживает несколько классов сервиса пакетов, то таблица маршрутов составляется и применяется отдельно для каждого вида сервиса (критерия выбора маршрута).

В табл. 5.1 расстояние между сетями измерялось хопами. Расстояние для сетей, непосредственно подключенных к портам маршрутизатора, здесь принимается равным 0, однако в некоторых реализациях отсчет расстояний начинается с 1.

Наличие нескольких маршрутов к одному узлу делают возможным передачу трафика к этому узлу

параллельно по нескольким каналам связи, это повышает пропускную способность и надежность сети.

Задачу маршрутизации решают не только промежуточные узлы-маршрутизаторы, но и конечные узлы – компьютеры. Средства сетевого уровня, установленные на конечном узле, при обработке пакета должны, прежде всего, определить, направляется ли он в другую сеть или адресован какому-нибудь узлу данной сети. Если номер сети назначения совпадает с номером данной сети, то для данного пакета не требуется решать задачу маршрутизации. Если же номера сетей отправления и назначения не совпадают, то маршрутизация нужна. Таблицы маршрутизации конечных узлов полностью аналогичны таблицам маршрутизации, хранящимся на маршрутизаторах.

Обратимся снова к сети, изображенной на рис. 5.2. Таблица маршрутизации для конечного узла B могла бы выглядеть следующим образом (табл. 5.2). Здесь MB – сетевой адрес порта компьютера B. На основании этой таблицы конечный узел B выбирает, на какой из двух имеющихся в локальной сети S3 маршрутизаторов следует посылать тот или иной пакет.

Таблица 5.2. Таблица маршрутизации конечного узла B

Номер сети назначения	Сетевой адрес следующего маршрутизатора	Сетевой адрес выходного порта	Расстояние до сети назначения
S1	M1(3)	MB	1
S2	M1(3)	MB	1
S3	–	MB	0
S4	M3(1)	MB	1
S5	M1(3)	MB	2
S6	M3(1)	MB	2
Default	M3(1)	MB	–

Конечные узлы в еще большей степени, чем маршрутизаторы, пользуются приемом маршрутизации по умолчанию. Хотя они также в общем случае имеют в своем распоряжении таблицу маршрутизации, ее объем обычно незначителен, что объясняется периферийным расположением всех конечных узлов. Конечный узел часто вообще работает без таблицы маршрутизации, имея только сведения об адресе маршрутизатора по умолчанию. При наличии одного маршрутизатора в локальной сети этот вариант – единственно возможный для всех конечных узлов. Но даже при наличии нескольких маршрутизаторов в локальной сети, когда перед конечным узлом стоит проблема их выбора, задание маршрута по умолчанию часто используется в компьютерах для сокращения объема их таблицы маршрутизации.

Ниже помещена таблица маршрутизации другого конечного узла составной сети – узла A (табл. 5.3). Компактный вид таблицы маршрутизации отражает тот факт, что все пакеты, направляемые из узла A, либо не выходят за пределы сети S12, либо непременно проходят через порт 1 маршрутизатора 17. Этот маршрутизатор и определен в таблице маршрутизации в качестве маршрутизатора по умолчанию.

Таблица 5.3. Таблица маршрутизации конечного узла A

Номер сети назначения	Сетевой адрес следующего маршрутизатора	Сетевой адрес выходного порта	Расстояние до сети назначения
S12	-	MA	0
Default	M17(1)	MA	-

Еще одним отличием работы маршрутизатора и конечного узла при выборе маршрута является способ построения таблицы маршрутизации. Если маршрутизаторы обычно автоматически создают таблицы маршрутизации, обмениваясь служебной информацией, то для конечных узлов таблицы маршрутизации часто создаются вручную администраторами и хранятся в виде постоянных файлов на дисках.

5.1.4. Протоколы маршрутизации

Задача маршрутизации решается на основе анализа таблиц маршрутизации, размещенных во всех

маршрутизаторах и конечных узлах сети. Каким же образом происходит формирование этих таблиц? Какими средствами обеспечивается адекватность содержащейся в них информации постоянно изменяющейся структуре сети? Основная работа по созданию таблиц маршрутизации выполняется автоматически, но и возможность вручную скорректировать или дополнить таблицу тоже, как правило, предусматривается.

Для автоматического построения таблиц маршрутизации маршрутизаторы обмениваются информацией о топологии составной сети в соответствии со специальным служебным протоколом. Протоколы этого типа называются протоколами маршрутизации (или маршрутизирующими протоколами). Протоколы маршрутизации (например, RIP, OSPF, NLSP) следует отличать от собственно сетевых протоколов (например, IP, IPX). И те и другие выполняют функции сетевого уровня модели OSI – участвуют в доставке пакетов адресату через разнородную составную сеть. Но в то время как первые собирают и передают по сети чисто служебную информацию, вторые предназначены для передачи пользовательских данных, как это делают протоколы канального уровня. Протоколы маршрутизации используют сетевые протоколы как транспортное средство. При обмене маршрутной информацией пакеты протокола маршрутизации помещаются в поле данных пакетов сетевого уровня или даже транспортного уровня, поэтому с точки зрения вложенности пакетов протоколы маршрутизации формально следовало бы отнести к более высокому уровню, чем сетевой.

В том, что маршрутизаторы для принятия решения о продвижении пакета обращаются к адресным таблицам, можно увидеть их некоторое сходство с мостами и коммутаторами. Однако природа используемых ими адресных таблиц сильно различается. Вместо MAC-адресов в таблицах маршрутизации указываются номера сетей, которые соединяются в интерсеть. Другим отличием таблиц маршрутизации от адресных таблиц мостов является способ их создания. В то время как мост строит таблицу, пассивно наблюдая за проходящими через него информационными кадрами, посылаемыми конечными узлами сети друг другу, маршрутизаторы по своей инициативе обмениваются специальными служебными пакетами, сообщая соседям об известных им сетях в интерсети, маршрутизаторах и о связях этих сетей с маршрутизаторами. Обычно учитывается не только топология связей, но и их пропускная способность и состояние. Это позволяет маршрутизаторам быстрее адаптироваться к изменениям конфигурации сети, а также правильно передавать пакеты в сетях с произвольной топологией, допускающей наличие замкнутых контуров.

С помощью протоколов маршрутизации маршрутизаторы составляют карту связей сети той или иной степени подробности. На основании этой информации для каждого номера сети принимается решение о том, какому следующему маршрутизатору надо передавать пакеты, направляемые в эту сеть, чтобы маршрут оказался рациональным. Результаты этих решений заносятся в таблицу маршрутизации. При изменении конфигурации сети некоторые записи в таблице становятся недействительными. В таких случаях пакеты, отправленные по ложным маршрутам, могут зацикливаться и теряться. От того, насколько быстро протокол маршрутизации приводит в соответствие содержимое таблицы реальному состоянию сети, зависит качество работы всей сети.

Протоколы маршрутизации могут быть построены на основе разных алгоритмов, отличающихся способами построения таблиц маршрутизации, способами выбора наилучшего маршрута и другими особенностями своей работы.

Во всех описанных выше примерах при выборе рационального маршрута определялся только следующий (ближайший) маршрутизатор, а не вся последовательность маршрутизаторов от начального до конечного узла. В соответствии с этим подходом маршрутизация выполняется по распределенной схеме – каждый маршрутизатор ответственен за выбор только одного шага маршрута, а окончательный маршрут складывается в результате работы всех маршрутизаторов, через которые проходит данный пакет. Такие алгоритмы маршрутизации называются *одношаговыми*.

Существует и прямо противоположный, многошаговый подход – *маршрутизация от источника* (*Source Routing*). В соответствии с ним узел-источник задает в отправляемом в сеть пакете полный маршрут его следования через все промежуточные маршрутизаторы. При использовании многошаговой маршрутизации нет необходимости строить и анализировать таблицы маршрутизации. Это ускоряет прохождение пакета по сети, разгружает маршрутизаторы, но при этом большая нагрузка ложится на конечные узлы. Эта схема в вычислительных сетях применяется сегодня гораздо реже, чем схема распределенной одношаговой маршрутизации. Однако в новой версии протокола IP наряду с классической одношаговой маршрутизацией будет разрешена и маршрутизация от источника.

Одношаговые алгоритмы в зависимости от способа формирования таблиц маршрутизации делятся на

три класса:

- алгоритмы фиксированной (или статической) маршрутизации;
- алгоритмы простой маршрутизации;
- алгоритмы адаптивной (или динамической) маршрутизации.

В алгоритмах *фиксированной маршрутизации* все записи в таблице маршрутизации являются статическими. Администратор сети сам решает, на какие маршрутизаторы надо передавать пакеты с теми или иными адресами, и вручную (например, с помощью утилиты route ОС Unix или Windows NT) заносит соответствующие записи в таблицу маршрутизации. Таблица, как правило, создается в процессе загрузки, в дальнейшем она используется без изменений до тех пор, пока ее содержимое не будет отредактировано вручную. Такие исправления могут понадобиться, например, если в сети отказывает какой-либо маршрутизатор и его функции возлагаются на другой маршрутизатор. Различают одномаршрутные таблицы, в которых для каждого адресата задан один путь, и многомаршрутные таблицы, определяющие несколько альтернативных путей для каждого адресата. В многомаршрутных таблицах должно быть задано правило выбора одного из маршрутов. Чаще всего один путь является основным, а остальные – резервными. Понятно, что алгоритм фиксированной маршрутизации с его ручным способом формирования таблиц маршрутизации приемлем только в небольших сетях с простой топологией. Однако этот алгоритм может быть эффективно использован и для работы на магистралях крупных сетей, так как сама магистраль может иметь простую структуру с очевидными наилучшими путями следования пакетов в подсети, присоединенные к магистрали.

В алгоритмах *простой маршрутизации* таблица маршрутизации либо вовсе не используется, либо строится без участия протоколов маршрутизации. Выделяют три типа простой маршрутизации:

- *случайная маршрутизация*, когда прибывший пакет посылается в первом попавшем случайном направлении, кроме исходного;
- *лавинная маршрутизация*, когда пакет широковещательно посылается по всем возможным направлениям, кроме исходного (аналогично обработке мостами кадров с неизвестным адресом);
- *маршрутизация по предыдущему опыту*, когда выбор маршрута осуществляется по таблице, но таблица строится по принципу моста путем анализа адресных полей пакетов, появляющихся на входных портах.

Самыми распространенными являются алгоритмы *адаптивной* (или *динамической*) *маршрутизации*. Эти алгоритмы обеспечивают автоматическое обновление таблиц маршрутизации после изменения конфигурации сети. Протоколы, построенные на основе адаптивных алгоритмов, позволяют всем маршрутизаторам собирать информацию о топологии связей в сети, оперативно отрабатывая все изменения конфигурации связей. В таблицах маршрутизации при адаптивной маршрутизации обычно имеется информация об интервале времени, в течение которого данный маршрут будет оставаться действительным. Это время называют *временем жизни маршрута* (*Time To Live, TTL*).

Адаптивные алгоритмы обычно имеют распределенный характер, который выражается в том, что в сети отсутствуют какие-либо выделенные маршрутизаторы, которые собирали бы и обобщали топологическую информацию: эта работа распределена между всеми маршрутизаторами.

ПРИМЕЧАНИЕ В последнее время наметилась тенденция использовать так называемые серверы маршрутов. Сервер маршрутов собирает маршрутную информацию, а затем раздает ее по запросам маршрутизаторам, которые освобождаются в этом случае от функции создания таблиц маршрутизации, либо создают только части этих таблиц. Появились специальные протоколы взаимодействия маршрутизаторов с серверами маршрутов, например Next Hop Resolution Protocol (NHRP).

Адаптивные алгоритмы маршрутизации должны отвечать нескольким важным требованиям. Во-первых, они должны обеспечивать, если не оптимальность, то хотя бы рациональность маршрута. Во-вторых, алгоритмы должны быть достаточно простыми, чтобы при их реализации не тратилось слишком много сетевых ресурсов, в частности они не должны требовать слишком большого объема вычислений или порождать интенсивный служебный трафик. И наконец, алгоритмы маршрутизации должны обладать свойством сходимости, то есть всегда приводить к однозначному результату за приемлемое время.

Адаптивные протоколы обмена маршрутной информацией, применяемые в настоящее время в

вычислительных сетях, в свою очередь делятся на две группы, каждая из которых связана с одним из следующих типов алгоритмов:

- дистанционно-векторные алгоритмы (Distance Vector Algorithms, DVA);
- алгоритмы состояния связей (Link State Algorithms, LSA).

В алгоритмах *дистанционно-векторного типа* каждый маршрутизатор периодически и широковещательно рассылает по сети вектор, компонентами которого являются расстояния от данного маршрутизатора до всех известных ему сетей. Под расстоянием обычно понимается число хопов. Возможна и другая метрика, учитывающая не только число промежуточных маршрутизаторов, но и время прохождения пакетов по сети между соседними маршрутизаторами. При получении вектора от соседа маршрутизатор наращивает расстояния до указанных в векторе сетей на расстояние до данного соседа. Получив вектор от соседнего маршрутизатора, каждый маршрутизатор добавляет к нему информацию об известных ему других сетях, о которых он узнал непосредственно (если они подключены к его портам) или из аналогичных объявлений других маршрутизаторов, а затем снова рассылает новое значение вектора по сети. В конце концов, каждый маршрутизатор узнает информацию обо всех имеющихся в интерсети сетях и о расстоянии до них через соседние маршрутизаторы.

Дистанционно-векторные алгоритмы хорошо работают только в небольших сетях. В больших сетях они засоряют линии связи интенсивным широковещательным трафиком, к тому же изменения конфигурации могут отрабатываться по этому алгоритму не всегда корректно, так как маршрутизаторы не имеют точного представления о топологии связей в сети, а располагают только обобщенной информацией – вектором дистанций, к тому же полученной через посредников. Работа маршрутизатора в соответствии с дистанционно-векторным протоколом напоминает работу моста, так как точной топологической картины сети такой маршрутизатор не имеет.

Наиболее распространенным протоколом, основанным на дистанционно-векторном алгоритме, является протокол RIP, который распространен в двух версиях – RIP IP, работающий с протоколом IP, и RIP IPX, работающий с протоколом IPX.

Алгоритмы состояния связей обеспечивают каждый маршрутизатор информацией, достаточной для построения точного графа связей сети. Все маршрутизаторы работают на основании одинаковых графов, что делает процесс маршрутизации более устойчивым к изменениям конфигурации. «Широковещательная» рассылка (то есть передача пакета всем непосредственным соседям маршрутизатора) используется здесь только при изменениях состояния связей, что происходит в надежных сетях не так часто. Вершинами графа являются как маршрутизаторы, так и объединяемые ими сети. Распространяемая по сети информация состоит из описания связей различных типов: маршрутизатор-маршрутизатор, маршрутизатор–сеть.

Чтобы понять, в каком состоянии находятся линии связи, подключенные к его портам, маршрутизатор периодически обменивается короткими пакетами HELLO со своими ближайшими соседями. Этот служебный трафик также засоряет сеть, но не в такой степени как, например, RIP-пакеты, так как пакеты HELLO имеют намного меньший объем.

Протоколами, основанными на алгоритме состояния связей, являются протоколы IS-IS (Intermediate System to Intermediate System) стека OSI, OSPF (Open Shortest Path First) стека TCP/IP и недавно реализованный протокол NLSP стека Novell.

5.1.5. Функции маршрутизатора

Основная функция маршрутизатора – чтение заголовков пакетов сетевых протоколов, принимаемых и буферизуемых по каждому порту (например, IPX, IP, AppleTalk или DECnet), и принятие решения о дальнейшем маршруте следования пакета по его сетевому адресу, включающему, как правило, номер сети и номер узла.

Функции маршрутизатора могут быть разбиты на 3 группы в соответствии с уровнями модели OSI (рис. 5.3).

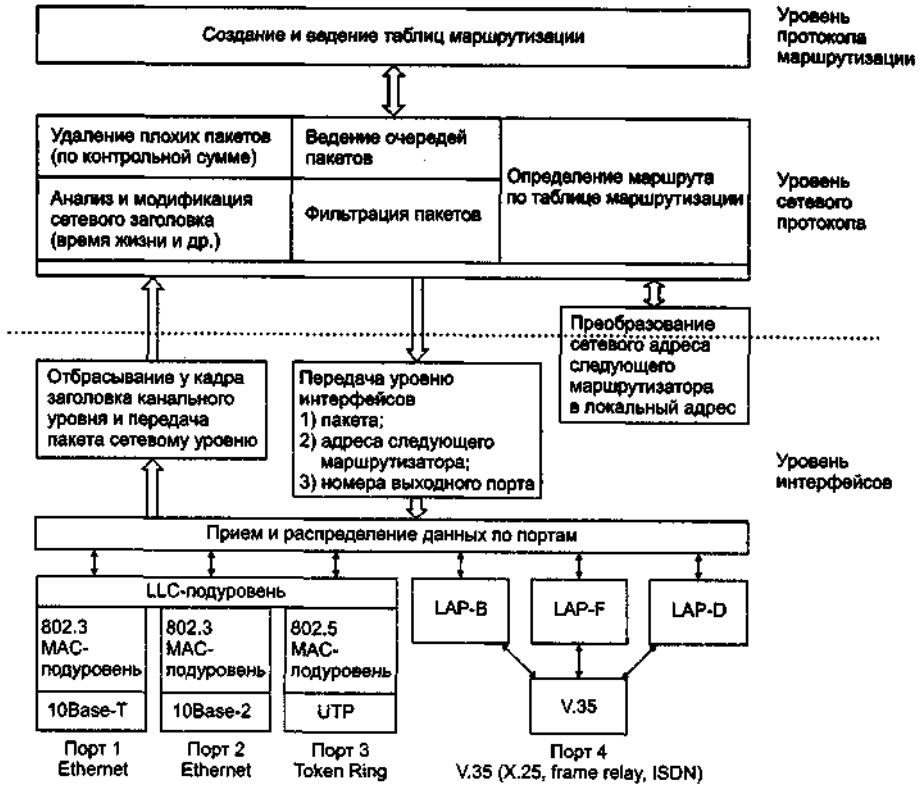


Рис. 5.3. Функциональная модель маршрутизатора

Уровень интерфейсов

На нижнем уровне маршрутизатор, как и любое устройство, подключенное к сети, обеспечивает физический интерфейс со средой передачи, включая согласование уровней электрических сигналов, линейное и логическое кодирование, оснащение определенным типом разъема. В разных моделях маршрутизаторов часто предусматриваются различные наборы физических интерфейсов, представляющих собой комбинацию портов для подсоединения локальных и глобальных сетей. С каждым интерфейсом для подключения локальной сети неразрывно связан определенный протокол канального уровня – например, Ethernet, Token Ring, FDDI. Интерфейсы для присоединения к глобальным сетям чаще всего определяют только некоторый стандарт физического уровня, над которым в маршрутизаторе могут работать различные протоколы канального уровня. Например, глобальный порт может поддерживать интерфейс V.35, над которым могут работать протоколы канального уровня: LAP-B (используемый в сетях X.25), LAP-F (используемый в сетях frame relay), LAP-D (используемый в сетях ISDN). Разница между интерфейсами локальных и глобальных сетей объясняется тем, что технологии локальных сетей работают по собственным стандартам физического уровня, которые не могут, как правило, использоваться в других технологиях, поэтому интерфейс для локальной сети представляет собой сочетание физического и канального уровней и носит название по имени соответствующей технологии – например, интерфейс Ethernet.

Интерфейсы маршрутизатора выполняют полный набор функций физического и канального уровней по передаче кадра, включая получение доступа к среде (если это необходимо), формирование битовых сигналов, прием кадра, подсчет его контрольной суммы и передачу поля данных кадра верхнему уровню, в случае если контрольная сумма имеет корректное значение.

ПРИМЕЧАНИЕ Как и любой конечный узел, каждый порт маршрутизатора имеет собственный аппаратный адрес (в локальных сетях MAC-адрес), по которому ему и направляются кадры, требующие маршрутизации, другими узлами сети.

Перечень физических интерфейсов, которые поддерживает та или иная модель маршрутизатора, является его важнейшей потребительской характеристикой. Маршрутизатор должен поддерживать все протоколы канального и физического уровней, используемые в каждой из сетей, к которым он будет

непосредственно присоединен. На рис. 5.3 показана функциональная модель маршрутизатора с четырьмя портами, реализующими следующие физические интерфейсы: 10Base-T и 10Base-2 для двух портов Ethernet, UTP для Token Ring и V.35, над которым могут работать протоколы LAP-B, LAP-D или LAP-F, обеспечивая подключение к сетям X.25, ISDN или frame relay.

Кадры, которые поступают на порты маршрутизатора, после обработки соответствующими протоколами физического и канального уровней, освобождаются от заголовков канального уровня. Извлеченные из поля данных кадра пакеты передаются модулю сетевого протокола.

Уровень сетевого протокола

Сетевой протокол в свою очередь извлекает из пакета заголовок сетевого уровня и анализирует содержимое его полей. Прежде всего проверяется контрольная сумма, и если пакет пришел поврежденным, то он отбрасывается. Выполняется проверка, не превысило ли время, которое провел пакет в сети (время жизни пакета), допустимой величины. Если превысило – то пакет также отбрасывается. На этом этапе вносятся корректировки в содержимое некоторых полей, например, наращивается время жизни пакета, пересчитывается контрольная сумма.

На сетевом уровне выполняется одна из важнейших функций маршрутизатора – фильтрация трафика. Маршрутизатор, обладая более высоким интеллектом, нежели мосты и коммутаторы, позволяет задавать и может отрабатывать значительно более сложные правила фильтрации. Пакет сетевого уровня, находящийся в поле данных кадра, для мостов/коммутаторов представляется неструктурированной двоичной последовательностью. Маршрутизаторы же, программное обеспечение которых содержит модуль сетевого протокола, способны производить разбор и анализ отдельных полей пакета. Они оснащаются развитыми средствами пользовательского интерфейса, которые позволяют администратору без особых усилий задавать сложные правила фильтрации. Они, например, могут запретить прохождение в корпоративную сеть всех пакетов, кроме пакетов, поступающих из подсетей этого же предприятия. Фильтрация в данном случае производится по сетевым адресам, и все пакеты, адреса которых не входят в разрешенный диапазон, отбрасываются. Маршрутизаторы, как правило, также могут анализировать структуру сообщений транспортного уровня, поэтому фильтры могут не пропускать в сеть сообщения определенных прикладных служб, например службы telnet, анализируя поле типа протокола в транспортном сообщении.

В случае если интенсивность поступления пакетов выше интенсивности, с которой они обрабатываются, пакеты могут образовать очередь. Программное обеспечение маршрутизатора может реализовать различные дисциплины обслуживания очередей пакетов: в порядке поступления по принципу «первый пришел - первым обслужен» (First Input First Output, FIFO), случайное раннее обнаружение, когда обслуживание идет по правилу FIFO, но при достижении длиной очереди некоторого порогового значения вновь поступающие пакеты отбрасываются (Random Early Detection, RED), а также различные варианты приоритетного обслуживания.

К сетевому уровню относится основная функция маршрутизатора – определение маршрута пакета. По номеру сети, извлеченному из заголовка пакета, модуль сетевого протокола находит в таблице маршрутизации строку, содержащую сетевой адрес следующего маршрутизатора, и номер порта, на который нужно передать данный пакет, чтобы он двигался в правильном направлении. Если в таблице отсутствует запись о сети назначения пакета и к тому же нет записи о маршрутизаторе по умолчанию, то данный пакет отбрасывается.

Перед тем как передать сетевой адрес следующего маршрутизатора на канальный уровень, необходимо преобразовать его в локальный адрес той технологии, которая используется в сети, содержащей следующий маршрутизатор. Для этого сетевой протокол обращается к *протоколу разрешения адресов*. Протоколы этого типа устанавливают соответствие между сетевыми и локальными адресами либо на основании заранее составленных таблиц, либо путем рассылки широковещательных запросов. Таблица соответствия локальных адресов сетевым адресам строится отдельно для каждого сетевого интерфейса. Протоколы разрешения адресов занимают промежуточное положение между сетевым и канальным уровнями.

С сетевого уровня пакет, локальный адрес следующего маршрутизатора и номер порта маршрутизатора передаются вниз, канальному уровню. На основании указанного номера порта осуществляется коммутация с одним из интерфейсов маршрутизатора, средствами которого выполняется упаковка пакета в кадр соответствующего формата. В поле адреса назначения заголовка

кадра помещается локальный адрес следующего маршрутизатора. Готовый кадр отправляется в сеть.

Уровень протоколов маршрутизации

Сетевые протоколы активно используют в своей работе таблицу маршрутизации, но ни ее построением, ни поддержанием ее содержимого не занимаются. Эти функции выполняют протоколы маршрутизации. На основании этих протоколов маршрутизаторы обмениваются информацией о топологии сети, а затем анализируют полученные сведения, определяя наилучшие по тем или иным критериям маршруты. Результаты анализа и составляют содержимое таблиц маршрутизации.

Помимо перечисленных выше функций, на маршрутизаторы могут быть возложены и другие обязанности, например операции, связанные с фрагментацией. Более детально работа маршрутизаторов будет описана при рассмотрении конкретных протоколов сетевого уровня.

5.1.6. Реализация межсетевого взаимодействия средствами TCP/IP

В настоящее время стек TCP/IP является самым популярным средством организации составных сетей. На рис. 5.4 показана доля, которую составляет тот или иной стек протоколов в общемировой инсталляционной сетевой базе. До 1996 года бесспорным лидером был стек IPX/SPX компании Novell, но затем картина резко изменилась – стек TCP/IP по темпам роста числа установок намного стал опережать другие стеки, а с 1998 года вышел в лидеры и в абсолютном выражении. Именно поэтому дальнейшее изучение функций сетевого уровня будет проводиться на примере стека TCP/IP.

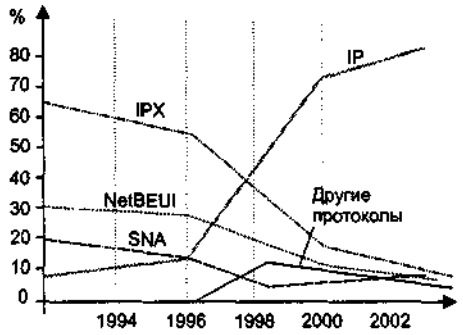


Рис. 5.4. *Стек TCP/IP становится основным средством построения составных сетей*

Многоуровневая структура стека TCP/IP

В стеке TCP/IP определены 4 уровня (рис. 5.5). Каждый из этих уровней несет на себе некоторую нагрузку по решению основной задачи – организации надежной и производительной работы составной сети, части которой построены на основе разных сетевых технологий.

Уровень I	Прикладной уровень
Уровень II	Основной (транспортный) уровень
Уровень III	Уровень межсетевого взаимодействия
Уровень IV	Уровень сетевых интерфейсов

Рис. 5.5. *Многоуровневая архитектура стека TCP/IP*

Уровень межсетевого взаимодействия

Стержнем всей архитектуры является *уровень межсетевого взаимодействия* который реализует концепцию передачи пакетов в режиме без установления соединений, то есть дейтаграммным способом. Именно этот уровень обеспечивает возможность перемещения пакетов по сети, используя тот маршрут, который в данный момент является наиболее рациональным. Этот уровень также называют уровнем internet, указывая тем самым на основную его функцию – передачу данных через составную сеть.

Основным протоколом сетевого уровня (в терминах модели OSI) в стеке является протокол IP

(Internet Protocol). Этот протокол изначально проектировался как протокол передачи пакетов в составных сетях, состоящих из большого количества локальных сетей, объединенных как локальными, так и глобальными связями. Поэтому протокол IP хорошо работает в сетях со сложной топологией, рационально используя наличие в них подсистем и экономно расходуя пропускную способность низкоскоростных линий связи. Так как протокол IP является дейтаграммным протоколом, он не гарантирует доставку пакетов до узла назначения, но старается это сделать.

К уровню межсетевого взаимодействия относятся и все протоколы, связанные с составлением и модификацией таблиц маршрутизации, такие как протоколы сбора маршрутной информации RIP (Routing Internet Protocol) и OSPF (Open Shortest Path First), а также протокол межсетевых управляющих сообщений ICMP (Internet Control Message Protocol). Последний протокол предназначен для обмена информацией об ошибках между маршрутизаторами сети и узлом-источником пакета. С помощью специальных пакетов ICMP сообщает о невозможности доставки пакета, о превышении времени жизни или продолжительности сборки пакета из фрагментов, об аномальных величинах параметров, об изменении маршрута пересылки и типа обслуживания, о состоянии системы и т. п.

Основной уровень

Поскольку на сетевом уровне не устанавливаются соединения, то нет никаких гарантий, что все пакеты будут доставлены в место назначения целыми и невредимыми или придут в том же порядке, в котором они были отправлены. Эту задачу – обеспечение надежной информационной связи между двумя конечными узлами – решает *основной уровень* стека TCP/IP, называемый также *транспортным*.

На этом уровне функционируют протокол управления передачей TCP (Transmission Control Protocol) и протокол дейтаграмм пользователя UDP (User Datagram Protocol). Протокол TCP обеспечивает надежную передачу сообщений между удаленными прикладными процессами за счет образования логических соединений. Этот протокол позволяет равноранговым объектам на компьютере-отправителе и компьютере-получателе поддерживать обмен данными в дуплексном режиме. TCP позволяет без ошибок доставить сформированный на одном из компьютеров поток байт в любой другой компьютер, входящий в составную сеть. TCP делит поток байт на части – *сегменты* и передает их ниже лежащему уровню межсетевого взаимодействия. После того как эти сегменты будут доставлены средствами уровня межсетевого взаимодействия в пункт назначения, протокол TCP снова соберет их в непрерывный поток байт.

Протокол UDP обеспечивает передачу прикладных пакетов дейтаграммным способом, как и главный протокол уровня межсетевого взаимодействия IP, и выполняет только функции связующего звена (мультиплексора) между сетевым протоколом и многочисленными службами прикладного уровня или пользовательскими процессами.

Прикладной уровень

Прикладной уровень объединяет все службы, предоставляемые системой пользовательским приложениям. За долгие годы использования в сетях различных стран и организаций стек TCP/IP накопил большое количество протоколов и служб прикладного уровня. Прикладной уровень реализуется программными системами, построенными в архитектуре клиент-сервер, базирующимися на протоколах нижних уровней. В отличие от протоколов остальных трех уровней, протоколы прикладного уровня занимаются деталями конкретного приложения и «не интересуются» способами передачи данных по сети. Этот уровень постоянно расширяется за счет присоединения к старым, прошедшим многолетнюю эксплуатацию сетевым службам типа Telnet, FTP, TFTP, DNS, SNMP сравнительно новых служб таких, например, как протокол передачи гипертекстовой информации HTTP.

Уровень сетевых интерфейсов

Идеологическим отличием архитектуры стека TCP/IP от многоуровневой организации других стеков является интерпретация функций самого нижнего уровня – *уровня сетевых интерфейсов*. Протоколы этого уровня должны обеспечивать интеграцию в составную сеть других сетей, причем задача ставится так: сеть TCP/IP должна иметь средства включения в себя любой другой сети, какую бы внутреннюю технологию передачи данных эта сеть не использовала. Отсюда следует, что этот уровень нельзя

определить раз и навсегда. Для каждой технологии, включаемой в составную сеть подсети, должны быть разработаны собственные интерфейсные средства. К таким интерфейсным средствам относятся протоколы инкапсуляции IP-пакетов уровня межсетевого взаимодействия в кадры локальных технологий. Например, документ RFC 1042 определяет способы инкапсуляции IP-пакетов в кадры технологий IEEE 802. Для этих целей должен использоваться заголовок LLC/ SNAP, причем в поле Type заголовка SNAP должен быть указан код 0x0800. Только для протокола Ethernet в RFC 1042 сделано исключение – помимо заголовка LLC/ SNAP разрешается использовать кадр Ethernet DIX, не имеющий заголовка LLC, зато имеющий поле Type. В сетях Ethernet предпочтительным является инкапсуляция IP-пакета в кадр Ethernet DIX.

Уровень сетевых интерфейсов в протоколах TCP/IP не регламентируется, но он поддерживает все популярные стандарты физического и канального уровней: для локальных сетей это Ethernet, Token Ring, FDDI, Fast Ethernet, Gigabit Ethernet, lOOVG-AnyLAN, для глобальных сетей – протоколы соединений «точка-точка» SLIP и PPP, протоколы территориальных сетей с коммутацией пакетов X.25, frame relay. Разработана также специальная спецификация, определяющая использование технологии ATM в качестве транспорта канального уровня. Обычно при появлении новой технологии локальных или глобальных сетей она быстро включается в стек TCP/IP за счет разработки соответствующего RFC, определяющего метод инкапсуляции IP-пакетов в ее кадры (спецификация RFC 1577, определяющая работу IP через сети ATM, появилась в 1994 году вскоре после принятия основных стандартов этой технологии).

Соответствие уровней стека TCP/IP семиуровневой модели ISO/OSI

Так как стек TCP/IP был разработан до появления модели взаимодействия открытых систем ISO/OSI, то, хотя он также имеет многоуровневую структуру, соответствие уровней-стека TCP/IP уровням модели OSI достаточно условно (рис. 5,6).

Рассматривая многоуровневую архитектуру TCP/IP, можно выделить в ней, подобно архитектуре OSI, уровни, функции которых зависят от конкретной технической реализации сети, и уровни, функции которых ориентированны на работу с приложениями (рис. 5.7).

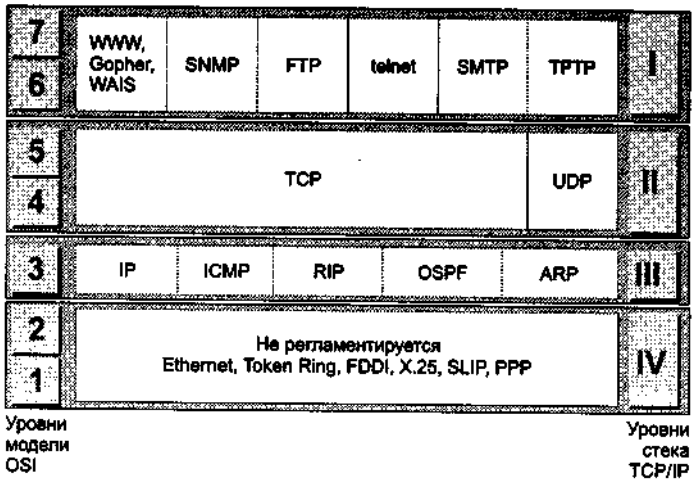


Рис. 5.6. Соответствие уровней стека TCP/IP семиуровневой модели OSI

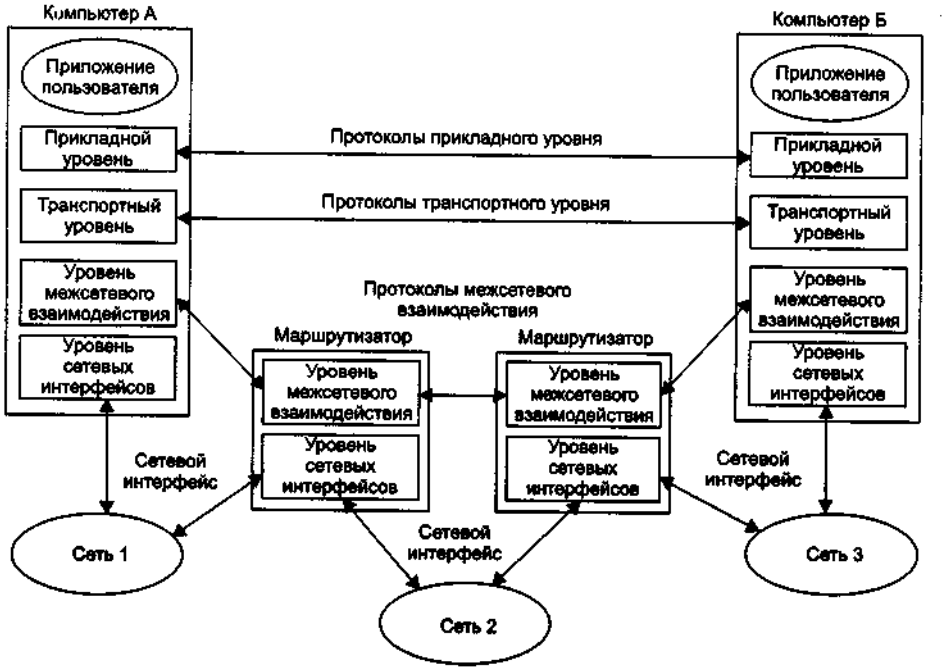


Рис. 5.7. Сетезависимые и сетенезависимые уровни стека TCP/IP

Протоколы прикладного уровня стека TCP/IP работают на компьютерах, выполняющих приложения пользователей. Даже полная смена сетевого оборудования в общем случае не должна влиять на работу приложений, если они получают доступ к сетевым возможностям через протоколы прикладного уровня.

Протоколы транспортного уровня уже более зависят от сети, так как они реализуют интерфейс к уровням, непосредственно организующим передачу данных по сети. Однако, подобно протоколам прикладного уровня, программные модули, реализующие протоколы транспортного уровня, устанавливаются только на конечных узлах. Протоколы двух нижних уровней являются сетезависимыми, а следовательно, программные модули протоколов межсетевого уровня и уровня сетевых интерфейсов устанавливаются как на конечных узлах составной сети, так и на маршрутизаторах.

Каждый коммуникационный протокол оперирует с некоторой единицей передаваемых данных. Названия этих единиц иногда закрепляются стандартом, а чаще просто определяются традицией В стеке TCP/IP за многие годы его существования образовалась устоявшаяся терминология в этой области (рис 5.8).

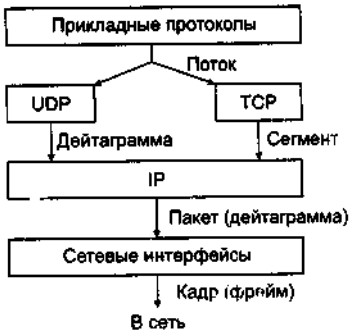


Рис. 5.8. Название единиц данных, используемые в TCP/IP

Потоком называют данные, поступающие от приложений на вход протоколов транспортного уровня TCP и UDP.

Протокол TCP нарезает из потока данных *сегменты*.

Единицу данных протокола UDP часто называют *дейтаграммой* (или датаграммой). Дейтаграмма – это общее название для единиц данных, которыми оперируют протоколы без установления соединений. К таким протоколам относится и протокол межсетевого взаимодействия IP.

Дейтаграмму протокола IP называют также *пакетом*.

В стеке TCP/IP принято называть *кадрами* (*фреймами*) единицы данных протоколов, на основе

которых 1Р-пакегы переносятся через подсети составной сети При этом не имеет значения, какое название используется для этой единицы данных в локальной технологии.

Выводы

- Составная сеть (internetwork или internet) – это совокупность нескольких сетей, называемых также подсетями (subnet), которые соединяются между собой маршрутизаторами. Организация совместной транспортной службы в составной сети называется межсетевым взаимодействием (internetworking).
- В функции сетевого уровня входит: передача пакетов между конечными узлами в составных сетях, выбор маршрута, согласование локальных технологий отдельных подсетей.
- Маршрут – это последовательность маршрутизаторов, которые должен пройти пакет от отправителя до пункта назначения. Задачу выбора маршрута из нескольких возможных решают маршрутизаторы и конечные узлы на основе таблиц маршрутизации. Записи в таблицу могут заноситься вручную администратором и автоматически протоколами маршрутизации.
- Протоколы маршрутизации (например, RIP или OSPF) следует отличать от собственно сетевых протоколов (например, IP или IPX). В то время как первые собирают и передают по сети чисто служебную информацию о возможных маршрутах, вторые предназначены для передачи пользовательских данных.
- Сетевые протоколы и протоколы маршрутизации реализуются в виде программных модулей на конечных узлах-компьютерах и на промежуточных узлах-маршрутизаторах.
- Маршрутизатор представляет собой сложное многофункциональное устройство, в задачи которого входит: построение таблицы маршрутизации, определение на ее основе маршрута, буферизация, фрагментация и фильтрация поступающих пакетов, поддержка сетевых интерфейсов. Функции маршрутизаторов могут выполнять как специализированные устройства, так и универсальные компьютеры с соответствующим программным обеспечением.
- Для алгоритмов маршрутизации характерны одношаговый и многошаговый подходы. Одношаговые алгоритмы делятся на алгоритмы фиксированной, простой и адаптивной маршрутизации. Адаптивные протоколы маршрутизации являются наиболее распространенными и в свою очередь могут быть основаны на дистанционно-векторных алгоритмах и алгоритмах состояния связей.
- Наибольшее распространение для построения составных сетей в последнее время получил стек TCP/IP. Стек TCP/IP имеет 4 уровня: прикладной, основной, уровень межсетевого взаимодействия и уровень сетевых интерфейсов. Соответствие уровней стека TCP/IP уровням модели OSI достаточно условно.
- *Прикладной уровень* объединяет все службы, предоставляемые системой пользовательским приложениям: традиционные сетевые службы типа telnet, FTP, TFTP, DNS, SNMP, а также сравнительно новые, такие, например, как протокол передачи гипертекстовой информации HTTP.
- *На основном уровне* стека TCP/IP, называемом также транспортным, функционируют протоколы TCP и UDP. Протокол управления передачей TCP решает задачу обеспечения надежной информационной связи между двумя конечными узлами. Дейтаграммный протокол UDP используется как экономичное средство связи уровня межсетевого взаимодействия с прикладным уровнем.
- *Уровень межсетевого взаимодействия* реализует концепцию коммутации пакетов в режиме без установления соединений. Основными протоколами этого уровня являются дейтаграммный протокол IP и протоколы маршрутизации (RIP, OSPF, BGP и др.). Вспомогательную роль выполняют протокол межсетевых управляющих сообщений ICMP, протокол группового управления IGMP и протокол разрешения адресов ARP.
- Протоколы *уровня сетевых интерфейсов* обеспечивают интеграцию в составную сеть других сетей. Этот уровень не регламентируется, но поддерживает все популярные стандарты физического и канального уровней: для локальных сетей – Ethernet, Token Ring, FDDI и т. д.,

для глобальных сетей – X.25, frame relay, PPP, ISDN и т. д.

- В стеке TCP/IP для именования единиц передаваемых данных на разных уровнях используют разные названия: поток, сегмент, дейтаграмма, пакет, кадр.

5.2. Адресация в IP-сетях

5.2.1. Типы адресов стека TCP/IP

В стеке TCP/IP используются три типа адресов: локальные (называемые также аппаратными), IP-адреса и символьные доменные имена.

В терминологии TCP/IP под *локальным адресом* понимается такой тип адреса, который используется средствами базовой технологии для доставки данных в пределах подсети, являющейся элементом составной интерсети. В разных подсетях допустимы разные сетевые технологии, разные стеки протоколов, поэтому при создании стека TCP/IP предполагалось наличие разных типов локальных адресов. Если подсетью интерсети является локальная сеть, то локальный адрес – это MAC-адрес. MAC-адрес назначается сетевым адаптерам и сетевым интерфейсам маршрутизаторов. MAC-адреса назначаются производителями оборудования и являются уникальными, так как управляются централизованно. Для всех существующих технологий локальных сетей MAC-адрес имеет формат 6 байт, например 11-AO-17-3D-BC-01. Однако протокол IP может работать и над протоколами более высокого уровня, например над протоколом IPX или X.25. В этом случае локальными адресами для протокола IP соответственно будут адреса IPX и X.25. Следует учесть, что компьютер в локальной сети может иметь несколько локальных адресов даже при одном сетевом адаптере. Некоторые сетевые устройства не имеют локальных адресов. Например, к таким устройствам относятся глобальные порты маршрутизаторов, предназначенные для соединений типа «точка-точка».

IP-адреса представляют собой основной тип адресов, на основании которых сетевой уровень передает пакеты между сетями. Эти адреса состоят из 4 байт, например 109.26.17.100. IP-адрес назначается администратором во время конфигурирования компьютеров и маршрутизаторов. IP-адрес состоит из двух частей: номера сети и номера узла. Номер сети может быть выбран администратором произвольно, либо назначен по рекомендации специального подразделения Internet (Internet Network Information Center, InterNIC), если сеть должна работать как составная часть Internet. Обычно поставщики услуг Internet получают диапазоны адресов у подразделений InterNIC, а затем распределяют их между своими абонентами. Номер узла в протоколе IP назначается независимо от локального адреса узла. Маршрутизатор по определению входит сразу в несколько сетей. Поэтому каждый порт маршрутизатора имеет собственный IP-адрес. Конечный узел также может входить в несколько IP-сетей. В этом случае компьютер должен иметь несколько IP-адресов, по числу сетевых связей. Таким образом, IP-адрес характеризует не отдельный компьютер или маршрутизатор, а одно сетевое соединение.

Символьные доменные имена. Символьные имена в IP-сетях называются доменными и строятся по иерархическому признаку. Составляющие полного символьного имени в IP-сетях разделяются точкой и перечисляются в следующем порядке: сначала простое имя конечного узла, затем имя группы узлов (например, имя организации), затем имя более крупной группы (поддомена) и так до имени домена самого высокого уровня (например, домена объединяющего организации по географическому принципу: RU – Россия, UK – Великобритания, SU – США). Примеров доменного имени может служить имя base2.sales.zil.ru. Между доменным именем и IP-адресом узла нет никакого алгоритмического соответствия, поэтому необходимо использовать какие-то дополнительные таблицы или службы, чтобы узел сети однозначно определялся как по доменному имени, так и по IP-адресу. В сетях TCP/IP используется специальная распределенная служба Domain Name System (DNS), которая устанавливает это соответствие на основании создаваемых администраторами сети таблиц соответствия. Поэтому доменные имена называют также DNS-именами.

5.2.2. Классы IP-адресов

IP-адрес имеет длину 4 байта и обычно записывается в виде четырех чисел, представляющих значения каждого байта в десятичной форме и разделенных точками, например, 128.10.2.30 – традиционная десятичная форма представления адреса, а 10000000 00001010 00000010 00011110 -

двоичная форма представления этого же адреса.

Адрес состоит из двух логических частей – номера сети и номера узла в сети. Какая часть адреса относится к номеру сети, а какая – к номеру узла, определяется значениями первых бит адреса. Значения этих бит являются также признаками того, к какому *классу* относится тот или иной IP-адрес.

На рис 5.9 показана структура IP-адреса разных классов.

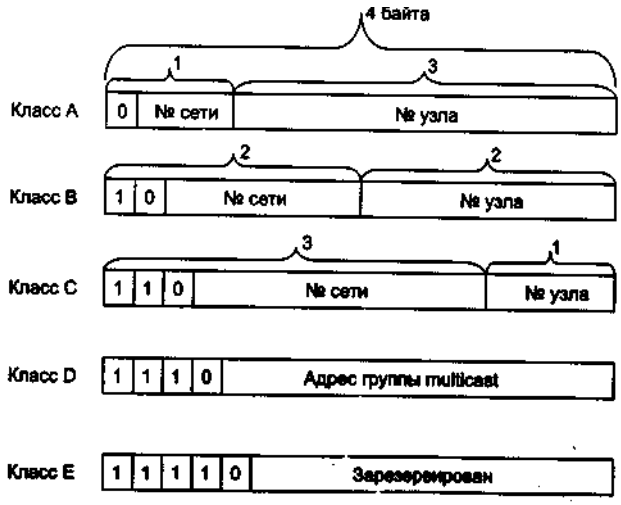


Рис. 5.9. Структура IP-адреса

Если адрес начинается с 0, то сеть относят к *классу A* и номер сети занимает один байт, остальные 3 байта интерпретируются как номер узла в сети. Сети класса А имеют номера в диапазоне от 1 до 126. (Номер 0 не используется, а номер 127 зарезервирован для специальных целей, о чем будет сказано ниже.) Сетей класса А немного, зато количество узлов в них может достигать 2^{24}, то есть 16 777 216 узлов.

Если первые два бита адреса равны 10, то сеть относится к *классу B*. В сетях класса В под номер сети и под номер узла отводится по 16 бит, то есть по 2 байта. Таким образом, сеть класса В является сетью средних размеров с максимальным числом узлов 2^{16}, что составляет 65 536 узлов.

Если адрес начинается с последовательности 110, то это сеть *класса C*. В этом случае под номер сети отводится 24 бита, а под номер узла – 8 бит. Сети этого класса наиболее распространены, число узлов в них ограничено 2^8, то есть 256 узлами.

Если адрес начинается с последовательности 1110, то он является адресом *класса D* и обозначает особый, групповой адрес – multicast. Если в пакете в качестве адреса назначения указан адрес класса D, то такой пакет должны получить все узлы, которым присвоен данный адрес.

Если адрес начинается с последовательности 11110, то это значит, что данный адрес относится к *классу E*. Адреса этого класса зарезервированы для будущих применений.

В табл. 5.4 приведены диапазоны номеров сетей и максимальное число узлов, соответствующих каждому классу сетей.

Таблица 5.4. Характеристики адресов разного класса

Класс	Первые биты	Наименьший номер сети	Наибольший номер сети	Максимальное число узлов в сети
A	0	1.0.0.0	126.0.0 0	2^{24}
B	10	128.0.0.0	191.255.0.0	2^{16}
C	110	192.0.1.0	223.255 255.0	2^8
D	1110	224.0.0.0	239.255. 255.255	Multicast
E	11110	240.0.0.0	247. 255. 255. 255	Зарезервирован

Большие сети получают адреса класса А, средние – класса В, а маленькие – класса С.

5.2.3. Особые IP-адреса

В протоколе IP существует несколько соглашений об особой интерпретации IP-адресов.

- Если весь IP-адрес состоит только из двоичных нулей, то он обозначает адрес того узла, который сгенерировал этот пакет; этот режим используется только в некоторых сообщениях ICMP.
- Если в поле номера сети стоят только нули, то по умолчанию считается, что узел назначения принадлежит той же самой сети, что и узел, который отправил пакет.
- Если все двоичные разряды IP-адреса равны 1, то пакет с таким адресом назначения должен рассылаться веем узлам, находящимся в той же сети, что и источник этого пакета. Такая рассылка называется *ограниченным широковещательным сообщением* (*limited broadcast*).
- Если в поле номера узла назначения стоят только единицы, то пакет, имеющий такой адрес, рассылается всем узлам сети с заданным номером сети. Например, пакет с адресом 192.190.21.255 доставляется всем узлам сети 192.190.21.0. Такая рассылка называется *широковещательным сообщением* (*broadcast*).

При адресации необходимо учитывать те ограничения, которые вносятся особым назначением некоторых IP-адресов. Так, ни номер сети, ни номер узла не может состоять только из одних двоичных единиц или только из одних двоичных нулей. Отсюда следует, что максимальное количество узлов, приведенное в таблице для сетей каждого класса, на практике должно быть уменьшено на 2. Например, в сетях класса С под номер узла отводится 8 бит, которые позволяют задавать 256 номеров: от 0 до 255. Однако на практике максимальное число узлов в сети класса С не может превышать 254, так как адреса 0 и 255 имеют специальное назначение. Из этих же соображений следует, что конечный узел не может иметь адрес типа 98.255.255.255, поскольку номер узла в этом адресе класса А состоит из одних двоичных единиц.

Особый смысл имеет IP-адрес, первый октет которого равен 127. Он используется для тестирования программ и взаимодействия процессов в пределах одной машины. Когда программа посылает данные по IP-адресу 127.0.0.1, то образуется как бы «петля». Данные не передаются по сети, а возвращаются модулям верхнего уровня как только что принятые. Поэтому в IP-сети запрещается присваивать машинам IP-адреса, начинающиеся со 127. Этот адрес имеет название *loopback*. Можно отнести адрес 127.0.0.0 ко внутренней сети модуля маршрутизации узла, а адрес 127.0.0.1 – к адресу этого модуля на внутренней сети. На самом деле любой адрес сети 127.0.0.0 служит для обозначения своего модуля маршрутизации, а не только 127.0.0.1, например 127.0.0.3.

В протоколе IP нет понятия широковещательности в том смысле, в котором оно используется в протоколах канального уровня локальных сетей, когда данные должны быть доставлены абсолютно всем узлам. Как ограниченный широковещательный IP-адрес, так и широковещательный IP-адрес имеют пределы распространения в интерсети – они ограничены либо сетью, к которой принадлежит узел-источник пакета, либо сетью, номер которой указан в адресе назначения. Поэтому деление сети с помощью маршрутизаторов на части локализует широковещательный шторм пределами одной из составляющих общую сеть частей просто потому, что нет способа адресовать пакет одновременно всем узлам всех сетей составной сети.

Уже упоминавшаяся форма группового IP-адреса - *multicast* – означает, что данный пакет должен быть доставлен сразу нескольким узлам, которые образуют группу с номером, указанным в поле адреса. Узлы сами идентифицируют себя, то есть определяют, к какой из групп они относятся. Один и тот же узел может входить в несколько групп. Члены какой-либо группы multicast не обязательно должны принадлежать одной сети. В общем случае они могут распределяться по совершенно различным сетям, находящимся друг от друга на произвольном количестве хопов. Групповой адрес не делится на поля номера сети и узла и обрабатывается маршрутизатором особым образом.

Основное назначение multicast-адресов – распространение информации по схеме «один-ко-многим». Хост, который хочет передавать одну и туже информацию многим абонентам, с помощью специального протокола IGMP (Internet Group Management Protocol) сообщает о создании в сети новой мультивещательной группы с определенным адресом. Машрутизаторы, поддерживающие мультивещательность, распространяют информацию о создании новой группы в сетях, подключенных к портам этого маршрутизатора. Хосты, которые хотят присоединиться к вновь создаваемой мультивещательной группе, сообщают об этом своим локальным маршрутизаторам и те передают эту

информацию хосту, инициатору создания новой группы.

Чтобы маршрутизаторы могли автоматически распространять пакеты с адресом multicast по составной сети, необходимо использовать в конечных маршрутизаторах модифицированные протоколы обмена маршрутной информацией, такие как, например, MOSPF (Multicast OSPF, аналог OSPF).

Групповая адресация предназначена для экономичного распространения в Internet или большой корпоративной сети аудио- или видеопрограмм, предназначенных сразу большой аудитории слушателей или зрителей. Если такие средства найдут широкое применение (сейчас они представляют в основном небольшие экспериментальные островки в общем Internet), то Internet сможет создать серьезную конкуренцию радио и телевидению.

5.2.4. Использование масок в IP-адресации

Традиционная схема деления IP-адреса на номер сети и номер узла основана на понятии класса, который определяется значениями нескольких первых бит адреса. Именно потому, что первый байт адреса 185.23.44.206 попадает в диапазон 128-191, мы можем сказать, что этот адрес относится к классу B, а значит, номером сети являются первые два байта, дополненные двумя нулевыми байтами – 185.23.0.0, а номером узла – 0.0.44.206.

А что если использовать какой-либо другой признак, с помощью которого можно было бы более гибко устанавливать границу между номером сети и номером узла? В качестве такого признака сейчас получили широкое распространение маски. *Маска* – это число, которое используется в паре с IP-адресом; двоичная запись маски содержит единицы в тех разрядах, которые должны в IP-адресе интерпретироваться как номер сети. Поскольку номер сети является цельной частью адреса, единицы в маске также должны представлять непрерывную последовательность.

Для стандартных классов сетей маски имеют следующие значения:

- класс A - 11111111. 00000000. 00000000. 00000000 (255.0.0.0);
- класс B - 11111111.11111111.00000000. 00000000 (255.255.0.0);
- класс C-11111111.11111111.11111111.00000000(255.255.255.0).

ПРИМЕЧАНИЕ	Для записи масок используются и другие форматы, например, удобно интерпретировать значение маски, записанной в шестнадцатеричном коде: FF.FF.OO.OO - маска для адресов класса B. Часто встречается и такое обозначение 185.23.44.206/16 - эта запись говорит о том, что маска для этого адреса содержит 16 единиц или что в указанном IP-адресе под номер сети отведено 16 двоичных разрядов.

Снабжая каждый IP-адрес маской, можно отказаться от понятий классов адресов и сделать более гибкой систему адресации. Например, если рассмотренный выше адрес 185.23.44.206 ассоциировать с маской 255.255.255.0, то номером сети будет 185.23.44.0, а не 185.23.0.0, как это определено системой классов.

В масках количество единиц в последовательности, определяющей границу номера сети, не обязательно должно быть кратным 8, чтобы повторять деление адреса на байты. Пусть, например, для IP-адреса 129.64.134.5 указана маска 255.255.128.0, то есть в двоичном виде:

IP-адрес 129.64.134.5 - 10000001. 01000000.10000110. 00000101

Маска 255.255.128.0- 11111111. 11111111. 10000000.00000000

Если игнорировать маску, то в соответствии с системой классов адрес 129.64.134.5 относится к классу B, а значит, номером сети являются первые 2 байта – 129.64.0.0 а номером узла-0.0.134.5.

Если же использовать для определения границы номера сети маску, то 17 последовательных единиц в маске, «наложенные» на IP-адрес, определяют в качестве номера сети в двоичном выражении число:

10000001. 01000000. 10000000. 00000000

или в десятичной форме записи - номер сети 129.64.128.0, а номер узла 0.0.6.5.

Механизм масок широко распространен в IP-маршрутизации, причем маски могут использоваться для самых разных целей. С их помощью администратор может структурировать свою сеть, не требуя от поставщика услуг дополнительных номеров сетей. На основе этого же механизма поставщики услуг могут объединять адресные пространства нескольких сетей путем введения так называемых «префиксов» с целью уменьшения объема таблиц маршрутизации и повышения за счет этого

производительности маршрутизаторов.

5.2.5. Порядок распределения IP-адресов

Номера сетей назначаются либо централизованно, если сеть является частью Internet, либо произвольно, если сеть работает автономно. Номера узлов и в том и в другом случае администратор волен назначать по своему усмотрению, не выходя, разумеется, из разрешенного для этого класса сети диапазона.

Координирующую роль в централизованном распределении IP-адресов до некоторого времени играла организация InterNIC, однако с ростом сети задача распределения адресов стала слишком сложной, и InterNIC делегировала часть своих функций другим организациям и крупным поставщикам услуг Internet,



Рис. 5.10. Нерациональное использование пространства IP-адресов

Уже сравнительно давно наблюдается дефицит IP-адресов. Очень трудно получить адрес класса B и практически невозможно стать обладателем адреса класса A. При этом надо отметить, что дефицит обусловлен не только ростом сетей, но и тем, что имеющееся множество IP-адресов используется нерационально. Очень часто владельцы сети класса C расходуют лишь небольшую часть из имеющихся у них 254 адресов. Рассмотрим пример, когда две сети необходимо соединить глобальной связью. В таких случаях в качестве канала связи используют два маршрутизатора, соединенных по схеме «точка-точка» (рис. 5.10). Для вырожденной сети, образованной каналом, связывающим порты двух смежных маршрутизаторов, приходится выделять отдельный номер сети, хотя в этой сети имеются всего 2 узла.

Если же некоторая IP-сеть создана для работы в «автономном режиме», без связи с Internet, тогда администратор этой сети волен назначить ей произвольно выбранный номер. Но и в этой ситуации для того, чтобы избежать каких-либо коллизий, в стандартах Internet определено несколько диапазонов адресов, рекомендуемых для локального использования. Эти адреса не обрабатываются маршрутизаторами Internet ни при каких условиях. Адреса, зарезервированные для локальных целей, выбраны из разных классов: в классе A – это сеть 10.0.0.0, в классе B – это диапазон из 16 номеров сетей 172.16.0.0-172.31.0.0, в классе C – это диапазон из 255 сетей - 192.168.0.0-192.168.255.0.

Для смягчения проблемы дефицита адресов разработчики стека TCP/IP предлагают разные подходы. Принципиальным решением является переход на новую версию IPv6, в которой резко расширяется адресное пространство за счет использования 16-байтных адресов. Однако и текущая версия IPv4 поддерживает некоторые технологии, направленные на более экономное расходование IP-адресов. Одной из таких технологий является технология *масок* и ее развитие – технология *бесклассовой междоменной маршрутизации* (*Classless Inter-Domain Routing, CIDR*). Технология CIDR отказывается от традиционной концепции разделения адресов протокола IP на классы, что позволяет получать в пользование столько адресов, сколько реально необходимо. Благодаря CIDR поставщик услуг получает возможность «нарезать» блоки из выделенного ему адресного пространства в точном соответствии с требованиями каждого клиента, при этом у него остается пространство для маневра на случай его будущего роста.

Другая технология, которая может быть использована для снятия дефицита адресов, это *трансляция адресов* (*Network Address Translator, NAT*). Узлам внутренней сети адреса назначаются произвольно (естественно, в соответствии с общими правилами, определенными в стандарте), так, как будто эта сеть работает автономно. Внутренняя сеть соединяется с Internet через некоторое промежуточное устройство (маршрутизатор, межсетевой экран). Это промежуточное устройство получает в свое распоряжение некоторое количество внешних «нормальных» IP-адресов, согласованных с поставщиком услуг или другой организацией, распределяющей IP-адреса. Промежуточное устройство способно

преобразовывать внутренние адреса во внешние, используя для этого некие таблицы соответствия. Для внешних пользователей все многочисленные узлы внутренней сети выступают под несколькими внешними IP-адресами. При получении внешнего запроса это устройство анализирует его содержимое и при необходимости пересылает его во внутреннюю сеть, заменяя IP-адрес на внутренний адрес этого узла. Процедура трансляции адресов определена в RFC 1631.

5.2.6. Автоматизация процесса назначения IP-адресов

Назначение IP-адресов узлам сети даже при не очень большом размере сети может представлять для администратора утомительную процедуру. Протокол *Dynamic Host Configuration Protocol* (*DHCP*) освобождает администратора от этих проблем, автоматизируя процесс назначения IP-адресов.

DHCP может поддерживать способ автоматического динамического распределения адресов, а также более простые способы ручного и автоматического статического назначения адресов. Протокол DHCP работает в соответствии с моделью клиент-сервер. Во время старта системы компьютер, являющийся DHCP-клиентом, посылает в сеть широковещательный запрос на получение IP-адреса. DHCP-сервер откликается и посылает сообщение-ответ, содержащее IP-адрес. Предполагается, что DHCP-клиент и DHCP-сервер находятся в одной IP-сети.

При динамическом распределении адресов DHCP-сервер выдает адрес клиенту на ограниченное время, называемое *временем аренды* (*lease duration*), что дает возможность впоследствии повторно использовать этот IP-адрес для назначения другому компьютеру. Основное преимущество DHCP - автоматизация рутинной работы администратора по конфигурированию стека TCP/IP на каждом компьютере. Иногда динамическое разделение адресов позволяет строить IP-сеть, количество узлов в которой превышает количество имеющихся в распоряжении администратора IP-адресов.

В ручной процедуре назначения статических адресов активное участие принимает администратор, который предоставляет DHCP-серверу информацию о соответствии IP-адресов физическим адресам или другим идентификаторам клиентов. DHCP-сервер, пользуясь этой информацией, всегда выдает определенному клиенту назначенный администратором адрес.

При автоматическом статическом способе DHCP-сервер присваивает IP-адрес из пула наличных IP-адресов без вмешательства оператора. Границы пула назначаемых адресов задает администратор при конфигурировании DHCP-сервера. Адрес дается клиенту из пула в постоянное пользование, то есть с неограниченным сроком аренды. Между идентификатором клиента и его IP-адресом по-прежнему, как и при ручном назначении, существует постоянное соответствие. Оно устанавливается в момент первого назначения DHCP-сервером IP-адреса клиенту. При всех последующих запросах сервер возвращает тот же самый IP-адрес.

DHCP обеспечивает надежный и простой способ конфигурации сети TCP/IP, гарантируя отсутствие дублирования адресов за счет централизованного управления их распределением. Администратор управляет процессом назначения адресов с помощью параметра «продолжительность аренды», которая определяет, как долго компьютер может использовать назначенный IP-адрес, перед тем как снова запросить его от DHCP-сервера в аренду.

Примером работы протокола DHCP может служить ситуация, когда компьютер, являющийся DHCP-клиентом, удаляется из подсети. При этом назначенный ему IP-адрес автоматически освобождается. Когда компьютер подключается к другой подсети, то ему автоматически назначается новый адрес. Ни пользователь, ни сетевой администратор не вмешиваются в этот процесс. Это свойство очень важно для мобильных пользователей.

DHCP-сервер может назначить клиенту не только IP-адрес клиента, но и другие параметры стека TCP/IP, необходимые для его эффективной работы, например, маску, IP-адрес маршрутизатора по умолчанию, IP-адрес сервера DNS, доменное имя компьютера и т. п.

5.2.7. Отображение IP-адресов на локальные адреса

Одной из главных задач, которая ставилась при создании протокола IP, являлось обеспечение совместной согласованной работы в сети, состоящей из подсетей, в общем случае использующих разные сетевые технологии. Непосредственно с решением этой задачи связан уровень межсетевых интерфейсов стека TCP/IP. На этом уровне определяются уже рассмотренные выше спецификации упаковки (инкапсуляции) IP-пакетов в кадры локальных технологий. Кроме этого, уровень межсетевых

интерфейсов должен заниматься также крайне важной задачей отображения IP-адресов в локальные адреса.

Для определения локального адреса по IP-адресу используется *протокол разрешения адреса* (*Address Resolution Protocol, ARP*). Протокол ARP работает различным образом в зависимости от того, какой протокол канального уровня работает в данной сети – протокол локальной сети (Ethernet, Token Ring, FDDI) с возможностью широковещательного доступа одновременно ко всем узлам сети или же протокол глобальной сети (X.25, frame relay), как правило не поддерживающий широковещательный доступ. Существует также протокол, решающий обратную задачу – нахождение IP-адреса по известному локальному адресу. Он называется реверсивным ARP (Reverse Address Resolution Protocol, RARP) и используется при старте бездисковых станций, не знающих в начальный момент своего IP-адреса, но знающих адрес своего сетевого адаптера.

Необходимость в обращении к протоколу ARP возникает каждый раз, когда модуль IP передает пакет на уровень сетевых интерфейсов, например драйверу Ethernet. IP-адрес узла назначения известен модулю IP. Требуется на его основе найти MAC-адрес узла назначения.

Работа протокола ARP начинается с просмотра так называемой *ARP-таблицы* (табл. 5.5). Каждая строка таблицы устанавливает соответствие между IP-адресом и MAC-адресом. Для каждой сети, подключенной к сетевому адаптеру компьютера или к порту маршрутизатора, строится отдельная ARP-таблица.

Таблица 5.5. Пример ARP-таблицы

IP-адрес	MAC-адрес	Тип записи
194.85.135.75	008048EB7E60	Динамический
194.85.135.70	08005A21A722	Динамический
194.85.60.21	008048EB7567	Статический

Поле «Тип записи» может содержать одно из двух значений – «динамический» или «статический». Статические записи создаются вручную с помощью утилиты arp и не имеют срока устаревания, точнее, они существуют до тех пор, пока компьютер или маршрутизатор не будут выключены. Динамические же записи создаются модулем протокола ARP, использующим широковещательные возможности локальных сетевых технологий. Динамические записи должны периодически обновляться. Если запись не обновлялась в течение определенного времени (порядка нескольких минут), то она исключается из таблицы. Таким образом, в ARP-таблице содержатся записи не обо всех узлах сети, а только о тех, которые активно участвуют в сетевых операциях. Поскольку такой способ хранения информации называют кэшированием, ARP-таблицы иногда называют ARP-кэш.

В глобальных сетях администратору сети чаще всего приходится вручную формировать ARP-таблицы, в которых он задает, например, соответствие IP-адреса адресу узла сети X.25, который имеет для протокола IP смысл локального адреса. В последнее время наметилась тенденция автоматизации работы протокола ARP и в глобальных сетях. Для этой цели среди всех маршрутизаторов, подключенных к какой-либо глобальной сети, выделяется специальный маршрутизатор, который ведет ARP-таблицу для всех остальных узлов и маршрутизаторов этой сети. При таком централизованном подходе для всех узлов и маршрутизаторов вручную нужно задать только IP-адрес и локальный адрес выделенного маршрутизатора. Затем каждый узел и маршрутизатор регистрирует свои адреса в выделенном маршрутизаторе, а при необходимости установления соответствия между IP-адресом и локальным адресом узел обращается к выделенному маршрутизатору с запросом и автоматически получает ответ без участия администратора. Работающий таким образом маршрутизатор называют ARP-сервором.

Итак, после того как модуль IP обратился к модулю ARP с запросом на разрешение адреса, происходит поиск в ARP-таблице указанного в запросе IP-адреса. Если таковой адрес в ARP-таблице отсутствует, то исходящий IP-пакет, для которого нужно было определить локальный адрес, ставится в очередь. Далее протокол ARP формирует свой запрос (ARP-запрос), вкладывает его в кадр протокола канального уровня и рассылает запрос широковещательно.

Все узлы локальной сети получают ARP-запрос и сравнивают указанный там IP-адрес с собственным. В случае их совпадения узел формирует ARP-ответ, в котором указывает свой IP-адрес и свой локальный адрес, а затем отправляет его уже направленно, так как в ARP-запросе отправитель

указывает свой локальный адрес. ARP-запросы и ответы используют один и тот же формат пакета. В табл. 5.6 приведены значения полей примера ARP запроса для передачи по сети Ethernet.

Таблица 5.6. Пример ARP-запроса

Тип сети	1 (0x1)
Тип протокола	2048 (0x800)
Длина локального адреса	6 (0x6)
Длина сетевого адреса	4 (0x4)
Опция	1 (0x1)
Локальный адрес отправителя	008048EB7E60
Сетевой адрес отправителя	194.85.135.75
Локальный (искомый) адрес получателя	000000000000
Сетевой адрес получателя	194.85.135.65

В поле «тип сети» для сетей Ethernet указывается значение 1.

Поле «тип протокола» позволяет использовать протокол ARP не только для протокола IP, но и для других сетевых протоколов. Для IP значение этого поля равно 0800_{16}.

Длина локального адреса для протокола Ethernet равна 6 байт, а длина IP-адреса – 4 байт. В поле операции для ARP-запросов указывается значение 1, если это запрос, и 2, если это ответ.

Из этого запроса видно, что в сети Ethernet узел с IP-адресом 194.85.135.75 пытается определить, какой MAC-адрес имеет другой узел той же сети, сетевой адрес которого 194.85.135.65. Поле искомого локального адреса заполнено нулями.

Ответ присылает узел, опознавший свой IP-адрес. Если в сети нет машины с искомым IP-адресом, то ARP-ответа не будет. Протокол IP уничтожает IP-пакеты, направляемые по этому адресу. (Заметим, что протоколы верхнего уровня не могут отличить случай повреждения сети Ethernet от случая отсутствия машины с искомым IP-адресом.) В табл. 5.7 помещены значения полей ARP-ответа, который мог бы поступить на приведенный выше пример ARP-запроса.

Таблица 5.7. Пример ARP-ответа

Тип сети	1 (0x1)
Тип протокола	2048 (0x800)
Длина локального адреса	6 (0x6)
Длина сетевого адреса	4 (0x4)
Опция	2 (0x2)
Локальный адрес отправителя	OOEOF77F1920
Сетевой адрес отправителя	194.85.135.65
Локальный (искомый) адрес получателя	008048EB7E60
Сетевой адрес получателя	194.85.135.75

Этот ответ получает машина, сделавшая ARP-запрос. Модуль ARP анализирует ARP-ответ и добавляет запись в свою ARP-таблицу (табл. 5.8). В результате обмена этими двумя ARP-сообщенпями модуль IP-узла 194.85.135.75 определил, что IP-адресу 194.85.135.65 соответствует MAC-адрес OOEOF77F1920. Новая запись в ARP-таблице появляется автоматически, спустя несколько миллисекунд после того, как она потребовалась.

Таблица 5.8. Обновленная ARP-таблица

IP-адрес	MAC-адрес	Тип записи
194.85.135.75	008048EB7E60	Динамический
194.85.135.70	08005A21A722	Динамический
194.85.60.21	008048EB7567	Статический
194.85.135.65	OOEOF77F1920	Динамический

ПРИМЕЧАНИЕ Некоторые реализации IP и ARP не ставят IP-пакеты в очередь на время ожидания ARP-ответов. Вместо этого IP-пакет просто уничтожается, а его восстановление возлагается на модуль TCP или прикладной процесс, работающий через UDP. Такое восстановление выполняется с помощью тайм-аутов и повторных передач. Повторная передача сообщения проходит успешно, так как первая попытка уже вызвала заполнение ARP-таблицы.

5.2.8. Отображение доменных имен на IP-адреса

Организация доменов и доменных имен

Для идентификации компьютеров аппаратное и программное обеспечение в сетях TCP/IP полагается на IP адреса, поэтому для доступа к сетевому ресурсу в параметрах программы вполне достаточно указать IP-адрес, чтобы программа правильно поняла, к какому хосту ей нужно обратиться. Например, команда ftp://192.45.66.17 будет устанавливать сеанс связи с нужным ftp-сервером, а команда http://203.23.106.33 откроет начальную страницу на корпоративном Web-сервере. Однако пользователи обычно предпочитают работать с символьными именами компьютеров, и операционные системы локальных сетей приучили их к этому удобному способу. Следовательно, в сетях TCP/IP должны существовать символьные имена хостов и механизм для установления соответствия между символьными именами и IP-адресами.

В операционных системах, которые первоначально разрабатывались для работы в локальных сетях, таких как Novell NetWare, Microsoft Windows или IBM OS/2, пользователи всегда работали с символьными именами компьютеров. Так как локальные сети состояли из небольшого числа компьютеров, то использовались так называемые плоские имена, состоящие из последовательности символов, не разделенных на части. Примерами таких имен являются: NW1_1, mail2, MOSCOW_SALES_2. Для установления соответствия между символьными именами и MAC-адресами в этих операционных системах применялся механизм широковещательных запросов, подобный механизму запросов протокола ARP. Так, широковещательный способ разрешения имен реализован в протоколе NetBIOS, на котором были построены многие локальные ОС. Так называемые NetBIOS-имена стали на долгие годы одним из основных типов плоских имен в локальных сетях.

Для стека TCP/IP, рассчитанного в общем случае на работу в больших территориально распределенных сетях, подобный подход оказывается неэффективным по нескольким причинам.

Плоские имена не дают возможности разработать единый алгоритм обеспечения уникальности имен в пределах большой сети. В небольших сетях уникальность имен компьютеров обеспечивает администратор сети, записывая несколько десятков имен в журнале или файле. При росте сети задачу решают уже несколько администраторов, согласовывая имена между собой неформальным способом. Однако если сеть расположена в разных городах или странах, то администраторам каждой части сети нужно придумать способ именования, который позволил бы им давать имена новым компьютерам независимо от других администраторов, обеспечивая в то же время уникальность имен для всей сети. Самый надежный способ решения этой задачи – отказ от плоских имен в принципе.

Широковещательный способ установления соответствия между символьными именами и локальными адресами хорошо работает только в небольшой локальной сети, не разделенной на подсети. В крупных сетях, где общая широковещательность не поддерживается, нужен другой способ разрешения символьных имен. Обычно хорошей альтернативой широковещательности является применение централизованной службы, поддерживающей соответствие между различными типами адресов всех компьютеров сети. Компания Microsoft для своей корпоративной операционной системы Windows NT разработала централизованную службу WINS, которая поддерживает базу данных NetBIOS-имен и соответствующих им IP-адресов.

Для эффективной организации именования компьютеров в больших сетях естественным является применение иерархических составных имен.

В стеке TCP/IP применяется доменная система имен, которая имеет иерархическую древовидную структуру, допускающую использование в имени произвольного количества составных частей (рис. 5.11).

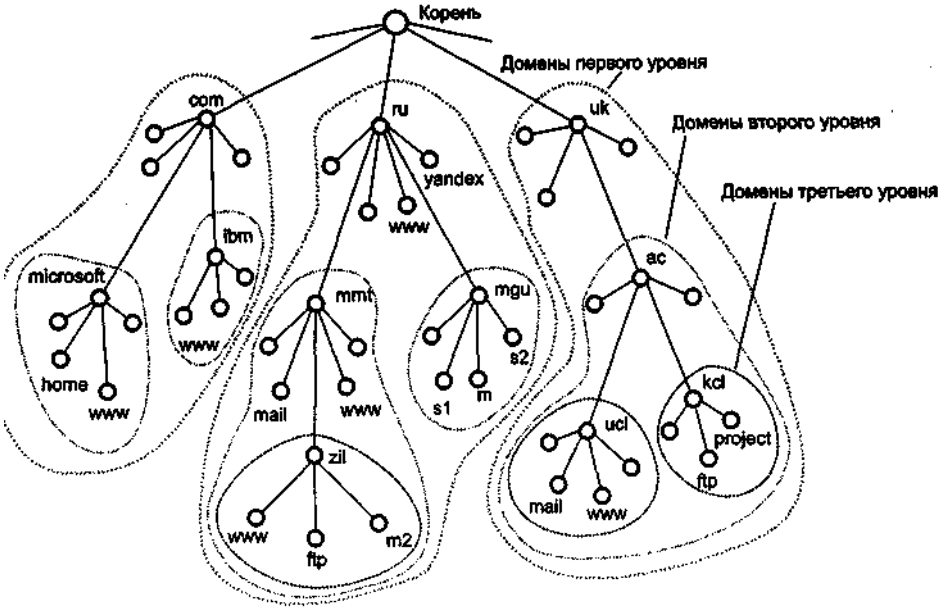


Рис. 5.11. Пространство доменных имен

Иерархия доменных имен аналогична иерархии имен файлов, принятой во многих популярных файловых системах. Дерево имен начинается с корня, обозначаемого здесь точкой (.). Затем следует старшая символьная часть имени, вторая по старшинству символьная часть имени и т. д. Младшая часть имени соответствует конечному узлу сети. В отличие от имен файлов, при записи которых сначала указывается самая старшая составляющая, затем составляющая более низкого уровня и т. д., запись доменного имени начинается с самой младшей составляющей, а заканчивается самой старшей. Составные части доменного имени отделяется друг от друга точкой. Например, в имени partnering.microsoft.com составляющая partnering является именем одного из компьютеров в домене microsoft.com.

Разделение имени на части позволяет *разделить административную ответственность* за назначение уникальных имен между различными людьми или организациями в пределах своего уровня иерархии. Так, для примера, приведенного на рис. 5.11, один человек может нести ответственность за то, чтобы все имена, которые имеют окончание «ru», имели уникальную следующую вниз по иерархии часть. Если этот человек справляется со своими обязанностями, то все имена типа www.ru, mail.mmt.ru или m2.zil.mmt.ru будут отличаться второй по старшинству частью.

Разделение административной ответственности позволяет решить проблему образования уникальных имен без взаимных консультаций между организациями, отвечающими за имена одного уровня иерархии. Очевидно, что должна существовать одна организация, отвечающая за назначение имен верхнего уровня иерархии.

Совокупность имен, у которых несколько старших составных частей совпадают, образуют *домен* имен (*domain*). Например, имена wwwl.zil.mmt.ru, ftp.zil.mmt.ru, yandex.ru и sl.mgu.ru входят в домен ru, так как все эти имена имеют одну общую старшую часть – имя ru. Другим примером является домен mgu.ru. Из представленных на рис. 5.11 имен в него входят имена sl.mgu.ru, s2.mgu.ru и m.mgu.ru. Этот домен образуют имена, у которых две старшие части всегда равны mgu.ru. Имя www.mmt.ru в домен mgu.ru не входит, так как имеет отличающуюся составляющую mmt

ВНИМАНИЕ Термин «домен» очень многозначен, поэтому его нужно трактовать в рамках определенного контекста. Кроме доменов имен стека TCP/IP в компьютерной литературе также часто упоминаются домены Windows NT, домены коллизий и некоторые другие. Общим у всех этих терминов является то, что они описывают некоторое множество компьютеров, обладающее каким-либо определенным свойством.

Если один домен входит в другой домен как его составная часть, то такой домен могут называть *поддоменом* (*subdomain*), хотя название домен за ним также остается. Обычно поддомен называют по

имени той его старшей составляющей, которая отличает его от других поддоменов. Например, поддомен mmt.ru обычно называют поддоменом (или доменом) mmt. Имя поддомену назначает администратор вышестоящего домена. Хорошей аналогией домена является каталог файловой системы.

Если в каждом домене и поддомене обеспечивается уникальность имен следующего уровня иерархии, то и вся система имен будет состоять из уникальных имен.

По аналогии с файловой системой, в доменной системе имен различают краткие имена, относительные имена и полные доменные имена. Краткое имя – это имя конечного узла сети: хоста или порта маршрутизатора. Краткое имя – это лист дерева имен. Относительное имя – это составное имя, начинающееся с некоторого уровня иерархии, но не самого верхнего. Например, wwwl.zil – это относительное имя. *Полное доменное имя* (*fully qualified domain name, FQDN*) включает составляющие всех уровней иерархии, начиная от краткого имени и кончая корневой точкой: wwwl.zil.mmt.ru.

Необходимо подчеркнуть, что компьютеры входят в домен в соответствии со своими составными именами, при этом они могут иметь совершенно различные IP-адреса, принадлежащие к различным сетям и подсетям. Например, в домен mgu.ru могут входить хосты с адресами 132.13.34.15, 201.22.100.33, 14.0.0.6. Доменная система имен реализована в сети Internet, но она может работать и как автономная система имен в крупной корпоративной сети, использующей стек TCP/IP, но не связанной с Internet.

В Internet корневой домен управляется центром InterNIC. Домены верхнего уровня назначаются для каждой страны, а также на организационной основе. Имена этих доменов должны следовать международному стандарту ISO 3166. Для обозначения стран используются трехбуквенные и двухбуквенные аббревиатуры, а для различных типов организаций – следующие обозначения:

- com – коммерческие организации (например, microsoft.com);
- edu – образовательные (например, mit.edu);
- gov – правительственные организации (например, nsf.gov);
- org – некоммерческие организации (например, fidonet.org);
- net – организации, поддерживающие сети (например, nsf.net).

Каждый домен администрируется отдельной организацией, которая обычно разбивает свой домен на поддомены и передает функции администрирования этих поддоменов другим организациям. Чтобы получить доменное имя, необходимо зарегистрироваться в какой-либо организации, которой InterNIC делегировал свои полномочия по распределению имен доменов. В России такой организацией является РосНИИРОС, которая отвечает за делегирование имен поддоменов в домене ru.

Система доменных имен DNS

Соответствие между доменными именами и IP-адресами может устанавливаться как средствами локального хоста, так и средствами централизованной службы. На раннем этапе развития Internet на каждом хосте вручную создавался текстовый файл с известным именем hosts. Этот файл состоял из некоторого количества строк, каждая из которых содержала одну пару «IP-адрес – доменное имя», например 102.54.94.97 - rhino.acme.com.

По мере роста Internet файлы hosts также росли, и создание масштабируемого решения для разрешения имен стало необходимостью.

Таким решением стала специальная служба – *система доменных имен* (*Domain Name System, DNS*). DNS – это централизованная служба, основанная на распределенной базе отображений «доменное имя – IP-адрес». Служба DNS использует в своей работе протокол типа «клиент-сервер». В нем определены DNS-серверы и DNS-клиенты. DNS-серверы поддерживают распределенную базу отображений, а DNS-клиенты обращаются к серверам с запросами о разрешении доменного имени в IP-адрес.

Служба DNS использует текстовые файлы почти такого формата, как и файл hosts, и эти файлы администратор также подготавливает вручную. Однако служба DNS опирается на иерархию доменов, и каждый сервер службы DNS хранит только часть имен сети, а не все имена, как это происходит при использовании файлов hosts При росте количества узлов в сети проблема масштабирования решается созданием новых доменов и поддоменов имен и добавлением в службу DNS новых серверов.

Для каждого домена имен создается свой DNS-сервер. Этот сервер может хранить отображения «доменное имя – IP-адрес» для всего домена включая все его поддомены. Однако при этом решение оказывается плохо масштабируемым, так как при добавлении новых поддоменов нагрузка на этот сервер может превысить его возможности. Чаще сервер домена хранит только имена, которые

заканчиваются на следующем ниже уровне иерархии по сравнению с именем домена. (Аналогично каталогу файловой системы, который содержит записи о файлах и подкаталогах, непосредственно в него «входящих».) Именно при такой организации службы DNS нагрузка по разрешению имен распределяется более-менее равномерно между всеми DNS-серверами сети. Например, в первом случае DNS-сервер домена mmt.ru будет хранить отображения для всех имен, заканчивающихся на mmt.ru: wwwl.zil.mmt.ru, ftp.zil.mmt.ru, mail.mmt.ru и т. д. Во втором случае этот сервер хранит отображения только имен типа mail.mmt.ru, www.mmt.ru, а все остальные отображения должны храниться на DNS-сервере поддомена zil.

Каждый DNS-сервер кроме таблицы отображений имен содержит ссылки на DNS-серверы своих поддоменов. Эти ссылки связывают отдельные DNS-серверы в единую службу DNS. Ссылки представляют собой IP-адреса соответствующих серверов. Для обслуживания корневого домена выделено несколько дублирующих друг друга DNS-серверов, IP-адреса которых являются широко известными (их можно узнать, например, в InterNIC).

Процедура разрешения DNS-имени во многом аналогична процедуре поиска файловой системой адреса файла по его символьному имени. Действительно, в обоих случаях составное имя отражает иерархическую структуру организации соответствующих справочников – каталогов файлов или таблиц DNS. Здесь домен и доменный DNS-сервер являются аналогом каталога файловой системы. Для доменных имен, так же как и для символьных имен файлов, характерна независимость именования от физического местоположения.

Процедура поиска адреса файла по символьному имени заключается в последовательном просмотре каталогов, начиная с корневого. При этом предварительно проверяется кэш и текущий каталог. Для определения IP-адреса по доменному имени также необходимо просмотреть все DNS-серверы, обслуживающие цепочку поддоменов, входящих в имя хоста, начиная с корневого домена. Существенным же отличием является то, что файловая система расположена на одном компьютере, а служба DNS по своей природе является распределенной.

Существуют две основные схемы разрешения DNS-имен. В первом варианте работу по поиску IP-адреса координирует DNS-клиент:

- DNS-клиент обращается к корневому DNS-серверу с указанием полного доменного имени;
- DNS-сервер отвечает, указывая адрес следующего DNS-сервера, обслуживающего домен верхнего уровня, заданный в старшей части запрошенного имени;
- DNS-клиент делает запрос следующего DNS-сервера, который отсылает его к DNS-серверу нужного поддомена, и т. д., пока не будет найден DNS-сервер, в котором хранится соответствие запрошенного имени IP-адресу. Этот сервер дает окончательный ответ клиенту.

Такая схема взаимодействия называется нерекурсивной или итеративной, когда клиент сам итеративно выполняет последовательность запросов к разным серверам имен. Так как эта схема загружает клиента достаточно сложной работой, то она применяется редко.

Во втором варианте реализуется рекурсивная процедура:

- DNS-клиент запрашивает локальный DNS-сервер, то есть тот сервер, который обслуживает поддомен, к которому принадлежит имя клиента;
- если локальный DNS-сервер знает ответ, то он сразу же возвращает его клиенту; это может соответствовать случаю, когда запрошенное имя входит в тот же поддомен, что и имя клиента, а также может соответствовать случаю, когда сервер уже узнавал данное соответствие для другого клиента и сохранил его в своем кэше;
- если же локальный сервер не знает ответ, то он выполняет итеративные запросы к корневому серверу и т. д. точно так же, как это делал клиент в первом варианте; получив ответ, он передает его клиенту, который все это время просто ждал его от своего локального DNS-сервера.

В этой схеме клиент перепоручает работу своему серверу, поэтому схема называется косвенной или рекурсивной. Практически все DNS-клиенты используют рекурсивную процедуру.

Для ускорения поиска IP-адресов DNS-серверы широко применяют процедуру кэширования проходящих через них ответов. Чтобы служба DNS могла оперативно отрабатывать изменения, происходящие в сети, ответы кэшируются на определенное время – обычно от нескольких часов до нескольких дней.

Выводы

- В стеке TCP/IP используются три типа адресов: локальные (называемые также аппаратными), IP-адреса и символьные доменные имена. Все эти типы адресов присваиваются узлам составной сети независимо друг от друга.
- IP-адрес имеет длину 4 байта и состоит из номера сети и номера узла. Для определения границы, отделяющей номер сети от номера узла, реализуются два подхода. Первый основан на понятии класса адреса, второй – на использовании масок.
- Класс адреса определяется значениями нескольких первых бит адреса. В адресах класса A под номер сети отводится один байт, а остальные три байта – под номер узла, поэтому они используются в самых больших сетях. Для небольших сетей больше подходят адреса класса C, в которых номер сети занимает три байта, а для нумерации узлов может быть использован только один байт. Промежуточное положение занимают адреса класса B.
- Другой способ определения, какая часть адреса является номером сети, а какая номером узла, основан на использовании маски. Маска – это число, которое используется в паре с IP-адресом; двоичная запись маски содержит единицы в тех разрядах, которые в IP-адресе должны интерпретироваться как номер сети.
- Номера сетей назначаются либо централизованно, если сеть является частью Internet, либо произвольно, если сеть работает автономно.
- Процесс распределения IP-адресов по узлам сети может быть автоматизирован с помощью протокола DHCP.
- Установление соответствия между IP-адресом и аппаратным адресом (чаще всего MAC-адресом) осуществляется протоколом разрешения адресов ARP, который для этой цели просматривает ARP-таблицы. Если нужный адрес отсутствует, то выполняется широковещательный ARP-запрос.
- В стеке TCP/IP применяется доменная система символьных имен, которая имеет иерархическую древовидную структуру, допускающую использование в имени произвольного количества составных частей. Совокупность имен, у которых несколько старших составных частей совпадают, образуют домен имен. Доменные имена назначаются централизованно, если сеть является частью Internet, в противном случае – локально.
- Соответствие между доменными именами и IP-адресами может устанавливаться как средствами локального хоста с использованием файла hosts, так и с помощью централизованной службы DNS, основанной на распределенной базе отображений «доменное имя – IP-адрес».

5.3. Протокол IP

5.3.1. Основные функции протокола IP

Основу транспортных средств стека протоколов TCP/IP составляет *протокол межсетевого взаимодействия* (*Internet Protocol, IP*). Он обеспечивает передачу дейтаграмм от отправителя к получателям через объединенную систему компьютерных сетей.

Название данного протокола – Intrenet Protocol – отражает его суть: он должен передавать пакеты *между сетями*. В каждой очередной сети, лежащей на пути перемещения пакета, протокол IP вызывает средства транспортировки, принятые в этой сети, чтобы с их помощью передать этот пакет на маршрутизатор, ведущий к следующей сети, или непосредственно на узел-получатель.

Протокол IP относится к протоколам без установления соединений. Перед IP не ставится задача надежной доставки сообщений от отправителя к получателю. Протокол IP обрабатывает каждый IP-пакет как независимую единицу, не имеющую связи ни с какими другими IP-пакетами. В протоколе IP нет механизмов, обычно применяемых для увеличения достоверности конечных данных: отсутствует квитирование – обмен подтверждениями между отправителем и получателем, нет процедуры упорядочивания, повторных передач или других подобных функций. Если во время продвижения пакета произошла какая-либо ошибка, то протокол IP по своей инициативе ничего не предпринимает

для исправления этой ошибки. Например, если на промежуточном маршрутизаторе пакет был отброшен по причине истечения времени жизни или из-за ошибки в контрольной сумме, то модуль IP не пытается заново послать испорченный или потерянный пакет. Все вопросы обеспечения надежности доставки данных по составной сети в стеке TCP/IP решает протокол TCP, работающий непосредственно над протоколом IP. Именно TCP организует повторную передачу пакетов, когда в этом возникает необходимость.

Важной особенностью протокола IP, отличающей его от других сетевых протоколов (например, от сетевого протокола IPX), является его способность выполнять динамическую фрагментацию пакетов при передаче их между сетями с различными, максимально допустимыми значениями поля данных кадров MTU. Свойство фрагментации во многом способствовало тому, что протокол IP смог занять доминирующие позиции в сложных составных сетях.

Имеется прямая связь между функциональной сложностью протокола и сложностью заголовка пакетов, которые этот протокол использует. Это объясняется тем, что основные служебные данные, на основании которых протокол выполняет то или иное действие, переносятся между двумя модулями, реализующими этот протокол на разных машинах, именно в полях заголовков пакетов. Поэтому очень полезно изучить назначение каждого поля заголовка IP-пакета, и это изучение дает не только формальные знания о структуре пакета, но и объясняет все основные режимы работы протокола по обработке и передаче IP-дейтаграмм.

5.3.2. Структура IP-пакета

IP-пакет состоит из заголовка и поля данных. Заголовок, как правило, имеющий длину 20 байт, имеет следующую структуру (рис. 5.12).

Поле *Номер версии* (*Version*), занимающее 4 бит, указывает версию протокола IP. Сейчас повсеместно используется версия 4 (IPv4), и готовится переход на версию 6 (IPv6).

Поле *Длина заголовка* (*IHL*) IP-пакета занимает 4 бит и указывает значение длины заголовка, измеренное в 32-битовых словах. Обычно заголовок имеет длину в 20 байт (пять 32-битовых слов), но при увеличении объема служебной информации эта длина может быть увеличена за счет использования дополнительных байт в поле *Опции* (*IP Options*). Наибольший заголовок занимает 60 октетов.

Поле *Тип сервиса* (*Type of Service*) занимает один байт и задает приоритетность пакета и вид критерия выбора маршрута. Первые три бита этого поля образуют подполе *приоритета* пакета (*Precedence*). Приоритет может иметь значения от самого низкого – 0 (нормальный пакет) до самого высокого – 7 (пакет управляющей информации). Маршрутизаторы и компьютеры могут принимать во внимание приоритет пакета и обрабатывать более важные пакеты в первую очередь. Поле *Тип сервиса* содержит также три бита, определяющие критерий выбора маршрута. Реально выбор осуществляется между тремя альтернативами: малой задержкой, высокой достоверностью и высокой пропускной способностью. Установленный бит D (delay) говорит о том, что маршрут должен выбираться для минимизации задержки доставки данного пакета, бит T – для максимизации пропускной способности, а бит R – для максимизации надежности доставки. Во многих сетях улучшение одного из этих параметров связано с ухудшением другого, кроме того, обработка каждого из них требует дополнительных вычислительных затрат. Поэтому редко, когда имеет смысл устанавливать одновременно хотя бы два из этих трех критериев выбора маршрута. Зарезервированные биты имеют нулевое значение.

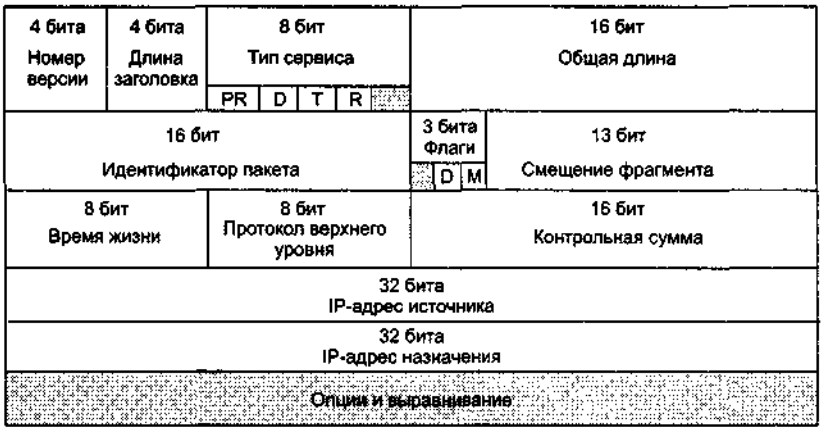


Рис. 5.12. Структура заголовка IP-пакета

Поле *Общая длина* (*Total Length*) занимает 2 байта и означает общую длину пакета с учетом заголовка и поля данных. Максимальная длина пакета ограничена разрядностью поля, определяющего эту величину, и составляет 65 535 байт, однако в большинстве хост-компьютеров и сетей столь большие пакеты не используются. При передаче по сетям различного типа длина пакета выбирается с учетом максимальной длины пакета протокола нижнего уровня, несущего IP-пакеты. Если это кадры Ethernet, то выбираются пакеты с максимальной длиной в 1500 байт, умещающиеся в поле данных кадра Ethernet. В стандарте предусматривается, что все хосты должны быть готовы принимать пакеты вплоть до 576 байт длиной (приходят ли они целиком или по фрагментам). Хостам рекомендуется отправлять пакеты размером более чем 576 байт, только если они уверены, что принимающий хост или промежуточная сеть готовы обслуживать пакеты такого размера.

Поле *Идентификатор пакета* (*Identification*) занимает 2 байта и используется для распознавания пакетов, образовавшихся путем фрагментации исходного пакета. Все фрагменты должны иметь одинаковое значение этого поля.

Поле *Флаги* (*Flags*) занимает 3 бита и содержит признаки, связанные с фрагментацией. Установленный бит DF (Do not Fragment) запрещает маршрутизатору фрагментировать данный пакет, а установленный бит MF (More Fragments) говорит о том, что данный пакет является промежуточным (не последним) фрагментом. Оставшийся бит зарезервирован.

Поле *Смещение фрагмента* (*Fragment Offset*) занимает 13 бит и задает смещение в байтах поля данных этого пакета от начала общего поля данных исходного пакета, подвергнутого фрагментации. Используется при сборке/разборке фрагментов пакетов при передачах их между сетями с различными величинами MTU. Смещение должно быть кратно 8 байт.

Поле *Время жизни* (*Time to Live*) занимает один байт и означает предельный срок, в течение которого пакет может перемещаться по сети. Время жизни данного пакета измеряется в секундах и задается источником передачи. На маршрутизаторах и в других узлах сети по истечении каждой секунды из текущего времени жизни вычитается единица; единица вычитается и в том случае, когда время задержки меньше секунды. Поскольку современные маршрутизаторы редко обрабатывают пакет дольше, чем за одну секунду, то время жизни можно считать равным максимальному числу узлов, которые разрешено пройти данному пакету до того, как он достигнет места назначения. Если параметр времени жизни станет нулевым до того, как пакет достигнет получателя, этот пакет будет уничтожен. Время жизни можно рассматривать как часовой механизм самоуничтожения. Значение этого поля изменяется при обработке заголовка IP-пакета.

Идентификатор *Протокол верхнего уровня* (*Protocol*) занимает один байт и указывает, какому протоколу верхнего уровня принадлежит информация, размещенная в поле данных пакета (например, это могут быть сегменты протокола TCP дейтаграммы UDP, пакеты ICMP или OSPF). Значения идентификаторов для различных протоколов приводятся в документе RFC «Assigned Numbers».

Контрольная сумма (*Header Checksum*) занимает 2 байта и рассчитывается только по заголовку. Поскольку некоторые поля заголовка меняют свое значение в процессе передачи пакета по сети (например, время жизни), контрольная сумма проверяется и повторно рассчитывается при каждой обработке IP-заголовка. Контрольная сумма - 16 бит - подсчитывается как дополнение к сумме всех 16-битовых слов заголовка. При вычислении контрольной суммы значение самого поля «контрольная сумма» устанавливается в нуль. Если контрольная сумма неверна, то пакет будет отброшен, как только

ошибка будет обнаружена.

Поля *IP-адрес источника* (*Source IP Address*) и *IP-адрес назначения* (*Destination IP Address*) имеют одинаковую длину - 32 бита - и одинаковую структуру.

Поле *Опции* (*IP Options*) является необязательным и используется обычно только при отладке сети. Механизм опций предоставляет функции управления, которые необходимы или просто полезны при определенных ситуациях, однако он не нужен при обычных коммуникациях. Это поле состоит из нескольких подполей, каждое из которых может быть одного из восьми предопределенных типов. В этих подполях можно указывать точный маршрут прохождения маршрутизаторов, регистрировать проходимые пакетом маршрутизаторы, помещать данные системы безопасности, а также временные отметки. Так как число подполей может быть произвольным, то в конце поля *Опции* должно быть добавлено несколько байт для выравнивания заголовка пакета по 32-битной границе.

Поле *Выравнивание* (*Padding*) используется для того, чтобы убедиться в том, что IP-заголовок заканчивается на 32-битной границе. Выравнивание осуществляется нулями.

Ниже приведена распечатка значений полей заголовка одного из реальных IP-пакетов, захваченных в сети Ethernet средствами анализатора протоколов Microsoft Network Monitor.

IP: Version = 4 (0x4)
IP: Header Length = 20 (0x14)
IP: Service Type = 0 (0x0)
IP: Precedence = Routine
IP: ...0.... = Normal Delay
IP: ….0... = Normal Throughput
IP: .....0.. = Normal Reliability
IP: Total Length = 54 (0x36)
IP: Identification = 31746 (Ox7C02)
IP: Flags Summary = 2 (0x2)
IP: ....0 = Last fragment in datagram
IP: ......1. = Cannot fragment datagram
IP: Fragment Offset = 0 (0x0) bytes
IP: Time to Live = 128 (0x80)
IP: Protocol = TCP - Transmission Control
IP: Checksum = OxEB86
IP: Source Address = 194.85.135.75
IP: Destination Address - 194.85.135.66
IP: Data: Number of data bytes remaining = 34 (0x0022)

5.3.3. Таблицы маршрутизации в IP-сетях

Программные модули протокола IP устанавливаются на всех конечных станциях и маршрутизаторах сети. Для продвижения пакетов они используют таблицы маршрутизации.

Примеры таблиц различных типов маршрутизаторов

Структура таблицы маршрутизации стека TCP/IP соответствует общим принципам построения таблиц маршрутизации, рассмотренным выше. Однако важно отметить, что вид таблицы IP-маршрутизации зависит от конкретной реализации стека TCP/IP. Приведем пример трех вариантов таблицы маршрутизации, с которыми мог бы работать маршрутизатор Ml в сети, представленной на рис. 5.13.

Если представить, что в качестве маршрутизатора M1 в данной сети работает штатный программный маршрутизатор MPR операционной системы Microsoft Windows NT, то его таблица маршрутизации могла бы иметь следующий вид (табл. 5.9).

Таблица 5.9. Таблица программного маршрутизатора MPR Windows NT

Network Address	Netmask	Gateway Address	Interface	Metric
127.0.0.0	255.0.0.0	127.0.0.1	127.0.0.1	1
0.0.0.0	0.0.0.0	198.21.17.7	198.21.17.5	1

56.0.0.0	255.0.0.0	213.34.12.4	213.34.12.3	15
116.0.0.0	255.0.0.0	213.34.12.4	213.34.12.3	13
129.13.0.0	255.255.0.0	198.21.17.6	198.21.17.5	2
198.21.17.0	255.255.255.0	198.21.17.5	198.21.17.5	1
198.21.17.5	255.255.255.255	127.0.0.1	127.0.0.1	1
198.21.17.255	255.255.255.255	198.21.17.5	198.21.17.5	1
213.34.12.0	255.255.255.0	213.34.12.3	213.34.12.3	1
213.34.12.3	255.255.255.255	127.0.0.1	127.0.0.1	1
213.34.12.255	255.255.255.255	213.34.12.3	213.34.12.3	1
224.0.0.0	224.0.0.0	198.21.17.5	198.21.17.5	1
224.0.0.0	224.0.0.0	213.34.12.3	213.34.12.3	1
255.255.255.255	255.255.255.255	198.21.17.5	198.21.17.5	1

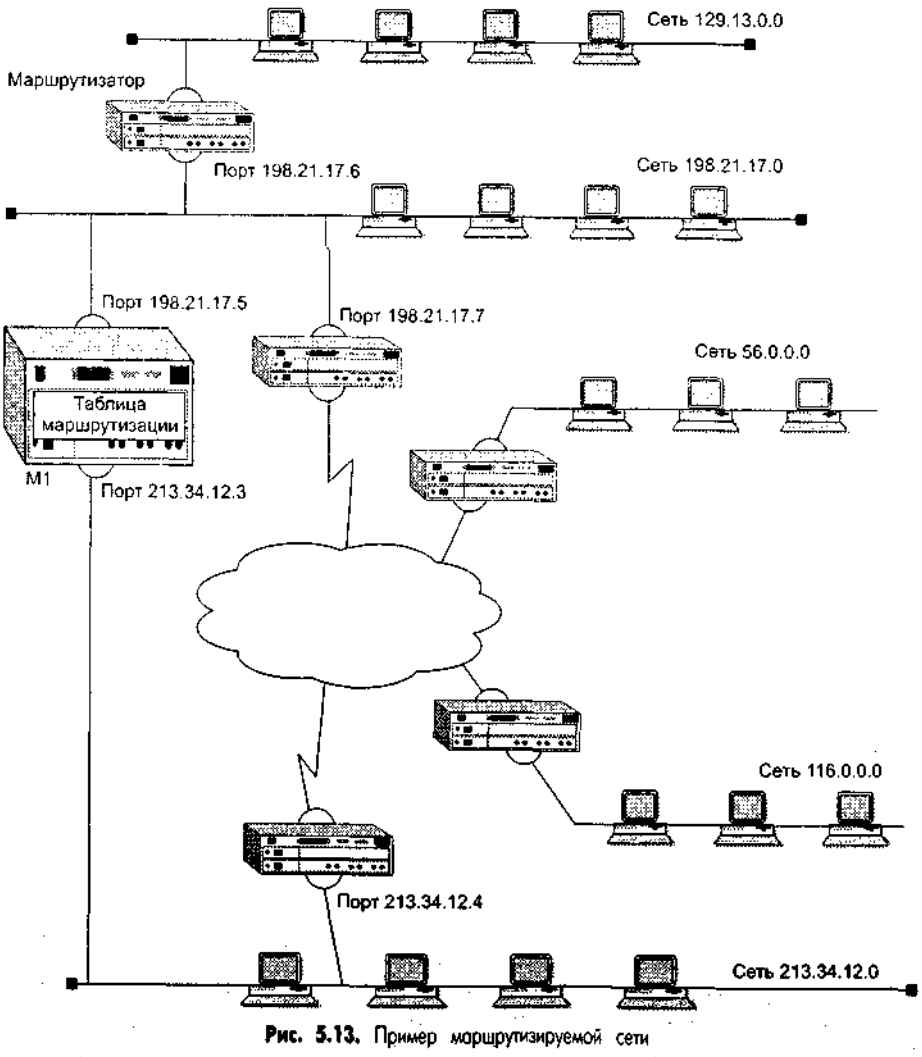


Рис. 5.13. Пример маршрутизируемой сети

Если на месте маршрутизатора Ml установить аппаратный маршрутизатор NetBuilder II компании 3Com, то его таблица маршрутизации для этой же сети может выглядеть так, как показано в табл 5.10.

Таблица 5.10. Таблица маршрутизации аппаратного маршрутизатора NetBuilder II компании 3Com
NetBuilder# Show – IP AllRoutes
Total Routes = 5 Total Direct Networks = 2

Destination	Mask	Gateway	Metric	Status	TTL	Source
198.21.17.0	255.255.255.0	198.21.17.5	0	Up	–	Connected
213.34.12.0	255.255.255.0	213.34.12.3	0	Up	–	Connected

56.0.0.0	255.0.0.0	213.34.12.4	14	Up	–	Static
116.0.0.0	255.0.0.0	213.34.12.4	12	Up	–	Static
129.13.0.0	255.255.0.0	198.21.17.6	1	Up	160	RIP

Таблица 5.11 представляет собой таблицу маршрутизации для маршрутизатора Ml, реализованного в виде программного маршрутизатора одной из версий операционной системы Unix.

Таблица 5.11. Таблица маршрутизации Unix-маршрутизатора

Destination	Gateway	Flags	Refcnt	Use	Interface
127.0.0.0	127.0.01	UH	1	154	loO
Default	198.21.17.7	UG	5	43270	leO
198.21.17.0	198.21.17.5	U	35	246876	leO
213.34.12.0	213.34.12.3	U	44	132435	le1
129.13.0.0	198.21.1.7.6	UG	6	16450	leO
56.0.0.0	213.34.12.4	UG	12	5764	le1
116.0.0.0	213.34.12.4	UG	21	23544	le1

ПРИМЕЧАНИЕ Заметим, что поскольку между структурой сети и таблицей маршрутизации в принципе нет однозначного соответствия, то и для каждого из приведенных вариантов таблицы можно предложить свои «подварианты», отличающиеся выбранным маршрутом к той или иной сети. В данном случае внимание концентрируется на существенных различиях в форме представления маршрутной информации разными реализациями маршрутизаторов

Назначение полей таблицы маршрутизации

Несмотря на достаточно заметные внешние различия, во всех трех таблицах есть все те ключевые параметры, необходимые для работы маршрутизатора, которые были рассмотрены ранее при обсуждении концепции маршрутизации.

К таким параметрам, безусловно, относятся адрес сети назначения (столбцы «Destination» в маршрутизаторах NetBuilder и Unix или «Network Address» в маршрутизаторе MPR) и адрес следующего маршрутизатора (столбцы «Gateway» в маршрутизаторах NetBuilder и Unix или «Gateway Address» в маршрутизаторе MPR).

Третий ключевой параметр – адрес порта, на который нужно направить пакет, в некоторых таблицах указывается прямо (поле «Interface» в таблице Windows NT), а в некоторых – косвенно. Так, в таблице Unix-маршрутизатора вместо адреса порта задается его условное наименование – 1eO для порта с адресом 198.21.17.5, le1 для порта с адресом 213.34.12.3 и 1oO для внутреннего порта с адресом 127.0.0.1.

В маршрутизаторе NetBuilder II поле, обозначающее выходной порт в какой-либо форме, вообще отсутствует. Это объясняется тем, что адрес выходного порта всегда можно косвенно определить по адресу следующего маршрутизатора. Например, попробуем определить по табл. 5.10 адрес выходного порта для сети 56.0.0.0. Из таблицы следует, что следующим маршрутизатором для этой сети будет маршрутизатор с адресом 213.34.12.4. Адрес следующего маршрутизатора должен принадлежать одной из непосредственно присоединенных к маршрутизатору сетей, и в данном случае это сеть 213.34.12.0. Маршрутизатор имеет порт, присоединенный к этой сети, и адрес этого порта 213.34.12.3 мы находим в поле «Gateway» второй строки таблицы маршрутизации, которая описывает непосредственно присоединенную сеть 213.34.12.0. Для непосредственно присоединенных сетей адресом следующего маршрутизатора всегда является адрес собственного порта маршрутизатора. Таким образом, адрес выходного порта для сети 56.0.0 – это адрес 213.34.12.3.

Остальные параметры, которые можно найти в представленных версиях таблицы маршрутизации, являются необязательными для принятия решения о пути следования пакета.

Наличие или отсутствие поля маски в таблице говорит о том, насколько современен данный маршрутизатор. Стандартным решением сегодня является использование поля маски в каждой записи

таблицы, как это сделано в таблицах маршрутизаторов MPR Windows NT (поле «Netmask») и NetBuilder (поле «Mask»). Обработка масок при принятии решения маршрутизаторами будет рассмотрена ниже. Отсутствие поля маски говорит о том, что либо маршрутизатор рассчитан на работу только с тремя стандартными классами адресов, либо он использует для всех записей одну и ту же маску, что снижает гибкость маршрутизации.

Метрика, как видно из примера таблицы Unix-маршрутизатора, является необязательным параметром. В остальных двух таблицах это поле имеется, однако оно используется только в качестве признака непосредственно подключенной сети. Действительно, если в таблице маршрутизации каждая сеть назначения упомянута только один раз, то поле метрики не будет приниматься во внимание при выборе маршрута, так как выбор отсутствует. А вот признак непосредственно подключенной сети маршрутизатору нужен, поскольку пакет для этой сети обрабатывается особым способом – он не передается следующему маршрутизатору, а отправляется узлу назначения. Поэтому метрика 0 для маршрутизатора NetBuilder или 1 для маршрутизатора MPR просто говорит маршрутизатору, что эта сеть непосредственно подключена к его порту, а другое значение метрики соответствует удаленной сети. Выбор значения метрики для непосредственно подключенной сети является достаточно произвольным, главное, чтобы метрика удаленной сети отсчитывалась с учетом этого выбранного начального значения. В Unix-маршрутизаторе используется поле признаков, где флаг G отмечает удаленную сеть, а его отсутствие – непосредственно подключенную.

Однако существуют ситуации, когда маршрутизатор должен обязательно хранить значение метрики для записи о каждой удаленной сети. Эти ситуации возникают, когда записи в таблице маршрутизации являются результатом работы некоторых протоколов маршрутизации, например протокола RIP. В таких протоколах новая информация о какой-либо удаленной сети сравнивается с имеющейся в таблице, и если метрика новой информации лучше имеющейся, то новая запись вытесняет имеющуюся. В таблице Unix-маршрутизатора поле метрики отсутствует, и это значит, что он не использует протокол RIP.

Флаги записей присутствуют только в таблице Unix-маршрутизатора. Они описывают характеристики записи.

- U – показывает, что маршрут активен и работоспособен. Аналогичный смысл имеет поле «Status» в маршрутизаторе NetBuilder.
- H – признак специфического маршрута к определенному хосту. Маршрут ко всей сети, к которой принадлежит данный хост, может отличаться от данного маршрута.
- G – означает, что маршрут пакета проходит через промежуточный маршрутизатор (gateway). Отсутствие этого флага отмечает непосредственно подключенную сеть.
- D – означает, что маршрут получен из сообщения Redirect (перенаправление) протокола ICMP. Этот признак может присутствовать только в таблице маршрутизации конечного узла. Признак означает, что конечный узел в какой-то предыдущей передаче пакета выбрал не самый рациональный следующий маршрутизатор на пути к данной сети, и этот маршрутизатор с помощью протокола ICMP сообщил, что все последующие пакеты к данной сети нужно отправлять через другой следующий маршрутизатор. Протокол ICMP может посылать сообщения только узлу-отправителю, поэтому у промежуточного маршрутизатора этот признак встретиться не может. Признак никак не влияет на процесс маршрутизации, он только указывает администратору источник появления записи.

В таблице Unix-маршрутизатора используются еще два поля, имеющих справочное значение. Поле «Refcnt» показывает, сколько раз на данный маршрут ссылались при продвижении пакетов. Поле «Use» отражает количество пакетов, переданных по данному маршруту.

В таблице маршрутизатора NetBuilder также имеются два справочных поля. Поле времени жизни «TTL» (Time To Live) имеет смысл для динамических записей, которые имеют ограниченный срок жизни. Текущее значение поля показывает оставшийся срок жизни записи в секундах. Поле «Source» отражает источник появления записи в таблице маршрутизации. Хотя это поле имеется не во всех маршрутизаторах, но практически для всех маршрутизаторов существуют три основных источника появления записи в таблице.

Источники и типы записей в таблице маршрутизации

Первым источником является программное обеспечение стека TCP/IP. При инициализации маршрутизатора это программное обеспечение автоматически заносит в таблицу несколько записей, в

результате чего создается так называемая *минимальная таблица маршрутизации.*

Это, во-первых, записи о непосредственно подключенных сетях и маршрутизаторах по умолчанию, информация о которых появляется в стеке при ручном конфигурировании интерфейсов компьютера или маршрутизатора. К таким записям в приведенных примерах относятся записи о сетях 213.34.12.0 и 198.21.17.0, а также запись о маршрутизаторе по умолчанию – default в Unix-маршрутизаторе и 0.0.0.0 в маршрутизаторе MPR Windows NT. В приведенном примере таблицы для маршрутизатора NetBuilder маршрутизатор по умолчанию не используется, следовательно, при поступлении пакета с адресом назначения, отсутствующим в таблице маршрутизации, этот пакет будет отброшен.

Во-вторых, программное обеспечение автоматически заносит в таблицу маршрутизации записи об адресах особого назначения. В приведенных примерах таблица маршрутизатора MPR Windows NT содержит наиболее полный набор записей такого рода. Несколько записей в этой таблице связаны с особым адресом 127.0.0.0 (loopback), который используется для локального тестирования стека TCP/IP. Пакеты, направленные в сеть с номером 127.0.0.0, не передаются протоколом IP на канальный уровень для последующей передачи в сеть, а возвращаются в источник – локальный модуль IP. Записи с адресом 224.0.0.0 требуются для обработки групповых адресов (multicast address). Кроме того, в таблицу могут быть занесены адреса, предназначенные для обработки широковещательных рассылок (например, записи 8 и 11 содержат адрес отправки широковещательного сообщения в соответствующих подсетях, а последняя запись в таблице – адрес ограниченной широковещательной рассылки сообщения). Заметим, что в некоторых таблицах записи об особых адресах вообще отсутствуют.

Вторым источником появления записи в таблице является администратор, непосредственно формирующий запись с помощью некоторой системной утилиты, например программы route, имеющейся в операционных системах Unix и Windows NT. В аппаратных маршрутизаторах также всегда имеется команда для ручного задания записей таблицы маршрутизации. Заданные вручную записи всегда являются статическими, то есть не имеют срока истечения жизни. Эти записи могут быть как постоянными, то есть сохраняющимися при перезагрузке маршрутизатора, так и временными, хранящимися в таблице только до выключения устройства. Часто администратор вручную заносит запись default о маршрутизаторе по умолчанию. Таким же образом в таблицу маршрутизации может быть внесена запись о специфичном для узла маршруте. Специфичный для узла маршрут содержит вместо номера сети полный IP-адрес, то есть адрес, имеющий ненулевую информацию не только в поле номера сети, но и в поле номера узла. Предполагается, что для такого конечного узла маршрут должен выбираться не так, как для всех остальных узлов сети, к которой он относится. В случае когда в таблице есть разные записи о продвижении пакетов для всей сети и ее отдельного узла, при поступлении пакета, адресованного узлу, маршрутизатор отдаст предпочтение записи с полным адресом узла.

И наконец, третьим источником записей могут быть протоколы маршрутизации, такие как RIP или OSPF. Такие записи всегда являются динамическими, то есть имеют ограниченный срок жизни. Программные маршрутизаторы Windows NT и Unix не показывают источник появления той или иной записи в таблице, а маршрутизатор NetBuilder использует для этой цели поле «Source». В приведенном в табл. 5.10 примере первые две записи созданы программным обеспечением стека на основании данных о конфигурации портов маршрутизатора – это показывает признак «Connected». Следующие две записи обозначены как «Static», что указывает на то, что их ввел вручную администратор. Последняя запись является следствием работы протокола RIP, поэтому в ее поле «TTL» имеется значение 160.

5.3.4. Маршрутизация без использования масок

Рассмотрим на примере IP-сети (рис. 5.14) алгоритм работы средств сетевого уровня по продвижению пакета в составной сети. При этом будем считать, что все узлы сети рассматриваемой в примере, имеют адреса, основанные на классах, без использования масок. Особое внимание будет уделено взаимодействию протокола IP с протоколами разрешения адресов ARP и DNS.

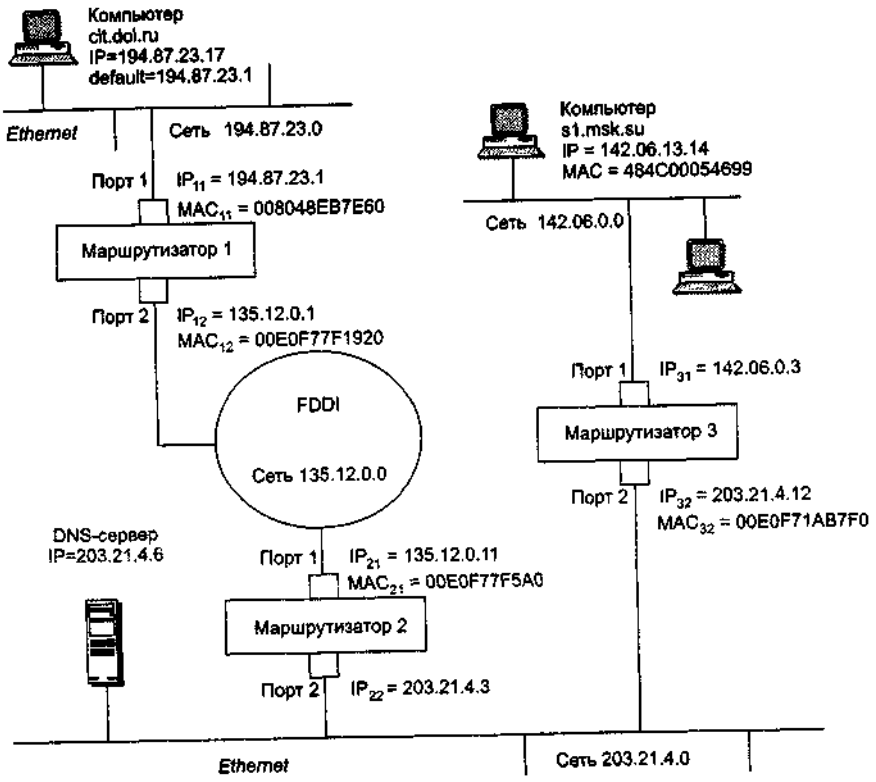


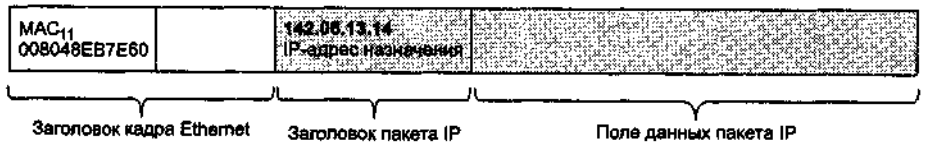
Рис. 5.14. Пример взаимодействия компьютеров через сеть

1. Итак пусть пользователь компьютера cit.dol.ru, находящегося в сети Etnernet и имеющего IP-адрес 194.87.23.17 (адрес класса C), обращается по протоколу FTP к компьютеру sl.msk.su, принадлежащему другой сети Ethernet и имеющему IP-адрес 142.06.13.14 (адрес класса B):
   > ftp sl.msk.su
   Модуль FTP упаковывает свое сообщение в сегмент транспортного протокола TCP который в свою очередь помещает свой сегмент в пакет протокола IP. В заголовке IP-пакета должен быть указан IP-адрес узла назначения. Так как пользователь компьютера cit.dol.ru использует символьное имя компьютера sl.msk.su, то стек TCP/IP должен определить IP-адрес узла назначения самостоятельно. При конфигурировании стека TCP/IP в компьютере cit.dol.ru был задан его собственный IP-адрес, IP-адрес маршрутизатора по умолчанию и IP-адрес DNS-сервера. Модуль IP может сделать запрос к серверу DNS, но обычно сначала просматривается локальная таблица соответствия символьных имен и IP-адресов. Такая таблица хранится чаще всего в виде текстового файла простой структуры – каждая его строка содержит запись об одном символьном имени и его IP-адресе. В ОС Unix такой файл традиционно носит имя hosts и находится в каталоге /etc.
2. Будем считать, что компьютер cit.dol.ru имеет файл hosts, а в нем есть строка 142.06.13.14 sl.msk.su
   Таким образом, разрешение имени выполняется локально, и протокол IP может теперь формировать IP-пакеты с адресом назначения 142.06.13.14 для взаимодействия с компьютером sl.msk.su.
3. Модуль IP компьютера cit.dol.ru проверяет, нужно ли маршрутизировать пакеты с адресом 142.06.13.14. Так как адрес сети назначения (142.06.0.0) не совпадает с адресом (194.87.23.0) сети, которой принадлежит компьютер-отправитель, то маршрутизация необходима.
4. Компьютер cit.dol.ru начинает формировать кадр Ethernet для отправки IP-пакета маршрутизатору по умолчанию, IP-адрес которого известен – 194.87.23.1, но неизвестен MAC-адрес, необходимый для перемещения кадра в локальной сети. Для определения MAC-адреса маршрутизатора протокол IP обращается к протоколу ARP, который просматривает ARP-таблицу. Если в последнее время компьютер cit.dol.ru выполнял какие-либо межсетевые обмены, то скорее всего искомая запись, содержащая соответствие между IP- и MAC-адресами маршрутизатора по умолчанию уже находится в кэш-таблице протокола ARP. Пусть в данном случае нужная запись была найдена именно в кэш-таблице:

194.87.23.1 008048EB7E60

Обозначим найденный MAC-адрес 008048EB7E60 в соответствии с номером маршрутизатора и его порта через MAC_{11}.

5. В результате компьютер cit.dol.ru отправляет по локальной сети пакет, упакованный в кадр Ethernet, имеющий следующие поля:

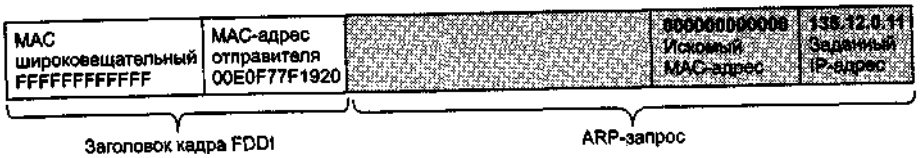


6. Кадр принимается портом 1 маршрутизатора 1 в соответствии с протоколом Ethernet, так как MAC-узел этого порта распознает свой адрес MAC_{11}. Протокол Ethernet извлекает из этого кадра IP-пакет и передает его программному обеспечению маршрутизатора, реализующему протокол IP. Протокол IP извлекает из пакета адрес назначения 142.06.13.14 и просматривает записи своей таблицы маршрутизации. Пусть маршрутизатор 1 имеет в своей таблице маршрутизации запись

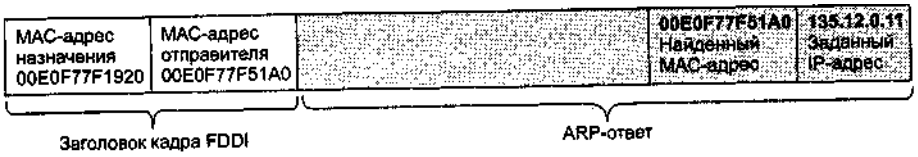
142.06.0.0 135.12.0.11 2.

которая говорит о том, что пакеты для сети 142.06.0.0 нужно передавать маршрутизатору 135.12.0.11, находящемуся в сети, подключенной к порту 2 маршрутизатора 1.

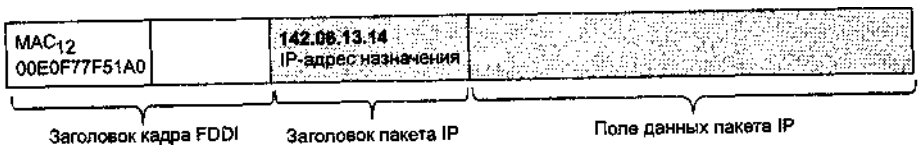
7. Маршрутизатор 1 просматривает параметры порта 2 и находит, что к нему подключена сеть FDDI. Так как сеть FDDI имеет значение MTU большее, чем сеть Ethernet, то фрагментация IP-пакета не требуется. Поэтому маршрутизатор 1 формирует кадр формата FDDI. На этом этапе модуль IP должен определить MAC-адрес следующего маршрутизатора по известному IP .адресу 135.12.0.11. Для этого он обращается к протоколу ARP. Допустим, что нужной записи в кэш-таблице не оказалось, тогда в сеть FDDI отправляется широковещательный ARP-запрос, содержащий наряду с прочей следующую информацию.



Порт 1 маршрутизатора 2 распознает свой IP-адрес и посылает ARP-ответ по адресу запросившего узла:



Теперь, зная MAC-адрес следующего маршрутизатора OOEOF77F51AO, маршрутизатор 1 отсылает кадр FDDI по направлению к маршрутизатору 2. Заметим, что в поле IP-адреса назначения никаких изменений не произошло.



8. Аналогично действует модуль IP на маршрутизаторе 2. Получив кадр FDDI, он отбрасывает его заголовок, а из заголовка IP извлекает IP-адрес сети назначения и просматривает свою таблицу

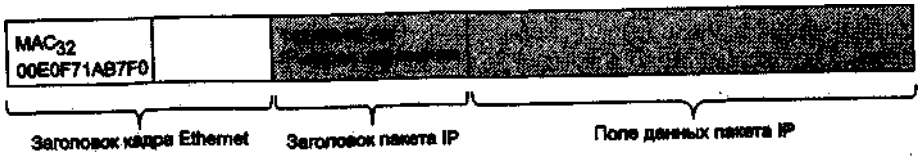
маршрутизации. Там он может найти запись о конкретной сети назначения:

142.06.0.0 203.21.4.12 2

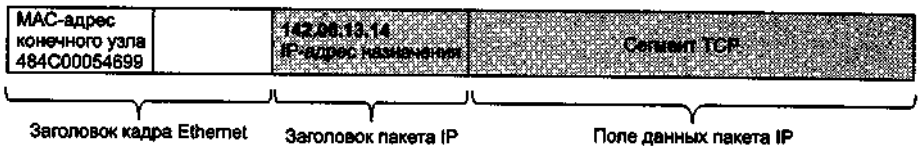
или при отсутствии такой записи будет использована запись о маршрутизаторе по умолчанию:

default 203.21.4.12 2.

Определив IP-адрес следующего маршрутизатора 203.21.4.12, модуль IP формирует кадр Ethernet для передачи пакета маршрутизатору 3 по сети Ethernet. С помощью протокола ARP он находит MAC-адрес этого маршрутизатора и помещает его в заголовок кадра. IP-адрес узла назначения, естественно, остается неизменным.



9. Наконец, после того как пакет поступил в маршрутизатор сети назначения – маршрутизатор 3, – появляется возможность передачи этого пакета компьютеру назначения. Маршрутизатор 3 определяет, что пакет нужно передать в сеть 142.06.0.0, которая непосредственно подключена к его первому порту. Поэтому он посылает ARP-запрос по сети Ethernet с IP-адресом компьютера sl.msk.su. ARP-ответ содержит MAC-адрес конечного узла, который модуль IP передает канальному протоколу для формирования кадра Ethernet:



10. Сетевой адаптер компьютера sl.msk.su захватывает кадр Ethernet, обнаруживает совпадение MAC-адреса, содержащегося в заголовке, со своим собственным адресом и направляет его модулю IP. После анализа полей IP-заголовка из пакета извлекаются данные, которые в свою очередь содержат сообщение выше лежащего протокола. Поскольку в данном примере рассматривается обмен данными по протоколу FTP, который использует в качестве транспортного протокола TCP, то в поле данных IP-пакета находится TCP-сегмент. Определив из TCP-заголовка номер порта, модуль IP переправляет сегмент в соответствующую очередь, из которой данный сегмент попадет программному модулю FTP-сервера.

5.3.5. Маршрутизация с использованием масок

Использование масок для структуризации сети

Алгоритм маршрутизации усложняется, когда в систему адресации узлов вносятся дополнительные элементы – маски. В чем же причина отказа от хорошо себя зарекомендовавшего в течение многих лет метода адресации, основанного на классах? Таких причин несколько, и одна из них – потребность в структуризации сетей.

Часто администраторы сетей испытывают неудобства из-за того, что количество централизованно выделенных им номеров сетей недостаточно для того, чтобы структурировать сеть надлежащим образом, например разместить все слабо взаимодействующие компьютеры по разным сетям. В такой ситуации возможны два пути. Первый из них связан с получением от InterNIC или поставщика услуг Internet дополнительных номеров сетей. Второй способ, употребляющийся чаще, связан с использованием технологии масок, которая позволяет разделять одну сеть на несколько сетей.

Допустим, администратор получил в свое распоряжение адрес класса B: 129.44.0.0. Он может организовать сеть с большим числом узлов, номера которых он может брать из диапазона 0.0.0.1-

0.0.255.254 (с учетом того, что адреса из одних нулей и одних единиц имеют специальное назначение и не годятся для адресации узлов). Однако ему не нужна одна большая неструктурированная сеть, производственная необходимость диктует администратору другое решение, в соответствии с которым сеть должна быть разделена на три отдельных подсети, при этом трафик в каждой подсети должен быть надежно локализован. Это позволит легче диагностировать сеть и проводить в каждой из подсетей особую политику безопасности.

Посмотрим, как решается эта проблема путем использования механизма масок.

Итак, номер сети, который администратор получил от поставщика услуг, – 129.44.0.0 (10000001 00101100 00000000 00000000). В качестве маски было выбрано значение 255.255.192.0 (11111111 11111111 11000000 00000000). После наложения маски на этот адрес число разрядов, интерпретируемых как номер сети, увеличилось с 16 (стандартная длина поля номера сети для класса B) до 18 (число единиц в маске), то есть администратор получил возможность использовать для нумерации подсетей два дополнительных бита. Это позволяет ему сделать из одного, централизованно заданного ему номера сети, четыре:

129.44.0.0 (10000001 00101100 00000000 00000000)
129.44.64.0 (10000001 00101100 01000000 00000000)
129.44.128.0 (10000001 00101100 10000000 00000000)
129.44.192.0 (10000001 00101100 11000000 00000000)

Два дополнительных последних бита в номере сети часто интерпретируются как номера подсетей (subnet), и тогда четыре перечисленных выше подсети имеют номера 0 (00), 1 (01), 2 (10) и 3 (11) соответственно.

ПРИМЕЧАНИЕ. Некоторые программные и аппаратные маршрутизаторы не поддерживают номера подсетей, которые состоят либо только из одних нулей, либо только из одних единиц. Например, для некоторых типов оборудования номер сети 129.44.0.0 с маской 255.255.192.0, использованный в нашем примере, окажется недопустимым, поскольку в этом случае разряды в поле номера подсети имеют значение 00. По аналогичным соображениям недопустимым может оказаться и номер сети 129.44.192.0 с тем же значением маски. Здесь номер подсети состоит только из единиц. Однако более современные маршрутизаторы свободны от этих ограничений. Поэтому, принимая решение об использовании механизма масок, необходимо выяснить характеристики того оборудования, которым вы располагаете, чтобы соответствующим образом сконфигурировать маршрутизаторы и компьютеры сети.

В результате использования масок была предложена следующая схема распределения адресного пространства (рис. 5.15).

Сеть, получившаяся в результате проведенной структуризации, показана на рис. 5.16. Весь трафик во внутреннюю сеть 129.44.0.0, направляемый из внешней сети, поступает через маршрутизатор Ml. В целях структуризации информационных потоков во внутренней сети установлен дополнительный маршрутизатор M2.

Все узлы были распределены по трем разным сетям, которым были присвоены номера 129.44.0.0, 129.44.64.0 и 129.44.128.0 и маски одинаковой длины – 255.255.192.0. Каждая из вновь образованных сетей была подключена к соответственно сконфигурированным портам внутреннего маршрутизатора M2. Кроме того, еще одна сеть (номер 129.44.192.0, маска 255.255.192.0) была выделена для создания соединения между внешним и внутренним маршрутизаторами. Особо отметим, что в этой сети для адресации узлов были заняты всего два адреса 129.44.192.1 (порт маршрутизатора M2) и 129.44.192.2 (порт маршрутизатора Ml), еще два адреса 129.44.192.0 и 129.44.192.255 являются особыми адресами. Следовательно, огромное число узлов (2^{14} - 4) в этой подсети никак не используются.

Извне сеть по-прежнему выглядит, как единая сеть класса B, а на местном уровне это полноценная составная сеть, в которую входят три отдельные сети. Приходящий общий трафик разделяется местным маршрутизатором M2 между этими сетями в соответствии с таблицей маршрутизации. (Заметим, что разделение большой сети, имеющей один адрес старшего класса, например A или B, с помощью масок несет в себе еще одно преимущество по сравнению с использованием нескольких адресов стандартных классов для сетей меньшего размера, например C. Оно позволяет скрыть внутреннюю структуру сети предприятия от внешнего наблюдения и тем повысить ее безопасность.)

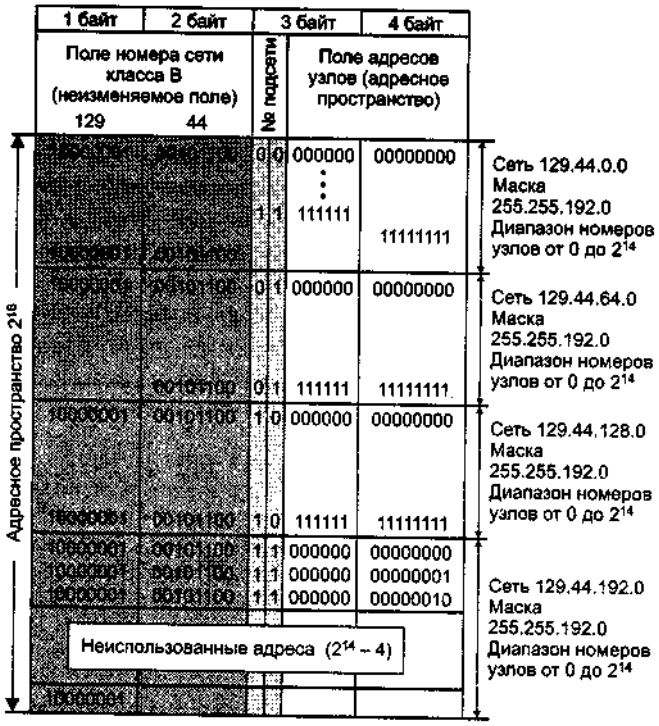


Рис. 5.15. Разделение адресного пространства сети класса B 129.44.0.0 на четыре равные части путем использования масок одинаковой длины 255.255.192.0

Рассмотрим, как изменяется работа модуля IP, когда становится необходимым учитывать наличие масок. Во-первых, в каждой записи таблицы маршрутизации появляется новое поле – поле маски.

Во-вторых, меняется алгоритм определения маршрута по таблице маршрутизации. После того как IP-адрес извлекается из очередного полученного IP-пакета, необходимо определить адрес следующего маршрутизатора, на который надо передать пакет с этим адресом. Модуль IP последовательно просматривает все записи таблицы маршрутизации. С каждой записью производятся следующие действия.

- Маска M, содержащаяся в данной записи, накладывается на IP-адрес узла назначения, извлеченный из пакета.
- Полученное в результате число является номером сети назначения обрабатываемого пакета. Оно сравнивается с номером сети, который помещен в данной записи таблицы маршрутизации.
- Если номера сетей совпадают, то пакет передается маршрутизатору, адрес которого помещен в соответствующем поле данной записи.

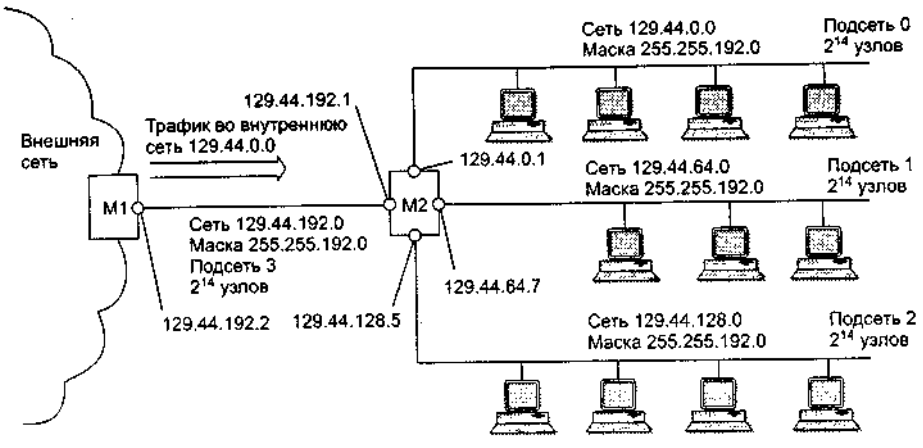


Рис. 5.16. Маршрутизация с использованием масок одинаковой длины

Теперь рассмотрим этот алгоритм на примере маршрутизации пакетов в сети, изображенной на рис. 5.16. Все маршрутизаторы внешней сети, встретив пакеты с адресами, начинающимися с 129.44

интерпретируют их как адреса класса B и направляют по маршрутам, ведущим к маршрутизатору Ml. Маршрутизатор Ml и свою очередь направляет весь входной трафик сети 129.44.0.0 на маршрутизатор M2, а именно на его порт 129.44.192.1.

Маршрутизатор M2 обрабатывает все поступившие на него пакеты в соответствии с таблицей маршрутизации (табл. 5.12).

Таблица 5.12. Таблица маршрутизатора M2 в сети с масками одинаковой длины

Номер сети	Маска	Адрес следующего маршрутизатора	Адрес порта	Расстояние
129.44.0.0	255.255.192.0	129.44.0.1	129.44.0.1	Подключена
129.44.64.0	255.255.192.0	129.44.64.7	129.44.64.7	Подключена
129.44.128.0	255.255.192.0	129.44.128.5	129.44.128.5	Подключена
129.44.192.0	255.255.192.0	129.44.192.1	129.44.192.1	Подключена
0.0.0.0	0.0.0.0	129.44.192.2	129.44.192.1	-

Первые четыре записи в таблице соответствуют внутренним подсетям, непосредственно подключенным к портам маршрутизатора M2.

Запись 0.0.0.0 с маской 0.0.0.0 соответствует маршруту по умолчанию. Действительно, любой адрес в пришедшем пакете после наложения на него маски 0.0.0.0 даст адрес сети 0.0.0.0, что совпадает с адресом, указанным в записи. Маршрутизатор выполняет сравнение с адресом 0.0.0.0 в последнюю очередь, в том случае когда пришедший адрес не дал совпадения ни с одной записью в таблице, отличающейся от 0.0.0.0. Записей с адресом 0.0.0.0 в таблице маршрутизации может быть несколько. В этом случае маршрутизатор передает пакет по всем таким маршрутам.

Пусть, например, с маршрутизатора Ml на порт 129.44.192.1 маршрутизатора M2 поступает пакет с адресом назначения 129.44.78.200. Модуль IP начинает последовательно просматривать все строки таблицы, до тех пор пока не найдет совпадения номера сети в адресе назначения и в строке таблицы. Маска из первой строки 255.255.192.0 накладывается на адрес 129.44.78.200, в результате чего получается номер сети 129.44.64.0.

В двоичном виде эта операция выглядит следующим образом:

10000001.00101100.01001110.11001000

11111111.11111111.11000000.00000000

10000001.00101100.01000000.00000000

Полученный номер 129.44.64.0 сравнивается с номером сети в первой строке таблицы 129.44.0.0. Поскольку они не совпадают, то происходит переход к следующей строке. Теперь извлекается маска из второй строки (в данном случае она имеет такое же значение, но в общем случае это совсем не обязательно) и накладывается на адрес назначения пакета 129.44.78.200. Понятно, что из-за совпадения длины масок будет получен тот же номер сети 129.44.64.0. Этот номер совпадает с номером сети во второй строке таблицы, а значит, найден маршрут для данного пакета – он должен быть отправлен на порт маршрутизатора 129.44.64.7 в сеть, непосредственно подключенную к данному маршрутизатору.

Вот еще пример. IP-адрес 129.44.141.15(10000001 00101100 10001101 00001111), который при использовании классов делится на номер сети 129.44.0.0 и номер узла 0.0.141.15, теперь, при использовании маски 255.255.192.0, будет интерпретироваться как пара: 129.44.128.0 – номер сети, 0.0.13.15 – номер узла.

Использование масок переменной длины

В предыдущем примере использования масок (см. рис. 5.15 и 5.16) все подсети имеют одинаковую длину поля номера сети – 18 двоичных разрядов, и, следовательно, для нумерации узлов в каждой из них отводится по 14 разрядов. То есть все сети являются очень большими и имеют одинаковый размер. Однако в этом случае, как и во многих других, более эффективным явилось бы разбиение сети на подсети разного размера. В частности, большое число узлов, вполне желательное для пользовательской подсети, явно является избыточным для подсети, которая связывает два маршрутизатора по схеме «точка-точка». В этом случае требуются всего два адреса для адресации двух портов соседних маршрутизаторов. В предыдущем же примере для этой вспомогательной сети Ml - M2 был использован

номер, позволяющий адресовать 2^{14} узлов, что делает такое решение неприемлемо избыточным. Администратор может более рационально распределить имеющееся в его распоряжении адресное пространство с помощью масок переменной длины.

На рис. 5.17 приведен пример распределения адресного пространства, при котором избыточность имеющегося множества IP-адресов может быть сведена к минимуму. Половина из имеющихся адресов (2^{15}) была отведена для создания сети с адресом 129.44.0.0 и маской 255.255.128.0. Следующая порция адресов, составляющая четверть всего адресного пространства (2^{14}), была назначена для сети 129.44.128.0 с маской 255.255.192.0. Далее в пространстве адресов был «вырезан» небольшой фрагмент для создания сети, предназначенной для связывания внутреннего маршрутизатора М2 с внешним маршрутизатором Ml.

В IP-адресе такой вырожденной сети для поля номера узла как минимум должны быть отведены два двоичных разряда. Из четырех возможных комбинаций номеров узлов: 00, 01, 10 и 11 два номера имеют специальное назначение и не могут быть присвоены узлам, но оставшиеся два 10 и 01 позволяет адресовать порты маршрутизаторов. В нашем примере сеть была выбрана с некоторым запасом - на 8 узлов Поле номера узла в таком случае имеет 3 двоичных разряда, маска в десятичной нотации имеет вид 255.255.255.248, а номер сети, как видно из рис. 5.17, равен в данном конкретном случае 129.44.192.0. Если эта сеть является локальной, то на ней могут быть расположены четыре узла помимо двух портов маршуртизаторов.

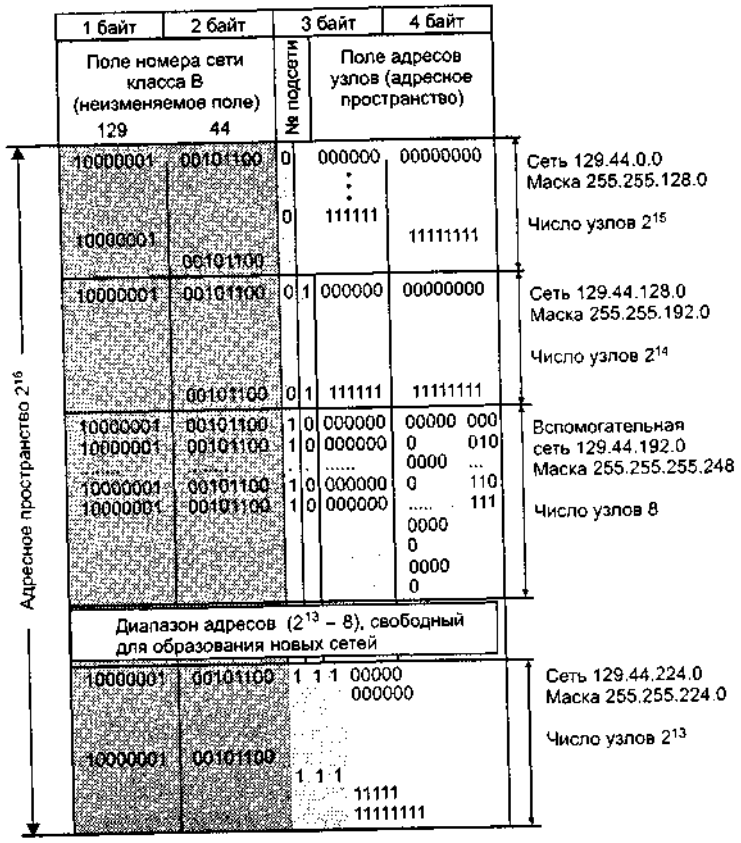


Рис. 5.17. Разделение адресного пространства сети класса В 129.44.0.0 на сети разного размера путем использования масок переменной длины

ПРИМЕЧАНИЕ Заметим, что глобальным связям между маршрутизаторами типа «точка-точки не обязательно давать IP-адреса, так как к такой сети не могут подключаться никакие другие узлы, кроме двух портов маршрутизаторов. Однако чаще всего такой вырожденной сети все же дают IP-адрес. Это делается, например, для того чтобы скрыть внутреннюю структуру сети и обращаться к ней по одному адресу входного порта маршрутизатора, в данном примере по адресу 129.44.192.1. Кроме того, этот адрес может понадобиться при туннелировании немаршрутизируемых протоколов в IP-пакеты, что будет рассмотрено ниже.

Оставшееся адресное пространство администратор может «нарезать», на разное количество сетей

разного объема в зависимости от своих потребностей. Из оставшегося пула (2^{14} - 4) адресов администратор может образовать еще одну достаточно большую сеть с числом узлов 2^{13}. При этом свободными останутся почти столько же адресов (2^{13} - 4), которые также могут быть использованы для создания новых сетей. К примеру, из этого «остатка» можно образовать 31 сеть, каждая из которых равна размеру стандартной сети класса С, и к тому же еще несколько сетей меньшего размера. Ясно, что разбиение может быть другим, но в любом случае с помощью масок переменного размера администратор всегда имеет возможность гораздо рациональнее использовать все имеющиеся у него адреса.

На рис. 5.18 показана схема сети, структурированной с помощью масок переменной длины.

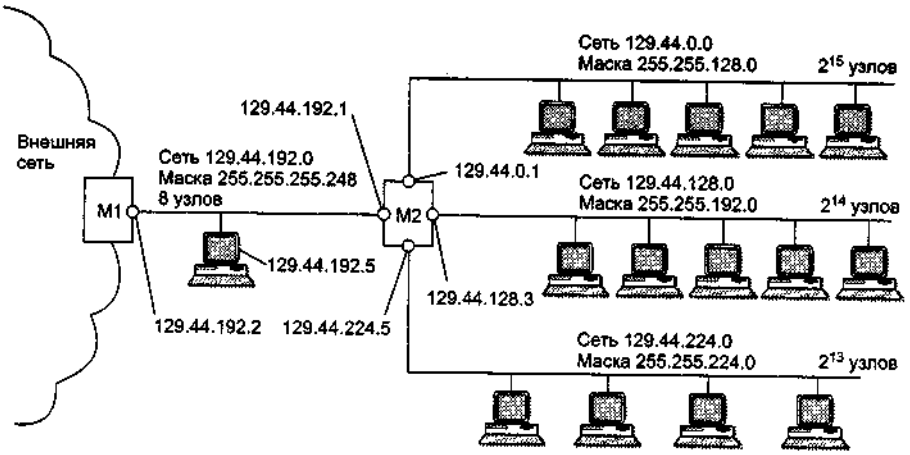


Рис. 5.18. Сеть, структурированная с использованием масок переменной длины

Таблица маршрутизации М2, соответствующая структуре сети, показанной на рис. 5.18, содержит записи о четырех непосредственно подключенных сетях и запись о маршрутизаторе по умолчанию (табл. 5.13). Процедура поиска маршрута при использовании масок переменной длины ничем не отличается от подобной процедуры, описанной ранее для масок одинаковой длины.

Таблица 5.13. Таблица маршрутизатора М2 в сети с масками переменной длины

Номер сети	Маска	Адрес следующего маршрутизатора	Адрес порта	Расстояние
129.44.0.0	255.255.128.0	129.44.0.1	129.44.0.1	Подключена
129.44.128.0	255.255.192.0	129.44.128.3	129.44.128.3	Подключена
129.44.192.0	255.255.255.248	129.44.192.1	129.44.191.1	Подключена
129.44.224.0	255.255.224.0	129.44.224.5	129.44.224.5	Подключена
0.0.0.0	0.0.0.0	129.44.192.2	129.44.192.1	-

Некоторые особенности масок переменной длины проявляются при наличии так называемых «перекрытий». Под перекрытием понимается наличие нескольких маршрутов к одной и той же сети или одному и тому же узлу. В этом случае адрес сети в пришедшем пакете может совпасть с адресами сетей, содержащихся сразу в нескольких записях таблицы маршрутизации.

Рассмотрим пример. Пусть пакет, поступивший из внешней сети на маршрутизатор Ml, имеет адрес назначения 129.44.192.5. Ниже приведен фрагмент таблицы маршрутизации маршрутизатора Ml. Первая из приведенных двух записей говорит о том, что все пакеты, адреса которых начинаются на 129.44, должны быть переданы на маршрутизатор М2. Эта запись выполняет *агрегирование* адресов всех подсетей, созданных на базе одной сети 129.44.0.0. Вторая строка говорит о том, что среди всех возможных подсетей сети 129.44.0.0 есть одна, 129.44.192.0, для которой пакеты можно направлять непосредственно, а не через маршрутизатор М2.

Номер сети	Маска	Адрес следующего маршрутизатора	Адрес порта	Расстояние

…….	…….	……	……	…….
129.44.0.0	255.255.0.0	129.44.192.1	129.44.191.2	2
129.44.192.0	255.255.255.248	129.44.192.2	129.44.192.2	Подключена
…….	…….	…….	…….	…….

Если следовать стандартному алгоритму поиска маршрута по таблице, то сначала на адрес назначения 129.44.192.5 накладывается маска из первой строки 255.255.0.0 и получается результат 129.44.0.0, который совпадает с номером сети в этой строке. Но и при наложении на адрес 129.44.192.5 маски из второй строки 255.255.255.248 полученный результат 129.44.192.0 также совпадает с номером сети во второй строке. В таких случаях должно быть применено следующее правило: «Если адрес принадлежит нескольким подсетям в базе данных маршрутов, то продвигающий пакет маршрутизатор использует наиболее специфический маршрут, то есть выбирается адрес подсети, дающий большее совпадение разрядов».

В данном примере будет выбран второй маршрут, то есть пакет будет передан в непосредственно подключенную сеть, а не пойдет кружным путем через маршрутизатор М2.

Механизм выбора самого специфического маршрута является обобщением понятия «маршрут по умолчанию». Поскольку в традиционной записи для маршрута по умолчанию 0.0.0.0 маска 0.0.0.0 имеет нулевую длину, то этот маршрут считается самым неспецифическим и используется только при отсутствии совпадений со всеми остальными записями из таблицы маршрутизации.

ПРИМЕЧАНИЕ В IP-пакетах при использовании механизма масок по-прежнему передается только IP-адрес назначения, а маска сети назначения не передается. Поэтому из IP-адреса пришедшего пакета невозможно выяснить, какая часть адреса относится к номеру сети, а какая - к номеру узла. Если маски во всех подсетях имеют один размер, то это не создает проблем. Если же для образования подсетей применяют маски переменной рины, то маршрутизатор должен каким-то образом узнавать, каким адресам сетей какие маски соответствуют. Для этого используются протоколы маршрутизации, переносящие между маршрутизаторами не только служебную информацию об адресах сетей, но и о масках, соответствующих этим номерам. К таким протоколам относятся протоколы RIPv2 и OSPF, а вот, например, протокол RIP маски не распространяет и для использования масок переменной длины не подходит.

Технология бесклассовой междоменной маршрутизации CIDR

За последние несколько лет в сети Internet многое изменилось: резко возросло число узлов и сетей, повысилась интенсивность трафика, изменился характер передаваемых данных. Из-за несовершенства протоколов маршрутизации обмен сообщениями об обновлении таблиц стал иногда приводить к сбоям магистральных маршрутизаторов из-за перегрузки при обработке большого объема служебной информации. Так, в 1994 году таблицы магистральных маршрутизаторов в Internet содержали до 70 000 маршрутов.

На решение этой проблемы была направлена, в частности, и технология *бесклассовой междоменной маршрутизации* (*Classless Inter-Domain Routing, CIDR*), впервые о которой было официально объявлено в 1993 году, когда были опубликованы RFC 1517, RFC 1518, RFC 1519 и RFC 1520.

Суть технологии CIDR заключается в следующем. Каждому поставщику услуг Internet должен назначаться непрерывный диапазон в пространстве IP-адресов. При таком подходе адреса всех сетей каждого поставщика услуг имеют общую старшую часть – *префикс*, поэтому маршрутизация на магистралях Internet может осуществляться на основе префиксов, а не полных адресов сетей. Агрегирование адресов позволит уменьшить объем таблиц в маршрутизаторах всех уровней, а следовательно, ускорить работу маршрутизаторов и повысить пропускную способность Internet.

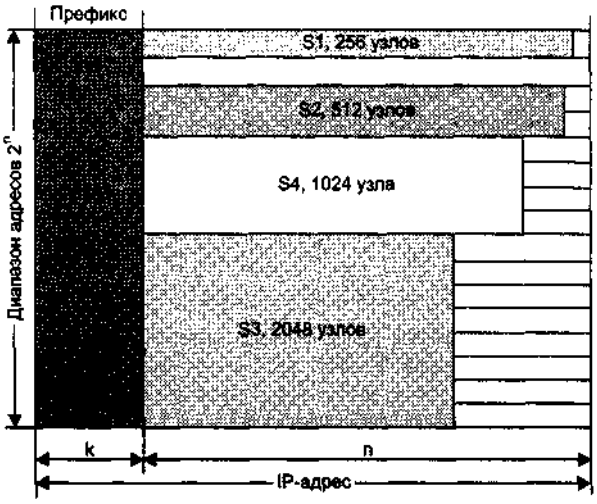


Рис. 5.19. Технологии CIDR

Деление IP-адреса на номер сети и номер узла в технологии CIDR происходит не на основе нескольких старших бит, определяющих класс сети (A, B или C), а на основе маски переменной длины, назначаемой поставщиком услуг. На рис. 5.19 показан пример некоторого пространства IP-адресов, которое имеется в распоряжении гипотетического поставщика услуг. Все адреса имеют общую часть в k старших разрядах – префикс. Оставшиеся n разрядов используются для дополнения неизменяемого префикса переменной частью адреса. Диапазон имеющихся адресов в таком случае составляет 2^n. Когда потребитель услуг обращается к поставщику услуг с просьбой о выделении ему некоторого количества адресов, то в имеющемся пуле адресов «вырезается» непрерывная область SI, S2, S3 или S4 соответствующего размера. Причем границы этой области выбираются такими, чтобы для нумерации требуемого числа узлов хватило некоторого числа младших разрядов, а значения всех оставшихся (старших) разрядов было одинаковым у всех адресов данного диапазона. Таким условиям могут удовлетворять только области, размер которых кратен степени двойки. А границы выделяемого участка должны быть кратны требуемому размеру.

Рассмотрим пример. Пусть поставщик услуг Internet располагает пулом адресов в диапазоне 193.20.0.0-193.23.255.255 (1100 0001.0001 0100.0000 0000.0000 0000-1100 0001.0001 0111.11111111.1111 1111) с общим префиксом 193.20 (1100 0001.0001 01) и маской, соответствующей этому префиксу 255.252.0.0.

Если абоненту этого поставщика услуг требуется совсем немного адресов, например 13, то поставщик мог бы предложить ему различные варианты: сеть 193.20.30.0, сеть 193.20.30.16 или сеть 193.21.204.48, все с одним и тем же значением маски 255.255.255.240. Во всех случаях в распоряжении абонента для нумерации узлов имеются 4 младших бита.

Рассмотрим другой вариант, когда к поставщику услуг обратился крупный заказчик, сам, возможно собирающийся оказывать услуги по доступу в Internet. Ему требуется блок адресов в 4000 узлов. В этом случае поставщик услуг мог бы предложить ему, например, диапазон адресов 193.22.160.0-193.22.175.255 с маской 255.255.240.0. Агрегированный номер сети (префикс) в этом случае будет равен 193.22.160.0.

Администратор маршрутизатора M2 (рис. 5.20) поместит в таблицу маршрутизации только по одной записи на каждого клиента, которому был выделен пул адресов, независимо от количества подсетей, организованных клиентом. Если клиент, получивший сеть 193.22.160.0, через некоторое время разделит ее адресное пространство в 4096 адресов на 8 подсетей, то в маршрутизаторе M2 первоначальная информация о выделенной ему сети не изменится.

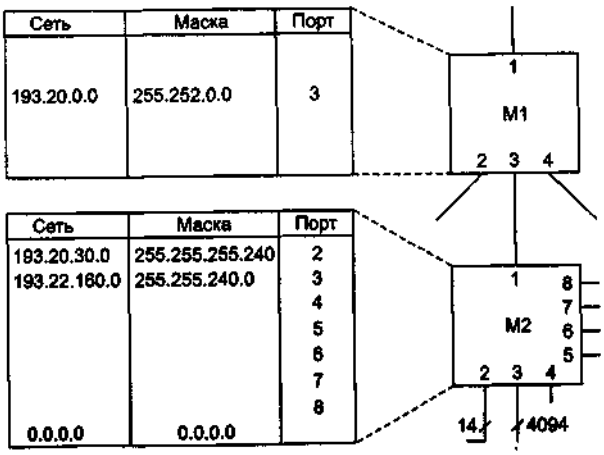


Рис. 5.20. Выигрыш в количестве записей в маршрутизаторе при использовании технологии CIDR

Для поставщика услуг верхнего уровня, поддерживающего клиентов через маршрутизатор Ml, усилия поставщика услуг нижнего уровня по разделению его адресного пространства также не будут заметны. Запись 193.20.0.0 с маской 255.252.0.0 полностью описывает сети поставщика услуг нижнего уровня в маршрутизаторе Ml.

Итак, внедрение технологии CIDR позволяет решить две основные задачи.

- Более экономное расходование адресного пространства. Действительно, получая в свое распоряжение адрес сети, например, класса С, некоторые организации не используют весь возможный диапазон адресов просто потому, что в их сети имеется гораздо меньше 255 узлов. Технология CIDR отказывается от традиционной концепции разделения адресов протокола IP на классы, что позволяет получать в пользование столько адресов, сколько реально необходимо. Благодаря технологии CIDR поставщики услуг получают возможность «нарезать» блоки из выделенного им адресного пространства в точном соответствии с требованиями каждого клиента, при этом у клиента остается пространство для маневра на случай его будущего роста.
- Уменьшение числа записей в таблицах маршрутизаторов за счет объединения маршрутов – одна запись в таблице маршрутизации может представлять большое количество сетей. Действительно, для всех сетей, номера которых начинаются с одинаковой последовательности цифр, в таблице маршрутизации может быть предусмотрена одна запись (см. рис. 5.20). Так, маршрутизатор М2 установленный в организации, которая использует технику CIDR для выделения адресов своим клиентам, должен поддерживать в своей таблице маршрутизации все 8 записей о сетях клиентов. А маршрутизатору Ml достаточно иметь одну запись о всех этих сетях, на основании которой он передает пакеты с префиксом 193.20 маршрутизатору М2, который их и распределяет по нужным портам.

Если все поставщики услуг Internet будут придерживаться стратегии CIDR, то особенно заметный выигрыш будет достигаться в магистральных маршрутизаторах.

Технология CIDR уже успешно используется в текущей версии IPv4 и поддерживается такими протоколами маршрутизации, как OSPF, RIP-2, BGP4. Предполагается, что эти же протоколы будут работать и с новой версией протокола IPv6. Следует отметить, что в настоящее время технология CIDR поддерживается магистральными маршрутизаторами Internet, а не обычными хостами в локальных сетях.

Использование CIDR в сетях IPv4 в общем случае требует перенумерации сетей. Поскольку эта процедура сопряжена с определенными временными и материальными затратами, для ее проведения пользователей нужно каким-либо образом стимулировать. В качестве таких стимулов рассматривается, например, введение оплаты за строку в таблице маршрутизации или же за количество узлов в сети. При использовании классов сетей абонент часто не полностью занимает весь допустимый диапазон адресов узлов – 254 адреса для сети класса С или 65 534 адреса для сети класса В. Часть адресов узлов обычно пропадает. Требование оплаты каждого адреса узла поможет пользователю решиться на перенумерацию, с тем чтобы получить ровно столько адресов, сколько ему нужно.

5.3.6. Фрагментация IP-пакетов

Протокол IP позволяет выполнять фрагментацию пакетов, поступающих на входные порты маршрутизаторов.

Следует различать фрагментацию сообщений в узле-отправителе и динамическую фрагментацию сообщений в транзитных узлах сети – маршрутизаторах. Практически во всех стеках протоколов есть протоколы, которые отвечают за фрагментацию сообщений прикладного уровня на такие части, которые укладываются в кадры канального уровня. В стеке TCP/IP эту задачу решает протокол TCP, который разбивает поток байтов, передаваемый ему с прикладного уровня на сообщения нужного размера (например, на 1460 байт для протокола Ethernet). Поэтому протокол IP в узле-отправителе не использует свои возможности по фрагментации пакетов.

А вот при необходимости передать пакет в следующую сеть, для которой размер пакета является слишком большим, IP-фрагментация становится необходимой. В функции уровня IP входит разбиение слишком длинного для конкретного типа составляющей сети сообщения на более короткие пакеты с созданием соответствующих служебных полей, нужных для последующей сборки фрагментов в исходное сообщение.

В большинстве типов локальных и глобальных сетей значения MTU, то есть максимальный размер поля данных, в которое должен инкапсулировать свой пакет протокол IP, значительно отличается. Сети Ethernet имеют значение MTU, равное 1500 байт, сети FDDI – 4096 байт, а сети X.25 чаще всего работают с MTU в 128 байт.

IP-пакет может быть помечен как не фрагментируемый. Любой пакет, помеченный таким образом, не может быть фрагментирован модулем IP ни при каких условиях. Если же пакет, помеченный как не фрагментируемый, не может достигнуть получателя без фрагментации, то этот пакет просто уничтожается, а узлу-отправителю посылается соответствующее ICMP-сообщение.

Протокол IP допускает возможность использования в пределах отдельной подсети ее собственных средств фрагментирования, невидимых для протокола IP. Например, технология ATM делит поступающие IP-пакеты на ячейки с полем данных в 48 байт с помощью своего уровня сегментирования, а затем собирает ячейки в исходные пакеты на выходе из сети. Но такие технологии, как ATM, являются скорее исключением, чем правилом.

Процедуры фрагментации и сборки протокола IP рассчитаны на то, чтобы пакет мог быть разбит на практически любое количество частей, которые впоследствии могли бы быть вновь собраны. Получатель фрагмента использует поле идентификации для того, чтобы не перепутать фрагменты различных пакетов. Модуль IP, отправляющий пакет, устанавливает в поле идентификации значение, которое должно быть уникальным для данной пары отправитель - получатель, а также время, в течение которого пакет может быть активным в сети.

Поле смещения фрагмента сообщает получателю положение фрагмента в исходном пакете. Смещение фрагмента и длина определяют часть исходного пакета, принесенную этим фрагментом. Флаг «more fragments» показывает появление последнего фрагмента. Модуль протокола IP, отправляющий неразбитый на фрагменты пакет, устанавливает в нуль флаг «more fragments» и смещение во фрагменте.

Эти поля дают достаточное количество информации для сборки пакета.

Чтобы разделить на фрагменты большой пакет, модуль протокола IP, установленный, например, на маршрутизаторе, создает несколько новых пакетов и копирует содержимое полей IP-заголовка из большого пакета в IP-заголовки всех новых пакетов. Данные из старого пакета делятся на соответствующее число частей, размер каждой из которых, кроме самой последней, обязательно должен быть кратным 8 байт. Размер последней части данных равен полученному остатку.

Каждая из полученных частей данных помещается в новый пакет. Когда происходит фрагментация, то некоторые параметры IP-заголовка копируются в заголовки всех фрагментов, а другие остаются лишь в заголовке первого фрагмента. Процесс фрагментации может изменить значения данных, расположенных в поле параметров, и значение контрольной суммы заголовка, изменить значение флага «more fragments» и смещение фрагмента, изменить длину IP-заголовка и общую длину пакета. В заголовок каждого пакета заносятся соответствующие значения в поле смещения «fragment offset», а в поле общей длины пакета помещается длина каждого пакета. Первый фрагмент будет иметь в поле «fragment offset» нулевое значение. Во всех пакетах, кроме последнего, флаг «more fragments»

устанавливается в единицу, а в последнем фрагменте – в нуль.

Чтобы собрать фрагменты пакета, модуль протокола IP (например, модуль на хост-компьютере) объединяет IP-пакеты, имеющие одинаковые значения в полях идентификатора, отправителя, получателя и протокола. Таким образом, отправитель должен выбрать идентификатор таким образом, чтобы он был уникален для данной пары отправитель-получатель, для данного протокола и в течение того времени, пока данный пакет (или любой его фрагмент) может существовать в составной IP-сети.

Очевидно, что модуль протокола IP, отправляющий пакеты, должен иметь таблицу идентификаторов, где каждая запись соотносится с каждым отдельным получателем, с которым осуществлялась связь, и указывает последнее значение максимального времени жизни пакета в IP-сети. Однако, поскольку поле идентификатора допускает 65 536 различных значений, некоторые хосты могут использовать просто уникальные идентификаторы, не зависящие от адреса получателя.

В некоторых случаях целесообразно, чтобы идентификаторы IP-пакетов выбирались протоколами более высокого, чем IP, уровня. Например, в протоколе TCP предусмотрена повторная передача TCP-сегментов, по каким-либо причинам не дошедшим до адресата. Вероятность правильного приема увеличивалась бы, если бы при повторной передаче идентификатор для IP-пакета был бы тем же, что и в исходном IP-пакете, поскольку его фрагменты могли бы использоваться для сборки правильного TCP-сегмента.

Процедура объединения заключается в помещении данных из каждого фрагмента в позицию, указанную в заголовке пакета в поле «fragment offset».

Каждый модуль IP должен быть способен передать пакет из 68 байт без дальнейшей фрагментации. Это связано с тем, что IP-заголовок может включать до 60 байт, а минимальный фрагмент данных – 8 байт. Каждый получатель должен быть в состоянии принять пакет из 576 байт в качестве единого куска либо в виде фрагментов, подлежащих сборке.

Если бит флага запрета фрагментации (Don't Fragment, DF) установлен, то фрагментация данного пакета запрещена, даже если в этом случае он будет потерян. Данное средство может использоваться для предотвращения фрагментации в тех случаях, когда хост-получатель не имеет достаточных ресурсов для сборки фрагментов.

Работа протокола IP по фрагментации пакетов в хостах и маршрутизаторах иллюстрируется на рис. 5.21.

Пусть компьютер 1 связан с сетью, имеющей значение MTU в 4096 байт, например с сетью FDDI. При поступлении на IP-уровень компьютера 1 сообщения от транспортного уровня размером в 5600 байт протокол IP делит его на два IP-пакета, устанавливая в первом пакете признак фрагментации и присваивая пакету уникальный идентификатор, например 486. В первом пакете величина поля смещения равна 0, а во втором – 2800. Признак фрагментации во втором пакете равен нулю, что показывает, что это последний фрагмент пакета. Общая величина IP-пакета составляет 2800 плюс 20 (размер IP-заголовка), то есть 2820 байт, что умещается в поле данных кадра FDDI. Далее модуль IP компьютера 1 передает эти пакеты своему сетевому интерфейсу (образуемому протоколами канального уровня К1 и физического уровня Ф1). Сетевой интерфейс отправляет кадры следующему маршрутизатору.

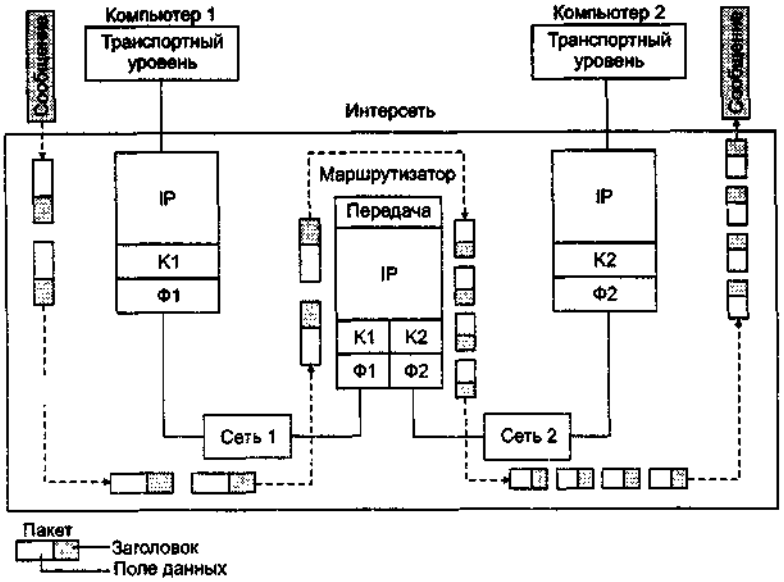


Рис. 5.21. Фрагментация IP-пакетов при передаче между сетями с разным максимальным размером пакетов: К1 и Ф1 — канальный и физический уровень сети 1; К2 и Ф2 — канальный и физический уровень сети 2

После того, как кадры пройдут уровень сетевого интерфейса маршрутизатора (К1 и Ф1) и освободятся от заголовков FDDI, модуль IP по сетевому адресу определяет, что прибывшие два пакета нужно передать в сеть 2, которая является сетью Ethernet и имеет значение MTU, равное 1500. Следовательно, прибывшие IP-пакеты необходимо фрагментировать. Маршрутизатор извлекает поле данных из каждого пакета и делит его еще пополам, чтобы каждая часть уместилась в поле данных кадра Ethernet. Затем он формирует новые IP-пакеты, каждый из которых имеет длину 1400 + 20 = 1420 байт, что меньше 1500 байт, поэтому они нормально помещаются в поле данных кадров Ethernet.

В результате в компьютер 2 по сети Ethernet приходят четыре IP-пакета с общим идентификатором 486, что позволяет протоколу IP, работающему в компьютере 2, правильно собрать исходное сообщение. Если пакеты пришли не в том порядке, в котором были посланы, то смещение укажет правильный порядок их объединения.

Отметим, что IP-маршрутизаторы не собирают фрагменты пакетов в более крупные пакеты, даже если на пути встречается сеть, допускающая такое укрупнение. Это связано с тем, что отдельные фрагменты сообщения могут перемещаться по интерсети по различным маршрутам, поэтому нет гарантии, что все фрагменты проходят через какой-либо промежуточный маршрутизатор на их пути.

При приходе первого фрагмента пакета узел назначения запускает таймер, который определяет максимально допустимое время ожидания прихода остальных фрагментов этого пакета. Таймер устанавливается на максимальное из двух значений: первоначальное установочное время ожидания и время жизни, указанное в принятом фрагменте. Таким образом, первоначальная установка таймера является нижней границей для времени ожидания при сборке. Если таймер истекает раньше прибытия последнего фрагмента, то все ресурсы сборки, связанные с данным пакетом, освобождаются, все полученные к этому моменту фрагменты пакета отбрасываются, а в узел, пославший исходный пакет, направляется сообщение об ошибке с помощью протокола ICMP.

5.3.7. Протокол надежной доставки TCP-сообщений

Протокол IP является дейтаграммным протоколом и поэтому по своей природе не может гарантировать надежность передачи данных. Эту задачу – обеспечение надежного канала обмена данными между прикладными процессами в составной сети – решает протокол TCP (Transmission Control Protocol), относящийся к транспортному уровню.

Протокол TCP работает непосредственно над протоколом IP и использует для транспортировки своих блоков данных потенциально ненадежный протокол IP. Надежность передачи данных протоколом TCP достигается за счет того, что он основан на установлении логических соединений между взаимодействующими процессами. До тех пор пока программы протокола TCP продолжают функционировать корректно, а составная сеть не распалась на несвязные части, ошибки в передаче данных на уровне протокола IP не будут влиять на правильное получение данных.

Протокол IP используется протоколом TCP в качестве транспортного средства. Перед отправкой своих блоков данных протокол TCP помещает их в оболочку IP-пакета. При необходимости протокол IP осуществляет любую фрагментацию и сборку блоков данных TCP, требующуюся для осуществления передачи и доставки через множество сетей и промежуточных шлюзов.

На рис. 5.22 показано, как процессы, выполняющиеся на двух конечных узлах, устанавливают с помощью протокола TCP надежную связь через составную сеть, все узлы которой используют для передачи сообщений дейтаграммный протокол IP.

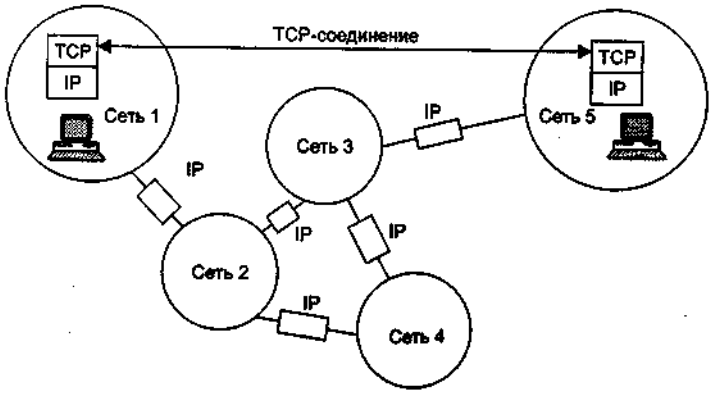


Рис. 5.22. TCP-соединение создает надежный канал связи между конечными узлами

Порты

Протокол TCP взаимодействует через межуровневые интерфейсы с ниже лежащим протоколом IP и с выше лежащими протоколами прикладного уровня или приложениями.

В то время как задачей сетевого уровня, к которому относится протокол IP, является передача данных между произвольными узлами сети, задача транспортного уровня, которую решает протокол TCP, заключается в передаче данных между любыми *прикладными процессами*, выполняющимися на любых узлах сети. Действительно, после того как пакет средствами протокола IP доставлен в компьютер-получатель, данные необходимо направить конкретному процессу-получателю. Каждый компьютер может выполнять несколько процессов, более того, прикладной процесс тоже может иметь несколько точек входа, выступающих в качестве адреса назначения для пакетов данных.

Пакеты, поступающие на транспортный уровень, организуются операционной системой в виде множества очередей к точкам входа различных прикладных процессов. В терминологии TCP/IP такие системные очереди называются *портами*. Таким образом, адресом назначения, который используется протоколом TCP, является идентификатор (номер) порта прикладной службы. Номер порта в совокупности с номером сети и номером конечного узла однозначно определяют прикладной процесс в сети. Этот набор идентифицирующих параметров имеет название *сокет* (*socket*).

Назначение номеров портов прикладным процессам осуществляется либо *централизованно*, если эти процессы представляют собой популярные общедоступные службы (например, номер 21 закреплен за службой удаленного доступа к файлам FTP, а 23 – за службой удаленного управления telnet), либо локально для тех служб, которые еще не стали столь распространенными, чтобы закреплять за ними стандартные (зарезервированные) номера. Централизованное присвоение службам номеров портов выполняется организацией *Internet Assigned Numbers Authority* (*IANA*). Эти номера затем закрепляются и опубликовываются в стандартах Internet (RFC 1700).

Локальное присвоение номера порта заключается в том, что разработчик некоторого приложения просто связывает с ним любой доступный, произвольно выбранный числовой идентификатор, обращая внимание на то, чтобы он не входил в число зарезервированных номеров портов. В дальнейшем все удаленные запросы к данному приложению от других приложений должны адресоваться с указанием назначенного ему номера порта.

Протокол TCP ведет для каждого порта две очереди: очередь пакетов, поступающих в данный порт из сети, и очередь пакетов, отправляемых данным портом в сеть. Процедура обслуживания протоколом TCP запросов, поступающих от нескольких различных прикладных служб, называется *мультиплексированием*. Обратная процедура распределения протоколом TCP поступающих от сетевого

уровня пакетов между набором высокоуровневых служб, идентифицированных номерами портов, называется *демультиплексированием* (рис. 5.23).

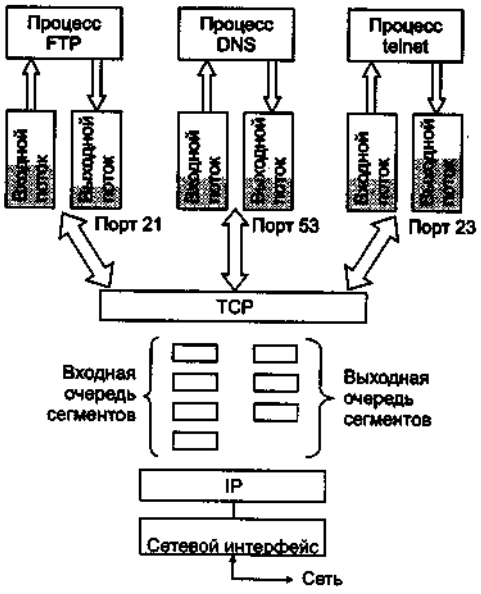


Рис. 5.23. Функции протокола TCP по мультиплексированию и демультиплексированию потоков

Сегменты и потоки

Единицей данных протокола TCP является *сегмент*. Информация, поступающая к протоколу TCP в рамках логического соединения от протоколов более высокого уровня, рассматривается протоколом TCP как неструктурированный *поток* байтов. Поступающие данные буферизуются средствами TCP. Для передачи на сетевой уровень из буфера «вырезается» некоторая непрерывная часть данных, которая и называется сегментом (см. рис. 5.23). В отличие от многих других протоколов, протокол TCP подтверждает получение не пакетов, а байтов потока.

Не все сегменты, посланные через соединение, будут одного и того же размера, однако оба участника соединения должны договориться о максимальном размере сегмента, который они будут использовать. Этот размер выбирается таким образом, чтобы при упаковке сегмента в IP-пакет он помещался туда целиком, то есть максимальный размер сегмента не должен превосходить максимального размера поля данных IP-пакета. В противном случае пришлось бы выполнять фрагментацию, то есть делить сегмент на несколько частей, чтобы разместить его в IP-пакете.

Соединения

Для организации надежной передачи данных предусматривается установление *логического соединения* между двумя прикладными процессами. Поскольку соединения устанавливаются через ненадежную коммуникационную систему, основанную на протоколе IP, то во избежание ошибочной инициализации соединений используется специальная многошаговая процедура подтверждения связи.

Соединение в протоколе TCP идентифицируется парой полных адресов обоих взаимодействующих процессов – сокетов. Каждый из взаимодействующих процессов может участвовать в нескольких соединениях.

Формально соединение можно определить как набор параметров, характеризующий процедуру обмена данными между двумя процессами. Помимо полных адресов процессов этот набор включает и параметры, значения которых определяются в результате переговорного процесса модулей TCP двух сторон соединения. К таким параметрам относятся, в частности, согласованные размеры сегментов, которые может посылать каждая из сторон, объемы данных, которые разрешено передавать без получения на них подтверждения, начальные и текущие номера передаваемых байтов. Некоторые из этих параметров остаются постоянными в течение всего сеанса связи, а некоторые адаптивно изменяются.

В рамках соединения осуществляется обязательное подтверждение правильности приема для всех

переданных сообщений и при необходимости выполняется повторная передача. Соединение в TCP позволяет вести передачу данных одновременно в обе стороны, то есть полнодуплексную передачу.

Реализация скользящего окна в протоколе TCP

В рамках установленного соединения правильность передачи каждого сегмента должна подтверждаться квитанцией получателя. *Квитирование* – это один из традиционных методов обеспечения надежной связи. В протоколе TCP используется частный случай квитирования – алгоритм скользящего окна. Идея этого алгоритма была изложена в главе 2, «Основы передачи дискретных данных».

Особенность использования алгоритма скользящего окна в протоколе TCP состоит в том, что, хотя единицей передаваемых данных является сегмент, окно определено на множестве нумерованных байтов неструктурированного потока данных, поступающих с верхнего уровня и буферизуемых протоколом TCP. Получающий модуль TCP отправляет «окно» посылающему модулю TCP. Данное окно задает количество байтов (начиная с номера байта, о котором уже была выслана квитанция), которое принимающий модуль TCP готов в настоящий момент принять.

Квитанция (подтверждение) посылается только в случае правильного приема данных, отрицательные квитанции не посылаются. Таким образом, отсутствие квитанции означает либо прием искаженного сегмента, либо потерю сегмента, либо потерю квитанции. В качестве квитанции получатель сегмента отсылает ответное сообщение (сегмент), в которое помещает число, на единицу превышающее максимальный номер байта в полученном сегменте. Это число часто называют *номером очереди*.

На рис. 5.24 показан поток байтов, поступающий на вход протокола TCP. Из потока байтов модуль TCP нарезает последовательность сегментов. Для определенности на рисунке принято направление перемещения данных справа налево. В этом потоке можно указать несколько логических границ. Первая граница отделяет сегменты, которые уже были отправлены и на которые уже пришли квитанции. Следующую часть потока составляют сегменты, которые также уже отправлены, так как входят в границы, определенные окном, но квитанции на них пока не получены. Третья часть потока – это сегменты, которые пока не отправлены, но могут быть отправлены, так как входят в пределы окна. И наконец, последняя граница указывает на начало последовательности сегментов, ни один из которых не может быть отправлен до тех пор, пока не придет очередная квитанция и окно не будет сдвинуто вправо.

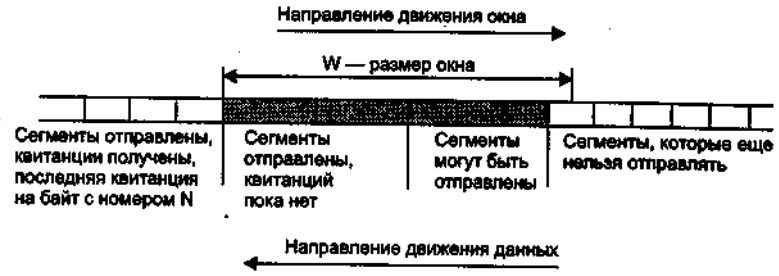


Рис. 5.24. Особенности реализации алгоритма скользящего окна в протоколе TCP

Если размер окна равен W, а последняя по времени квитанция содержала значение N, то отправитель может посылать новые сегменты до тех пор, пока в очередной сегмент не попадет байт с номером N+W. Этот сегмент выходит за рамки окна, и передачу в таком случае необходимо приостановить до прихода следующей квитанции.

Надежность передачи достигается благодаря подтверждениям и номерам очереди. Концептуально каждому байту данных присваивается номер очереди. Номер очереди для первого байта данных в сегменте передается вместе с этим сегментом и называется номером очереди для сегмента. Сегменты также несут номер подтверждения, который является номером для следующего ожидаемого байта данных, передаваемого в обратном направлении. Когда протокол TCP передает сегмент с данными, он помещает его копию в очередь повторной передачи и запускает таймер. Когда приходит подтверждение для этих данных, соответствующий сегмент удаляется из очереди. Если подтверждение не приходит до истечения срока, то сегмент посылается повторно.

Выбор времени ожидания (тайм-аута) очередной квитанции является важной задачей, результат

решения которой влияет на производительность протокола TCP. Тайм-аут не должен быть слишком коротким, чтобы по возможности исключить избыточные повторные передачи, которые снижают полезную пропускную способность системы. Но он не должен быть и слишком большим, чтобы избежать длительных простоев, связанных с ожиданием несуществующей или «заблудившейся» квитанции.

При выборе величины тайм-аута должны учитываться скорость и надежность физических линий связи, их протяженность и многие другие подобные факторы. В протоколе TCP тайм-аут определяется с помощью достаточно сложного адаптивного алгоритма, идея которого состоит в следующем. При каждой передаче засекается время от момента отправки сегмента до прихода квитанции о его приеме (время оборота). Получаемые значения времени оборота усредняются с весовыми коэффициентами, возрастающими от предыдущего замера к последующему. Это делается с тем, чтобы усилить влияние последних замеров. В качестве тайм-аута выбирается среднее время оборота, умноженное на некоторый коэффициент. Практика показывает, что значение этого коэффициента должно превышать 2. В сетях с большим разбросом времени оборота при выборе тайм-аута учитывается и дисперсия этой величины.

Поскольку каждый байт пронумерован, то каждый из них может быть опознан. Приемлемый механизм опознавания является накопительным, поэтому опознавание номера X означает, что все байты с предыдущими номерами уже получены. Этот механизм позволяет регистрировать появление дубликатов в условиях повторной передачи. Нумерация байтов в пределах сегмента осуществляется так, чтобы первый байт данных сразу вслед за заголовком имел наименьший номер, а следующие за ним байты имели номера по возрастающей.

Окно, посылаемое с каждым сегментом, определяет диапазон номеров очереди, которые отправитель окна (он же получатель данных) готов принять в настоящее время. Предполагается, что такой механизм связан с наличием в данный момент места в буфере данных.

Варьируя величину окна, можно влиять на загрузку сети. Чем больше окно, тем большую порцию неподтвержденных данных можно послать в сеть. Но если пришло большее количество данных, чем может быть принято программой TCP, данные будут отброшены. Это приведет к излишним пересылкам информации и ненужному увеличению нагрузки на сеть и программу TCP.

С другой стороны, указание окна малого размера может ограничить передачу данных скоростью, которая определяется временем путешествия по сети каждого посылаемого сегмента. Чтобы избежать применения малых окон, получателю данных предлагается откладывать изменение окна до тех пор, пока свободное место не составит 20-40 % от максимально возможного объема памяти для этого соединения. Но и отправителю не стоит спешить с посылкой данных, пока окно не станет достаточно большим. Учитывая эти соображения, разработчики протокола TCP предложили схему, согласно которой при установлении соединения заявляется большое окно, но впоследствии его размер существенно уменьшается.

Если сеть не справляется с нагрузкой, то возникают очереди в промежуточных узлах-маршрутизаторах и в конечных узлах-компьютерах.

При переполнении приемного буфера конечного узла «перегруженный» протокол TCP, отправляя квитанцию, помещает в нее новый, уменьшенный размер окна. Если он совсем отказывается от приема, то в квитанции указывается окно нулевого размера. Однако даже после этого приложение может послать сообщение на отказавшийся от приема порт. Для этого сообщение должно сопровождаться пометкой «срочно». В такой ситуации порт обязан принять сегмент, даже если для этого придется вытеснить из буфера уже находящиеся там данные. После приема квитанции с нулевым значением окна протокол-отправитель время от времени делает контрольные попытки продолжить обмен данными. Если протокол-приемник уже готов принимать информацию, то в ответ на контрольный запрос он посылает квитанцию с указанием ненулевого размера окна.

Другим проявлением перегрузки сети является переполнение буферов в маршрутизаторах. В таких случаях они могут централизованно изменить размер окна, посылая управляющие сообщения некоторым конечным узлам, что позволяет им дифференцированно управлять интенсивностью потока данных в разных частях сети.

Выводы

- Протокол IP решает задачу доставки сообщений между узлами составной сети. Протокол IP относится к протоколам без установления соединений, поэтому он не дает никаких гарантий

надежной доставки сообщений. Все вопросы обеспечения надежности доставки данных в составной сети в стеке TCP/IP решает протокол TCP, основанный на установлении логических соединений между взаимодействующими процессами.

- IP-пакет состоит из заголовка и поля данных. Максимальная длина пакета 65 535 байт. Заголовок обычно имеет длину 20 байт и содержит информацию о сетевых адресах отправителя и получателя, о параметрах фрагментации, о времени жизни пакета, о контрольной сумме и некоторых других. В поле данных IP-пакета находятся сообщения более высокого уровня, например TCP или UDP.
- Вид таблицы IP-маршрутизации зависит от конкретной реализации маршрутизатора, но, несмотря на достаточно сильные внешние различия, в таблицах всех типов маршрутизаторов есть все ключевые поля, необходимые для выполнения маршрутизации.
- Существует несколько источников, поставляющих записи в таблицу маршрутизации. Во-первых, при инициализации программное обеспечение стека TCP/ IP заносит в таблицу записи о непосредственно подключенных сетях и маршрутизаторах по умолчанию, а также записи об особых адресах типа 127.0.0.0. Во-вторых, администратор вручную заносит статические записи о специфичных маршрутах или о маршрутизаторе по умолчанию. В-третьих, протоколы маршрутизации автоматически заносят в таблицу динамические записи о имеющихся маршрутах.
- Эффективным средством структуризации IP-сетей являются маски. Маски позволяют разделить одну сеть на несколько подсетей. Маски одинаковой длины используются для деления сети на подсети равного размера, а маски переменной длины – для деления сети на подсети разного размера. Использование масок модифицирует алгоритм маршрутизации, поэтому в этом случае предъявляются особые требования к протоколам маршрутизации в сети, к техническим характеристикам маршрутизаторов и процедурам их конфигурирования.
- Значительная роль в будущем IP-сетей отводится технологии бесклассовой междоменной маршрутизации (CIDR), которая решает две основные задачи. Первая состоит в более экономном расходование адресного пространства – благодаря CIDR поставщики услуг получают возможность «нарезать» блоки разных размеров из выделенного им адресного пространства в точном соответствии с требованиями каждого клиента. Вторая задача заключается в уменьшении числа записей в таблицах маршрутизации за счет объединения маршрутов – одна запись в таблице маршрутизации может представлять большое количество сетей с общим префиксом.
- Важной особенностью протокола IP, отличающей его от других сетевых протоколов, является его способность выполнять динамическую фрагментацию пакетов при передаче их между сетями с различными MTU. Это свойство во многом способствовало тому, что протокол IP смог занять доминирующие позиции в сложных составных сетях.

5.4. Протоколы маршрутизации в IP-сетях

5.4.1. Внутренние и внешние протоколы маршрутизации Internet

Большинство протоколов маршрутизации, применяемых в современных сетях с коммутацией пакетов, ведут свое происхождение от сети Internet и ее предшественницы – сети ARPANET. Для того чтобы понять их назначение и особенности, полезно сначала познакомиться со структурой сети Internet, которая наложила отпечаток на терминологию и типы протоколов.

Internet изначально строилась как сеть, объединяющая большое количество существующих систем. С самого начала в ее структуре выделяли *магистральную сеть* (*core backbone network*), а сети, присоединенные к магистрали, рассматривались как *автономные системы* (*autonomous systems, AS*). Магистральная сеть и каждая из автономных систем имели свое собственное административное управление и собственные протоколы маршрутизации. Необходимо подчеркнуть, что автономная система и домен имен Internet – это разные понятия, которые служат разным целям. Автономная система объединяет сети, в которых под общим административным руководством одной организации осуществляется маршрутизация, а домен объединяет компьютеры (возможно, принадлежащие разным сетям), в которых под общим административным руководством одной организации осуществляется

назначение уникальных символьных имен. Естественно, области действия автономной системы и домена имен могут в частном случае совпадать, если одна организация выполняет обе указанные функции.

Общая схема архитектуры сети Internet показана на рис. 5.25. Далее маршрутизаторы мы будем называть шлюзами, чтобы оставаться в русле традиционной терминологии Internet.

Шлюзы, которые используются для образования сетей и подсетей внутри автономной системы, называются *внутренними шлюзами* (*interior gateways*), а шлюзы, с помощью которых автономные системы присоединяются к магистрали сети, называются *внешними шлюзами* (*exterior gateways*). Магистраль сети также является автономной системой. Все автономные системы имеют уникальный 16-разрядный номер, который выделяется организацией, учредившей новую автономную систему, InterNIC.

Соответственно протоколы маршрутизации внутри автономных систем называются *протоколами внутренних шлюзов* (*interior gateway protocol, IGP*), а протоколы, определяющие обмен маршрутной информацией между внешними шлюзами и шлюзами магистральной сети – *протоколами внешних шлюзов* (*exterior gateway protocol, EGP*). Внутри магистральной сети также допустим любой собственный внутренний протокол IGP.

Смысл разделения всей сети Internet на автономные системы – в ее многоуровневом модульном представлении, что необходимо для любой крупной системы, способной к расширению в больших масштабах. Изменение протоколов маршрутизации внутри какой-либо автономной системы никак не должно влиять на работу остальных автономных систем. Кроме того, деление Internet на автономные системы должно способствовать агрегированию информации в магистральных и внешних шлюзах. Внутренние шлюзы могут использовать для внутренней маршрутизации достаточно подробные графы связей между собой, чтобы выбрать наиболее рациональный маршрут. Однако если информация такой степени детализации будет храниться во всех маршрутизаторах сети, то топологические базы данных так разрастутся, что потребуют наличия памяти гигантских размеров, а время принятия решений о маршрутизации станет неприемлемо большим.

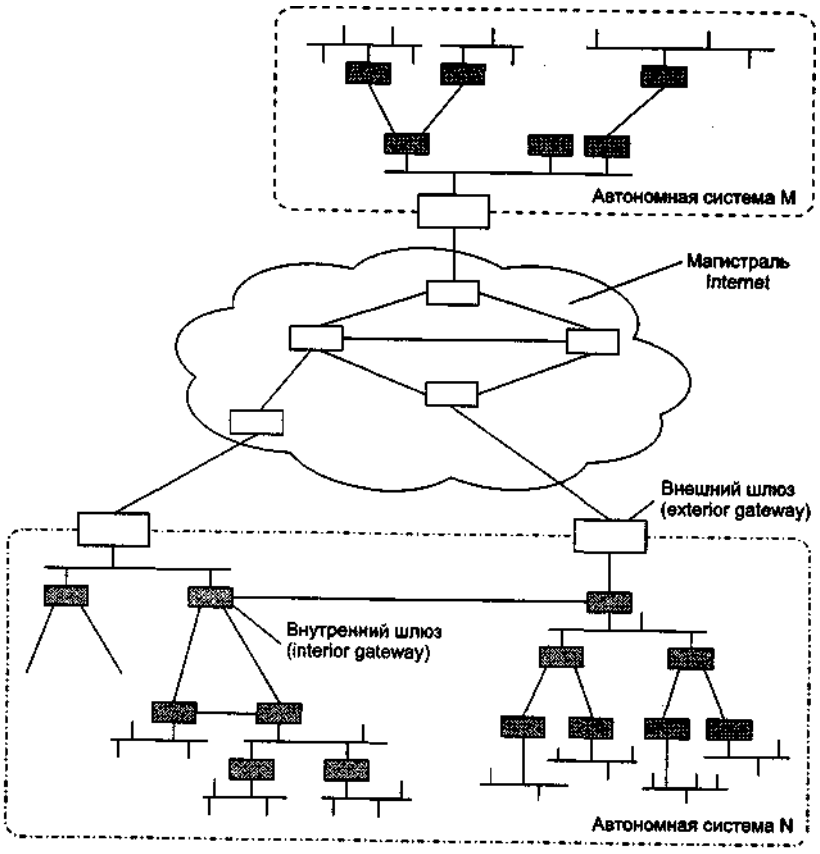


Рис. 5.25. Магистраль и автономные системы Internet

Поэтому детальная топологическая информация остается внутри автономной системы, а автономную систему как единое целое для остальной части Internet представляют внешние шлюзы, которые

сообщают о внутреннем составе автономной системы минимально необходимые сведения - количество IP-сетей, их адреса и внутреннее расстояние до этих сетей от данного внешнего шлюза.

Техника бесклассовой маршрутизации CIDR может значительно сократить объемы маршрутной информации, передаваемой между автономными системами. Так, если все сети внутри некоторой автономной системы начинаются с общего префикса, например 194.27.0.0/16, то внешний шлюз этой автономной системы должен делать объявления только об этом адресе, не сообщая отдельно о существовании внутри данной автономной системы, например, сети 194.27.32.0/19 или 194.27.40.O/21, так как эти адреса агрегируются в адрес 194.27.0.0/16.

Приведенная на рис. 5.25 структура Internet с единственной магистралью достаточно долго соответствовала действительности, поэтому специально для нее был разработан протокол обмена маршрутной информации между автономными системами, названный EGP. Однако по мере развития сетей поставщиков услуг структура Internet стала гораздо более сложной, с произвольным характером связей между автономными системами. Поэтому протокол EGP уступил место протоколу BGP, который позволяет распознать наличие петель между автономными системами и исключить их из межсистемных маршрутов. Протоколы EGP и BGP используются только во внешних шлюзах автономных систем, которые чаще всего организуются поставщиками услуг Internet. В маршрутизаторах корпоративных сетей работают внутренние протоколы маршрутизации, такие как RIP и OSPF.

5.4.2. Дистанционно-векторный протокол RIP

Построение таблицы маршрутизации

Протокол RIP (Routing Information Protocol) является внутренним протоколом маршрутизации дистанционно-векторного типа, он представляет собой один из наиболее ранних протоколов обмена маршрутной информацией и до сих пор чрезвычайно распространен в вычислительных сетях ввиду простоты реализации. Кроме версии RIP для сетей TCP/IP существует также версия RIP для сетей IPX/SPX компании Novell.

Для IP имеются две версии протокола RIP: первая и вторая. Протокол RIPvl не поддерживает масок, то есть он распространяет между маршрутизаторами только информацию о номерах сетей и расстояниях до них, а информацию о масках этих сетей не распространяет, считая, что все адреса принадлежат к стандартными классам A, B или C. Протокол RIPv2 передает информацию о масках сетей, поэтому он в большей степени соответствует требованиям сегодняшнего дня. Так как при построении таблиц маршрутизации работа версии 2 принципиально не отличается от версии 1, то в дальнейшем для упрощения записей будет описываться работа первой версии.

В качестве расстояния до сети стандарты протокола RIP допускают различные виды метрик: хопы, метрики, учитывающие пропускную способность, вносимые задержки и надежность сетей (то есть соответствующие признакам D, T и R в поле «Качество сервиса» IP-пакета), а также любые комбинации этих метрик. Метрика должна обладать свойством аддитивности – метрика составного пути должна быть равна сумме метрик составляющих этого пути. В большинстве реализаций RIP используется простейшая метрика – количество хопов, то есть количество промежуточных маршрутизаторов, которые нужно преодолеть пакету до сети назначения.

Рассмотрим процесс построения таблицы маршрутизации с помощью протокола RIP на примере составной еети, изображенной на рис. 5.26.

Этап1 - создание минимальных таблиц

В этой сети имеется восемь IP-сетей, связанных четырьмя маршрутизаторами с идентификаторами: Ml, M2, M3 и M4. Маршрутизаторы, работающие по протоколу RIP, могут иметь идентификаторы, однако для работы протокола они не являются необходимыми. В RIP-сообщениях эти идентификаторы не передаются.

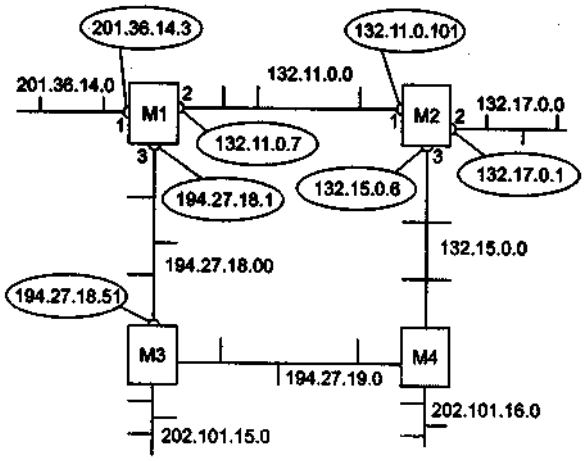


Рис. 5.26. Сеть, объединенная RIP-маршрутизаторами

В исходном состоянии в каждом маршрутизаторе программным обеспечением стека TCP/IP автоматически создается минимальная таблица маршрутизации, в которой учитываются только непосредственно подсоединенные сети. На рисунке адреса портов маршрутизаторов в отличие от адресов сетей помещены в овалы.

Таблица 5.14 позволяет оценить примерный вид минимальной таблицы маршрутизации маршрутизатора Ml.

Таблица 5.14. Минимальная таблица маршрутизации маршрутизатора Ml

Номер сети	Адрес следующего маршрутизатора	Порт	Расстояние
201.36.14.0	201.36.14.3	1	1
132.11.0.0	132.11.0.7	2	1
194.27.18.0	194.27.18.1	3	1

Минимальные таблицы маршрутизации в других маршрутизаторах будут выглядеть соответственно, например, таблица маршрутизатора М2 будет состоять из трех записей (табл. 5.15).

Таблица 5.15. Минимальная таблица маршрутизации маршрутизатора М2

Номер сети	Адрес следующего маршрутизатора	Порт	Расстояние
132.11.0.0	132.11.0.101	1	1
132.17.0.0	132.17.0.1	2	1
132.15.0.0	132.15.0.6	3	1

Этап 2 – рассылка минимальных таблиц соседям

После инициализации каждого маршрутизатора он начинает посылать своим соседям сообщения протокола RIP, в которых содержится его минимальная таблица.

RIP-сообщения передаются в пакетах протокола UDP и включают два параметра для каждой сети: ее IP-адрес и расстояние до нее от передающего сообщение маршрутизатора.

Соседями являются те маршрутизаторы, которым данный маршрутизатор непосредственно может передать IP-пакет по какой-либо своей сети, не пользуясь услугами промежуточных маршрутизаторов. Например, для маршрутизатора Ml соседями являются маршрутизаторы М2 и М3, а для маршрутизатора М4 – маршрутизаторы М2 и М3.

Таким образом, маршрутизатор Ml передает маршрутизатору М2 и М3 следующее сообщение:

сеть 201.36.14.0, расстояние 1;

сеть 132.11.0.0, расстояние 1;

сеть 194.27.18.0, расстояние 1.

Этап 3 – получение RIP-сообщений от соседей и обработка полученной информации

После получения аналогичных сообщений от маршрутизаторов М2 и М3 маршрутизатор Ml наращивает каждое полученное поле метрики на единицу и запоминает, через какой порт и от какого маршрутизатора получена новая информация (адрес этого маршрутизатора будет адресом следующего маршрутизатора, если эта запись будет внесена в таблицу маршрутизации). Затем маршрутизатор начинает сравнивать новую информацию с той, которая хранится в его таблице маршрутизации (табл. 5.16).

Таблица 5.16. Таблица маршрутизации маршрутизатора Ml

Номер сети	Адрес следующего маршрутизатора	Порт	Расстояние
201.36.14.0	201.36.14.3	1	1
132.11.0.0	132.11.0.7	2	1
194.27.18.0	194.27.18.1	3	1
132.17.0.0	132.11.0.101	2	2
132.15.0.0	132.11.0.101	2	2
194.27.19.0	194.27.18.51	3	2
202.101.15.0	194.27.18.51	3	2
~~132.11.0.0~~	~~132.11.0.101~~	~~2~~	~~2~~
~~194.27.10.0~~	~~194.27.10.51~~	~~3~~	~~2~~

Записи с четвертой по девятую получены от соседних маршрутизаторов, и они претендуют на помещение в таблицу. Однако только записи с четвертой по седьмую попадают в таблицу, а записи восьмая и девятая – нет. Это происходит потому, что они содержат данные об уже имеющихся в таблице Ml сетях, а расстояние до них хуже, чем в существующих записях.

Протокол RIP замещает запись о какой-либо сети только в том случае, если новая информация имеет лучшую метрику (расстояние в хопах меньше), чем имеющаяся. В результате в таблице маршрутизации о каждой сети остается только одна запись; если же имеется несколько равнозначных в отношении расстояния путей к одной и той же сети, то всё равно в таблице остается одна запись, которая пришла в маршрутизатор первая по времени. Для этого правила существует исключение – если худшая информация о какой-либо сети пришла от того же маршрутизатора, на основании сообщения которого была создана данная запись, то худшая информация замещает лучшую.

Аналогичные операции с новой информацией выполняют и остальные маршрутизаторы сети.

Этап 4 – рассылка новой, уже не минимальной, таблицы соседям

Каждый маршрутизатор отсылает новое RIP-сообщение всем своим соседям. В этом сообщении он помещает данные о всех известных ему сетях – как непосредственно подключенных, так и удаленных, о которых маршрутизатор узнал из RIP-сообщений.

Этап 5 – получение RIP-сообщений от соседей и обработка полученной информации

Этап 5 повторяет этап 3 – маршрутизаторы принимают RIP-сообщения, обрабатывают содержащуюся в них информацию и на ее основании корректируют свои таблицы маршрутизации.

Посмотрим, как это делает маршрутизатор Ml (табл. 5.17).

Таблица 5.17. Таблица маршрутизации маршрутизатора Ml

Номер сети	Адрес следующего маршрутизатора	Порт	Расстояние

201.36.14.0	201.36.14.3	1	1
132.11.0.0	132.11.0.7	2	1
194.27.18.0	194.27.18.1	3	1
132.17.0.0	132.11.0.101	2	2
132.15.0.0	132.11.0.101	2	2
~~132.15.0.0~~	~~194.27.18.51~~	~~3~~	~~3~~
194.27.19.0	194.27.18.51	3	2
~~194.27.19.0~~	~~132.11.0.101~~	~~2~~	~~3~~
~~202.101.15.0~~	~~194.27.18.51~~	~~3~~	~~2~~
202.101.16.0	132.11.0.101	2	3
~~202.101.16.0~~	~~194.27.18.51~~	~~3~~	~~3~~

На этом этапе маршрутизатор Ml получил от маршрутизатора М3 информацию о сети 132.15.0.0, которую тот в свою очередь на предыдущем цикле работы подучил от маршрутизатора М4. Маршрутизатор уже знает о сети 132.15.0.0, причем старая информация имеет лучшую метрику, чем новая, поэтому новая информация об этой сети отбрасывается.

О сети 202.101.16.0 маршрутизатор Ml узнает на этом этапе впервые, причем данные о ней приходят от двух соседей – от М3 и М4. Поскольку метрики в этих сообщениях указаны одинаковые, то в таблицу попадают данные, которые пришли первыми. В нашем примере считается, что маршрутизатор М2 опередил маршрутизатор М3 и первым переслал свое RIP-сообщение маршрутизатору Ml.

Если маршрутизаторы периодически повторяют этапы рассылки и обработки RIP-сообщений, та за конечное время в сети установится корректный режим маршрутизации. Под корректным режимом маршрутизации здесь понимается такое состояние таблиц маршрутизации, когда все сети будут достижимы из любой сети с помощью некоторого рационального маршрута. Пакеты будут доходить до адресатов и не зацикливаться в петлях, подобных той, которая образуется на рис. 5.26, маршрутизаторами M1-M2-M3-M4.

Очевидно, если в сети все маршрутизаторы, их интерфейсы и соединяющие их каналы связи постоянно работоспособны, то объявления по протоколу RIP можно делать достаточно редко, например, один раз в день. Однако в сетях постоянно происходят изменения - изменяется как работоспособность маршрутизаторов и каналов, так и сами маршрутизаторы и каналы могут добавляться в существующую сеть или же выводиться из ее состава.

Для адаптации к изменениям в сети протокол RIP использует ряд механизмов.

Адаптация RIP-маршрутизаторов к изменениям состояния сети

К новым маршрутам RIP-маршрутизаторы приспосабливаются просто - они передают новую информацию в очередном сообщении своим соседям и постепенно эта информация становится известна всем маршрутизаторам сети. А вот к отрицательным изменениям, связанным с потерей какого-либо маршрута, RIP-маршрутизаторы приспосабливаются сложнее. Это связано с тем, что в формате сообщений протокола RIP нет поля, которое бы указывало на то, что путь к данной сети больше не существует.

Вместо этого используются два механизма уведомления о том, что некоторый маршрут более недействителен:

- истечение времени жизни маршрута;
- указание специального расстояния (бесконечности) до сети, ставшей недоступной.

Для отработки первого механизма каждая запись таблицы маршрутизации (как и записи таблицы продвижения моста/коммутатора), полученная по протоколу RIP, имеет время жизни (TTL). При поступлении очередного RIP-сообщения, которое подтверждает справедливость данной записи, таймер TTL устанавливается в исходное состояние, а затем из него каждую секунду вычитается единица. Если за время тайм-аута не придет новое маршрутное сообщение об этом маршруте, то он помечается как недействительный.

Время тайм-аута связано с периодом рассылки векторов по сети. В RIP IP период рассылки выбран равным 30 секундам, а в качестве тайм-аута выбрано шестикратное значение периода рассылки, то есть 180 секунд. Выбор достаточно малого времени периода рассылки объясняется несколькими причинами, которые станут понятны из дальнейшего изложения. Шестикратный запас времени нужен для

уверенности в том, что сеть действительно стала недоступна, а не просто произошли потери RIP-сообщений (а это возможно, так как RIP использует транспортный протокол UDP, который не обеспечивает надежной доставки сообщений).

Если какой-либо маршрутизатор отказывает и перестает слать своим соседям сообщений о сетях, которые можно достичь через него, то через 180 секунд все записи, которые породил этот маршрутизатор, станут недействительными у его ближайших соседей. После этого процесс повторится уже для соседей ближайших соседей –они вычеркнут подобные записи уже через 360 секунд, так как первые 180 секунд ближайшие соседи еще передавали сообщения об этих записях.

Как видно из объяснения, сведения о недоступных через отказавший маршрутизатор сетях распространяются по сети не очень быстро, время распространения кратно времени жизни записи, а коэффициент кратности равен количеству хопов между самыми дальними маршрутизаторами сети. В этом заключается одна из причин выбора в качестве периода рассылки небольшой величины в 30 секунд.

Если отказывает не маршрутизатор, а интерфейс или сеть, связывающие его с каким-либо соседом, то ситуация сводится к только что описанной – снова начинает работать механизм тайм-аута и ставшие недействительными маршруты постепенно будут вычеркнуты из таблиц всех маршрутизаторов сети.

Тайм-аут работает в тех случаях, когда маршрутизатор не может послать соседям сообщение об отказавшем маршруте, так как либо он сам неработоспособен, либо неработоспособна линия связи, по которой можно было бы передать сообщение.

Когда же сообщение послать можно, RIP-маршрутизаторы не используют специальный признак в сообщении, а указывают бесконечное расстояние до сети, причем в протоколе RIP оно выбрано равным 16 хопам (при другой метрике необходимо указать маршрутизатору ее значение, считающееся бесконечностью). Получив сообщение, в котором некоторая сеть сопровождается расстоянием 16 (или 15, что приводит к тому же результату, так как маршрутизатор наращивает полученное значение на 1), маршрутизатор должен проверить, исходит ли эта «плохая» информация о сети от того же маршрутизатора, сообщение которого послужило в свое время основанием для записи о данной сети в таблице маршрутизации. Если это тот же маршрутизатор, то информация считается достоверной и маршрут помечается как недоступный.

Такое небольшое значение «бесконечного» расстояния вызвано тем, что в некоторых случаях отказы связей в сети вызывают длительные периоды некорректной работы RIP-маршрутизаторов, выражающейся в зацикливании пакетов в петлях сети. И чем меньше расстояние, используемое в качестве «бесконечного», тем такие периоды становятся короче.

Рассмотрим случай зацикливания пакетов на примере сети, изображенной на рис. 5.26.

Пусть маршрутизатор Ml обнаружил, что его связь с непосредственно подключенной сетью 201.36.14.0 потеряна (например, по причине отказа интерфейса 201.36.14.3). Ml отметил в своей таблице маршрутизации, что сеть 201.36.14.0 недоступна. В худшем случае он обнаружил это сразу же после отправки очередных RIP-сообщений, так что до начала нового цикла его объявлений, в котором он должен сообщить соседям, что расстояние до сети 201.36.14.0 стало равным 16, остается почти 30 секунд.

Каждый маршрутизатор работает на основании своего внутреннего таймера, не синхронизируя работу по рассылке объявлений с другими маршрутизаторами. Поэтому весьма вероятно, маршрутизатор М2 опередил маршрутизатор Ml и передал ему свое сообщение раньше, чем Ml успел передать новость о недостижимости сети 201.36.14.0. А в этом сообщении имеются данные, порожденные следующей записью в таблице маршрутизации М2 (табл. 5.18).

Таблица 5.18. Таблица маршрутизации маршрутизатора М2

Номер сети	Адрес следующего маршрутизатора	Порт	Расстояние
201.36.14.0	132.11.0.7	1	2

Эта запись была получена от маршрутизатора Ml и корректна до отказа интерфейса 201.36.14.3, а теперь она устарела, но маршрутизатор М2 об этом не узнал.

Теперь маршрутизатор Ml получил новую информацию о сети 201.36.14.0 – эта сеть достижима через маршрутизатор М2 с метрикой 2. Раньше Ml также получал эту информацию от М2. Но игнорировал ее,

так как его собственная метрика для 201.36.14.0 была лучше. Теперь Ml должен принять данные о сети 201.36.14.0, полученные от М2, и заменить запись в таблице маршрутизации о недостижимости этой сети (табл. 5.19).

Таблица 5.19. Таблица маршрутизации маршрутизатора Ml

Номер сети	Адрес следующего маршрутизатора	Порт	Расстояние
201.36.14.0	132.11.0.101	2	3

В результате в сети образовалась маршрутная петля: пакеты, направляемые узлам сети 201.36.14.0, будут передаваться маршрутизатором М2 маршрутизатору Ml, а маршрутизатор Ml будет возвращать их маршрутизатору М2. IP-пакеты будут циркулировать по этой петле до тех пор, пока не истечет время жизни каждого пакета.

Маршрутная петля будет существовать в сети достаточно долго. Рассмотрим периоды времени, кратные времени жизни записей в таблицах маршрутизаторов.

- Время 0-180 с. После отказа интерфейса в маршрутизаторах Ml и М2 будут сохраняться некорректные записи, приведенные выше. Маршрутизатор М2 по-прежнему снабжает маршрутизатор Ml своей записью о сети 201.36.14.0 с метрикой 2, так как ее время жизни не истекло. Пакеты зацикливаются.
- Время 180-360 с. В начале этого периода у маршрутизатора М2 истекает время жизни записи о сети 201.36.14.0 с метрикой 2, так как маршрутизатор Ml в предыдущий период посылал ему сообщения о сети 201.36.14.0 с худшей метрикой, чем у М2, и они не могли подтверждать эту запись. Теперь маршрутизатор М2 принимает от маршрутизатора Ml запись о сети 201.36.14.0 с метрикой 3 и трансформирует ее в запись с метрикой 4. Маршрутизатор Ml не получает новых сообщений от маршрутизатора М2 о сети 201.36.14.0 с метрикой 2, поэтому время жизни его записи начинает уменьшаться. Пакеты продолжают зацикливаться.
- Время 360-540 с. Теперь у маршрутизатора Ml истекает время жизни записи о сети 201.36.14.0 с метрикой 3. Маршрутизаторы Ml и М2 опять меняются ролями – М2 снабжает Ml устаревшей информацией о пути к сети 201.36.14.0, уже с метрикой 4, которую Ml преобразует в метрику 5. Пакеты продолжают зацикливаться.

Если бы в протоколе RIP не было выбрано расстояние 16 в качестве недостижимого, то описанный процесс длился бы до бесконечности (вернее, пока не была бы исчерпана разрядная сетка поля расстояния и не было бы зафиксировано переполнения при очередном наращивании расстояния).

В результате маршрутизатор М2 на очередном этапе описанного процесса получает от маршрутизатора Ml метрику 15, которая после наращивания, превращаясь в метрику 16, фиксирует недостижимость сети. Период нестабильной работы сети длился 36 минут!

Ограничение в 15 хопов сужает область применения протокола RIP до сетей, в которых число промежуточных маршрутизаторов не может быть больше 15. Для более масштабных сетей нужно применять другие протоколы маршрутизации, например OSPF, или разбивать сеть на автономные области.

Приведенный пример хорошо иллюстрирует главную причину нестабильной работы маршрутизаторов, работающих по протоколу RIP. Эта причина коренится в самом принципе работы дистанционно-векторных протоколов – пользовании информацией, полученной из вторых рук. Действительно, маршрутизатор М2 передал маршрутизатору Ml информацию о достижимости сети 201.36.14.0, за достоверность которой он сам не отвечает. Искоренить эту причину полностью нельзя, ведь сам способ построения таблиц маршрутизации связан с передачей чужой информации без указания источника ее происхождения.

Не следует думать, что при любых отказах интерфейсов и маршрутизаторов в сетях возникают маршрутные петли. Если бы маршрутизатор Ml успел передать сообщение о недостижимости сети 201.36.14.0 раньше ложной информации маршрутизатора М2, то маршрутная петля не образовалась бы. Так что маршрутные петли даже без дополнительных методов борьбы с ними, описанными в следующем разделе, возникают в среднем не более чем в половине потенциально возможных случаев.

Методы борьбы с ложными маршрутами в протоколе RIP

Несмотря на то что протокол RIP не в состоянии полностью исключить переходные состояния в сети, когда некоторые маршрутизаторы пользуются устаревшей информацией об уже несуществующих маршрутах, имеется несколько методов, которые во многих случаях решают подобные проблемы.

Ситуация с петлей, образующейся между соседними маршрутизаторами, описанная в предыдущем разделе, надежно решается с помощью метода, получившем название *расщепления горизонта* (*split horizon*). Метод заключается в том, что маршрутная информация о некоторой сети, хранящаяся в таблице маршрутизации, никогда не передается тому маршрутизатору, от которого она получена (это следующий маршрутизатор в данном маршруте). Если маршрутизатор М2 в рассмотренном выше примере поддерживает технику расщепления горизонта, то он не передаст маршрутизатору Ml устаревшую информацию о сети 201.36.14.0, так как получил ее именно от маршрутизатора Ml.

Практически все сегодняшние маршрутизаторы, работающие по протоколу RIP, используют технику расщепления горизонта.

Однако расщепление горизонта не помогает в тех случаях, когда петли образуются не двумя, а несколькими маршрутизаторами. Рассмотрим более детально ситуацию, которая возникнет в сети, приведенной на рис. 5.26, в случае потери связи маршрутизатора Ml с сетью 129.36.14.0. Пусть все маршрутизаторы этой сети поддерживают технику расщепления горизонта. Маршрутизаторы М2 и М3 не будут возвращать маршрутизатору в этой ситуации данные о сети 201.36.14.0 с метрикой 2, так как они получили эту информацию от маршрутизатора Ml. Однако они будут передавать маршрутизатору информацию о достижимости сети 201.36.14.0 с метрикой 4 через себя, так как получили эту информацию по сложному маршруту а не от маршрутизатора Ml непосредственно. Например, маршрутизатор М2 получил эту информацию по цепочке М4-М3-М1. Поэтому маршрутизатор Ml снова может быть обманут, пока каждый из маршрутизаторов в цепочке М3-М4-М2 не вычеркнет запись о достижимости сети 1 (а это произойдет через период 3 х 180 секунд).

Для предотвращения зацикливания пакетов по составным петлям при отказах связей применяются два других приема, называемые *триггерными обновлениями* (*triggered updates*) и *замораживанием изменений* (*hold down*).

Способ триггерных обновлений состоит в том, что маршрутизатор, получив данные об изменении метрики до какой-либо сети, не ждет истечения периода передачи таблицы маршрутизации, а передает данные об изменившемся маршруте немедленно. Этот прием может во многих случаях предотвратить передачу устаревших сведений об отказавшем маршруте, но он перегружает сеть служебными сообщениями, поэтому триггерные объявления также делаются с некоторой задержкой. Поэтому возможна ситуация, когда регулярное обновление в каком-либо маршрутизаторе чуть опередит по времени приход триггерного обновления от предыдущего в цепочке маршрутизатора и данный маршрутизатор успеет передать по сети устаревшую информацию о несуществующем маршруте.

Второй прием позволяет исключить подобные ситуации. Он связан с введением тайм-аута на принятие новых данных о сети, которая только что стала недоступной. Этот тайм-аут предотвращает принятие устаревших сведений о некотором маршруте от тех маршрутизаторов, которые находятся на некотором расстоянии от отказавшей связи и передают устаревшие сведения о ее работоспособности. Предполагается, что в течение тайм-аута «замораживания изменений» эти маршрутизаторы вычеркнут данный маршрут из своих таблиц, так как не получат о нем новых записей и не будут распространять устаревшие сведения по сети.

5.4.3. Протокол «состояния связей» OSPF

Протокол OSPF (*Open Shortest Path First, открытый протокол «кратчайший путь первым»*) является достаточно современной реализацией алгоритма состояния связей (он принят в 1991 году) и обладает многими особенностями, ориентированными на применение в больших гетерогенных сетях.

В OSPF процесс построения таблицы маршрутизации разбивается на два крупных этапа. На первом этапе каждый маршрутизатор строит граф связей сети, в котором вершинами графа являются маршрутизаторы и IP-сети, а ребрами – интерфейсы маршрутизаторов. Все маршрутизаторы для этого обмениваются со своими соседями той информацией о графе сети, которой они располагают к данному моменту времени. Этот процесс похож на процесс распространения векторов расстояний до сетей в

протоколе RIP, однако сама информация качественно другая – это информация о топологии сети. Эти сообщения называются *router links advertisement – объявление о связях маршрутизатора*. Кроме того, при передаче топологической информации маршрутизаторы ее не модифицируют, как это делают RIP-маршрутизаторы, а передают в неизменном виде. В результате распространения топологической информации все маршрутизаторы сети располагают идентичными сведениями о графе сети, которые хранятся в *топологической базе данных* маршрутизатора.

Второй этап состоит в нахождении оптимальных маршрутов с помощью полученного графа. Каждый маршрутизатор считает себя центром сети и ищет оптимальный маршрут до каждой известной ему сети. В каждом найденном таким образом маршруте запоминается только один шаг – до следующего маршрутизатора, в соответствии с принципом одношаговой маршрутизации. Данные об этом шаге и попадают в таблицу маршрутизации. Задача нахождения оптимального пути на графе является достаточно сложной и трудоемкой. В протоколе OSPF для ее решения используется итеративный алгоритм Дийкстры. Если несколько маршрутов имеют одинаковую метрику до сети назначения, то в таблице маршрутизации запоминаются первые шаги всех этих маршрутов.

После первоначального построения таблицы маршрутизации необходимо отслеживать изменения состояния сети и вносить коррективы в таблицу маршрутизации. Для контроля состояния связей и соседних маршрутизаторов OSPF-маршрутизаторы не используют обмен полной таблицей маршрутизации, как это не очень рационально делают RIP-маршрутизаторы. Вместо этого они передают специальные короткие сообщения HELLO. Если состояние сети не меняется, то OSPF-маршрутизаторы корректировкой своих таблиц маршрутизации не занимаются и не посылают соседям объявления о связях. Если же состояние связи изменилось, то ближайшим соседям посылается новое объявление, касающееся только данной связи, что, конечно, экономит пропускную способность сети. Получив новое объявление об изменении состояния связи, маршрутизатор перестраивает граф сети, заново ищет оптимальные маршруты (не обязательно все, а только те, на которых отразилось данное изменение) и корректирует свою таблицу маршрутизации. Одновременно маршрутизатор ретранслирует объявление каждому из своих ближайших соседей (кроме того, от которого он получил это объявление).

При появлении новой связи или нового соседа маршрутизатор узнает об этом из новых сообщений HELLO. В сообщениях HELLO указывается достаточно детальная информация о том маршрутизаторе, который послал это сообщение, а также о его ближайших соседях, чтобы данный маршрутизатор можно было однозначно идентифицировать. Сообщения HELLO отправляются через каждые 10 секунд, чтобы повысить скорость адаптации маршрутизаторов к изменениям, происходящим в сети. Небольшой объем этих сообщений делает возможной такое частое тестирование состояния соседей и связей с ними.

Так как маршрутизаторы являются одними из вершин графа, то они обязательно должны иметь идентификаторы.

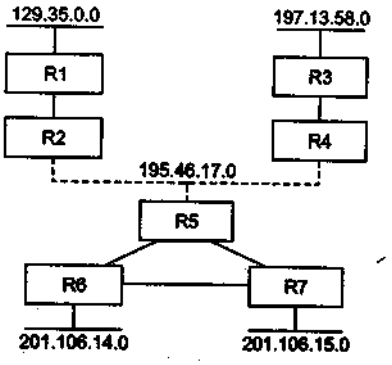


Рис. 5.27. Построение таблицы маршрутизации по протоколу OSPF

Протокол OSPF обычно использует метрику, учитывающую пропускную способность сетей. Кроме того, возможно использование двух других метрик, учитывающих требования к качеству обслуживания в IP-пакете, – задержки передачи пакетов и надежности передачи пакетов сетью. Для каждой из метрик протокол OSPF строит отдельную таблицу маршрутизации. Выбор нужной таблицы происходит в зависимости от требований к качеству обслуживания пришедшего пакета (см. рис. 5.27).

Маршрутизаторы соединены как с локальными сетями, так и непосредственно между собой глобальными каналами типа «точка-точка».

Данной сети соответствует граф, приведенный на рис. 5.28.

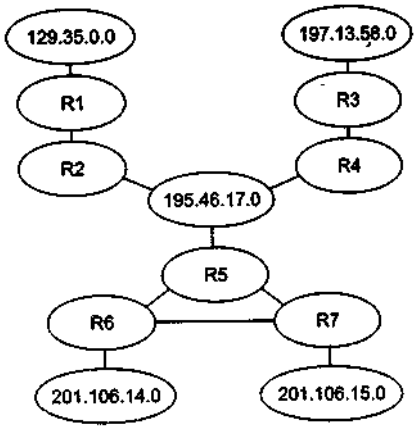


Рис. 5.28. Граф сети, построенный протоколом OSPF

Протокол OSPF в своих объявлениях распространяет информацию о связях двух типов: маршрутизатор - маршрутизатор и маршрутизатор - сеть. Примером связи первого типа служит связь «R3 - R4», а второго – связь «R4 - 195.46.17.0». Если каналам «точка-точка» дать IP-адреса, то они станут дополнительными вершинами графа, как и локальные сети. Вместе с IP-адресом сети передается также информация о маске сети.

После инициализации OSPF-маршрутизаторы знают только о связях с непосредственно подключенными сетями, как и RIP-маршрутизаторы. Они начинают распространять эту информацию своим соседям. Одновременно они посылают сообщения HELLO по всем своим интерфейсам, так что почти сразу же маршрутизатор узнает идентификаторы своих ближайших соседей, что пополняет его топологическую базу новой информацией, которую он узнал непосредственно. Далее топологическая информация начинает распространяться по сети от соседа к соседу и через некоторое время достигает самых удаленных маршрутизаторов.

Каждая связь характеризуется метрикой. Протокол OSPF поддерживает стандартные для многих протоколов (например, для протокола Spanning Tree) значения расстояний для метрики, отражающей производительность сетей: Ethernet – 10 единиц, Fast Ethernet – 1 единица, канал T1 – 65 единиц, канал 56 Кбит/с – 1785 единиц и т. д.

При выборе оптимального пути на графе с каждым ребром графа связана метрика, которая добавляется к пути, если данное ребро в него входит. Пусть на приведенном примере маршрутизатор R5 связан с R6 и R7 каналами Tl, а R6 и R7 связаны между собой каналом 56 Кбит/с. Тогда R7 определит оптимальный маршрут до сета 201.106.14.0 как составной, проходящий сначала через маршрутизатор R5, а затем через R6, поскольку у этого маршрута метрика будет равна 65 + 65 = 130 единиц. Непосредственный маршрут через R6 не будет оптимальным, так как его метрика равна 1785. При использовании хопов был бы выбран маршрут через R6, что не было бы оптимальным.

Протокол OSPF разрешает хранить в таблице маршрутизации несколько маршрутов к одной сети, если они обладают равными метриками. Если такие записи образуются в таблице маршрутизации, то маршрутизатор реализует режим баланса загрузки маршрутов (load balancing), отправляя пакеты попеременно по каждому из маршрутов.

У каждой записи в топологической базе данных имеется срок жизни, как и у маршрутных записей протокола RIP. С каждой записью о связях связан таймер, который используется для контроля времени жизни записи. Если какая-либо запись топологической базы маршрутизатора, полученная от другого маршрутизатора, устаревает, то он может запросить ее новую копию с помощью специального сообщения Link-State Request протокола OSPF, на которое должен поступить ответ Link-State Update от маршрутизатора, непосредственно тестирующего запрошенную связь.

При инициализации маршрутизаторов, а также для более надежной синхронизации топологических баз маршрутизаторы периодически обмениваются всеми записями базы, но этот период существенно больше, чем у RIP-маршрутизаторов.

Так как информация о некоторой связи изначально генерируется только тем маршрутизатором, который выяснил фактическое состояние этой связи путем тестирования с помощью сообщений HELLO, а остальные маршрутизаторы только ретранслируют эту информацию без преобразования, то

недостоверная информация о достижимости сетей, которая может появляться в RIP-маршрутизаторах, в OSPF-маршрутизаторах появиться не может, а устаревшая информация быстро заменяется новой, так как при изменении состояния связи новое сообщение генерируется сразу же.

Периоды нестабильной работы в OSPF-сетях могут возникать. Например, при отказе связи, когда информация об этом не дошла до какого-либо маршрутизатора и он отправляет пакеты сети назначения, считая эту связь работоспособной. Однако эти периоды продолжаются недолго, причем пакеты не зацикливаются в маршрутных петлях, а просто отбрасываются при невозможности их передать через неработоспособную связь.

К недостаткам протокола OSPF следует отнести его вычислительную сложность, которая быстро растет с увеличением размерности сети, то есть количества сетей, маршрутизаторов и связей между ними. Для преодоления этого недостатка в протоколе OSPF вводится понятие *области сети* (*area*) (не нужно путать с автономной системой Internet). Маршрутизаторы, принадлежащие некоторой области, строят граф связей только для этой области, что сокращает размерность сети. Между областями информация о связях не передается, а пограничные для областей маршрутизаторы обмениваются только информацией об адресах сетей, имеющихся в каждой из областей, и расстоянием от пограничного маршрутизатора до каждой сети. При передаче пакетов между областями выбирается один из пограничных маршрутизаторов области, а именно тот, у которого расстояние до нужной сети меньше. Этот стиль напоминает стиль работы протокола RIP, но нестабильность здесь устраняется тем, что петлевидные связи между областями запрещены. При передаче адресов в другую область OSPF-маршрутизаторы агрегируют несколько адресов в один, если обнаруживают у них общий префикс.

OSPF-маршрутизаторы могут принимать адресную информацию от других протоколов маршрутизации, например от протокола RIP, что полезно для работы в гетерогенных сетях. Такая адресная информация обрабатывается так же, как и внешняя информация между разными областями.

Выводы

- Крупные сети разбивают на автономные системы, в которых проводится общая политика маршрутизации IP-пакетов. Если сеть подключена к Internet, то идентификатор автономной системы назначается в InterNIC.
- Протоколы маршрутизации делятся на внешние и внутренние. Внешние протоколы (EGP, BGP) переносят маршрутную информацию между автономными системами, а внутренние (RIP, OSPF) применяются только в пределах определенной автономной системы.
- Протокол RIP является наиболее заслуженным и распространенным протоколом маршрутизации сетей TCP/IP. Несмотря на его простоту, определенную использованием дистанционно-векторного алгоритма, RIP успешно работает в небольших сетях с количеством промежуточных маршрутизаторов не более 15.
- RIP-маршрутизаторы при выборе маршрута обычно используют самую простую метрику – количество промежуточных маршрутизаторов между сетями, то есть хопов.
- Версия RIPvl не распространяет маски подсетей, что вынуждает администраторов использовать маски фиксированной длины во всей составной сети. В версии RIPv2 это ограничение снято.
- В сетях, использующих RIP и имеющих петлевидные маршруты, могут наблюдаться достаточно длительные периоды нестабильной работы, когда пакеты зацикливаются в маршрутных петлях и не доходят до адресатов. Для борьбы с этими явлениями в RIP-маршрутизаторах предусмотрено несколько приемов (Split Horizon, Hold Down, Triggered Updates), которые сокращают в некоторых случаях периоды нестабильности.
- Протокол OSPF был разработан для эффективной маршрутизации IP-пакетов в больших сетях со сложной топологией, включающей петли. Он основан на алгоритме состояния связей, который обладает высокой устойчивостью к изменениям топологии сети.
- При выборе маршрута OSPF-маршрутизаторы используют метрику, учитывающую пропускную способность составных сетей.
- Протокол OSPF является первым протоколом маршрутизации для IP-сетей, который учитывает биты качества обслуживания (пропускная способность, задержка и надежность) в заголовке IP-пакета. Для каждого типа качества обслуживания строится отдельная таблица

маршрутизации.

- Протокол OSPF обладает высокой вычислительной сложностью, поэтому чаще всего работает на мощных аппаратных маршрутизаторах.

5.5. Средства построения составных сетей стека Novell

5.5,1. Общая характеристика протокола IPX

Протокол *Internetwork Packet Exchange* (*IPX*) является оригинальным протоколом сетевого уровня стека Novell, разработанным в начале 80-х годов на основе протокола Internetwork Datagram Protocol (IDP) компании Xerox.

Протокол IPX соответствует сетевому уровню модели ISO/OSI (рис. 5.29) и поддерживает, как и протокол IP, только дейтаграммный (без установления соединений) способ обмена сообщениями. В сети NetWare наиболее быстрая передача данных при наиболее экономном использовании памяти реализуется именно протоколом IPX.

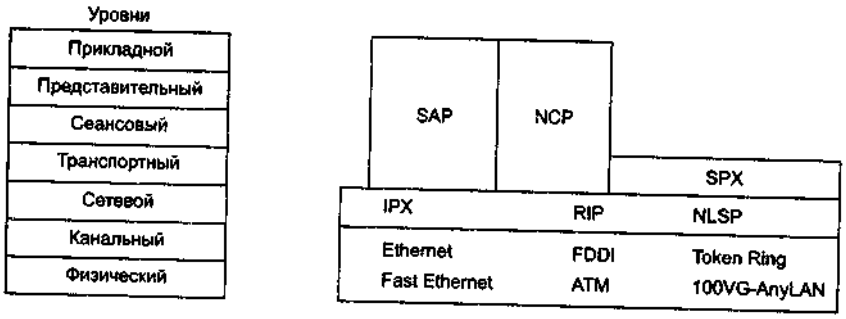


Рис. 5.29. Соответствие протоколов IPX/SPX семиуровневой модели OSI

Надежную передачу пакетов может осуществлять транспортный протокол SPX (Sequenced Packet Exchange Protocol), который работает с установлением соединения и восстанавливает пакеты при их потере или повреждении. Как видно из рис. 5.29, использование протокола SPX не является обязательным при выполнении операций передачи сообщений протоколами прикладного уровня.

Прикладной уровень стека IPX/SPX составляют два протокола: NCP и SAP. Протокол NCP (NetWare Core Protocol) поддерживает все основные службы операционной системы Novell NetWare – файловую службу, службу печати и т. д. Протокол SAP (Service Advertising Protocol) выполняет вспомогательную роль. С помощью протокола SAP каждый компьютер, который готов предоставить какую-либо службу для клиентов сети, объявляет об этом широковещательно по сети, указывая в SAP-пакетах тип службы (например, файловая), а также свой сетевой адрес. Наличие протокола SAP позволяет резко уменьшить административные работы по конфигурированию клиентского программного обеспечения, так как всю необходимую информацию для работы клиенты узнают из объявлений SAP (кроме маршрутизаторов по умолчанию, о которых можно узнать с помощью протокола IPX).

В отличие от протокола IP, который изначально разрабатывался для глобальных сетей, протокол IPX создавался для применения в локальных сетях. Именно поэтому он является одним из самых экономичных протоколов в отношении требований к вычислительным ресурсам и хорошо работает в сравнительно небольших локальных сетях.

Специфика адресации в протоколе IPX является источником как достоинств, так и недостатков этого протокола. Протокол IPX работает с сетевыми адресами, включающими три компонента:

- номер сети (4 байта);
- номер узла (6 байт);
- номер сокета (2 байта).

Номер сети в отличие от протокола IP имеет всегда фиксированную длину – 4 байта. В принципе для корпоративных сетей эта длина является избыточной, так как вряд ли у предприятия возникнет потребность разделить свою сеть на 4 миллиарда подсетей. В период доминирования сетей IPX/SPX компания Novell рассматривала возможность создания единого всемирного центра по распределению IPX-адресов, аналогичного центру InterNIC. Однако стремительный рост популярности сети Internet лишил это начинание смысла. Хотя протоколы IPX/SPX по-прежнему работают в огромном количестве

корпоративных сетей, заменить IP во всемирной сети они уже не смогут. Надо отметить, что специалисты компании Novell приложили немало усилий, чтобы в новой версии 6 протокол IP приобрел некоторые черты, свойственные протоколу IPX, и тем самым облегчил переход пользователей IPX на IPv6 (когда это станет практически необходимым). Обычно все три составляющие IPX-адреса, в том числе и номер сети, записываются в шестнадцатеричной форме.

Под номером узла в протоколе IPX понимается аппаратный адрес узла. В локальных сетях это MAC-адрес узла – сетевого адаптера или порта маршрутизатора. Размер адреса узла в 6 байт отражает происхождение этого поля, но в него можно поместить любой аппаратный адрес, если он укладывается в размер этого поля.

Номер сокета (socket) идентифицирует приложение, которое передает свои сообщения по протоколу IPX. Сокет выполняет в стеке IPX/SPX ту же роль, что порт в протоколах TCP/UDP стека TCP/IP. Наличие этого поля в протоколе сетевого уровня, которым является IPX, объясняется тем, что в стеке Novell прикладные протоколы NCP и SAP взаимодействует с сетевым уровнем непосредственно, минуя транспортный протокол SPX. Поэтому роль мультиплексора-демультиплексора прикладных протоколов приходится выполнять протоколу IPX, для чего в его пакете необходимо передавать номер сокета прикладного протокола. Протоколы NCP и SAP не пользуются услугами SPX для ускорения работы стека, а скорость работы на маломощных персональных компьютерах начала 80-х годов была одной из основных целей компании Novell. Каждый дополнительный уровень в стеке, хотя бы и такой простой, как UDP, замедляет работу стека. За отказ от транспортного уровня компании Novell пришлось реализовывать средства восстановления утерянных пакетов в протоколе NCP. Тем не менее прикладные программисты, разрабатывающие свои собственные сетевые приложения для стека IPX/ SPX, могут пользоваться протоколом SPX, если не захотят встраивать достаточно сложные алгоритмы скользящего окна в свои программы.

Протокол IPX является одним из наиболее легко настраиваемых протоколов сетевого уровня. Номер сети задается администратором только на серверах, а номер узла автоматически считывается из сетевого адаптера компьютера. На клиентском компьютере номер сети не задается – клиент узнает эту информацию из серверных объявлений SAP или локального маршрутизатора.

Адрес маршрутизатора по умолчанию также не нужно задавать вручную на каждом клиентском компьютере. В протоколе IPX есть специальный запрос, который передается на заранее определенный номер сокета. Если в сети клиента есть маршрутизатор или сервер, выполняющий роль программного маршрутизатора, то клиент при старте системы выдает такой запрос широковещательно, и все маршрутизаторы сообщают ему свои MAC-адреса, которые используются в качестве адреса следующего маршрутизатора.

Как видно из описания, административные издержки при конфигурировании сети IPX/SPX сводятся к минимуму. При этом отпадает необходимость в протоколе типа ARP, выясняющего соответствие между сетевыми адресами узлов и их MAC-адресами. Однако при смене сетевого адаптера нужно скорректировать адрес узла, если для его выяснения используются не широковещательные запросы-ответы, а справочная служба типа Novell NDS, в которой фиксируются сетевые адреса серверов. Отсутствие протокола ARP повышает производительность сети, так как позволяет не тратить время на выполнение ARP-запросов и ARP-ответов.

5.5.2. Формат пакета протокола IPX

Пакет протокола IPX имеет гораздо более простую структуру по сравнению с пакетом IP, что, собственно, и отражает меньшие функциональные возможности протокола IPX.

IPX-пакет имеет следующие поля.

- *Контрольная сумма* (*Checksum*) – это 2-байтовое поле, являющееся «пережитком прошлого», которое протокол IPX ведет от протокола IDP стека Xerox. Так как низкоуровневые протоколы (например, Ethernet) всегда выполняют проверку контрольных сумм, то IPX не использует это поле и всегда устанавливает его в единицы.
- *Длина* (*Length*) занимает 2 байта и задает размер всего пакета, включая IPX-заголовок и поле данных. Самый короткий пакет – 30 байт – включает только IPX-заголовок, а рекомендуемый максимально большой – 576 байт – включает IPX-заголовок плюс 546 байт данных. Максимальный размер пакета в 576 байт соответствует рекомендациям стандартов Internet для составных сетей. Протокол IPX вычисляет значение этого поля, основываясь на информации,

предоставляемой прикладной программой при вызове функции IPX. IPX-пакет может превосходить рекомендуемый максимум в 576 байт, что и происходит в локальных сетях Ethernet, где используются IPX-пакеты в 1500 байт с полем данных в 1470 байт.

- *Управление транспортом* (*Transport control*) имеет длину 8 бит. Это поле определяет время жизни пакета в хопах. IPX-пакет может пересечь до 15 маршрутизаторов. Протокол IPX устанавливает это однобайтовое поле в 0 до начала передачи, а затем увеличивает его на 1 каждый раз, когда пакет проходит через маршрутизатор. Если счетчик превысит 15, то пакет аннулируется.
- *Тип пакета* (*Packet type*) имеет длину 8 бит. Фирма Xerox определила в свое время определенные значения для различных типов пакетов: прикладные программы, посылающие IPX-пакеты, должны устанавливать это поле в значение, равное 4. Значение 5 соответствует служебным IPX-пакетам, используемым протоколом SPX в качестве служебных сообщений. Значение 17 указывает на то, что в поле данных IPX-пакета находится сообщение протокола NetWare Core Protocol (NCP) – основного протокола файловой службы NetWare.
- *Адрес назначения* (*Destination address*) – состоит из трех полей: номера сети назначения, номера узла назначения, номера сокета назначения. Эти поля занимают соответственно 4, 6 и 2 байта.
- *Адрес отправителя* (*Source address*) – номер исходной сети, номер исходного узла, номер исходного сокета. Аналогичны адресным полям назначения.
- *Поле данных* (*Data*). Может занимать от 0 до 546 байт. Поле данных нулевой длины может использоваться в служебных пакетах, например, для подтверждения получения предыдущего пакета.

  Из анализа формата пакета можно сделать некоторые выводы об ограничениях протокола IPX.
- *Отсутствует возможность динамической фрагментации на сетевом уровне*. В IPX-пакете нет полей, с помощью которых маршрутизатор может разбить слишком большой пакет на части. При передаче пакета в сеть с меньшим значением MTU IPX-маршрутизатор отбрасывает пакет. Протокол верхнего уровня, например NCP, должен последовательно уменьшать размер пакета до тех пор, пока не получит на него положительную квитанцию.
- *Большие накладные расходы на служебную информацию*. Сравнительно небольшая максимальная длина поля данных IPX-пакета (546 байт при длине заголовка 30 байт) приводит к тому, что как минимум 5 % данных являются служебными.
- *Время жизни пакета ограничено числом 15*, что может оказаться недостаточным для большой сети (для сравнения, в IP-сетях пакет может пройти до 255 промежуточных маршрутизаторов).
- *Отсутствует поле качества сервиса*, что не позволяет маршрутизаторам автоматически подстраиваться к требованиям приложения к качеству передачи трафика.

Кроме того, некоторые недостатки сетей Novell связаны не с протоколом IPX, а со свойствами других протоколов стека IPX/SPX. Многие недостатки проявляются при работе стека IPX/SPX на медленных глобальных линиях связи, и это закономерно, так как ОС NetWare оптимизировалась для работы в локальной сети.

Например, неэффективная работа по восстановлению потерянных и искаженных пакетов на низкоскоростных глобальных каналах обусловлена тем, что протокол NCP, который выполняет эту работу, использует метод получения квитанций с простоями. В локальных сетях со скоростью 10 Мбит/с такой метод работал вполне эффективно, а на медленных каналах время ожидания квитанции заметно тормозит работу передающего узла.

В версиях ОС NetWare до 4.0 соответствие символьных имен серверов их сетевым адресам устанавливалось только с помощью широковещательного протокола Service Advertising Protocol (SAP). Однако широковещательные рассылки заметно засоряют медленные глобальные каналы. Модернизируя свой стек для применения в крупных корпоративных сетях, компания Novell использует теперь справочную службу NDS (NetWare Directory Services) для нахождения разнообразной информации об имеющихся в сети ресурсах и службах, в том числе и о соответствии имени сервера его сетевому адресу. Так как служба NDS поддерживается только серверами с версией NetWare 4.x и выше, то для работы с версиями NetWare 3.x маршрутизаторы распознают SAP-пакеты по номеру их сокета и передают их на все порты, имитируя широковещательные рассылки локальной сети, на что тратится значительная часть пропускной способности медленных глобальных линий. Кроме того, такая

«псевдоширококовещательность» сводит на нет изоляцию сетей от некорректных SAP-пакетов.

В последних версиях своей операционной системы NetWare компания Novell значительно модифицировала свой стек для того, чтобы он мог более эффективно использоваться в крупных составных сетях.

- Служба NDS позволяет отказаться от широковещательного протокола SAP. Служба NDS основана на иерархической распределенной базе данных, хранящей информацию о пользователях и разделяемых ресурсах сети. Приложения обращаются к этой службе по протоколу прикладного уровня NDS.
- Добавлен модуль для реализации метода скользящего окна – так называемый Burst Mode Protocol NLM.
- Добавлен модуль для поддержки длинных IPX-пакетов в глобальных сетях – Large Internet Packet NLM.

Кроме того, постоянное повышение быстродействия глобальных служб уменьшает недостатки оригинальных протоколов стека IPX/SPX, что позволяет некоторым обозревателям говорить об успешной работе операционной системы NetWare в глобальных сетях и без указанных нововведений.

5.5.3. Маршрутизация протокола IPX

В целом маршрутизация протокола IPX выполняется аналогично маршрутизации протокола IP. Каждый IPX-маршрутизатор поддерживает таблицу маршрутизации, на основании которой принимается решение о продвижении пакета. IPX-маршрутизаторы поддерживает одношаговую маршрутизацию, при которой каждый маршрутизатор принимает решение только о выборе следующего на пути маршрутизатора. Возможности маршрутизации от источника в протоколе IPX отсутствуют. Рассмотрим типичную таблицу маршрутизации (табл. 5.20) для протокола IPX.

Таблица 5.20. Таблица маршрутизации протокола IPX

Номер сети	Следующий маршрутизатор	Порт	Задержка	Хопы
A0000010	-	1	0	0
A0000011	-	2	0	0
000013F4	A0000010-008100E30067	1	3	2
00000120	A0000011-C000023300FA	2	2	1
00000033	A0000010-008100E30055	1	10	5

В поле «Номер сета» указывается шестнадцатеричный адрес сети назначения, а в поле «Следующий маршрутизатор» – полный сетевой адрес следующего маршрутизатора, то есть пара «номер сети-MAC-адрес». MAC-адрес из этой записи переносится в поле адреса назначения кадра канального уровня, например Ethernet, который и переносит IPX-пакет следующему маршрутизатору. IPX-пакет при передаче между промежуточными маршрутизаторами изменений не претерпевает.

Если IPX-маршрутизатор обнаруживает, что сеть назначения – это его непосредственно подключенная сеть, то из заголовка IPX-пакета извлекается номер узла назначения, который является MAC-адресом узла назначения. Этот MAC-адрес переносится в адрес назначения кадра канального уровня, например FDDI. Кадр непосредственно отправляется в сеть, и протокол FDDI доставляет его по этому адресу узлу назначения.

IPX-маршрутизаторы обычно используют два типа метрики при выборе маршрута: расстояние в хопах и задержку в некоторых условных единицах - тиках (ticks). Расстояние в хопах имеет обычный смысл - это количество промежуточных маршрутизаторов, которые нужно пересечь IPX-пакету для достижения сети назначения. Задержка также часто используется в маршрутизаторах и мостах/коммутаторах для более точного сравнения маршрутов. Однако в IPX-маршрутизаторах традиционно задержка измеряется в тиках таймера персонального компьютера, который выдает сигнал прерывания 18,21 раза в секунду. Эта традиция ведется от первых программных IPX-маршрутизаторов, которые работали в составе операционной системы NetWare и пользовались таймером персонального компьютера для измерения интервалов времени. Напомним, что IP-маршрутизаторы, а также мосты/коммутаторы, поддерживающие протокол Spanning Tree, измеряют задержку, вносимую какой-

либо сетью в 10-наносекундных единицах передачи одного бита информации, так что сеть Ethernet оценивается задержкой в 10 единиц. Кроме этого, IPX-маршрутизаторы оценивают задержку не одного бита, а стандартного для IPX-пакета в 576 байт.

Поэтому задержка в тиках для сети Ethernet получается равной 0,00839 тика, а для канала 64 Кбит/с – 1,31 тика. Задержка в тиках всегда округляется до целого числа тиков в большую сторону, так что сеть Ethernet вносит задержку в один тик, а канал 64 Кбит/с – в 2 тика. При вычислении метрики в тиках для составного маршрута задержки в тиках складываются.

Две метрики в записях таблицы маршрутизации протокола IPX используются в порядке приоритетов. Наибольшим приоритетом обладает метрика, измеренная в задержках, а если эта метрика совпадает для каких-либо маршрутов, то во внимание принимается расстояние в хопах.

Несмотря на традиции измерения задержки в тиках, IPX-маршрутизаторы могут использовать и стандартные задержки сетей, измеренные в 10-наносекундных интервалах.

IPX-маршрутизаторы могут поддерживать как статические маршруты, так и динамические, полученные с помощью протоколов RIP IPX и NLSP.

Протокол RIP IPX очень близок к протоколу RIP IP. Так как в IPX-сетях маски не применяются, то RIP IPX не имеет аналога RIPv2, передающего маски. Интервал между объявлениями у протокола RIP IPX равен 60 с (в отличие от 30 с у RIP IP). В пакетах RIP IPX для каждой сети указываются обе метрики - в хопах и тиках. Для исключения маршрутных петель IPX-маршрутизаторы используют прием расщепления горизонта.

Время жизни динамической записи составляет 180 секунд. Недостижимость сети указывается значением числа хопов в 15 (0xF), а тиков – в 0xFFFF.

IPX-маршрутизаторы, как и IP-маршрутизаторы, не передают из сети в сеть пакеты, имеющие широковещательный сетевой адрес. Однако для некоторых типов таких пакетов IPX-маршрутизаторы делают исключения. Это пакеты службы. SAP, с помощью которой серверы NetWare объявляют о себе по сети. IPX-маршрутизаторы передают SAP-пакеты во все непосредственно подключенные сети, кроме той, от которой этот пакет получен (расщепление горизонта). Если бы IPX-маршрутизаторы не выполняли таких передач, то клиенты NetWare не смогли бы взаимодействовать с серверами в сети, разделенной маршрутизаторами, в привычном стиле, то есть путем просмотра имеющихся серверов с помощью команды SLIST.

IPX-маршрутизаторы всегда используют внутренний номер сети, который относится не к интерфейсам маршрутизатора, а к самому модулю маршрутизации. Внутренний номер сети является некоторым аналогом сети 127.0.0.0 узлов IP-сетей, однако каждый IPX-маршрутизатор должен иметь уникальный внутренний номер сети, причем его уникальность должна распространяться и на внешние номера IPX-сетей в составной сети.

IPX-маршрутизаторы выполняют также функцию согласования форматов кадров Ethernet. В составных IPX-сетях каждая сеть может работать только с одним из 4-х возможных типов кадров IPX. Поэтому если в разных сетях используются разные типы кадров Ethernet, то маршрутизатор посылает в каждую сеть тот тип кадра, который установлен для этой сети.

Протокол NLSP (NetWare Link Services Protocol) представляет собой реализацию алгоритма состояния связей для IPX-сетей. В основном он работает аналогично протоколу OSPF сетей TCP/IP.

Выводы

- Стек Novell состоит из четырех уровней: канального, который собственно стеком Novell не определяется; сетевого, представленного протоколом дейтаграммного типа IPX; транспортного, на котором работает протокол надежной передачи данных SPX; прикладного, на котором работает протокол NCP, поддерживающий файловую службу и службу печати, а также протоколы SAP и NDS, выполняющие служебные функции по поиску в сети разделяемых ресурсов.
- Особенностью стека Novell является то, что основной прикладной протокол NCP не пользуется транспортным протоколом SPX, а обращается непосредственно к сетевому протоколу IPX. Это значительно ускоряет работу стека, но усложняет прикладной протокол NCP.
- Сетевой IPX-адрес состоит из номера сети, назначаемого администратором, и номера узла, который в локальных сетях совпадает с аппаратным адресом узла, то есть MAC-адресом.

Использование аппаратных адресов узлов на сетевом уровне ускоряет работу протокола, так как при этом отпадает необходимость в выполнении протокола типа ARP. Также упрощается конфигурирование компьютеров сети, так как они узнают свой номер сети от локального маршрутизатора, а номер узла извлекается из сетевого адаптера.

- Недостатком IPX-адресации является ограничение в 6 байт, накладываемое на адрес узла на сетевом уровне. Если какая-либо составная сеть использует,аппаратные адреса большего размера (это может произойти, например, в сети X.25), то протокол IPX не сможет доставить пакет конечному узлу такой сети.
- IPX-маршрутизаторы используют протоколы динамической маршрутизации RIP IPX, являющийся аналогом RIP IP, и NLSP, который во многом похож на протокол OSPF сетей TCP/IP.

5.6. Основные характеристики маршрутизаторов и концентраторов

5.6.1. Маршрутизаторы

Основная задача маршрутизатора – выбор наилучшего маршрута в сети – часто является достаточно сложной с математической точки зрения. Особенно интенсивных вычислений требуют протоколы, основанные на алгоритме состояния связей, вычисляющие оптимальный путь на графе, – OSPF, NLSP, IS-IS. Кроме этой основной функции в круг ответственности маршрутизатора входят и другие задачи, такие как буферизация, фильтрация и фрагментация перемещаемых пакетов. При этом очень важна производительность, с которой маршрутизатор выполняет эти задачи.

Поэтому типичный маршрутизатор является мощным вычислительным устройством с одним или даже несколькими процессорами, часто специализированными или построенными на RISC-архитектуре, со сложным программным обеспечением. То есть сегодняшний маршрутизатор – это специализированный компьютер, имеющий скоростную внутреннюю шину или шины (с пропускной способностью 600-2000 Мбит/с), часто использующий симметричное или асимметричное мультипроцессирование и работающий под управлением специализированной операционной системы, относящейся к классу систем реального времени. Многие разработчики маршрутизаторов построили в свое время такие операционные системы на базе операционной системы Unix, естественно, значительно ее переработав.

Маршрутизаторы могут поддерживать как один протокол сетевого уровня (например, IP, IPX или DECnet), так и множество таких протоколов. В последнем случае они называются *многопротокольными* маршрутизаторами. Чем больше протоколов сетевого уровня поддерживает маршрутизатор, тем лучше он подходит для корпоративной сети.

Большая вычислительная мощность позволяет маршрутизаторам наряду с основной работой по выбору оптимального маршрута выполнять и ряд вспомогательных высокоуровневых функций.

Классификация маршрутизаторов по областям применения

По областям применения маршрутизаторы делятся на несколько классов.

Магистральные маршрутизаторы (*backbone routers*) предназначены для построения центральной сети корпорации. Центральная сеть может состоять из большого количества локальных сетей, разбросанных по разным зданиям и использующих самые разнообразные сетевые технологии, типы компьютеров и операционных систем. Магистральные маршрутизаторы – это наиболее мощные устройства, способные обрабатывать несколько сотен тысяч или даже несколько миллионов пакетов в секунду, имеющие большое количество интерфейсов локальных и глобальных сетей. Поддерживаются не только среднескоростные интерфейсы глобальных сетей, такие как T1/E1, но и высокоскоростные, например, ATM или SDH со скоростями 155 Мбит/с или 622 Мбит/с. Чаще всего магистральный маршрутизатор конструктивно выполнен по модульной схеме на основе шасси с большим количеством слотов – до 12-14. Большое внимание уделяется в магистральных моделях надежности и отказоустойчивости маршрутизатора, которая достигается за счет системы терморегуляции, избыточных источников питания, заменяемых «на ходу» (hot swap) модулей, а также симметричного мультипроцессирования. Примерами магистральных маршрутизаторов могут служить маршрутизаторы Backbone Concentrator Node (BCN) компании Nortel Networks (ранее Bay Networks), Cisco 7500, Cisco

12000.

Маршрутизаторы региональных отделений соединяют региональные отделения между собой и с центральной сетью. Сеть регионального отделения, так же как и центральная сеть, может состоять из нескольких локальных сетей. Такой маршрутизатор обычно представляет собой некоторую упрощенную версию магистрального маршрутизатора. Если он выполнен на основе шасси, то количество слотов его шасси меньше: 4-5. Возможен также конструктив с фиксированным количеством портов. Поддерживаемые интерфейсы локальных и глобальных сетей менее скоростные. Примерами маршрутизаторов региональных отделений могут служить маршрутизаторы BLN, ASN компании Nortel Networks, Cisco 3600, Cisco 2500, NetBuilder II компании 3Com. Это наиболее обширный класс выпускаемых маршрутизаторов, характеристики которых могут приближаться к характеристикам магистральных маршрутизаторов, а могут и опускаться до характеристик маршрутизаторов удаленных офисов.

Маршрутизаторы удаленных офисов соединяют, как правило, единственную локальную сеть удаленного офиса с центральной сетью или сетью регионального отделения по глобальной связи. В максимальном варианте такие маршрутизаторы могут поддерживать и два интерфейса локальных сетей. Как правило, интерфейс локальной сети – это Ethernet 10 Мбит/с, а интерфейс глобальной сети – выделенная линия со скоростью 64 Кбит/с, 1,544 или 2 Мбит/с. Маршрутизатор удаленного офиса может поддерживать работу по коммутируемой телефонной линии в качестве резервной связи для выделенного канала. Существует очень большое количество типов маршрутизаторов удаленных офисов. Это объясняется как массовостью потенциальных потребителей, так и специализацией такого типа устройств, проявляющейся в поддержке одного конкретного типа глобальной связи. Например, существуют маршрутизаторы, работающие только по сети ISDN, существуют модели только для аналоговых выделенных линий и т. п. Типичными представителями этого класса являются маршрутизаторы Nautika компании Nortel Networks, Cisco 1600, Office Connect компании 3Com, семейство Pipeline компании Ascend.

Маршрутизаторы локальных сетей (*коммутаторы 3-го уровня*) предназначены для разделения крупных локальных сетей на подсети. Основное требование, предъявляемое к ним, – высокая скорость маршрутизации, так как в такой конфигурации отсутствуют низкоскоростные порты, такие как модемные порты 33,6 Кбит/с или цифровые порты 64 Кбит/с. Все порты имеют скорость по крайней мере 10 Мбит/с, а многие работают на скорости 100 Мбит/с. Примерами коммутаторов 3-го уровня служат коммутаторы CoreBuilder 3500 компании 3Com, Accelar 1200 компании Nortel Networks, Waveswitch 9000 компании Plaintree, Turboiron Switching Router компании Foudry Networks.

В зависимости от области применения маршрутизаторы обладают различными основными и дополнительными техническими характеристиками.

Основные технические характеристики маршрутизатора

Основные технические характеристики маршрутизатора связаны с тем, как он решает свою главную задачу – маршрутизацию пакетов в составной сети. Именно эти характеристики прежде всего определяют возможности и сферу применения того или иного маршрутизатора.

Перечень поддерживаемых сетевых протоколов. Магистральный маршрутизатор должен поддерживать большое количество сетевых протоколов и протоколов маршрутизации, чтобы обеспечивать трафик всех существующих на предприятии вычислительных систем (в том числе и устаревших, но все еще успешно эксплуатирующихся, так называемых унаследованных – legacy), а также систем, которые могут появиться на предприятии в ближайшем будущем. Если центральная сеть образует отдельную автономную систему Internet, то потребуется поддержка и специфических протоколов маршрутизации этой сети, таких как EGP и BGP. Программное обеспечение магистральных маршрутизаторов обычно строится по модульному принципу, поэтому при возникновении потребности можно докупать и добавлять программные модули, реализующие недостающие протоколы.

Перечень поддерживаемых сетевых протоколов обычно включает протоколы IP, CONS и CLNS OSI, IPX, AppleTalk, DECnet, Banyan VINES, Xerox XNS.

Перечень протоколов маршрутизации составляют протоколы IP RIP, IPX RIP, NLSP, OSPF, IS-IS OSI, EGP, BGP, VINES RTP, AppleTalk RTMP.

Перечень поддерживаемых интерфейсов локальных и глобальных сетей. Для локальных сетей – это интерфейсы, реализующие физические и канальные протоколы сетей Ethernet, Token Ring, FDDI, Fast

Ethernet, Gigabit Ethernet, lOOVG-AnyLAN и ATM.

Для глобальных связей – это интерфейсы физического уровня для связи с аппаратурой передачи данных, а также протоколы канального и сетевого уровней, необходимые для подключения к глобальным сетям с коммутацией каналов и пакетов.

Поддерживаются интерфейсы последовательных линий (serial lines) RS-232, RS-449/422, V.35 (для передачи данных со скоростями до 2-6 Мбит/с), высокоскоростной интерфейс HSSI, обеспечивающий скорость до 52 Мбит/с, а также интерфейсы с цифровыми каналами T1/E1, T3/E3 и интерфейсами BRI и PRI цифровой сети ISDN. Некоторые маршрутизаторы имеют аппаратуру связи с цифровыми глобальными каналами, что исключает необходимость использования внешних устройств сопряжения с этими каналами.

В набор поддерживаемых глобальных технологий обычно входят технологии X.25, frame relay, ISDN и коммутируемых аналоговых телефонных сетей, сетей ATM, а также поддержка протокола канального уровня PPP.

Общая производительность маршрутизатора. Высокая производительность маршрутизации важна для работы с высокоскоростными локальными сетями, а также для поддержки новых высокоскоростных глобальных технологий, таких как, frame relay, T3/E3, SDH и ATM. Общая производительность маршрутизатора зависит от многих факторов, наиболее важными из которых являются: тип используемых процессоров, эффективность программной реализации протоколов, архитектурная организация вычислительных и интерфейсных модулей. Общая производительность маршрутизаторов колеблется от нескольких десятков тысяч пакетов в секунду до нескольких миллионов пакетов в секунду. Наиболее производительные маршрутизаторы имеют мультипроцессорную архитектуру, сочетающую симметричные и асимметричные свойства – несколько мощных центральных процессоров по симметричной схеме выполняют функции вычисления таблицы маршрутизации, а менее мощные процессоры в интерфейсных модулях занимаются передачей пакетов на подключенные к ним сети и пересылкой пакетов на основании части таблицы маршрутизации, кэшированной в локальной памяти интерфейсного модуля.

Магистральные маршрутизаторы обычно поддерживают максимальный набор протоколов и интерфейсов и обладают высокой общей производительностью в один-два миллиона пакетов в секунду. Маршрутизаторы удаленных офисов поддерживают один-два протокола локальных сетей и низкоскоростные глобальные протоколы, общая производительность таких маршрутизаторов обычно составляет от 5 до 20-30 тысяч пакетов в секунду.

Маршрутизаторы региональных отделений занимают промежуточное положение, поэтому их иногда не выделяют в отдельный класс устройств.

Наиболее высокой производительностью обладают коммутаторы 3-го уровня, особенности которых рассмотрены ниже.

Дополнительные функциональные возможности маршрутизаторов

Наряду с функцией маршрутизации многие маршрутизаторы обладают следующими важными дополнительными функциональными возможностями, которые значительно расширяют сферу применения этих устройств.

Поддержка одновременно нескольких протоколов маршрутизации. В протоколах маршрутизации обычно предполагается, что маршрутизатор строит свою таблицу на основе работы только этого одного протокола. Деление Internet на автономные системы также направлено на исключение использования в одной автономной системе нескольких протоколов маршрутизации. Тем не менее иногда в большой корпоративной сети приходится поддерживать одновременно несколько таких протоколов, чаще всего это складывается исторически. При этом таблица маршрутизации может получаться противоречивой – разные протоколы маршрутизации могут выбрать разные следующие маршрутизаторы для какой-либо сети назначения. Большинство маршрутизаторов решает эту проблему за счет придания приоритетов решениям разных протоколов маршрутизации. Высший приоритет отдается статическим маршрутам (администратор всегда прав), следующий приоритет имеют маршруты, выбранные протоколами состояния связей, такими как OSPF или NLSP, а низшим приоритетов обладают маршруты дистанционно-векторных протоколов, как самых несовершенных.

Приоритеты сетевых протоколов. Можно установить приоритет одного.протокола сетевого уровня над другими. На выбор маршрутов эти приоритеты не оказывают никакого влияния, они влияют только

на порядок, в котором многопротокольный маршрутизатор обслуживает пакеты разных сетевых протоколов. Это свойство бывает полезно в случае недостаточной полосы пропускания кабельной системы и существования трафика, чувствительного к временным задержкам, например трафика SNA или голосового трафика, передаваемого одним из сетевых протоколов.

Поддержка политики маршрутных объявлений. В большинстве протоколов обмена маршрутной информации (RIP, OSPF, NLSP) предполагается, что маршрутизатор объявляет в своих сообщениях обо всех сетях, которые ему известны. Аналогично предполагается, что маршрутизатор при построении своей таблицы учитывает все адреса сетей, которые поступают ему от других маршрутизаторов сети. Однако существуют ситуации, когда администратор хотел бы скрыть существование некоторых сетей в определенной части своей сети от других администраторов, например, по соображениям безопасности. Или же администратор хотел бы запретить некоторые маршруты, которые могли бы существовать в сети. При статическом построении таблиц маршрутизации решение таких проблем не составляет труда. Динамические же протоколы маршрутизации не позволяют стандартным способом реализовывать подобные ограничения. Существует только один широко используемый протокол динамической маршрутизации, в котором описана возможность существования *правил* (*policy*), ограничивающих распространение некоторых адресов в объявлениях, – это протокол BGP. Необходимость поддержки таких правил в протоколе BGP понятна, так как это протокол обмена маршрутной информацией между автономными системами, где велика потребность в административном регулировании маршрутов (например, некоторый поставщик услуг Internet может не захотеть, чтобы через него транзитом проходил трафик другого поставщика услуг). Разработчики маршрутизаторов исправляют этот недостаток стандартов протоколов, вводя в маршрутизаторы поддержку правил передачи и использования маршрутной информации, подобных тем, которые рекомендует BGP.

Защита от широковещательных штормов (*broadcast storm*). Одна из характерных неисправностей сетевого программного обеспечения – самопроизвольная генерация с высокой интенсивностью широковещательных пакетов. Широковещательным штормом считается ситуация, в которой процент широковещательных пакетов превышает 20 % от общего количества пакетов в сети. Обычный коммутатор или мост слепо передает такие пакеты на все свои порты, как того требует его логика работы, засоряя, таким образом, сеть. Борьба с широковещательным штормом в сети, соединенной коммутаторами, требует от администратора отключения портов, генерирующих широковещательные пакеты. Маршрутизатор не распространяет такие поврежденные пакеты, поскольку в круг его задач не входит копирование широковещательных пакетов во все объединяемые им сети. Поэтому маршрутизатор является прекрасным средством борьбы с широковещательным штормом, правда, если сеть разделена на достаточное количество подсетей.

Поддержка немаршрутизируемых протоколов, таких как NetBIOS, NetBEUI или DEC LAT, которые не оперируют с таким понятием, как сеть. Маршрутизаторы могут обрабатывать пакеты таких протоколов двумя способами.

- В первом случае они могут работать с пакетами этих протоколов как мосты, то есть передавать их на основании изучения MAC-адресов. Маршрутизатор необходимо сконфигурировать особым способом, чтобы по отношению к некоторым немаршрутизируемым протоколам на некоторых портах он выполнял функции моста, а по отношению к маршрутизируемым протоколам – функции маршрутизатора. Такой мост/маршрутизатор иногда называют brouter (bridge плюс router).
- Другим способом передачи пакетов немаршрутизируемых протоколов является *инкапсуляция* этих пакетов в пакеты какого-либо сетевого протокола. Некоторые производители маршрутизаторов разработали собственные протоколы, специально предназначенные для инкапсуляции немаршрутизируемых пакетов. Кроме того, существуют стандарты для инкапсуляции некоторых протоколов в другие, в основном в IP. Примером такого стандарта является протокол DLSw, определяющий методы инкапсуляции пакетов SDLC и NetBIOS в IP-пакеты, а также протоколы PPTP и L2TP, инкапсулирующие кадры протокола PPP в IP-пакеты. Более подробно технология инкапсуляции рассматривается в главе, посвященной межсетевому взаимодействию.

Разделение функций построения и использования таблицы маршрутизации. Основная вычислительная работа проводится маршрутизатором при составлении таблицы маршрутизации с маршрутами ко всем известным ему сетям. Эта работа состоит в обмене пакетами протоколов маршрутизации, такими как RIP или OSPF, и вычислении оптимального пути к каждой целевой сети по

некоторому критерию. Для вычисления оптимального пути на графе, как того требуют протоколы состояния связей, необходимы значительные вычислительные мощности. После того как таблица маршрутизации составлена, функция продвижения пакетов происходит весьма просто – осуществляется просмотр таблицы и поиск совпадения полученного адреса с адресом целевой сети. Если совпадение есть, то пакет передается на соответствующий порт маршрутизатора. Некоторые маршрутизаторы поддерживают только функции продвижения пакетов по готовой таблице маршрутизации. Такие маршрутизаторы являются усеченными маршрутизаторами, так как для их полноценной работы требуется наличие полнофункционального маршрутизатора, у которого можно взять готовую таблицу маршрутизации. Этот маршрутизатор часто называется сервером маршрутов. Отказ от самостоятельного выполнения функций построения таблицы маршрутизации резко удешевляет маршрутизатор и повышает его производительность. Примерами такого подхода являются маршрутизаторы NetBuilder компании 3Com, поддерживающие фирменную технологию Boundary Routing, маршрутизирующие коммутаторы Catalyst 5000 компании Cisco Systems.

5.6.2. Корпоративные модульные концентраторы

Большинство крупных фирм-производителей сетевого оборудования предлагает модульные концентраторы в качестве «коммутационного центра» корпоративной сети. Такие концентраторы отражают тенденцию перехода от полностью распределенных локальных сетей 70-х годов на коаксиальном кабеле к централизованным коммуникационным решениям, активно воздействующим на передачу пакетов между сегментами и сетями. Модульные корпоративные концентраторы представляют собой многофункциональные устройства, которые могут включать несколько десятков модулей различного назначения: повторителей разных технологий, коммутаторов, удаленных мостов, маршрутизаторов и т. п., которые объединены в одном устройстве с модулями-агентами протокола SNMP, и, следовательно, позволяют централизованно объединять, управлять и обслуживать большое количество устройств и сегментов, что очень удобно в сетях большого размера.

Модульный концентратор масштаба предприятия обычно обладает внутренней шиной или набором шин очень высокой производительности – до нескольких десятков гигабит в секунду, что позволяет реализовать одновременные соединения между модулями с высокой скоростью, гораздо большей, чем скорость внешних интерфейсов модулей. Основная идея разработчиков таких устройств заключается в создании программно настраиваемой конфигурации связей в сети, причем сами связи между устройствами и сегментами могут также поддерживаться с помощью различных методов: побитовой передачи данных повторителями, передачи кадров коммутаторами и передачи пакетов сетевых протоколов маршрутизаторами.

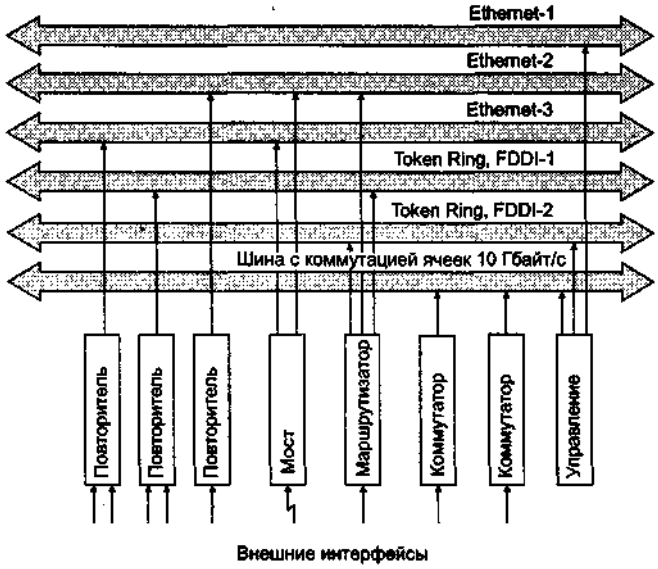


Рис. 5.30. Структура корпоративного модульного концентратора

Пример структуры корпоративного концентратора приведен на рис. 5.30. Он имеет несколько шин для образования независимых разделяемых сегментов Ethernet 10 Мбит/с, Token Ring и FDDI, а также

высокоскоростную шину в 10 Гбит/с для передач кадров и пакетов между модулями коммутации и маршрутизации. Каждый из модулей имеет внешние интерфейсы для подключения конечных узлов и внешних коммуникационных устройств – повторителей, коммутаторов, а также несколько интерфейсов с внутренними шинами концентратора. Концентратор рассчитан на подключение конечных узлов в основном к внешним интерфейсам повторителей (для .образования разделяемых сегментов) и коммутаторов (для поддержки микросегментации). Уже готовые сегменты, то есть образованные внешними повторителями и коммутаторами, могут подключаться к внешним интерфейсам коммутаторов и маршрутизаторов корпоративного концентратора. Дальнейшее соединение разделяемых сегментов и коммутируемых узлов и сегментов происходит модулями коммутации и маршрутизации концентратора по внутренним связям и с помощью высокоскоростной шины. Конечно, модули могут связываться между собой и через внешние интерфейсы, но такой способ нерационален, так как скорость обмена ограничивается при этом скоростью протокола интерфейса, например 10 Мбит/с или 100 Мбит/с. Внутренняя же шина соединяет модули на гораздо более высокой скорости, примерно 10/N Гбит/с, где N – количество портов, одновременно требующих обмена. Внешние связи между модулями превращают корпоративный концентратор просто в стойку с установленными модулями, а внутренний обмен делает эту стойку единым устройством с общей системой программного управления трафиком. Обычно для конфигурирования модулей и связей между ними производители корпоративных концентраторов сопровождают их удобным программным обеспечением с графическим интерфейсом. Отдельный модуль управления выполняет общие для всего концентратора функции: управления по протоколу SNMP, согласование таблиц коммутации и маршрутизации в разных модулях, возможно использование этого модуля как межмодульной коммутационной фабрики вместо общей шины.

Примерами корпоративных многофункциональных концентраторов могут служить устройства System 5000 компании Nortel Networks, MMAC-Plus компании Cabletron Systems, CoreBuilder 6012 компании 3Com.

Ввиду того, что отказ корпоративного модульного концентратора приводит к очень тяжелым последствиям, в их конструкцию вносится большое количество средств обеспечения отказоустойчивости.

5.6.3. Стирание граней между коммутаторами и маршрутизаторами

В классическом понимании терминов коммутатор – это устройство, принимающее решение о продвижении пакетов на основании заголовков протоколов 2-го уровня, то есть протоколов типа Ethernet или FDDI, а маршрутизатор – устройство, принимающее аналогичное решение на основании заголовков протоколов 3-го уровня, то есть уровня протоколов IP или IPX. В настоящее время наблюдается отчетливая тенденция по совмещению в одном устройстве функций коммутатора и маршрутизатора.

Соотношение коммутации и маршрутизации в корпоративных сетях

До недавнего времени сложившимся информационным потокам корпоративной сети наилучшим образом соответствовала следующая иерархическая структура. На нижнем уровне (уровне отделов) располагались сегменты сети, построенные на быстро работающих повторителях и коммутаторах. Сегменты включали в себя как рабочие станции так и серверы. В большинстве случаев было справедливо эмпирическое соотношение 80/20, в соответствии с которым основная часть трафика (80 %) циркулировала внутри сегмента, то есть порождалась запросами пользователей рабочих станций к серверам своего же сегмента.

На более высоком уровне располагался маршрутизатор, к которому подключалось сравнительно небольшое количество внутренних сетей, построенные на коммутаторах. Через порты маршрутизатора проходил трафик обращений рабочих станций одних сетей к серверам других сетей. Известно, что маршрутизатор затрачивает больше времени на обработку каждого пакета, чем коммутатор, поскольку он выполняет более сложную обработку трафика, включая интелектуальные алгоритмы фильтрации, выбор маршрута при наличии нескольких возможных путей и т. п. С другой стороны, трафик, проходящий через порты маршрутизатора был менее интенсивный, чем внутрисегментный, поэтому сравнительно низкая производительность маршрутизатора не делала его узким местом.

Сегодня ситуация в корпоративных сетях быстро меняется. Количество пользователей стремительно растет. Пользователи избавляются от устаревающих текстовых приложений, отдавая предпочтение Web-интерфейсу. А завтра эти же пользователи будут работать с аудио, видео, push и другими, абсолютно новыми приложениями, основанными на новых технологиях распространения пакетов, таких как IP Multicast и RSVP. Не работает и старое правило 80/20, сегодня большое количество информации берется из публичных серверов Internet, а также из Web-серверов других подразделений предприятия, создавая большой межсетевой трафик. Существующие сети не оптимизировались для таких непредсказуемых потоков трафика, когда каждый может общаться почти с каждым. А с проникновением в корпоративные сети технологии Gigabit Ethernet эта проблема обострится еще больше.

Таким образом, сегодня образовался большой разрыв между производительностью типичного маршрутизатора и типичного коммутатора. В этой ситуации возможны два решения: либо отказаться вообще от маршрутизации, либо увеличить ее производительность.

Отказ от маршрутизации

За последние годы основные усилия были сосредоточены в первом направлении: применять маршрутизацию как можно реже, только там, где от нее никак нельзя отказаться. Например, на границе между локальной и глобальной сетью. Отказ от маршрутизаторов означает переход к так называемой плоской сети, то есть сети, построенной только на коммутаторах, а значит, и отказ от всех интеллектуальных возможностей обработки трафика, присущих маршрутизаторам. Такой подход повышает производительность, но приводит к потере всех преимуществ, которые давали маршрутизаторы, а именно:

- маршрутизаторы более надежно, чем коммутаторы, изолируют части большой составной сети друг от друга, защищая их от ошибочных кадров, порождаемых неисправным программным или аппаратным обеспечением других сетей (например, от широковещательных штормов);
- маршрутизаторы обладают более развитыми возможностями защиты от несанкционированного доступа за счет функций анализа и фильтрации трафика на более высоких уровнях: сетевом и транспортном;
- сеть, не разделенная маршрутизаторами, имеет ограничения на число узлов (для популярного протокола IP это ограничение составляет 255 узлов для сетей самого доступного класса C).

Из этого следует, что в сети необходимо сохранять функции маршрутизации в привычном смысле этого слова.

Что касается второго направления – повышение производительности маршрутизаторов, – сложилось так, что самые активные действия в этом направлении были предприняты производителями коммутаторов, наделявшими свои продукты некоторыми возможностями маршрутизаторов. Именно в модифицированных коммутаторах были впервые достигнуты скорости маршрутизации в 5-7 миллионов пакетов в секунду, а также опробованы многие важные концепции ускорения функции маршрутизации.

Коммутаторы 3-го уровня с классической маршрутизацией

Термин «коммутатор 3-го уровня» употребляется для обозначения целого спектра коммутаторов различного типа, в которые встроены функции маршрутизации пакетов. Функции коммутации и маршрутизации могут быть совмещены двумя способами.

- Классическим, когда маршрутизация выполняется по каждому пакету, требующему передачи из сети в сеть, а коммутация выполняется для пакетов, принадлежащих одной сети.
- Нестандартным способом ускоренной маршрутизации, когда маршрутизируется несколько первых пакетов устойчивого потока, а все остальные пакеты этого потока коммутируются. Рассмотрим первый способ.

Классический коммутатор 3-го уровня подобно обычному коммутатору захватывает все кадры своими портами независимо от их MAC-адресов, а затем принимает решение о коммутации или маршрутизации каждого кадра. Если кадр имеет MAC-адрес назначения, отличный от MAC-адреса порта маршрутизатора, то этот кадр коммутируется. Если устройство поддерживает технику VLAN, то перед передачей кадра проверяется принадлежность адресов назначения и источника одной виртуальной сети.

Если же кадр направлен непосредственно MAC-адресу какого-либо порта маршрутизатора, то он маршрутизируется стандартным образом. Коммутатор 3-го уровня может поддерживать динамические протоколы маршрутизации, такие как RIP или OSPF, а может полагаться на статическое задание маршрутов или на получение таблицы маршрутизации от другого маршрутизатора.

Такие комбинированные устройства появились сразу после разработки коммутаторов, поддерживающих виртуальные локальные сети (VLAN). Для связи VLAN требовался маршрутизатор. Размещение маршрутизатора в одном корпусе с коммутатором позволяло получить некоторый выигрыш в производительности, например, за счет исключения одного этапа буферизации пакета, когда он передается из коммутатора в маршрутизатор. Хотя такие устройства с равным успехом можно называть маршрутизирующими коммутаторами или коммутирующими маршрутизаторами, за ними закрепилось название коммутаторов 3-го уровня.

Примерами таких коммутаторов могут служить хорошо известные коммутаторы LANplex (теперь CoreBuilder) 6000 и 2500 компании 3Com. В этих устройствах совместно используются специализированные большие интегральные микросхемы (ASIC), RISC- и CISC-процессоры. Микросхемы ASIC обеспечивают коммутацию пакетов и их первичный анализ при маршрутизации, RISC-процессоры выполняют основную работу по маршрутизации, а CISC-процессоры реализуют функции управления. За счет такого распараллеливания процесса функционирования подсистем коммутации и маршрутизации достигается достаточно высокий уровень производительности. Так, система CoreBuilder 2500, имеющая один блок коммутации/маршрутизации, способна маршрутизировать 98 тысяч IP-пакетов в секунду (без их потери) на полной скорости каналов связи. Более мощная система CoreBuilder 6000 по данным компании 3Com в конфигурации с 88 портами Fast Ethernet маршрутизирует до 3 миллионов пакетов в секунду.

Более быстродействующей реализацией данного подхода являются устройства, в которых функции маршрутизации перенесены из универсального центрального процессора в специализированные заказные микросхемы портов. При этом ускорение процесса маршрутизации происходит не только за счет распараллеливания работы между несколькими процессорами, но и за счет использования специализированных процессоров вместо универсальных процессоров типа Motorola или Intel. Примеры этого подхода – коммутатор CoreBuilder 3500 компании 3Com, маршрутизирующий коммутатор Accelar 1200 компании Nortel Networks.

По данным фирм-производителей, коммутаторы 3-го уровня CoreBuilder 3500 и Accelar 1200 способны маршрутизировать соответственно до 4 и 7 миллионов пакетов в секунду. С такой же скоростью они коммутируют поступающие кадры, что говорит о высокой эффективности реализованных в ASIC алгоритмах маршрутизации.

Подход, связанный с переносом процедур маршрутизации из программируемых процессоров, пусть и специализированных, в работающие по жестким алгоритмам БИС, имеет один принципиальный недостаток – ему недостает гибкости. При необходимости изменения протокола или набора протоколов требуется перепроектировать БИС, что очевидно подразумевает очень большие затраты времени и средств по сравнению с изменением программного обеспечения маршрутизатора. Поэтому быстродействующие маршрутизаторы переносят в БИС только несколько базовых протоколов сетевого уровня, чаще всего IP и IPX, делая такие маршрутизаторы узко специализированными.

Маршрутизация потоков

Еще один тип коммутаторов 3-го уровня – это коммутаторы, которые ускоряют процесс маршрутизации за счет выявления устойчивых потоков в сети и обработки по схеме маршрутизации только нескольких первых пакетов потока. Многие фирмы разработали подобные схемы, однако до сих пор они являются нестандартными, хотя работа над стандартизацией этого подхода идет в рамках одной из рабочих групп IETF. Существуют компании, которые считают эти попытки ошибочными, вносящими ненужную путаницу в и так непростую картину работы стека протоколов в сети. Наиболее известной компанией, занявшей такую позицию, является компания Nortel Networks, маршрутизаторы которой Accelar 1200 работают по классической схеме. Тем не менее количество компаний, разработавших протоколы ускоренной маршрутизации, в основном ускоренной IP-маршрутизации, довольно велико, туда входят такие известные компании, как 3Com, Cisco, Cabletron, Digital, Ipsilon.

Поток – это последовательность пакетов, имеющих некоторые общие свойства, по меньшей мере у них должны совпадать адрес отправителя и адрес получателя, и тогда их можно отправлять по одному и

тому же маршруту. Желательно, чтобы пакеты потока имели одно и то же требование к качеству обслуживания.

Ввиду разнообразия предложенных схем опишем только основную идею, лежащую в их основе.

Если бы все коммутаторы/маршрутизаторы, изображенные на рис. 5.31, работали по классической схеме, то каждый пакет, отправляемый из рабочей станции, принадлежащей одной IP-сети, серверу, принадлежащему другой IP-сети, проходил бы через блоки маршрутизации всех трех устройств.

В схеме ускоренной маршрутизации такую обработку проходит только несколько первых пакетов долговременного потока, то есть классическая схема работает до тех пор, пока долговременный поток не будет выявлен.

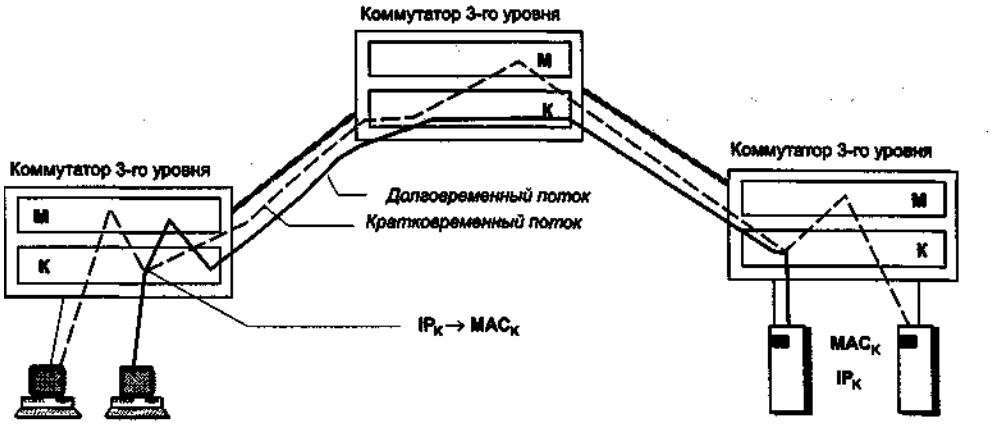


Рис. 5.31. Ускоренная маршрутизация потока пакетов

После этого первый коммутатор на пути следования потока выполняет нестандартную обработку пакета – он помещает в кадр канального протокола, например Ethernet, не MAC-адрес порта следующего маршрутизатора, а MAC-адрес узла назначения, который на рисунке обозначен как MAC_K. Как только эта замена произведена, кадр с таким MAC-адресом перестает поступать на блоки маршрутизации второго и третьего коммутатора/маршрутизатора, а проходит только через блоки коммутации этих устройств. Процесс передачи пакетов действительно ускоряется, так как они не проходят многократно повторяющиеся этапы маршрутизации. В то же время защитные свойства маршрутизаторы сохраняют, так как первые пакеты проверяются на допустимость передачи в сеть назначения, поэтому сохраняются фильтрация широковещательного шторма, защита от несанкционированного доступа и другие преимущества сети, разделенной на подсети.

Для реализации описанной схемы нужно решить несколько проблем. Первая – на основании каких признаков определяется долговременный поток. Это достаточно легкая проблема, и основные подходы к ее решению очевидны – совпадение адресов и портов соединения, общие признаки качества обслуживания, некоторый порог одинаковых пакетов для фиксации долговременности. Вторая проблема более серьезная. На основании какой информации первый маршрутизатор узнает MAC-адрес узла назначения. Этот узел находится за пределами непосредственно подключенных к первому маршрутизатору сетей, поэтому использование протокола ARP здесь не поможет. Именно здесь расходятся пути большинства фирменных технологий ускоренной маршрутизации. Многие компании разработали собственные служебные протоколы, с помощью которых маршрутизаторы запрашивают этот MAC-адрес друг у друга, пока последний на пути маршрутизатор не выяснит его с помощью протокола ARP.

Фирменные протоколы используют как распределенный подход, когда все маршрутизаторы равны в решении проблемы нахождения MAC-адреса, так и централизованный, когда в сети существует выделенный маршрутизатор, который помогает ее решить для всех.

Примерами коммутаторов 3-го уровня, работающими по схеме ускоренной IP-маршрутизации, являются коммутаторы SmartSwitch компании Cabletron, а также коммутатор Catalyst 5000 компании Cisco, выполняющий свои функций совместно с маршрутизаторами Cisco 7500 по технологии Cisco NetFlow для распознавания потоков и определения их адресной информации,и ряд других.

Выводы

- Типичный маршрутизатор представляет собой сложный специализированный компьютер, который работает под управлением специализированной операционной системы, оптимизированной для выполнения операций построения таблиц маршрутизации и продвижения пакетов на их основе.
- Маршрутизатор часто строится по мультипроцессорной схеме, причем используется симметричное мультипроцессирование, асимметричное мультипроцессирование и их сочетание. Наиболее рутинные операции обработки пакетов выполняются программно специализированными процессорами или аппаратно большими интегральными схемами (БИС/ASIC). Более высокоуровневые действия выполняют программно универсальные процессоры.
- По областям применения маршрутизаторы делятся на: магистральные маршрутизаторы, маршрутизаторы региональных подразделений, маршрутизаторы удаленных офисов и маршрутизаторы локальных сетей – коммутаторы 3-го уровня.
- Основными характеристиками маршрутизаторов являются: общая производительность в пакетах в секунду, набор поддерживаемых сетевых протоколов и протоколов маршрутизации, набор поддерживаемых сетевых интерфейсов глобальных и локальных сетей.
- К числу дополнительных функций маршрутизатора относится одновременная поддержка сразу нескольких сетевых протоколов и нескольких протоколов маршрутизации, возможность приоритетной обработки трафика, разделение функций построения таблиц маршрутизации и продвижения пакетов между маршрутизаторами разного класса на основе готовых таблиц маршрутизации.
- Основной особенностью коммутаторов 3-го уровня является высокая скорость выполнения операций маршрутизации за счет их перенесения на аппаратный уровень – уровень БИС/ASIC.
- Многие фирмы разработали собственные протоколы ускоренной маршрутизации долговременных потоков в локальных сетях, которые маршрутизируют только несколько первых пакетов потока, а остальные пакеты коммутируют на основе MAC-адресов.
- Корпоративные многофункциональные концентраторы представляют собой устройства, в которых на общей внутренней шине объединяются модули разного типа – повторители, мосты, коммутаторы и маршрутизаторы. Такое объединение дает возможность программного конфигурирования сети с определением состава подсетей и сегментов вне зависимости от их физического подключения к тому или иному порту устройства.

Вопросы и упражнения

1. В чем состоит отличие задач, решаемых протоколами сетевого уровня в локальных и глобальных сетях?
2. Сравните таблицу моста/коммутатора с таблицей маршрутизатора. Каким образом они формируются? Какую информацию содержат? От чего зависит их объем?
3. Таблица маршрутизации содержит записи о сетях назначения. Должна ли она содержать записи обо всех сетях составной сети или только о некоторых? Если только о некоторых, то о каких именно?
4. Может ли в таблице маршрутизации иметься несколько записей о маршрутизаторах по умолчанию?
5. На рис. 5.32 изображен компьютер с двумя сетевыми адаптерами, к которым подсоединены сегменты сети. Компьютер работает под управлением Windows NT. Может ли компьютер A обмениваться данными с компьютером B?
   - (A) Да, всегда.
   - (B) Нет, всегда.
   - (C) Все зависит от того, как сконфигурирована система Windows NT.

   Может ли повлиять на ответ тот факт, что в сегментах используются разные канальные протоколы, например Ethernet и Token Ring?

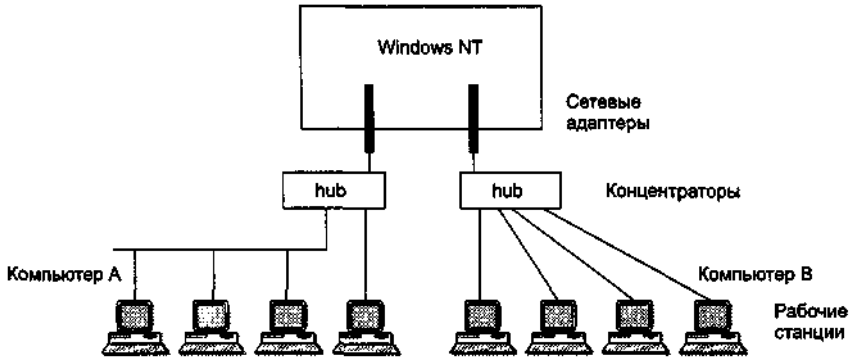


Рис. 5.32. Режимы работы компьютера с двумя сетевыми картами

6. Сколько уровней имеет стек протоколов TCP/IP? Каковы их функции? Какие особенности этого стека обусловливают его лидирующее положение в мире сетевых технологий?
7. Какие протоколы стека TCP/IP относятся к уровню Internet (уровню межсетевого взаимодействия)?
8. В чем проявляется ненадежность протокола IP?
9. Могут ли быть обнаружены ошибки на уровне Internet? Могут ли они быть исправлены средствами этого уровня?
10. В чем особенности реализации алгоритма скользящего окна в протоколе TCP?
11. В составных сетях используются три вида адресов: символьные, сетевые и локальные. Какие из приведенных ниже адресов могли бы в составной IP-сети являться локальными, а какие нет?
    - (A) 6-байтовый MAC-адрес (например, 12-B3-3B-51-A2-10);
    - (B) адрес X.25 (например, 25012112654987);
    - (C) 12-байтовый IPX-адрес(Наример, 13.34.B4.0A.C5.10.11.32.54.C5.3B.01);
    - (D) адрес VPI/VCI сети ATM.
12. Какие из следующих утверждений верны всегда?
    - (A) Каждый порт моста/коммутатора имеет MAC-адрес.
    - (B) Каждый мост/коммутатор имеет сетевой адрес.
    - (C) Каждый порт моста/коммутатора имеет сетевой адрес.
    - (D) Каждый маршрутизатор имеет сетевой адрес.
    - (E) Каждый порт маршрутизатора имеет MAC-адрес.
    - (F) Каждый порт маршрутизатора имеет сетевой адрес.
13. Какую долю всего множества IP-адресов составляют адреса класса A? Класса B? Класса C?
14. Какие из ниже приведенных адресов не могут быть использованы в качестве IP-адреса конечного узла сети, подключенной к Internet? Для синтаксически правильных адресов определите их класс: A, B, C, D или E.

(A) 127.0.0.1	(E) 10.234.17.25	(I) 193.256.1.16
(B) 201.13.123.245	(F) 154.12.255.255	(J) 194.87.45.0
(C) 226.4.37.105	(G) 13.13.13.13	(K) 195.34.116.255
(D) 103.24.254.0	(H) 204.0.3.1	(L) 161.23.45.305

15. Пусть IP-адрес некоторого узла подсети равен 198.65.12.67, а значение маски для этой подсети – 255.255.255.240. Определите номер подсети. Какое максимальное число узлов может быть в этой подсети?
16. Пусть поставщик услуг Internet имеет в своем распоряжении адрес сети класса B. Для адресации узлов своей собственной сети он использует 254 адреса. Определите максимально возможное число абонентов этого поставщика услуг, если размеры требуемых для них сетей соответствуют классу C? Какая маска должна быть установлена на маршрутизаторе поставщика услуг, соединяющем его сеть с сетями абонентов?
17. Какое максимальное количество подсетей теоретически возможно организовать, если в вашем распоряжении имеется сеть класса C? Какое значение должна при этом иметь маска?
18. Почему даже в тех случаях, когда используются маски, в IP-пакете маска не передается?
19. Какие преимущества дает технология CIDR? Что мешает ее широкому внедрению?
20. Имеется ли связь между длиной префикса пула IP-адресов и числом адресов, входящих в этот пул?

21. Почему в записи о маршрутизаторе по умолчанию в качестве адреса сети назначения указывается 0.0.0.0 с маской 0.0.0.0?
22. Отличается ли обработка поля MAC-адреса кадра маршрутизатором и коммутатором?
23. Сравните функции маршрутизаторов, которые поддерживают маршрутизацию от источника, с функциями маршрутизаторов, поддерживающих протоколы адаптивной маршрутизации.
24. Какие метрики расстояния могут быть использованы в алгоритмах сбора маршрутной информации?
25. Сравните интенсивность широковещательного трафика, порождаемого протоколами RIP и OSPF.
26. Какие элементы сети могут выполнять фрагментацию?
    - (A) только компьютеры;
    - (B) только маршрутизаторы;
    - (C) компьютеры, маршрутизаторы, мосты, коммутаторы;
    - (D) компьютеры и маршрутизаторы.
27. Что произойдет, если при передаче пакета он был фрагментирован и один из фрагментов не дошел до узла назначения после истечения тайм-аута?
    - (A) модуль IP узла-отправителя повторит передачу недошедшего фрагмента;
    - (B) модуль IP узла-отправителя повторит передачу всего пакета, в состав которого входил недошедший фрагмент;
    - (C) модуль IP узла-получателя отбросит все полученные фрагменты пакета, в котором потерялся один фрагмент; модуль IP узла-отправителя не будет предпринимать никаких действий по повторной передаче пакета данного пакета.
28. Какие особенности протоколов сетевого уровня стека Novell ограничивают их использование на глобальных линиях?
29. При образовании сетевого адреса в протоколе IPX в качестве номера узла используется MAC-адрес сетевого адаптера этого узла, а в протоколе IP номер узла назначается администратором произвольно. Какой, по вашему мнению, вариант является более эффективным и почему?
30. Каким образом должен быть сконфигурирован маршрутизатор, чтобы он предотвращал «широковещательный шторм»?
31. За счет чего коммутаторы третьего уровня ускоряют процесс маршрутизации?

ГЛАВА 6. ГЛОБАЛЬНЫЕ СЕТИ

Глобальные сети (*Wide Area Networks, WAN*), которые также называют территориальными компьютерными сетями, служат для того, чтобы предоставлять свои сервисы большому количеству конечных абонентов, разбросанных по большой территории – в пределах области, региона, страны, континента или всего земного шара. Ввиду большой протяженности каналов связи построение глобальной сети требует очень больших затрат, в которые входит стоимость кабелей и работ по их прокладке, затраты на коммутационное оборудование и промежуточную усилительную аппаратуру, обеспечивающую необходимую полосу пропускания канала, а также эксплуатационные затраты на постоянное поддержание в работоспособном состоянии разбросанной по большой территории аппаратуры сети.

Типичными абонентами глобальной компьютерной сети являются локальные сети предприятий, расположенные в разных городах и странах, которым нужно обмениваться данными между собой. Услугами глобальных сетей пользуются также и отдельные компьютеры. Крупные компьютеры класса мэйнфреймов обычно обеспечивают доступ к корпоративным данным, в то время как персональные компьютеры используются для доступа к корпоративным данным и публичным данным Internet.

Глобальные сети обычно создаются крупными телекоммуникационными компаниями для оказания платных услуг абонентам. Такие сети называют публичными или общественными. Существуют также такие понятия, как оператор сети и поставщик услуг сети. *Оператор* сети (*network operator*) – это та компания, которая поддерживает нормальную работу сети. *Поставщик услуг*, часто называемый также провайдером (*service provider*), – та компания, которая оказывает платные услуги абонентам сети. Владелец, оператор и поставщик услуг могут объединяться в одну компанию, а могут представлять и разные компании.

Гораздо реже глобальная сеть полностью создается какой-нибудь крупной корпорацией (такой,

например, как Dow Jones или «Транснефть») для своих внутренних нужд. В этом случае сеть называется частной. Очень часто встречается и промежуточный вариант – корпоративная сеть пользуется услугами или оборудованием общественной глобальной сети, но дополняет эти услуги или оборудование своими собственными. Наиболее типичным примером здесь является аренда каналов связи, на основе которых создаются собственные территориальные сети.

Кроме вычислительных глобальных сетей существуют и другие виды территориальных сетей передачи информации. В первую очередь это телефонные и телеграфные сети, работающие на протяжении многих десятков лет, а также телексная сеть.

Ввиду большой стоимости глобальных сетей существует долговременная тенденция создания единой глобальной сети, которая может передавать данные любых типов: компьютерные данные, телефонные разговоры, факсы, телеграммы, телевизионное изображение, телетекс (передача данных между двумя терминалами), видеотекс (получение хранящихся в сети данных на свой терминал) и т. д., и т. п. На сегодня существенного прогресса в этой области не достигнуто, хотя технологии для создания таких сетей начали разрабатываться достаточно давно – первая технология для интеграции телекоммуникационных услуг ISDN стала развиваться с начала 70-х годов. Пока каждый тип сети существует отдельно и наиболее тесная их интеграция достигнута в области использования общих первичных сетей – сетей PDH и SDH, с помощью которых сегодня создаются постоянные каналы в сетях с коммутацией абонентов. Тем не менее каждая из технологий, как компьютерных сетей, так и телефонных, старается сегодня передавать «чужой» для нее трафик с максимальной эффективностью, а попытки создать интегрированные сети на новом витке развития технологий продолжаются под преемственным названием Broadband ISDN (B-ISDN), то есть широкополосной (высокоскоростной) сети с интеграцией услуг. Сети B-ISDN будут основываться на технологии ATM, как универсальном транспорте, и поддерживать различные службы верхнего уровня для распространения конечным пользователям сети разнообразной информации – компьютерных данных, аудио- и видеоинформации, а также организации интерактивного взаимодействия пользователей.

6.1. Основные понятия и определения

Хотя в основе локальных и глобальных вычислительных сетей лежит один и тот же метод – метод коммутации пакетов, глобальные сети имеют достаточно много отличий от локальных сетей. Эти отличия касаются как принципов работы (например, принципы маршрутизации почти во всех типах глобальных сетей, кроме сетей TCP/IP, основаны на предварительном образовании виртуального канала), так и терминологии. Поэтому целесообразно изучение глобальных сетей начать с основных понятий и определений.

6.1.1. Обобщенная структура и функции глобальной сети

Транспортные функции глобальной сети

В идеале глобальная вычислительная сеть должна передавать данные абонентов любых типов, которые есть на предприятии и нуждаются в удаленном обмене информацией. Для этого глобальная сеть должна предоставлять комплекс услуг: передачу пакетов локальных сетей, передачу пакетов мини-компьютеров и мейнфреймов, обмен факсами, передачу трафика офисных АТС, выход в городские, междугородные и международные телефонные сети, обмен видеоизображениями для организации видеоконференций, передачу трафика кассовых аппаратов, банкоматов и т. д. и т. п. Основные типы потенциальных потребителей услуг глобальной компьютерной сети изображены на рис. 6.1.

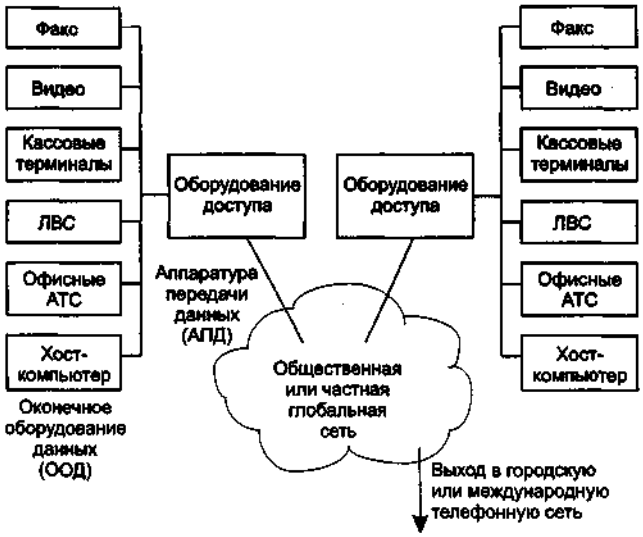


Рис. 6.1. Абоненты глобальной сети

Нужно подчеркнуть, что когда идет речь о передаче трафика офисных АТС, то имеется в виду обеспечение разговоров только между сотрудниками различных филиалов одного предприятия, а не замена городской, национальной или международной телефонной сети. Трафик внутренних телефонных разговоров имеет невысокую интенсивность и невысокие требования к качеству передачи голоса, поэтому многие компьютерные технологии глобальных сетей, например frame relay, справляются с такой упрощенной задачей.

Большинство территориальных компьютерных сетей в настоящее время обеспечивают только передачу компьютерных данных, но количество сетей, которые могут передавать остальные типы данных, постоянно растет.

ПРИМЕЧАНИЕ Отметим, что термин «передача данных» в территориальных сетях используется в узком смысле и означает передачу только компьютерных данных, а передачу речи и изображения обычно к передаче данных не относят.

Высокоуровневые услуги глобальных сетей

Из рассмотренного списка услуг, которые глобальная сеть предоставляет конечным пользователям, видно, что в основном она используется как транзитный транспортный механизм, предоставляющий только услуги трех нижних уровней модели OSI. Действительно, при построении корпоративной сети сами данные хранятся и вырабатываются в компьютерах, принадлежащих локальным сетям этого предприятия, а глобальная сеть их только переносит из одной локальной сети в другую. Поэтому в локальной сети реализуются все семь уровней модели OSI, включая прикладной, которые предоставляют доступ к данным, преобразуют их форму, организуют защиту информации от несанкционированного доступа.

Однако в последнее время функции глобальной сети, относящиеся к верхним уровням стека протоколов, стали играть заметную роль в вычислительных сетях. Это связано в первую очередь с популярностью информации, предоставляемой публично сетью Internet. Список высокоуровневых услуг, который предоставляет Internet, достаточно широк. Кроме доступа к гипертекстовой информации Web-узлов с большим количеством перекрестных ссылок, которые делают источником данных не отдельные компьютеры, а действительно всю глобальную сеть, здесь нужно отметить и широковещательное распространение звукозаписей, составляющее конкуренцию радиовещанию, организацию интерактивных «бесед» – chat, организацию конференций по интересам (служба News), поиск информации и ее доставку по индивидуальным заказам и многое другое.

Эти информационные (а не транспортные) услуги оказывают большое влияние не только на домашних пользователей, но и на работу сотрудников предприятий, которые пользуются профессиональной информацией, публикуемой другими предприятиями в Internet, в своей повседневной деятельности, общаются с коллегами с помощью конференций и chat, часто таким

образом достаточно быстро выясняя наболевшие нерешенные вопросы.

Информационные услуги Internet оказали влияние на традиционные способы доступа к разделяемым ресурсам, на протяжении многих лет применявшиеся в локальных сетях. Все больше корпоративной информации «для служебного пользования» распространяется среди сотрудников предприятия с помощью Web-службы, заменив многочисленные индивидуальные программные надстройки над базами данных, в больших количествах разрабатываемые на предприятиях. Появился специальный термин – *intranet*, который применяется в тех случаях, когда технологии Internet переносятся в корпоративную сеть. К технологиям intranet относят не только службу Web, но и использование Internet как глобальной транспортной сети, соединяющей локальные сети предприятия, а также все информационные технологии верхних уровней, появившиеся первоначально в Internet и поставленные на службу корпоративной сети.

В результате глобальные и локальные сети постепенно сближаются за счет взаимопроникновения технологий разных уровней – от транспортных до прикладных.

В данной книге основное внимание уделяется транспортным технологиям глобальных сетей, как основе любой высокоуровневой службы верхнего уровня. Кроме того, глобальные сети при построении корпоративных сетей в основном пока используются именно в этом качестве.

Структура глобальной сети

Типичный пример структуры глобальной компьютерной сети приведен на рис. 6.2. Здесь используются следующие обозначения: S (switch) – коммутаторы, К – компьютеры, R (router) – маршрутизаторы, MUX (multiplexor) – мультиплексор, UNI (User-Network Interface) – интерфейс пользователь - сеть и NNI (Network-Network Interface) – интерфейс сеть - сеть. Кроме того, офисная АТС обозначена аббревиатурой PBX, а маленькими черными квадратиками – устройства DCE, о которых будет рассказано ниже.

Сеть строится на основе некоммутируемых (выделенных) каналов связи, которые соединяют коммутаторы глобальной сети между собой. Коммутаторы называют также *центрами коммутации пакетов* (*ЦКП*), то есть они являются коммутаторами пакетов, которые в разных технологиях глобальных сетей могут иметь и другие названия – кадры, ячейки (cell). Как и в технологиях локальных сетей принципиальной разницы между этими единицами данных нет, однако в некоторых технологиях есть традиционные названия, которые к тому же часто отражают специфику обработки пакетов. Например, кадр технологии frame relay редко называют пакетом, поскольку од не инкапсулируется в кадр или пакет более низкого уровня и обрабатывается протоколом канального уровня.

Коммутаторы устанавливаются в тех географических пунктах, в которых требуется ответвление или слияние потоков данных конечных абонентов или магистральных каналов, переносящих данные многих абонентов. Естественно, выбор мест расположения коммутаторов определяется многими соображениями, в которые включается также возможность обслуживания коммутаторов квалифицированным персоналом, наличие выделенных каналов связи в данном пункте, надежность сети, определяемая избыточными связями между коммутаторами.

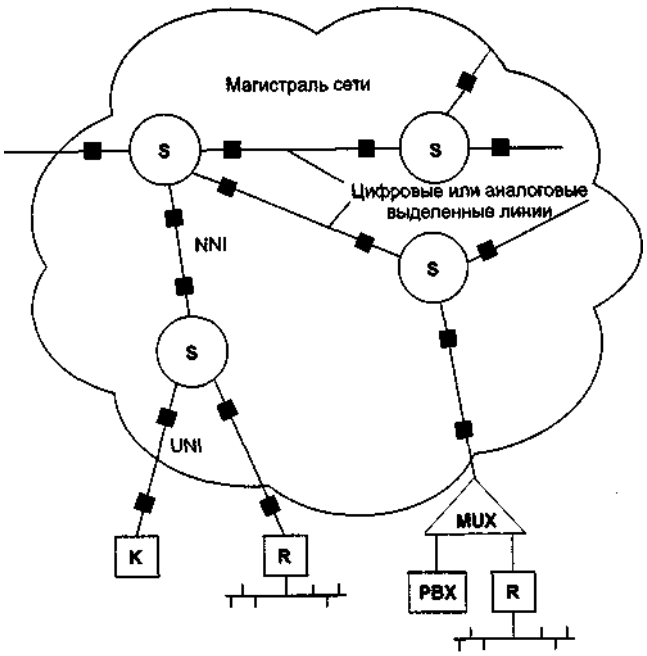


Рис. 6.2. Пример структуры глобальной сети

Абоненты сети подключаются к коммутаторам в общем случае также с помощью выделенных каналов связи. Эти каналы связи имеют более низкую пропускную способность, чем магистральные каналы, объединяющие коммутаторы, иначе сеть бы не справилась с потоками данных своих многочисленных пользователей. Для подключения конечных пользователей допускается использование коммутируемых каналов, то есть каналов телефонных сетей, хотя в таком случае качество транспортных услуг обычно ухудшается. Принципиально замена выделенного канала на коммутируемый ничего не меняет, но вносятся дополнительные задержки, отказы и разрывы канала по вине сети с коммутацией каналов, которая в таком случае становится промежуточным звеном между пользователем и сетью с коммутацией пакетов. Кроме того, в аналоговых телефонных сети канал обычно имеет низкое качество из-за высокого уровня шумов. Применение коммутируемых каналов на на магистральных связях коммутатор–коммутатор также возможно, но по тем же причинам весьма нежелательно.

В глобальной сети наличие большого количества абонентов с невысоким средним уровнем трафика весьма желательно – именно в этом случае начинают в наибольшей степени проявляться выгоды метода коммутации пакетов. Если же абонентов мало и каждый из них создает трафик большой интенсивности (по сравнению с возможностями каналов и коммутаторов сети), то равномерное распределение во времени пульсаций трафика становится маловероятным и для качественного обслуживания абонентов необходимо использовать сеть с низким коэффициентом нагрузки.

Конечные узлы глобальной сети более разнообразны, чем конечные узлы локальной сети. На рис. 6.2. показаны основные типы конечных узлов глобальной сети: отдельные компьютеры К, локальные сети, маршрутизаторы R и мультиплексоры MUX, которые используются для одновременной передачи по компьютерной сети данных и голоса (или изображения). Все эти устройства вырабатывают данные для передачи в глобальной сети, поэтому являются для нее устройствами типа DTE (Data Terminal Equipment). Локальная сеть отделена от глобальной маршрутизатором или удаленным мостом (который на рисунке не показан), поэтому для глобальной сети она представлена единым устройством DTE – портом маршрутизатора или моста.

При передаче данных через глобальную сеть *мосты и маршрутизаторы* работают в соответствии с той же логикой, что и при соединении локальных сетей. Мосты, которые в этом случае называются *удаленными мостами* (*remote bridges*), строят таблицу MAC-адресов на основании проходящего через них трафика, и по данным этой таблицы принимают решение – передавать кадры в удаленную сеть или нет. В отличие от своих локальных собратьев, удаленные мосты выпускаются и сегодня, привлекая сетевых интеграторов тем, что их не нужно конфигурировать, а в удаленных офисах, где нет квалифицированного обслуживающего персонала, это свойство оказывается очень полезным. Маршрутизаторы принимают решение на основании номера сети пакета какого-либо протокола сетевого уровня (например, IP или IPX) и, если пакет нужно переправить следующему маршрутизатору

по глобальной сети, например frame relay, упаковывают его в кадр этой сети, снабжают соответствующим аппаратным адресом следующего маршрутизатора и отправляют в глобальную сеть. *Мультиплексоры «голос-данные»* предназначены для совмещения в рамках одной территориальной сети компьютерного и голосового трафиков. Так как рассматриваемая глобальная сеть передает данные в виде пакетов, то мультиплексоры «голос-данные», работающие на сети данного типа, упаковывают голосовую информацию в кадры или пакеты территориальной сети и передают их ближайшему коммутатору точно так же, как и любой конечный узел глобальной сети, то есть мост или маршрутизатор. Если глобальная сеть поддерживает приоритезацию трафика, то кадрам голосового трафика мультиплексор присваивает наивысший приоритет, чтобы коммутаторы обрабатывали и продвигали их в первую очередь. Приемный узел на другом конце глобальной сети также должен быть мультиплексором «голос-данные», который должен понять, что за тип данных находится в пакете – замеры голоса или пакеты компьютерных данных, – и отсортировать эти данные по своим выходам. Голосовые данные направляются офисной АТС, а компьютерные данные поступают через маршрутизатор в локальную сеть. Часто модуль мультиплексора «голос-данные» встраивается в маршрутизатор. Для передачи голоса в наибольшей степени подходят технологии, работающие с предварительным резервированием полосы пропускания для соединения абонентов, – frame relay, ATM.

Так как конечные узлы глобальной сети должны передавать данные по каналу связи определенного стандарта, то каждое устройство типа DTE требуется оснастить устройством типа DCE (Data Circuit terminating Equipment), которое обеспечивает необходимый протокол физического уровня данного канала. В зависимости от типа канала для связи с каналами глобальных сетей используются DCE трех основных типов: модемы для работы по выделенным и коммутируемым аналоговым каналам, устройства DSU/CSU для работы по цифровым выделенным каналам сетей технологии TDM и терминальные адаптеры (TA) для работы по цифровым каналам сетей ISDN. Устройства DTE и DCE обобщенно называют оборудованием, размещаемым на территории абонента глобальной сети – Customer Premises Equipment, CPE.

Если предприятие не строит свою территориальную сеть, а пользуется услугами общественной, то внутренняя структура этой сети его не интересует. Для абонента общественной сети главное – это предоставляемые сетью услуги и четкое определение интерфейса взаимодействия с сетью, чтобы его оконечное оборудование и программное обеспечение корректно сопрягались с соответствующим оборудованием и программным обеспечением общественной сети.

Поэтому в глобальной сети обычно строго описан и стандартизован *интерфейс «пользователь-сеть»* (*User-to-Network Interface, UNI*). Это необходимо для того, чтобы пользователи могли без проблем подключаться к сети с помощью коммуникационного оборудования любого производителя, который соблюдает стандарт UNI данной технологии (например, X.25).

Протоколы взаимодействия коммутаторов внутри глобальной сети, называемые *интерфейсом «сеть-сеть»* (*Network-to-Network Interface, NNI*), стандартизуются не всегда. Считается, что организация, создающая глобальную сеть, должна иметь свободу действий, чтобы самостоятельно решать, как должны взаимодействовать внутренние узлы сети между собой. В связи с этим внутренний интерфейс, в случае его стандартизации, носит название «сеть-сеть», а не «коммутатор-коммутатор», подчеркивая тот факт, что он должен использоваться в основном при взаимодействии двух территориальных сетей различных операторов. Тем не менее если стандарт NNI принимается, то в соответствии с ним обычно организуется взаимодействие всех коммутаторов сети, а не только пограничных.

Интерфейсы DTE-DCE

Для подключения устройств DCE к аппаратуре, вырабатывающей данные для глобальной сети, то есть к устройствам DTE, существует несколько стандартных интерфейсов, которые представляют собой стандарты физического уровня. К этим стандартам относятся стандарты серии V CCITT, а также стандарты EIA серии RS (Recomended Standards). Две линии стандартов во многом дублируют одни и те же спецификации, но с некоторыми вариациями. Данные интерфейсы позволяют передавать данные со скоростями от 300 бит/с до нескольких мегабит в секунду на небольшие расстояния (15-20 м), достаточные для удобного размещения, например, маршрутизатора и модема.

Интерфейс RS-232C/V.24 является наиболее популярным низкоскоростным интерфейсом. Первоначально он был разработан для передачи данных между компьютером и модемом со скоростью не выше 9600 бит/с на расстояние до 15 метров. Позднее практические реализации этого интерфейса

стали работать и на более высоких скоростях – до 115 200 бит/с. Интерфейс поддерживает как асинхронный, так и синхронный режим работы. Особую популярность этот интерфейс получил после его реализации в персональных компьютерах (его поддерживают COM-порты), где он работает, как правило, только в асинхронном режиме и позволяет подключить к компьютеру не только коммуникационное устройство (такое, как модем), но и многие другие периферийные устройства – мышь, графопостроитель и т. д.

Интерфейс использует 25-контактный разъем или в упрощенном варианте – 9-контактный разъем (рис. 6.3).

DTE				DCE
Номер контакта	Назначение	Обозначение CCITT		
22	Индикатор вызова — RI	125	←	RI
20	Готовность терминала — DTR	108	→	DTR
16	Синхронизация передаваемых данных — TxClk (источник — DTE)	111	→	TxClk
15	Синхронизация передаваемых данных — TxClk (источник — DCE)	114	←	TxClk
17	Синхронизация принимаемых данных — RxClk	115	←	RxClk
8	Обнаружение несущей — CD	109	←	CD
7	Земля сигнала — SIG	102	—	SIG
6	Готовность DCE — DSR	107	←	
5	Готовность к передаче — CTS	106	←	CTS
4	Запрос передачи — RTS	105	→	RTS
3	Принимаемые данные DTE — RxD	104	←	RxD
2	Передаваемые данные DTE — TxD	103	→	TxD
1	Земля экрана — SHG	101	—	SHG

Рис. 6.3. Сигналы интерфейса RS-232C/V.24

Для обозначения сигнальных цепей используется нумерация CCITT, которая получила название «серия 100». Существуют также двухбуквенные обозначения EIA, которые на рисунке не показаны.

В интерфейсе реализован биполярный потенциальный код (+V, -V) на линиях между DTE и DCE. Обычно используется довольно высокий уровень сигнала: 12 или 15 В, чтобы более надежно распознавать сигнал на фоне шума.

При асинхронной передаче данных синхронизирующая информация содержится в самих кодах данных, поэтому сигналы синхронизации TxClk и RxClk отсутствуют. При синхронной передаче данных модем (DCE) передает на компьютер (DTE) сигналы синхронизации, без которых компьютер не может правильно интерпретировать потенциальный код, поступающий от модема по линии RxD. В случае когда используется код с несколькими состояниями (например, QAM), то один тактовый сигнал соответствует нескольким битам информации.

Нуль-модемный интерфейс характерен для прямой связи компьютеров на небольшом расстоянии с помощью интерфейса RS-232C/V.24. В этом случае необходимо применить специальный нуль-модемный кабель, так как каждый компьютер будет ожидать приема данных по линии RxD, что в случае применения модема будет корректно, но в случае прямого соединения компьютеров - нет. Кроме того, нуль-модемный кабель должен имитировать процесс соединения и разрыва через модемы, в котором используется несколько линий (RI, CB и т. д.). Поэтому для нормальной работы двух непосредственно соединенных компьютеров нуль-модемный кабель должен выполнять следующие соединения:

- RI-1+DSR-1 - DTR-2;
- DTR-1 - RI-2+DSR-2;
- CD-1 - CTS-2+RTS-2;
- CTS-1+RTS-1 - CD-2;
- RxD-l-TxD-2;
- TxD-l-RxD-2;
- SIG-l-SIG-1;
- SHG-l-SHG-2.

Знак «+» обозначает соединение соответствующих контактов на одной стороне кабеля.

Иногда при изготовлении нуль-модемного кабеля ограничиваются только перекрестным соединением линий приемника RxD и передатчика TxD, что для некоторого программного обеспечения бывает достаточно, но в общем случае может привести к некорректной работе программ, рассчитанных на реальные модемы.

Интерфейс RS-449/V.10/V.11 поддерживает более высокую скорость обмена данными и большую удаленность DCE от DTE. Этот интерфейс имеет две отдельные спецификации электрических сигналов. Спецификация RS-423/V.10 (аналогичные параметры имеет спецификация X.26) поддерживает скорость обмена до 100 000 бит/с на расстоянии до 10 м и скорость до 10 000 бит/с на расстоянии до 100 м. Спецификация RS-422/V.ll(X.27) поддерживает скорость до 10 Мбит/с на расстоянии до 10 м и скорость до 1 Мбит/с на расстоянии до 100 м. Как и RS-232C, интерфейс RS-449 поддерживает асинхронный и синхронный режимы обмена между DTE и DCE. Для соединения используется 37-контактный разъем.

Интерфейс V.35 был разработан для подключения синхронных модемов. Он обеспечивает только синхронный режим обмена между DTE и DCE на скорости до 168 Кбит/с. Для синхронизации обмена используются специальные тактирующие линии. Максимальное расстояние между DTE и DCE не превышает 15 м, как и в интерфейсе RS-232C.

Интерфейс X.21 разработан для синхронного обмена данными между DTE и DCE в сетях с коммутацией пакетов X.25. Это достаточно сложный интерфейс, который поддерживает процедуры установления соединения в сетях с коммутацией пакетов и каналов. Интерфейс был рассчитан на цифровые DCE. Для поддержки синхронных модемов была разработана версия интерфейса X.21 bis, которая имеет несколько вариантов спецификации электрических сигналов: RS-232C, V.10, V.11 и V.35.

Интерфейс «токовая петля 20 мА» используется для увеличения расстояния между DTE и DCE. Сигналом является не потенциал, а ток величиной 20 мА, протекающий в замкнутом контуре передатчика и приемника. Дуплексный обмен реализован на двух токовых петлях. Интерфейс работает только в асинхронном режиме. Расстояние между DTE и DCE может составлять несколько километров, а скорость передачи – до 20 Кбит/с.

Интерфейс HSSI (High-Speed Serial Interface) разработан для подключения к устройствам DCE, работающим на высокоскоростные каналы, такие как каналы Т3 (45 Мбит/с), SONET OC-1 (52 Мбит/с). Интерфейс работает в синхронном режиме и поддерживает передачу данных в диапазоне скоростей от 300 Кбит/с до 52 Мбит/с.

6.1.2. Типы глобальных сетей

Приведенная на рис. 6.2 глобальная вычислительная сеть работает в наиболее подходящем для компьютерного трафика режиме – режиме коммутации пакетов. Оптимальность этого режима для связи локальных сетей доказывают не только данные о суммарном трафике, передаваемом сетью в единицу времени, но и стоимость услуг такой территориальной сети. Обычно при равенстве предоставляемой скорости доступа сеть с коммутацией пакетов оказывается в 2-3 раза дешевле, чем сеть с коммутацией каналов, то есть публичная телефонная сеть.

Поэтому при создании корпоративной сети необходимо стремиться к построению или использованию услуг территориальной сети со структурой, подобной структуре, приведенной на рис. 6.2, то есть сети с территориально распределенными коммутаторами пакетов.

Однако часто такая вычислительная глобальная сеть по разным причинам оказывается недоступной в том или ином географическом пункте. В то же время гораздо более распространены и доступны услуги, предоставляемые телефонными сетями или первичными сетями, поддерживающими услуги выделенных каналов. Поэтому при построении корпоративной сети можно дополнить недостающие компоненты услугами и оборудованием, арендуемыми у владельцев первичной или телефонной сети.

В зависимости от того, какие компоненты приходится брать в аренду, принято различать корпоративные сети, построенные с использованием:

- выделенных каналов;
- коммутации каналов;
- коммутации пакетов.

Последний случай соответствует наиболее благоприятному случаю, когда сеть с коммутацией пакетов доступна во всех географических точках, которые нужно объединить в общую корпоративную сеть. Первые два случая требуют проведения дополнительных работ, чтобы на основании взятых в аренду средств построить сеть с коммутацией пакетов.

Выделенные каналы

Выделенные (или арендуемые – leased) каналы можно получить у телекоммуникационных компаний, которые владеют каналами дальней связи (таких, например, как «РОСТЕЛЕКОМ»), или от телефонных компаний, которые обычно сдают в аренду каналы в пределах города или региона.

Использовать выделенные линии можно двумя способами. Первый состоит в построении с их помощью территориальной сети определенной технологии, например frame relay, в которой арендуемые выделенные линии служат для соединения промежуточных, территориально распределенных коммутаторов пакетов, как в случае, приведенном на рис. 6.2.

Второй вариант – соединение выделенными линиями только объединяемых локальных сетей или конечных абонентов другого типа, например мэйнфреймов, без установки транзитных коммутаторов пакетов, работающих по технологии глобальной сети (рис. 6.4). Второй вариант является наиболее простым с технической точки зрения, так как основан на использовании маршрутизаторов или удаленных мостов в объединяемых локальных сетях и отсутствии протоколов глобальных технологий, таких как X.25 или frame relay. По глобальным каналам передаются те же пакеты сетевого или канального уровня, что и в локальных сетях.

Именно второй способ использования глобальных каналов получил специальное название «услуги выделенных каналов», так как в нем действительно больше ничего из технологий собственно глобальных сетей с коммутацией пакетов не используется.

Выделенные каналы очень активно применялись совсем в недалеком прошлом и применяются сегодня, особенно при построении ответственных магистральных связей между крупными локальными сетями, так как эта услуга гарантирует пропускную способность арендуемого канала. Однако при большом количестве географически удаленных точек и интенсивном смешанном трафике между ними использование этой службы приводит к высоким затратам за счет большого количества арендуемых каналов.

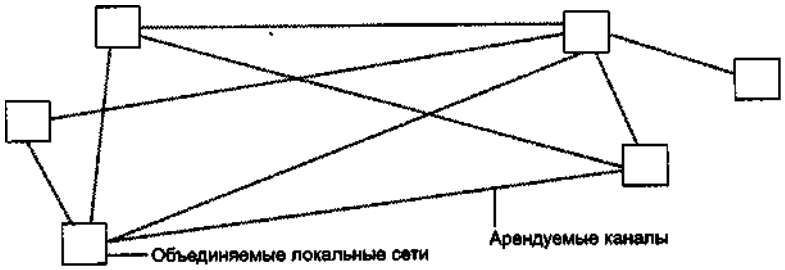


Рис. 6.4. Использование выделенных каналов

Сегодня существует большой выбор выделенных каналов – от аналоговых каналов тональной частоты с полосой пропускания 3,1 кГц до цифровых каналов технологии SDH с пропускной способностью 155 и 622 Мбит/с.

Глобальные сети с коммутацией каналов

Сегодня для построения глобальных связей в корпоративной сети доступны сети с коммутацией каналов двух типов – традиционные аналоговые телефонные сети и цифровые сети с интеграцией услуг ISDN. Достоинством сетей с коммутацией каналов является их распространенность, что характерно особенно для аналоговых телефонных сетей. В последнее время сети ISDN во многих странах также стали вполне доступны корпоративному пользователю, а в России это утверждение относится пока только к крупным городам.

Известным недостатком аналоговых телефонных сетей является низкое качество составного канала, которое объясняется использованием телефонных коммутаторов устаревших моделей, работающих по принципу частотного уплотнения каналов (FDM-технологии). На такие коммутаторы сильно воздействуют внешние помехи (например, грозовые разряды или работающие электродвигатели), которые трудно отличить от полезного сигнала. Правда, в аналоговых телефонных сетях все чаще используются цифровые АТС, которые между собой передают голос в цифровой форме. Аналоговым в таких сетях остается только абонентское окончание. Чем больше цифровых АТС в телефонной сети, тем

выше качество канала, однако до полного вытеснения АТС, работающих по принципу FDM-коммутации, в нашей стране еще далеко. Кроме качества каналов, аналоговые телефонные сети также обладают таким недостатком, как большое время установления соединения, особенно при импульсном способе набора номера, характерного для нашей страны.

Телефонные сети, полностью построенные на цифровых коммутаторах, и сети ISDN свободны от многих недостатков традиционных аналоговых телефонных сетей. Они предоставляют пользователям высококачественные линии связи, а время установления соединения в сетях ISDN существенно сокращено.

Однако даже при качественных каналах связи, которые могут обеспечить сети с коммутацией каналов, для построения корпоративных глобальных связей эти сети могут оказаться экономически неэффективными. Так как в таких сетях пользователи платят не за объем переданного трафика, а за время соединения, то при трафике с большими пульсациями и, соответственно, большими паузами между пакетами оплата идет во многом не за передачу, а за ее отсутствие. Это прямое следствие плохой приспособленности метода коммутации каналов для соединения компьютеров.

Тем не менее при подключении массовых абонентов к корпоративной сети, например сотрудников предприятия, работающих дома, телефонная сеть оказывается единственным подходящим видом глобальной службы из соображений доступности и стоимости (при небольшом времени связи удаленного сотрудника с корпоративной сетью).

Глобальные сети с коммутацией пакетов

В 80-е годы для надежного объединения локальных сетей и крупных компьютеров в корпоративную сеть использовалась практически одна технология глобальных сетей с коммутацией пакетов – X.25. Сегодня выбор стал гораздо шире, помимо сетей X.25 он включает такие технологии, как frame relay, SMDS и ATM. Кроме этих технологий, разработанных специально для глобальных компьютерных сетей, можно воспользоваться услугами территориальных сетей TCP/IP, которые доступны сегодня как в виде недорогой и очень распространенной сети Internet, качество транспортных услуг которой пока практически не регламентируется и оставляет желать лучшего, так и в виде коммерческих глобальных сетей TCP/IP, изолированных от Internet и предоставляемых в аренду телекоммуникационными компаниями.

В табл. 6.1 приводятся характеристики этих сетей, причем в графе «Трафик» указывается тип трафика, который наиболее подходит для данного типа сетей, а в графе «Скорость доступа» – наиболее типичный диапазон скоростей, предоставляемых поставщиками услуг этих сетей.

Таблица 6.1. Характеристики сетей с коммутацией пакетов

Тип сети	Скорость доступа	Трафик	Примечания
X.25	1,2-64 Кбит/с	Терминальный	Большая избыточность протоколов, хорошо работают на каналах низкого качества

Frame Relay	от 64 Кбит/с до 2 Мбит/с	Компьютерный	Сравнительно новые сети, хорошо передают пульсации трафика, в основном поддерживают службу постоянных виртуальных каналов
SMDS	1,544-45 Мбит/с	Компьютерный, графика, голос, видео	Сравнительно новые сети, распространены в крупных городах Америки, вытесняются сетями ATM
ATM	1,544-155 Мбит/с	Компьютерный, графика, голос, видео	Новые сети, коммерческая эксплуатация началась с 1996 года, пока используются в основном для передачи компьютерного трафика
TCP/IP	1,2-2,048 Кбит/с	Терминальный, компьютерный	Широко распространены в некоммерческом варианте – сети Internet, коммерческие услуги пока слабые

Принципы работы сетей TCP/IP уже были подробно рассмотрены в главе 5. Эти принципы остаются неизменными и при включении в состав этих сетей глобальных сетей различных технологий. Для остальных технологий, кроме SMDS, будут рассмотрены принципы доставки пакетов, пользовательский интерфейс и типы оборудования доступа к сетям данных технологий.

Технология SMDS (Switched Multi-megabit Data Service) была разработана в США для объединения локальных сетей в масштабах мегаполиса, а также предоставления высокоскоростного выхода в глобальные сети. Эта технология поддерживает скорости доступа до 45 Мбит/с и сегментирует кадры MAC-уровня в ячейки фиксированного размера 53 байт, имеющие, как и ячейки технологии ATM, поле данных в 48 байт. Технология SMDS основана на стандарте IEEE 802.6, который описывает несколько более широкий набор функций, чем SMDS. Стандарты SMDS приняты компанией Bellcore, но международного статуса не имеют. Сети SMDS были реализованы во многих крупных городах США, однако в других странах эта технология распространения не получила. Сегодня сети SMDS вытесняются сетями ATM, имеющими более широкие функциональные возможности, поэтому в данной книге технология SMDS подробно не рассматривается.

Магистральные сети и сети доступа

Целесообразно делить территориальные сети, используемые для построения корпоративной сети, на две большие категории:

- магистральные сети;
- сети доступа.

Магистральные территориальные сети (*backbone wide-area networks*) используются для образования одноранговых связей между крупными локальными сетями, принадлежащими большим подразделениям предприятия. Магистральные территориальные сети должны обеспечивать высокую пропускную способность, так как на магистрали объединяются потоки большого количества подсетей. Кроме того магистральные сети должны быть постоянно доступны, то есть обеспечивать очень высокий коэффициентом готовности, так как по ним передается трафик многих критически важных для успешной работы предприятия приложений (business-critical applications). Ввиду особой важности магистральных средств им может «прощаться» высокая стоимость. Так как у предприятия обычно имеется не так уж много крупных сетей, то к магистральным сетям не предъявляются требования поддержания разветвленной инфраструктуры доступа.

Обычно в качестве магистральных сетей используются цифровые выделенные каналы со скоростями

от 2 до 622 Мбит/с, по которым передается трафик IP, IPX или протоколов архитектуры SNA компании IBM, сети с коммутацией пакетов frame relay, ATM, X.25 или TCP/IP. При наличии выделенных каналов для обеспечения высокой готовности магистрали используется смешанная избыточная топология связей, как это показано на рис. 6.5.

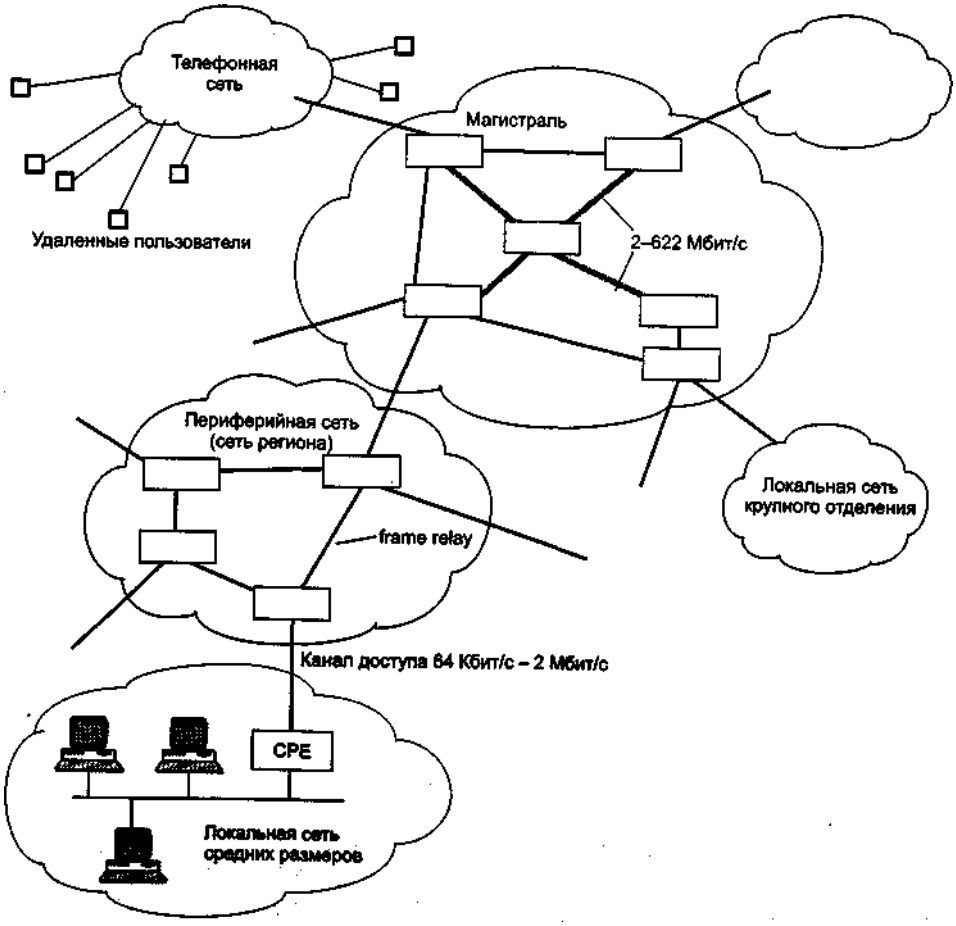


Рис. 6.5. Структура глобальной сети предприятия

Под *сетями доступа* понимаются территориальные сети, необходимые для связи небольших локальных сетей и отдельных удаленных компьютеров с центральной локальной сетью предприятия. Если организации магистральных связей при создании корпоративной сети всегда уделялось большое внимание, то организация удаленного доступа сотрудников предприятия перешла в разряд стратегически важных вопросов только в последнее время. Быстрый доступ к корпоративной информации из любой географической точки определяет для многих видов деятельности предприятия качество принятия решений его сотрудниками. Важность этого фактора растет с увеличением числа сотрудников, работающих на дому (telecommuters – телекоммьютеров), часто находящихся в командировках, и с ростом количества небольших филиалов предприятий, находящихся в различных городах и, может быть, разных странах.

В качестве отдельных удаленных узлов могут также выступать банкоматы или кассовые аппараты, требующие доступа к центральной базе данных для получения информации о легальных клиентах банка, пластиковые карточки которых необходимо авторизовать на месте. Банкоматы или кассовые аппараты обычно рассчитаны на взаимодействие с центральным компьютером по сети X.25, которая в свое время специально разрабатывалась как сеть для удаленного доступа неинтеллектуального терминального оборудования к центральному компьютеру.

К сетям доступа предъявляются требования, существенно отличающиеся от требований к магистральным сетям. Так как точек удаленного доступа у предприятия может быть очень много, одним из основных требований является наличие разветвленной инфраструктуры доступа, которая может использоваться сотрудниками предприятия как при работе дома, так и в командировках. Кроме того, стоимость удаленного доступа должна быть умеренной, чтобы экономически оправдать затраты на подключение десятков или сотен удаленных абонентов. При этом требования к пропускной

способности у отдельного компьютера или локальной сети, состоящей из двух-трех клиентов, обычно укладываются в диапазон нескольких десятков килобит в секунду (если такая скорость и не вполне удовлетворяет удаленного клиента, то обычно удобствами его работы жертвуют ради экономии средств предприятия).

В качестве сетей доступа обычно применяются телефонные аналоговые сети, сети ISDN и реже – сети frame relay. При подключении локальных сетей филиалов также используются выделенные каналы со скоростями от 19,2 до 64 Кбит/с. Качественный скачок в расширении возможностей удаленного доступа произошел в связи со стремительным ростом популярности и распространенности Internet. Транспортные услуги Internet дешевле, чем услуги междугородных и международных телефонных сетей, а их качество быстро улучшается.

Программные и аппаратные средства, которые обеспечивают подключение компьютеров или локальных сетей удаленных пользователей к корпоративной сети, называются *средствами удаленного доступа*. Обычно на клиентской стороне эти средства представлены модемом и соответствующим программным обеспечением.

Организацию массового удаленного доступа со стороны центральной локальной сети обеспечивает *сервер удаленного доступа* (*Remote Access Server, RAS*). Сервер удаленного доступа представляет собой программно-аппаратный комплекс, который совмещает функция маршрутизатора, моста и шлюза. Сервер выполняет ту или иную функцию в зависимости от типа протокола, по которому работает удаленный пользователь или удаленная сеть. Серверы удаленного доступа обычно имеют достаточно много низкоскоростных портов для подключения пользователей через аналоговые телефонные сети или ISDN.

Показанная на рис. 6.5. структура глобальной сети, используемой для объединения в корпоративную сеть отдельных локальных сетей и удаленных пользователей, достаточно типична. Она имеет ярко выраженную иерархию территориальных транспортных средств, включающую высокоскоростную магистраль (например, каналы SDH 155-622 Мбит/с), более медленные территориальные сети доступа для подключения локальных сетей средних размеров (например, frame relay) и телефонную сеть общего назначения для удаленного доступа сотрудников.

Выводы

- Глобальные компьютерные сети (WAN) используются для объединения абонентов разных типов: отдельных компьютеров разных классов – от мэйнфреймов до персональных компьютеров, локальных компьютерных сетей, удаленных терминалов.
- Ввиду большой стоимости инфраструктуры глобальной сети существует острая потребность передачи по одной сети всех типов трафика, которые возникают на предприятии, а не только компьютерного: голосового трафика внутренней телефонной сети, работающей на офисных АТС (PBX), трафика факс-аппаратов, видеокамер, кассовых аппаратов, банкоматов и другого производственного оборудования.
- Для поддержки мультимедийных видов трафика создаются специальные технологии: ISDN, B-ISDN. Кроме того, технологии глобальных сетей, которые разрабатывались для передачи исключительно компьютерного трафика, в последнее время адаптируются для передачи голоса и изображения. Для этого пакеты, переносящие замеры голоса или данные изображения, приоритезируются, а в тех технологиях, которые это допускают, для их переноса создается соединение с заранее резервируемой пропускной способностью. Имеются специальные устройства доступа – мультиплексоры «голос-данные» или «видео-данные», которые упаковывают мультимедийную информацию в пакеты и отправляют ее по сети, а на приемном конце распаковывают и преобразуют в исходную форму – голос или видеоизображение.
- Глобальные сети предоставляют в основном транспортные услуги, транзитом перенося данные между локальными сетями или компьютерами. Существует нарастающая тенденция поддержки служб прикладного уровня для абонентов глобальной сети: распространение публично-доступной аудио-, видео- и текстовой информации, а также организация интерактивного взаимодействия абонентов сети в реальном масштабе времени. Эти службы появились в Internet и успешно переносятся в корпоративные сети, что называется технологией intranet.

- Все устройства, используемые для подключения абонентов к глобальной сети, делятся, на два класса: DTE, собственно вырабатывающие данные, и DCE, служащие для передачи данных в соответствии с требованиями интерфейса глобального канала и заверяшющие канал.
- Технологии глобальных сетей определяют два типа интерфейса: «пользователь-сеть» (UNI) и «сеть-сеть» (NNI). Интерфейс UNI всегда глубоко детализирован для обеспечения подключения к сети оборудования доступа от разных производителей. Интерфейс NNI может быть детализирован не так подробно, так как взаимодействие крупных сетей может обеспечиваться на индивидуальной основе.
- Глобальные компьютерные сети работают на основе технологии коммутации пакетов, кадров и ячеек. Чаще всего глобальная компьютерная сеть принадлежит телекоммуникационной компании, которая предоставляет службы своей сети в аренду. При отсутствии такой сети в нужном регионе предприятия самостоятельно создают глобальные сети, арендуя выделенные или коммутируемые каналы у телекоммуникационных или телефонных компаний.
- На арендованных каналах можно построить сеть с промежуточной коммутацией на основе какой-либо технологии глобальной сети (X.25, frame relay, ATM) или же соединять арендованными каналами непосредственно маршрутизаторы или мосты локальных сетей. Выбор способа использования арендованных каналов зависит от количества и топологии связей между локальными сетями.
- Глобальные сети делятся на магистральные сети и сети доступа.

6.2. Глобальные связи на основе выделенных линий

Выделенный канал – это канал с фиксированной полосой пропускания или фиксированной пропускной способностью, постоянно соединяющий двух абонентов. Абонентами могут быть как отдельные устройства (компьютеры или терминалы), так и целые сети.

Выделенные каналы обычно арендуются у компаний – операторов территориальных сетей, хотя крупные корпорации могут прокладывать свои собственные выделенные каналы.

Выделенные каналы делятся на аналоговые и цифровые в зависимости от того, какого типа коммутационная аппаратура применена для постоянной коммутации абонентов – FDM или TDM. На аналоговых выделенных линиях для аппаратуры передачи данных физический и канальный протоколы жестко не определены. Отсутствие физического протокола приводит к тому, что пропускная способность аналоговых каналов зависит от пропускной способности модемов, которые использует пользователь канала. Модем собственно и устанавливает нужный ему протокол физического уровня для канала.

На цифровых выделенных линиях протокол физического уровня зафиксирован – он задан стандартом G.703.

На канальном уровне аналоговых и цифровых выделенных каналов обычно используется один из протоколов семейства HDLC или же более поздний протокол PPP, построенный на основе HDLC для связи многопротокольных сетей.

6.2.1. Аналоговые выделенные линии

Типы аналоговых выделенных линий

Выделенные аналоговые каналы предоставляются пользователю с 4-проводным или 2-проводным окончанием. На каналах с 4-проводным окончанием организация полнодуплексной связи, естественно, выполняется более простыми способами.

Выделенные линии могут быть разделены на две группы по другому признаку – наличию промежуточной аппаратуры коммутации и усиления или ее отсутствию.

Первую группу составляют так называемые нагруженные линии, проходящие через оборудование частотного уплотнения (FDM-коммутаторы и мультиплексоры), расположенное, например, на АТС. Телефонные компании обычно предоставляют в аренду два типа выделенных каналов: канал тональной частоты с полосой пропускания 3,1 кГц и широкополосный канал с полосой 48 кГц, который представляет собой *базовую группу* из 12 каналов тональной частоты. Широкополосный канал имеет границы полосы пропускания от 60 до 108 кГц. Так как широкополосный канал используется для связи

АТС между собой, то получение его в аренду более проблематично, чем канала тональной частоты.

Выделенные нагруженные каналы также классифицируются на категории в зависимости от их качества. От категории качества зависит и арендная месячная плата за канал.

Вторая группа выделенных линий – это ненагруженные физические проводные линии. Они могут кроссироваться, но при этом не проходят через аппаратуру частотного уплотнения. Часто такие линии используются для связи между близко стоящими зданиями. Разветвленные сети каналов, представляющих собой ненагруженные линии, используются, например, муниципальными службами (энергонадзора, водопровода, пожарной охраны и др.) для передачи технологической информации. При небольшой длине ненагруженной выделенной линии она обладает достаточно широкой полосой пропускания, иногда до 1 МГц, что позволяет передавать импульсные немодулированные сигналы. На первый взгляд может показаться, что ненагруженные линии не имеют отношения к глобальным сетям, так как их можно использовать при протяженности максимум в несколько километров, иначе затухание становится слишком большим для передачи данных. Однако в последнее время именно этот вид выделенных каналов привлекает пристальное внимание разработчиков средств удаленного доступа. Дело в том, что телефонные абонентские окончания – отрезок витой пары от АТС до жилого или производственного здания – представляют собой именно такой вид каналов. Широкая (хотя и заранее точно неизвестная) полоса пропускания этих каналов позволяет развить на коротком отрезке линии высокую скорость – до нескольких мегабит в секунду. В связи с этим до ближайшей АТС данные от удаленного компьютера или сети можно передавать гораздо быстрее, чем по каналам тональной частоты, которые начинаются в данной АТС. Использование выделенных ненагруженных каналов подробно рассматривается в разделе 6.5, посвященном удаленному доступу.

Модемы для работы на выделенных каналах

Для передачи данных по выделенным нагруженным аналоговым линиям используются модемы, работающие на основе методов аналоговой модуляции сигнала, рассмотренных в главе 2. Протоколы и стандарты модемов определены в рекомендациях CCITT серии V. Эти стандарты делятся на три группы:

- стандарты, определяющие скорость передачи данных и метод кодирования;
- стандарты исправления ошибок;
- стандарты сжатия данных.

Эти стандарты определяют работу модемов как для выделенных, так и коммутируемых линий. Модемы можно также классифицировать в зависимости от того, какой режимы работы они поддерживают (асинхронный, синхронный или оба этих режима), а также к какому окончанию (4-проводному или 2-проводному) они подключены.

В отношении режима работы модемы делятся на три группы:

- модемы, поддерживающие только асинхронный режим работы;
- модемы поддерживающие асинхронный и синхронный режимы работы;
- модемы, поддерживающие только синхронный режим работы.

Модемы, работающие *только в асинхронном режиме*, обычно поддерживают низкую скорость передачи данных – до 1200 бит/с. Так, модемы, работающие по стандарту V.23, могут обеспечивать скорость 1200 бит/с на 4-проводной выделенной линии в дуплексном асинхронном режиме, а по стандарту V.21 – на скорости 300 бит/с по 2-проводной выделенной линии также в дуплексном асинхронном режиме. Дуплексный режим на 2-проводном окончании обеспечивается частотным разделением канала. Асинхронные модемы представляют наиболее дешевый вид модемов, так как им не требуются высокоточные схемы синхронизации сигналов на кварцевых генераторах. Кроме того, асинхронный режим работы неприхотлив к качеству линии.

Модемы, работающие *только в синхронном режиме*, могут подключаться только к 4-проводному окончанию. Синхронные модемы используют для выделения сигнала высокоточные схемы синхронизации и поэтому обычно значительно дороже асинхронных модемов. Кроме того, синхронный режим работы предъявляет высокие требования к качеству линии.

Для выделенного канала тональной частоты с 4-проводным окончанием разработано достаточно много стандартов серии V. Все они поддерживают дуплексный режим:

- V.26 – скорость передачи 2400 бит/с;
- V.27 – скорость передачи 4800 бит/с;

- V.29 – скорость передачи 9600 бит/с;
- V.32 ter – скорость передачи 19 200 бит/с.

Для выделенного широкополосного канала 60-108 кГц существуют три стандарта:

- V.35 – скорость передачи 48 Кбит/с;
- V.36 – скорость передачи 48-72 Кбит/с;
- V.37 –скорость передачи 96-168 Кбит/с.

Коррекция ошибок в синхронном режиме работы обычно реализуется по протоколу HDLC, но допустимы и устаревшие протоколы SDLC и BSC компании IBM. Модемы стандартов V.35, V.36 и V.37 используют для-евяаи с DTE интерфейс V.35.

Модемы, *работающие в асинхронном и синхронном режимах*, являются наиболее универсальными устройствами. Чаще всего они могут работать как по выделенным, так и по коммутируемым каналам, обеспечивая дуплексный режим работы. На выделенных каналах они поддерживают в основном 2-проводное окончание и гораздо реже – 4-проводное.

Для асинхронно-синхронных модемов разработан рад стандартов серии V:

- V.22 – скорость передачи до 1200 бит/с;
- V.22 bis – скорость передачи до 2400 бит/с;
- V.26 ter – скорость передачи до 2400 бит/с;
- V.32 – скорость передачи до 9600 бит/с;
- V.32 bis – скорость передачи 14 400 бит/с;
- V.34 – скорость передачи до 28,8 Кбит/с;
- V.34+ – скорость передачи до 33,6 Кбит/с.

Стандарт V.34, принятый летом 1994 года, знаменует новый подход к передаче данных по каналу тональной частоты. Этот стандарт разрабатывался CCITT довольно долго – с 1990 года. Большой вклад в его разработку внесла компания Motorola, которая является одним из признанных лидеров этой отрасли. Стандарт V.34 разрабатывался для передачи информации по каналам практически любого качества. Особенностью стандарта являются процедуры динамической адаптации к изменениям характеристик канала во время обмена информацией. Адаптация осуществляется в ходе сеанса связи – без прекращения и без разрыва установленного соединения.

Основное отличие V.34 от предшествующих стандартов заключается в том, что в нем определено 10 процедур, по которым модем после тестирования линии выбирает свои основные параметры: несущую и полосу пропускания (выбор проводится из 11 комбинаций), фильтры передатчика, оптимальный уровень передачи и другие. Первоначальное соединение модемов проводится по стандарту V.21 на минимальной скорости 300 бит/с, что позволяет работать на самых плохих линиях. Для кодирования данных используются избыточные коды квадратурной амплитудной модуляции QAM. Применение адаптивных процедур сразу позволило поднять скорость передачи данных более чем в 2 раза по сравнению с предыдущим стандартом – V.32 bis.

Принципы адаптивной настройки к параметрам линии были развиты в стандарте V.34+, который является усовершенствованным вариантом стандарта V.34. Стандарт V.34+ позволил несколько повысить скорость передачи данных за счет усовершенствования метода кодирования. Один передаваемый кодовый символ несет в новом стандарте в среднем не 8,4 бита, как в протоколе V.34, а 9,8. При максимальной скорости передачи кодовых символов в 3429 бод (это ограничение преодолеть нельзя, так как оно определяется полосой пропускания канала тональной частоты) усовершенствованный метод кодирования дает скорость передачи данных в 33,6 Кбит/с (3429 x 9,8 = 33604). Правда, специалисты отмечают, что даже в Америке только 30 % телефонных линий смогут обеспечить такой низкий уровень помех, чтобы модемы V.34+ смогли работать на максимальной скорости. Тем не менее модемы стандарта V.34+ имеют преимущества по сравнению с модемами V.34 даже на зашумленных линиях – они лучше «держат» связь, чем модемы V.34.

Протоколы V.34 и V.34+ позволяют работать на 2-проводной выделенной линии в дуплексном режиме. Дуплексный режим передачи в стандартах V.32, V.34, V.34+ обеспечивается не с помощью частотного разделения канала, а с помощью одновременной передачи данных в обоих направлениях. Принимаемый сигнал определяется вычитанием с помощью сигнальных процессоров (DSP) передаваемого сигнала из общего сигнала в канале. Для этой операции используются также процедуры эхо-подавления, так как передаваемый сигнал, отражаясь от ближнего и дальнего концов канала, вносит искажения в общий сигнал (метод передачи данных, описанный в проекте стандарта 802.3ab, определяющего работу технологии Gigabit Ethernet на витой паре категории 5, взял многое из

стандартов V.32-V.34+).

На высокой скорости модемы V.32-V.34+ фактически всегда используют в канале связи синхронный режим. При этом они могут работать с DTE как по асинхронному интерфейсу, так и по синхронному. В первом случае модем преобразует асинхронные данные в синхронные.

Модемы различаются не только поддерживаемыми протоколами, но и определенной ориентацией на область применения. Различают профессиональные модемы, которые предназначены для работы в модемных пулах корпоративных сетей, и модемы для применения в небольших офисах или на дому.

Профессиональные модемы отличаются высокой надежностью, способностью устойчиво работать в непрерывном режиме и поддержкой средств удаленного централизованного управления. Обычно система управления модемными стойками поставляется отдельно и оправдывает себя в условиях большого предприятия. Стандарт V.34 выделяет в общей полосе пропускания линии отдельную полосу для управления модемом по тому же каналу, по которому передаются и пользовательские данные.

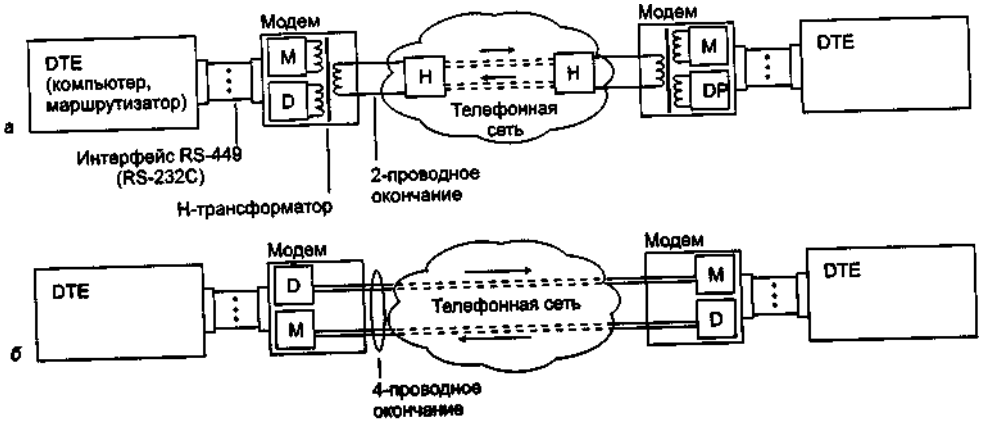


Рис. 6.6. Соединение локальных сетей или компьютеров по выделенному каналу

Типовая структура соединения двух компьютеров или локальных сетей через маршрутизатор с помощью выделенной аналоговой линии приведена на рис. 6.6. В случае 2-проводного окончания (см. рис. 6.6, *а*) для обеспечения дуплексного режима модем использует трансформаторную развязку. Телефонная сеть благодаря своей схеме развязки обеспечивает разъединение потоков данных, циркулирующих в разных направлениях. При наличии 4-проводного окончания (см. рис. 6.6, б) схема модема упрощается.

6.2.2. Цифровые выделенные линии

Цифровые выделенные линии образуются путем постоянной коммутации в первичных сетях, построенных на базе коммутационной аппаратуры, работающей на принципах разделения канала во времени – TDM, описанного в главе 2. Существуют два поколения технологий цифровых первичных сетей – технология плезио-хронной («плезио» означает «почти», то есть почти синхронной) цифровой иерархии (Plesiochronic Digital Hierarchy, PDH) и более поздняя технология – синхронная цифровая иерархия (Synchronous Digital Hierarchy, SDH). В Америке технологии SDH соответствует стандарт SONET.

Технология плезиохронной цифровой иерархии PDH

Цифровая аппаратура мультиплексирования и коммутации была разработана в конце 60-х годов компанией AT&T для решения проблемы связи крупных коммутаторов телефонных сетей между собой. Каналы с частотным уплотнением, применяемые до этого на участках АТС-АТС, исчерпали свои возможности по организации высокоскоростной многоканальной связи по одному кабелю. В технологии FDM для одновременной передачи данных 12 или 60 абонентских каналов использовалась витая пара, а для повышения скорости связи приходилось прокладывать кабели с большим количеством пар проводов или более дорогие коаксиальные кабели. Кроме того, метод частотного уплотнения высоко чувствителен к различного рода помехам, которые всегда присутствуют в территориальных кабелях, да и высокочастотная несущая речи сама создает помехи в приемной аппаратуре, будучи плохо

отфильтрована.

Для решения этой задачи была разработана аппаратура Т1, которая позволяла в цифровом виде мультиплексировать, передавать и коммутировать (на постоянной основе) данные 24 абонентов. Так как абоненты по-прежнему пользовались обычными телефонными аппаратами, то есть передача голоса шла в аналоговой форме, то мультиплексоры Т1 сами осуществляли оцифровывание голоса с частотой 8000 Гц и кодировали голос с помощью импульсно-кодовой модуляции (Pulse Code Modulation, PCM). В результате каждый абонентский канал образовывал цифровой поток данных 64 Кбит/с. Для соединения магистральных АТС каналы Т1 представляли собой слишком слабые средства мультиплексирования, поэтому в технологии была реализована идея образования каналов *с иерархией скоростей*. Четыре канала типа Т1 объединяются в канал следующего уровня цифровой иерархии – Т2, передающий данные со скоростью 6,312 Мбит/с, а семь каналов Т2 дают при объединении канал ТЗ, передающий данные со скоростью 44,736 Мбит/с. Аппаратура Т1, Т2 и ТЗ может взаимодействовать между собой, образуя иерархическую сеть с магистральными и периферийными каналами трех уровней скоростей.

С середины 70-х годов выделенные каналы, построенные на аппаратуре Т1, стали сдаваться телефонными компаниями в аренду на коммерческих условиях, перестав быть внутренней технологией этих компаний. Сети Т1, а также более скоростные сети Т2 и ТЗ позволяют передавать не только голос, но и любые данные, представленные в цифровой форме, – компьютерные данные, телевизионное изображение, факсы и т. п.

Технология цифровой иерархии была позже стандартизована CCITT. При этом в нее были внесены некоторые изменения, что привело к несовместимости американской и международной версий цифровых сетей. Американская версия распространена сегодня кроме США также в Канаде и Японии (с некоторыми различиями), а в Европе применяется международный стандарт. Аналогом каналов Т в международном стандарте являются каналы типа El, Е2 и Е3 с другими скоростями – соответственно 2,048 Мбит/с, 8,488 Мбит/с и 34,368 Мбит/с. Американский вариант технологии также был стандартизован ANSI.

Несмотря на различия американской и международных версий технологии цифровой иерархии, для обозначения иерархии скоростей принято использовать одни и те же обозначения – DSn (Digital Signal n). В табл. 6.2 приводятся значения для всех введенных стандартами уровней скоростей обеих технологий.

Таблица 6.2, Иерархия цифровых скоростей

Америка				CCITT (Европа)		
Обозначение скорости	Количество голосовых каналов	Количество каналов предыдущего уровня	Скорость, Мбит/с го	Количество голосовых каналов	Количество каналов предыдущего уровня	Скорость, Мбит/с
DS-0	1	1	64 Кбит/с	1	1	64 Кбит/с
DS-1	24	24	1,544	30	30	2,048
DS-2	96	4	6,312	120	4	8,488
DS-3	672	7	44,736	480	4	34,368
DS-4	4032	6	274,176	1920	4	139,264

На практике в основном используются каналы Т1/Е1 и ТЗ/Е3.

Мультиплексор Т1 обеспечивает передачу данных 24-х абонентов со скоростью 1,544 Мбит/с в кадре, имеющем достаточно простой формат. В этом кадре последовательно передается по одному байту каждого абонента, а после 24-х байт вставляется один бит синхронизации. Первоначально устройства Т1 (которые дали имя также и всей технологии, работающей на скорости 1,544 Мбит/с) работали только на внутренних тактовых генераторах, и каждый кадр с помощью битов синхронизации мог передаваться асинхронно. Аппаратура Т1, а также более скоростная аппаратура Т2 и ТЗ за долгие годы существования претерпела значительные изменения. Сегодня мультиплексоры и коммутаторы первичной сети работают на централизованной тактовой частоте, распределяемой из одной точки всей сети. Однако принцип формирования кадра остался, поэтому биты синхронизации в кадре по-прежнему присутствуют. Суммарная скорость пользовательских каналов составляет 24 х 64 = 1,536 Мбит/с, а еще

8 Кбит/с добавляют биты синхронизации.

В аппаратуре T1 назначение восьмого бита каждого байта в кадре разное и зависит от типа передаваемых данных и поколения аппаратуры.

При передаче голоса в сетях T1 все 24 канала являются абонентскими, поэтому управляющая и контрольная информация передается восьмым (наименее значащим) битом замеров голоса. В ранних версиях сетей T1 служебным был 8-й бит каждого байта кадра, поэтому реальная скорость передачи пользовательских данных составляла 56 Кбит/с (обычно восьмой бит отводился под такие служебные данные, как номер вызываемого телефонного абонента, сигнал занятости линии, сигнал снятия трубки и т. п,). Затем технология была улучшена и для служебных целей стали использовать только каждый шестой кадр. Таким образом, в пяти кадрах из шести пользовательские данные представлены всеми восемью битами, а в шестом – только семью.

При передаче компьютерных данных канал T1 предоставляет для пользовательских данных только 23 канала, а 24-й канал отводится для служебных целей, в основном – для восстановления искаженных кадров. Для одновременной передачи как голосовых, так и компьютерных данных используются все 24 канала, причем компьютерные данные передаются со скоростью 56 Кбит/с. Техника использования восьмого бита для служебных целей получила название «кражи бита» (bit robbing).

При мультиплексирования 4-х каналов T1 в один канал T2 между кадрами DS-1 по-прежнему используется один бит синхронизации, а кадры DS-2 (которые состоят из 4-х последовательных кадров DS-1) разделяются 12 служебными битами, которые предназначены не только для разделения кадров, но и для их синхронизации. Соответственно, кадры DS-3 состоят из 7 кадров DS-2, разделенных служебными битами.

Международная версия этой технологии описана в стандартах G.700-G.706. Она более логична, так как не использует схему «кражи бита». Кроме того, она основана на постоянном коэффициенте кратности скорости 4 при переходе к следующему уровню иерархии. Вместо восьмого бита в канале E1 на служебные цели отводятся 2 байта из 32. Для голосовых каналов или каналов данных остается 30 каналов со скоростью передачи 64 Кбит/с каждый.

Пользователь может арендовать несколько каналов 64 Кбит/с (56 Кбит/с) в канале T1/E1. Такой канал называется «дробным» (fractional) каналом T1/E1. В этом случае пользователю отводится несколько тайм-слотов работы мультиплексора.

Физический уровень технологии PDH поддерживает различные виды кабелей: витую пару, коаксиальный кабель и волоконно-оптический кабель. Основным вариантом абонентского доступа к каналам T1/E1 является кабель из двух витых пар с разъемами RJ-48. Две пары требуются для организации дуплексного режима передачи данных со скоростью 1,544/2,048 Мбит/с. Для представления сигналов используется: в каналах T1 биполярный потенциальный код B8ZS, в каналах E1 – биполярный потенциальный код HDB3. Для усиления сигнала на линиях T1 через каждые 1800 м (одна миля) устанавливаются регенераторы и аппаратура контроля линии.

Коаксиальный кабель благодаря своей широкой полосе пропускания поддерживает канал T2/E2 или 4 канала T1/E1. Для работы каналов T3/E3 обычно используется либо коаксиальный кабель, либо волоконно-оптический кабель, либо каналы СВЧ.

Физический уровень международного варианта технологии определяется стандартом G.703, названием которого обозначается тип интерфейса маршрутизатора или моста, подключаемого к каналу E1. Американский вариант интерфейса носит название T1.

Как американский, так и международный варианты технологии PDH обладают несколькими недостатками.

Одним из основных недостатков является сложность операций мультиплексирования и демультиплексирования пользовательских данных. Сам термин «плезио-хронный», используемый для этой технологии, говорит о причине такого явления – отсутствии полной синхронности потоков данных при объединении низкоскоростных каналов в более высокоскоростные. Изначально асинхронный подход к передаче кадров породил вставку бита или нескольких бит синхронизации между кадрами. В результате для извлечения пользовательских данных из объединенного канала необходимо полностью демультиплексировать кадры этого объединенного канала. Например, если требуется получить данные одного абонентского канала 64 Кбит/с из кадров канала T3, необходимо произвести демультиплексирование этих кадров до уровня кадров T2, затем – до уровня кадров T1, а затем демультиплексировать и сами кадры T1. Для преодоления этого недостатка в сетях PDH реализуют некоторые дополнительные приемы, уменьшающие количество операций демультиплексирования при

извлечения пользовательских данных из высокоскоростных каналов. Например, одним из таких приемов является «обратная доставка» (back hauling). Пусть коммутатор 1 канала Т3 принимает поток данных, состоящий из 672 пользовательских каналов, при этом он должен передать данные одного из этих каналов пользователю, подключенному к низкоскоростному выходу коммутатора, а весь остальной поток данных направить транзитом через другие коммутаторы в некоторый конечный демультиплексор 2, где поток Т3 полностью демультиплексируется на каналы 64 Кбит/с. Для экономии коммутатор 1 не выполняет операцию демультиплексирования своего потока, а получает данные своего пользователя только при их «обратном проходе», когда конечный демультиплексор выполнит операцию разбора кадров и вернет данные одного из каналов коммутатору 1. Естественно, такие сложные взаимоотношения коммутаторов усложняют работу сети, требуют ее тонкого конфигурирования, что ведет к большому объему ручной работы и ошибкам.

Другим существенным недостатком технологии PDH является отсутствие развитых встроенных процедур контроля и управления сетью. Служебные биты дают мало информации о состоянии канала, не позволяют его конфигурировать и т. п. Нет в технологии и процедур поддержки отказоустойчивости, которые очень полезны для первичных сетей, на основе которых строятся ответственные междугородные и международные сети. В современных сетях управлению уделяется большое внимание, причем считается, что управляющие процедуры желательно встраивать в основной протокол передачи данных сети.

Третий недостаток состоит в слишком низких по современным понятиям скоростях иерархии PDH. Волоконно-оптические кабели позволяют передавать данные со скоростями в несколько гигабит в секунду по одному волокну, что обеспечивает консолидацию в одном кабеле десятков тысяч пользовательских каналов, но это свойство технология PDH не реализует – ее иерархия скоростей заканчивается уровнем 139 Мбит/с.

Все эти недостатки устранены в новой технологии первичных цифровых сетей, получившей название *синхронной цифровой иерархии – Synchronous Digital Hierarchy, SDH*.

Технология синхронной цифровой иерархии SONET/SDH

Технология синхронной цифровой иерархии первоначально была разработана компанией Bellcore иод названием «Синхронные оптические сети» – Synchronous Optical NETs, SONET. Первый вариант стандарта появился в 1984 году. Затем эта технология была стандартизована комитетом Tl ANSI. Международная стандартизация технологии проходила под эгидой Европейского института телекоммуникационных стандартов (ETSI) и CCITT совмесно с ANSI и ведущими телекоммуникационными компаниями Америки, Европы и Японии. Основной целью разработчиков международного стандарта было создание такой технологии, которая позволяла бы передавать трафик всех существующих цифровых каналов (как американских Т1 - Т3, так и европейских Е1 - Е3) в рамках высокоскоростной магистральной сети на волоконно-оптических кабелях и обеспечила бы иерархию скоростей, продолжающую иерархию технологии PDH, до скорости в несколько гигабит в секунду.

В результате длительной работы удалось разработать международный стандарт Synchronous Digital Hierarchy, SDH (спецификации G.707-G.709), а также доработать стандарты SONET таким образом, что аппаратура и стеки SDH и SONET стали совместимыми и могут мультиплексировать входные потоки практически любого стандарта PDH – как американского, так и европейского. В терминологии и начальной скорости технологии SDH и SONET остались расхождения, но это не мешает совместимости аппаратуре разных производителей, а технология SONET/ SDH фактически стала считаться единой технологией. В России применяются стандарты и адаптированная терминология SDH.

Иерархия скоростей при обмене данными между аппаратурой SONET/SDH, которую поддерживает технология SONET/SDH, представлена в табл. 6.3.

Таблица 6.3. Скорости технологии SONET/SDH

SDH	SONET	Скорость

-	STS-1, OC-1	51,840 Мбит/с
STM-1	STS-3, OC-3	155,520 Мбит/с
STM-3	STS-9, OC-9	466,560 Мбит/с
STM-4	STS-12, OC-12	622,080 Мбит/с
STM-6	STS-18, OC-18	933,120 Мбит/с
STM-8	STS-24, OC-24	1,244 Гбит/с
STM-12	STS-36, OC-36	1,866 Гбит/с
STM-16	STS-48, OC-48	2,488 Гбит/с

В стандарте SDH все уровни скоростей (и, соответственно, форматы кадров для этих уровней) имеют общее название: STM-n – Synchronous Transport Module level n. В технологии SONET существуют два обозначения для уровней скоростей: STS-n – Synchronous Transport Signal level n, употребляемое при передаче данных электрическим сигналом, и OC-n – Optical Carrier level n, употребляемое при передаче данных световым лучом по волоконно-оптическому кабелю. Форматы кадров STS и OC идентичны.

Как видно из таблицы, стандарт SONET начинается со скорости 51,84 Мбит/с, а стандарт SDH – со скорости 155,52 Мбит/с, равной утроенной начальной скорости SONET. Международный стандарт определил начальную скорость иерархии в 155,52 Мбит/с, чтобы сохранялась стройность и преемственность технологии SDH с технологией PDH – в этом случае канал SDH может передавать данные уровня DS-4, скорость которых равна 139,264 Мбит/с. Любая скорость технологии SONET/ SDH кратна скорости STS-1. Некоторая избыточность скорости 155,52 Мбит/с для передачи данных уровня DS-4 объясняется большими накладными расходами на служебные заголовки кадров SONET/SDH.

Кадры данных технологий SONET и SDH, называемые также циклами, по форматам совпадают, естественно начиная с общего уровня STS-3/STM-1. Эти кадры обладают весьма большой избыточностью, так как передают большое количество служебной информации, которая нужна для:

- обеспечения гибкой схемы мультиплексирования потоков данных разных скоростей, позволяющих вставлять (add) и извлекать (drop) пользовательскую информацию любого уровня скорости, не демультиплексируя весь поток;
- обеспечения отказоустойчивости сети;
- поддержки операций контроля и управления на уровне протокола сети;
- синхронизации кадров в случае небольшого отклонения частот двух сопрягаемых сетей.

Стек протоколов и основные структурные элементы сети SONET/SDH показаны на рис. 6.7.

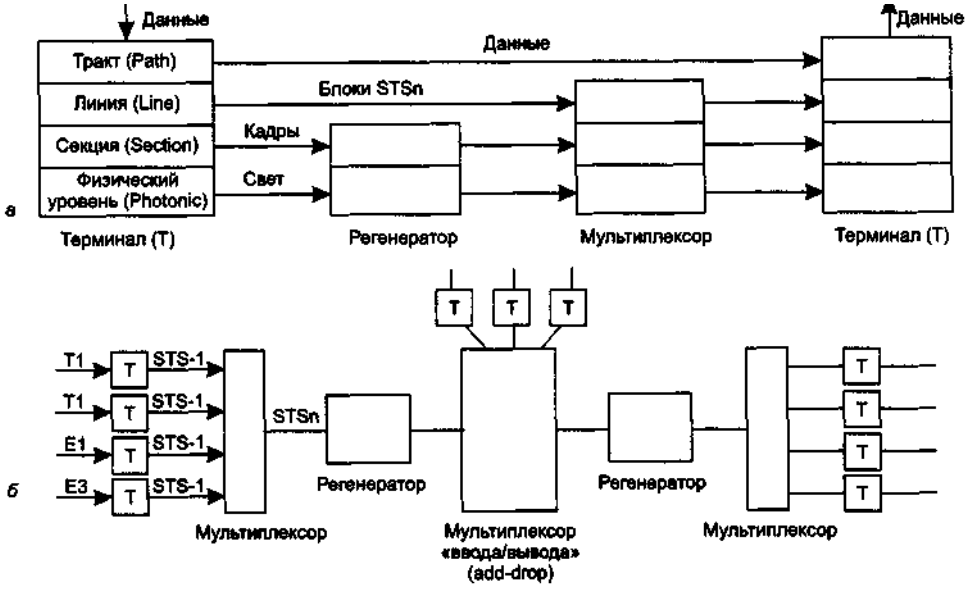


Рис. 6.7. Стек протоколов и структура сети SONET/SDH

Ниже перечислены устройства, которые могут входить в сеть технологии SONET/ SDH.

- *Терминальные устройства* (*Terminal, T*), называемые также сервисными адаптерами (Service Adapter, SA), принимают пользовательские данные от низкоскоростных каналов технологии PDH (типа T1/E1 или T3/E3) и преобразуют их в кадры STS-n. (Далее аббревиатура STS-n

используется как общее обозначение для кадров SONET/SDH.)

- *Мультиплексоры* (*Mutiplexers*) принимают данные от терминальных устройств и мультиплексируют потоки кадров разных скоростей STS-n в кадры более высокой иерархии STS-m.
- *Мультиплексоры «ввода-вывода»* (*Add Drop Mutiplexers*) могут принимать и передавать транзитом поток определенной скорости STS-n, вставляя или удаляя «на ходу», без полного демультиплексирования, пользовательские данные, принимаемые с низкоскоростных входов.
- *Цифровые кросс-коннекторы* (*Digital Cross-Connect, DCC*), называемые также аппаратурой оперативного переключения (АОП), предназначены для мультиплексирования и постоянной коммутации высокоскоростных потоков STS-n различного уровня между собой (на рис. 6.7 .не показаны). Кросс-коннектор представляет собой разновидность мультиплексора, основное назначение которого – коммутация высокоскоростных потоков данных, возможно, разной скорости. Кросс-коннекторы образуют магистраль сети SONET/SDH.
- *Регенераторы сигналов*, используемые для восстановления мощности и формы сигналов, прошедших значительное расстояние по кабелю.

На практике иногда сложно провести четкую грань между описанными устройствами, так как многие производители выпускают многофункциональные устройства, которые включают терминальные модули, модули «ввода-вывода», а также модули кросс-коннекторов.

Стек протоколов состоит из протоколов 4-х уровней.

- *Физический уровень*, названный в стандарте *фотонным* (*photonic*), имеет дело с кодированием бит информации с помощью модуляции света. Для кодирования сигнала применяется метод NRZ (благодаря внешней тактовой частоте его плохие самосинхронизирующие свойства недостатком не являются).
- *Уровень секции* (*section*) поддерживает физическую целостность сети. Секцией в технологии называется каждый непрерывный отрезок волоконно-оптического кабеля, который соединяет пару устройств SONET/SDH между собой, например мультиплексор и регенератор. Протокол секции имеет дело с кадрами и на основе служебной информации кадра может проводить тестирование секции и поддерживать операции административного контроля. В заголовке протокола секции имеются байты, образующие звуковой канал 64 Кбит/с, а также канал передачи данных управления сетью со скоростью 192 Кбит/с. Заголовок секции всегда начинается с двух байт 11110110 00101000, которые являются флагами начала кадра. Следующий байт определяет уровень кадра: STS-1, STS-2 и т. д. За каждым типом кадра закреплен определенный идентификатор.
- *Уровень линии* (*line*) отвечает за передачу данных между двумя мультиплексорами сети. Протокол этого уровня работает с кадрами разных уровней STS-n для выполнения различных операций мультиплексирования и демультиплексирования, а также вставки и удаления пользовательских данных. Таким образом, линией называется поток кадров одного уровня между двумя мультиплексорами. Протокол линии также ответственен за проведения операций реконфигурирования линии в случае отказа какого-либо ее элемента – оптического волокна, порта или соседнего мультиплексора.
- *Уровень тракта* (*path – путь*) отвечает за доставку данных между двумя конечными пользователями сети. Тракт (путь) – это составное виртуальное соединение между пользователями. Протокол тракта должен принять данные, поступающие в пользовательском формате, например формате T1, и преобразовать их в синхронные кадры STS-n/STM-m.

Как видно из рис. 6.7, регенераторы работают только с протоколами двух нижних уровней, отвечая за качество сигнала и поддержания операций тестирования и управления сетью. Мультиплексоры работают с протоколами трех нижних уровней, выполняя, кроме функций регенерации сигнала и реконфигурации секций, функцию мультиплексирования кадров STS-n разных уровней. Кросс-коннектор представляет собой пример мультиплексора, который поддерживает протоколы трех уровней. И наконец, функции всех четырех уровней выполняют терминалы, а также мультиплексоры «ввода-вывода», то есть устройства, работающие с пользовательскими потоками данных.

Формат кадра STS-1 представлен на рис. 6.8. Кадры технологии SONET/SDH принято представлять в виде матрицы, состоящей из n строк и m столбцов. Такое представление хорошо отражает структуру кадра со своего рода подкадрами, называемыми виртуальными контейнерами (Virtual Container, VC – термин SDH) или виртуальными притоками (Virtual Tributaries, VT – термин SONET). Виртуальные

контейнеры – это подкадры, которые переносят потоки данных, скорости которых ниже, чем начальная скорость технологии SONET/SDH в 51,84 Мбит/с (например, поток данных T1 со скоростью 1,544 Мбит/с).

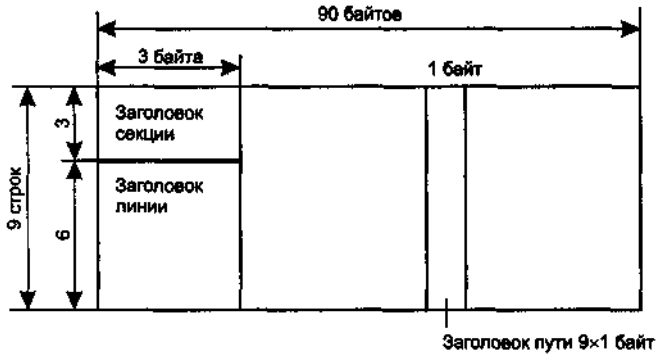


Рис. 6.8. Формат кадра STS-1

Кадр STS-1 состоит из 9 строк и 90 столбцов, то есть из 810 байт данных. Между устройствами сети кадр передается последовательно по байтам – сначала первая строка слева направо, затем вторая и т. д. Первые 3 байта каждой строки представляют собой служебные заголовки. Первые 3 строки представляют собой заголовок из 9 байт протокола уровня секции и содержат данные, необходимые для контроля и реконфигурации секции. Остальные 6 строк составляют заголовок протокола линии, который используется для реконфигурации, контроля и управления линией. Устройства сети SONET/SDH, которые работают с кадрами, имеют достаточный буфер для размещения в нем всех байт кадра, протекающих синхронно через устройство, поэтому устройство для анализа информации на некоторое время имеет полный доступ ко всем частям кадра. Таким образом, размещение служебной информации в несмежных байтах не представляет сложности для обработки кадра.

Еще один столбец представляет собой заголовок протокола пути. Он используется для указания местоположения виртуальных контейнеров внутри кадра, если кадр переносит низкоскоростные данные пользовательских каналов типа T1/E1. Местоположение виртуальных контейнеров задается не жестко, а с помощью системы *указателей* (*pointers*).

Концепция указателей является ключевой в технологии SONET/SDH. Указатель призван обеспечить синхронную передачу байт кадров с асинхронным характером вставляемых и удаляемых пользовательских данных.

Указатели используются на разных уровнях. Рассмотрим, как с помощью указателя выполняется выделение поля данных кадра из синхронного потока байт. Несмотря на питание всех устройств сети SONET/SDH тактовой частотой синхронизации из одного центрального источника, синхронизация между различными сетями может незначительно нарушаться. Для компенсации этого эффекта началу поля данных кадра (называемого в стандарте SPE – Synchronous Payload Environment) разрешается смещаться относительно начала кадра произвольным образом. Реальное начало поля SPE задается указателем H1, размещенным в заголовке протокола линии. Каждый узел, поддерживающий протокол линии, обязан следить за частотой поступающих данных и компенсировать ее несовпадение с собственной частотой за счет вставки или удаления одного байта из служебного заголовка. Затем узел должен нарастить или уменьшить значения указателя первого байта поля данных CPE относительно начала кадра STS-1. В результате поле данных может размещаться в двух последовательных кадрах, как это показано на рис. 6.9.

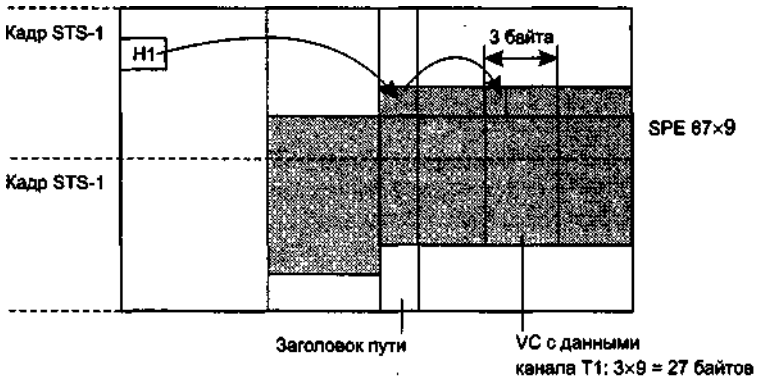


Рис. 6.9. Использование указателей для поиска данных в кадре

Тот же прием применяется для вставки или удаления пользовательских данных в потоке кадров STS-n. Пользовательские данные каналов типа T1/E1 или T3/E3 асинхронны по отношению к потоку байтов кадра STS-n. Мультиплексор формирует виртуальный контейнер и, пользуясь указателем H1, находит начало очередного поля данных. Затем мультиплексор анализирует заголовок пути и находит в нем указатель H4, который описывает структуру содержащихся в кадре виртуальных контейнеров. Обнаружив свободный виртуальный контейнер нужного формата, например для 24 байт канала T1, он вставляет эти байты в нужное место поля данных кадра STS-1. Аналогично производится поиск начала данных этого канала при выполнении операции удаления пользовательских данных.

Таким образом, кадры STS-n всегда образуют синхронный поток байтов, но с помощью изменения значения соответствующего указателя можно вставить и извлечь из этого потока байты низкоскоростного канала, не выполняя полного демультиплексирования высокоскоростного канала.

Виртуальные контейнеры также содержат дополнительную служебную информацию по отношению к данным пользовательского канала, который они переносят. Поэтому виртуальный контейнер для переноса данный канала T1 требует скорости передачи данных не 1,544 Мбит/с, а 1,728 Мбит/с.

В технологии SONET/SDH существует гибкая, но достаточно сложная схема использования поля данных кадров STS-n. Сложность этой схемы в том, что нужно «уложить» в кадр наиболее рациональным способом мозаику из виртуальных контейнеров разного уровня. Поэтому в технологии SONET/SDH стандартизовано шесть типов виртуальных контейнеров, которые хорошо сочетаются друг с другом при образовании кадра STS-n. Существует ряд правил, по которым контейнеры каждого вида могут образовывать группы контейнеров, а также входить в состав контейнеров более высокого уровня.

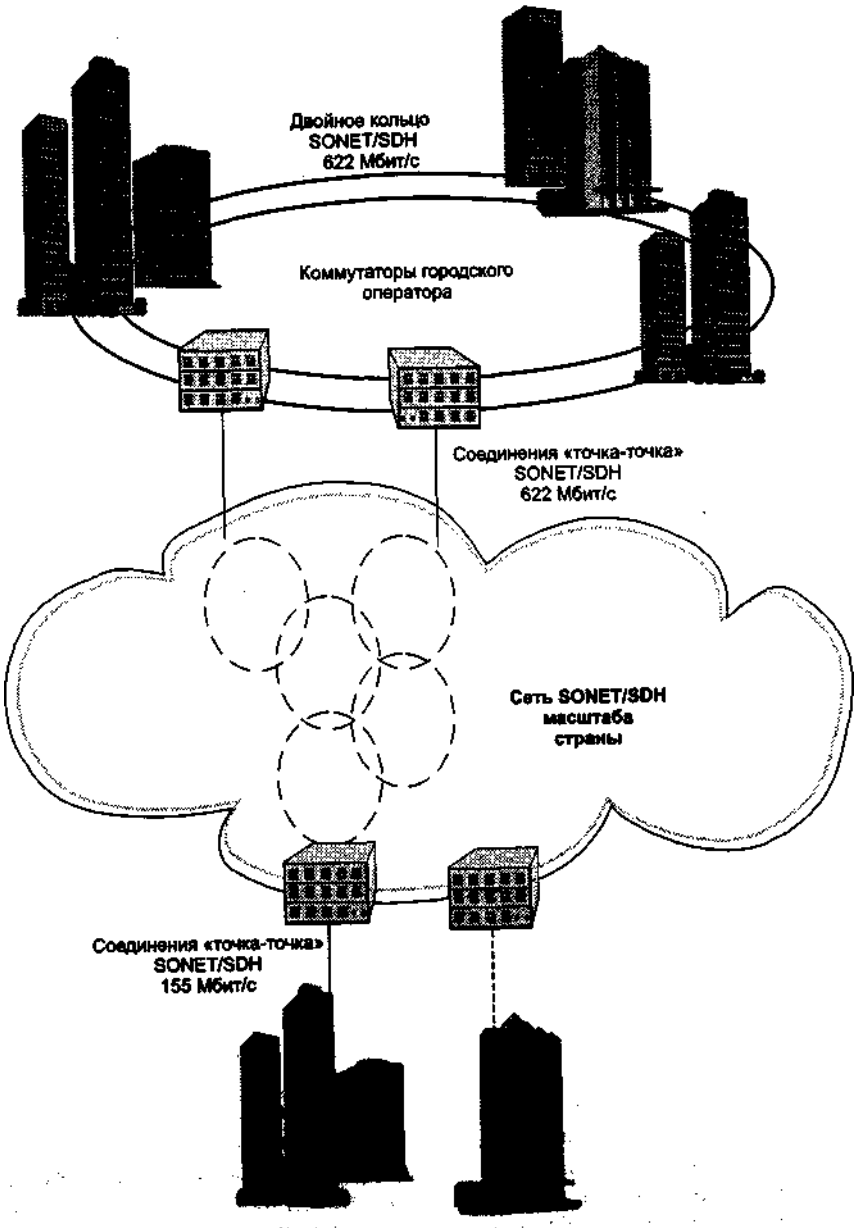


Рис. 6.10. Использование двойных колец для обеспечения отказоустойчивости сети SONET/SDH

Отказоустойчивость сети SONET/SDH встроена в ее основные протоколы. Этот механизм называется автоматическим защитным переключением – Automatic Protection Switching, APS. Существуют два способа его работы. В первом способе защита осуществляется по схеме 1:1. Для каждого рабочего волокна (и обслуживающего его порта) назначается резервное волокно. Во втором способе, называемом 1:n, для защиты n волокон назначается только одно защитное волокно.

В схеме защиты 1:1 данные передаются как по рабочему, так и по резервному волокну. При выявлении ошибок принимающий мультиплексор сообщает передающему, какое волокно должно быть рабочим. Обычно при защите 1:1 используется схема двух колец, похожая на двойные кольца FDDI (рис. 6.10), но только с одновременной передачей данных в противоположных направлениях. При обрыве кабеля между двумя мультиплексорами происходит сворачивание колец, и, как и в сетях FDDI, из двух колец образуется одно рабочее.

Применение схемы резервирования 1:1 не обязательно требует кольцевого соединения мультиплексоров, можно применять эту схему и при радиальном подключении устройств, но кольцевые структуры решают проблемы отказоустойчивости эффективнее – если в сети нет колец, радиальная схема не сможет ничего сделать при обрыве кабеля между устройствами.

Управление, конфигурирование и администрирование сети SONET/SDH также встроено в протоколы. Служебная информация протокола позволяет централизованно и дистанционно конфигурировать пути между конечными пользователями сети, изменять режим коммутации потоков в кросс-коннекторах, а также собирать подробную статистику о работе сети. Существуют мощные системы управления сетями

SDH, позволяющие прокладывать новые каналы простым перемещением мыши по графической схеме сети.

Применение цифровых первичных сетей

Сети SDH и сети плезиохронной цифровой иерархии очень широко используются для построения как публичных, так и корпоративных сетей. Особенно популярны их услуги в США, где большинство крупных корпоративных сетей построено на базе выделенных цифровых каналов. Эти каналы непосредственно соединяют маршрутизаторы, размещаемые на границе локальных сетей отделений корпорации.

При аренде выделенного канала сетевой интегратор всегда уверен, что между локальными сетями существует канал вполне определенной пропускной способности. Это положительная черта аренды выделенных каналов. Однако при относительно небольшом количестве объединяемых локальных сетей пропускная способность выделенных каналов никогда не используется на 100 %, и это недостаток монопольного владения каналом – предприятие всегда платит не за реальную пропускную способность. В связи с этим обстоятельством в последнее время все большую популярность приобретает служба сетей frame relay, в которых каналы разделяют несколько предприятий.

На основе первичной сети SDH можно строить сети с коммутацией пакетов, например frame или ATM, или же сети с коммутацией каналов, например ISDN. Технология ATM облегчила эту задачу, приняв стандарты SDH в качестве основных стандартов, физического уровня. Поэтому при существовании инфраструктуры SDH для образования сети ATM достаточно соединить ATM-коммутаторы жестко сконфигурированными в сети SDH-каналами.

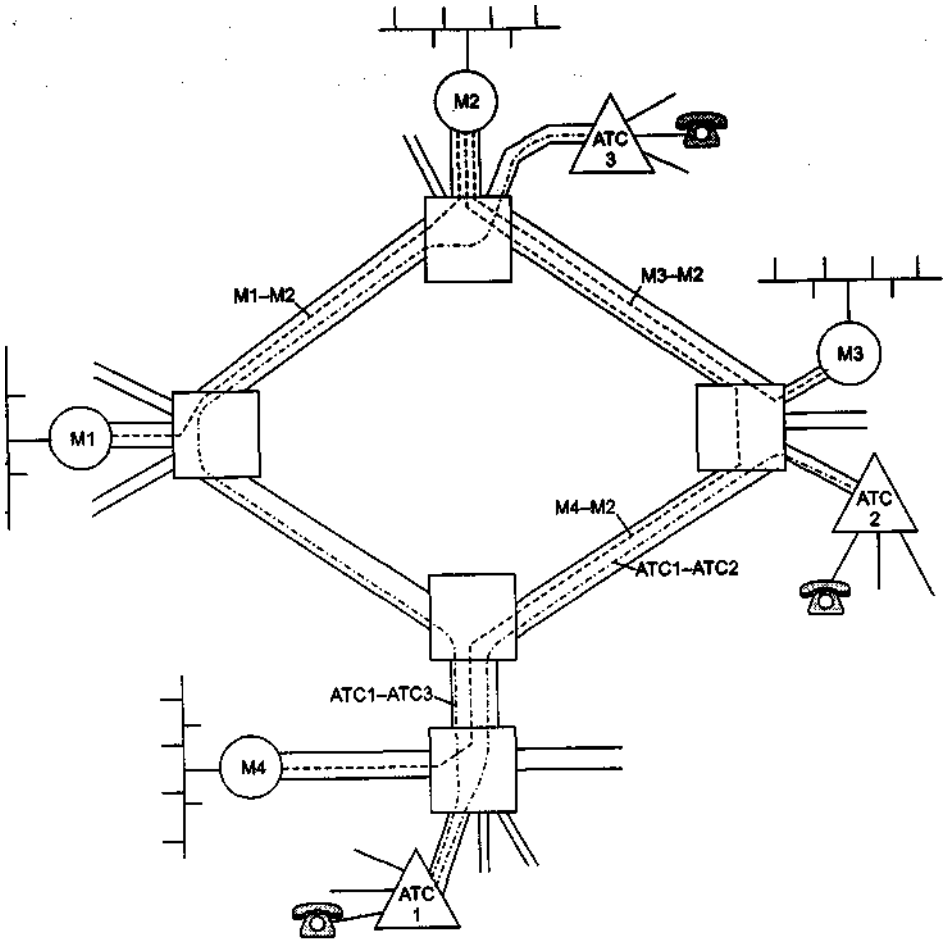


Рис. 6.11. Использование цифровой первичной сети для организации двух наложенных сетей – вычислительной и телефонной

Телефонные коммутаторы также могут использовать технологию цифровой иерархии, поэтому построение телефонной сети с помощью каналов PDH или SONET/SDH не представляет труда. На рис. 6.11. показан пример сосуществования двух сетей - компьютерной и телефонной - на основе выделенных каналов одной и той же первичной цифровой сети.

Технология SONET/SDH очень экономично решает задачу мультиплексирования и коммутации потоков различной скорости, поэтому сегодня она, несмотря на невозможность динамического перераспределения пропускной способности между абонентскими каналами, является наиболее распространенной технологией создания первичных сетей. Технология ATM, которая хотя и позволяет динамически перераспределять пропускную способность каналов, получилась значительно сложнее, и уровень накладных расходов у нее гораздо выше.

Примером российских сетей SDH могут служит сети «Макомнет», «Метроком» и «Раском», построенные совместными предприятиями с участием американской компании Andrew Corpdration.

Начало создания сети «Макомнет» относится к 1991 году, когда было образовано совместное предприятие, учредителями которого выступили Московский метрополитен и компания Andrew Corporation.

Транспортной средой сети стали одномодовые 32-, 16- и 8-жильные волоконно-оптические кабели фирмы Pirelli, проложенные в туннелях метрополитена. В метро было уложено более 350 км кабеля. Постоянно расширяясь, сегодня кабельная система «Макомнет» с учетом соединений «последней мили» имеет длину уже более 1000 километров.

Изначально в сети «Макомнет» использовалось оборудование SDH только 1 уровня (155 Мбит/с) – мультиплексоры TN-1X фирмы Northern Telecom (Nortel), обладающие функциями коммутации 63 каналов E1 по 2 Мбит/с каждый. Из данных мультиплексоров были организованы две кольцевые топологии «Восточная» и «Западная» (они разделили кольцевую линию метрополитена на два полукольца вдоль Сокольнической линии) и несколько отрезков «точка-точка», протянувшихся к ряду клиентов, абонировавших сравнительно большие емкости сети. Эти кольца образовали магистраль сети, от которой ответвлялись связи с абонентами.

Растущие день ото дня потребности заказчиков заставляли создавать новые топологии и переконфигурировать старые. В течение двух лет в сети «Макомнет» задача увеличения пропускной способности решалась за счет прокладки новых кабелей и установки нового оборудования, что позволило утроить количество топологий по кольцевой линии. Число узлов коммутации возросло до семидесяти. Но настал момент, когда остро встал вопрос о количестве резервных оптических волокон на некоторых участках сети, и с учетом прогнозов на развитие было принято решение о построении нового, 4-го уровня SDH (622 Мбит/с).

Подготовительные работы по переконфигурированию и введению действующих потоков в сеть нового уровня происходили без прекращения работы сети в целом. В качестве оборудования 4 уровня (622 Мбит/с) были установлены мультиплексоры TN-4X фирмы Nortel. Вместе с новым оборудованием была приобретена принципиально новая высокоинтеллектуальная система управления NRM (Network Resource Manager). Эта система является надстройкой над системами управления оборудования 1 и 4 уровней. Она обладает не только всеми функциями контроля оборудования, присущими каждой из систем, но и рядом дополнительных возможностей: автоматической прокладки канала по сети, когда оператору требуется лишь указать начальную и конечную точки; функциями инвентаризации каналов, обеспечивающих их быстрый поиск в системе, и рядом других.

Ввод всего шести узлов TN-4X значительно увеличил транспортную емкость сети, а высвободившиеся волокна сделали возможным ее дальнейшее наращивание.

На первых порах клиентами «Макомнет» стали телекоммуникационные компании, использующие каналы «Макомнет» для строительства собственных сетей. Однако со временем круг клиентов значительно расширился: банки, различные коммерческие и государственные структуры. Оборудование компании расположено на территории многих городских, а также основных международных и междугородных телефонных станций.

Устройства DSU/CSU для подключения к выделенному каналу

Связь компьютера или маршрутизатора с цифровой выделенной линией осуществляется с помощью пары устройств, обычно выполненных в одном корпусе или же совмещенных с маршрутизатором. Этими устройствами являются: *устройство обслуживания данных* (*УОД*) и *устройство обслуживания канала* (*УОК*). В англоязычной литературе эти устройства называются соответственно Data Service Unit (DSU) и Channel Service Unit (CSU). DSU преобразует сигналы, поступающие от DTE (обычно по интерфейсу RS-232C, RS-449 или V.35). DSU выполняет всю синхронизацию, формирует кадры каналов T1/E1, усиливает сигнал и осуществляет выравнивание загрузки канала. CSU выполняет более узкие

функции, в основном это устройство занимается созданием оптимальных условий передачи в линии. Эти устройства, как и модуляторы-демодуляторы, часто обозначаются одним словом DSU/CSU (рис. 6.12).

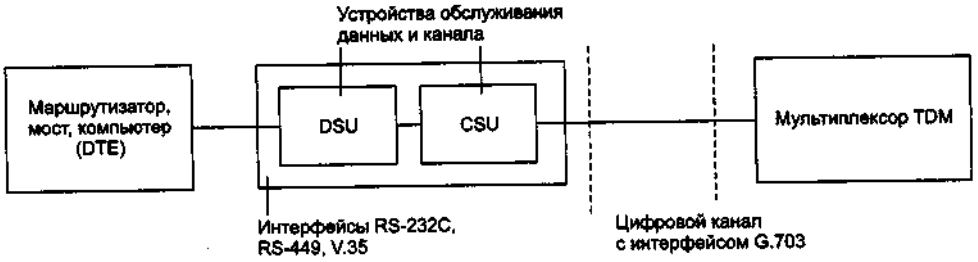


Рис. 6.12. Использование DSU/CSU для подключения к цифровой выделенной линии

Нередко под устройством DSU/CSU понимают более сложные устройства, которые кроме согласования интерфейсов выполняют функции мультиплексора T1/E1. В состав такого устройства может входит модуль мультиплексирования низкоскоростных потоков голоса и данных в канал 64 Кбит/с или в несколько таких каналов (голос при этом обычно компрессируется до скорости 8-16 Кбит/с).

6.2.3. Протоколы канального уровня для выделенных линий

Выделенные каналы используются для прямой связи между собой локальных сетей или отдельных компьютеров. Для маршрутизатора или моста выделенная линия предоставляет чаще всего либо канал с известной полосой пропускания, как в случае выделенных аналоговых линий, либо канал с известным протоколом физического уровня, как в случае цифровых выделенных каналов. Правда, так как аналоговый канал требует модема для передачи данных, протокол физического уровня также определен для этой линия – это протокол модема. Поэтому для передачи данных между маршрутизаторами, мостами или отдельными компьютерами с помощью выделенного канала необходимо решить, какие протоколы уровней выше физического необходимы для передачи сообщений с нужной степенью надежности и с возможностями управления потоком кадров для предотвращения переполненния соседних узлов.

Если выделенный канал соединяет сети через маршрутизаторы, то протокол сетевого уровня определен, а протокол канального уровня маршрутизатор может использовать любой, в том числе и протокол канального уровня локальной сети, например Ethernet. Мост должен передавать кадры канального протокола из одной локальной сети в другую, при этом ему тоже можно непосредственно использовать протокол локальной сети (Ethernet, Token Ring, FDDI) поверх физического уровня канала.

Однако ни мосты, ни маршрутизаторы на выделенных каналах с протоколами канального уровня локальных сетей не работают. Они, с одной стороны, избыточны, а с другой стороны, в них отсутствуют некоторые необходимые процедуры, очень полезные при объединении сетей по глобальному выделенному каналу.

Избыточность проявляется в процедурах получения доступа к разделяемой среде, а так как выделенная линия постоянно находится в распоряжении соединяющихся с ее помощью конечных узлов, процедура получения доступа к ней не имеет смысла. Среди отсутствующих процедур можно назвать процедуру управления потоком данных, процедуру взаимной аутентификации удаленных устройств, что часто необходимо для защиты сети от «подставного» маршрутизатора или моста, отводящего корпоративный трафик не по назначению. Кроме того, существует ряд параметров, которые полезно автоматически согласовывать при удаленном взаимодействии, – например, максимальный размер поля данных (MTU), IP-адрес партнера (как для безопасности, так и для автоматического конфигурирования стека TCP/IP на удаленных одиночных компьютерах).

Протокол SLIP

Протокол SLIP (Serial Line IP) был первым стандартом де-факто, позволяющим устройствам, соединенным последовательной линией связи, работать по протоколам TCP/IP. Он был создан в начале 80-х годов и в 1984 году встроен Риком Адамсом (Rick Adams) в операционную систему 4.2 Berkley

Unix. Позднее SLIP был поддержан в других версиях Unix и реализован в программном обеспечении для ПК.

Правда, ввиду его функциональной простоты, SLIP использовался и используется в основном на коммутируемых линиях связи, которые не характерны для ответственных и скоростных сетевых соединений. Тем не менее коммутируемый канал отличается от некоммутируемого только более низким качеством и необходимостью выполнять процедуру вызова абонента, поэтому SLIP вполне применим и на выделенных каналах.

Протокол SLIP выполняет единственную функцию – он позволяет в потоке бит, которые поступают по выделенному (или коммутируемому) каналу, распознать начало и конец IP-пакета. Помимо протокола IP, другие протоколы сетевого уровня SLIP не поддерживает.

Чтобы распознать границы IP-пакетов, протокол SLIP предусматривает использование специального символа END, значение которого в шестнадцатеричном представлении равно CO, Применение специального символа может породить конфликт: если байт пересылаемых данных тождественен символу END, то он будет ошибочно определен как признак конца пакета. Чтобы предотвратить такую ситуацию, байт данных со значением, равным значению символа END, заменяется составной двухбайтовой последовательностью, состоящей из специального символа ESC (DB) и кода DC. Если же байт данных имеет тот же код, что и символ SLIP ESC, то он заменяется двухбайтовой последовательностью, состоящей из собственно символа SLIP ESC и кода DD. После последнего байта пакета передается символ END.

Механизм формирования составных последовательностей показан на рис. 6.13. Здесь приведены стандартный IP-пакет (один байт которого тождественен символу END, а другой – символу SLIP ESC) и соответствующий ему SLIP-пакет, который больше на 2 байта.

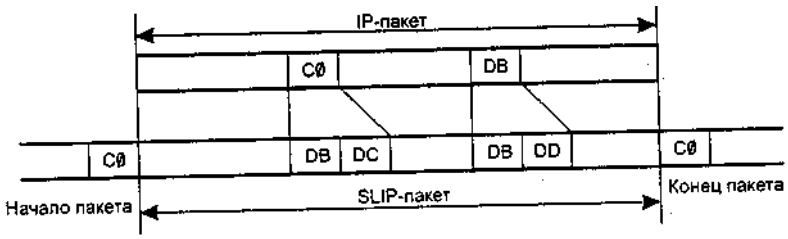


Рис. 6.13. Инкапсуляция IP-пакетов в SLIP-пакеты

Хотя в спецификации протокола SLIP не определена максимальная длина передаваемого пакета, реальный размер IP-пакета не должен превышать 1006 байт. Данное ограничение связано с первой реализацией протокола SLIP в соответствующем драйвере для Berkley Unix, и его соблюдение необходимо для поддержки совместимости разных реализаций SLIP (большинство современных реализаций позволяют администратору самому установить размер пакета, а по умолчанию используют размер 1500 байт).

Для установления связи по протоколу SLIP компьютеры должны иметь информацию об IP-адресах друг друга. Однако возможна ситуация, когда, скажем, при осуществлении соединения между хостом и маршрутизатором последнему понадобится передать хосту информацию о его IP-адресе. В протоколе SLIP нет механизмов, дающих возможность обмениваться адресной информацией. Это ограничение не позволяет использовать SLIP для некоторых видов сетевых служб.

Другой недостаток SLIP – отсутствие индикации типа протокола, пакет которого инкапсулируется в SLIP-пакет. Поэтому через последовательную линию по протоколу SLIP можно передавать трафик лишь одного сетевого протокола – IP. При работе с реальными телефонными линиями, зашумленными и поэтому искажающими пакеты при пересылке, требуются процедуры обнаружения и коррекции ошибок. В протоколе SLIP такие процедуры не предусмотрены. Эти функции обеспечивают вышележащие протоколы: протокол IP проводит тестирование целостности пакета по заголовку IP, а один из двух транспортных протоколов (UDP или TCP) проверяет целостность всех данных по контрольным суммам.

Низкая пропускная способность последовательных линий связи вынуждает сокращать время передачи пакетов, уменьшая объем содержащейся в них служебной информации. Эта задача решается с помощью протокола Compressed SLIP (CSLIP), поддерживающего сжатие заголовков пакетов. Появление CSLIP объясняется тем фактом, что при использований программ типа Telnet, Rlogin и других для пересылки одного байта данных потребуется переслать 20-байтбвый заголовок IP-пакета и

20-байтовый заголовок TCP-пакета (итого 40 байт). Спецификация CSLIP обеспечивает сжатие 40-байтового заголовка до 3-5 байт. На сегодняшний момент большинство реализаций протокола SLIP поддерживают спецификацию CSLIP.

Таким образом, протокол SLIP выполняет работу по выделению из последовательности передаваемых по последовательному каналу бит границ IP-пакета. Протокол не имеет механизмов передачи адресной информации, идентификации типа протокола сетевого уровня, определения и коррекции ошибок.

Протоколы семейства HDLC

Долгое время основным протоколом выделенных линий был протокол HDLC (High-level Data Link Control), имеющий статус стандарта ISO. Протокол HDLC на самом деле представляет собой семейство протоколов, в которое входят известные протоколы: LAP-B, образующий канальный уровень сетей X.25, LAP-D – канальный уровень сетей ISDN, LAP-M – канальный уровень асинхронно-синхронных модемов, LAP-F – канальный уровнеь сетей frame relay.

Основные принципы работы протокола HDLC: режим логического соединения, контроль искаженных и потерянных кадров с помощью метода скользящего окна, управление потоком кадров с помощью команд RNR и RR, а также различные типы кадров этого протокола были уже рассмотрены в главе 3 при изучении еще одного представителя семейства HDLC – протокола LLC2.

Однако сегодня протокол HDLC на выделенных каналах вытеснил *протокол «точка-точка», Point-to-Point Protocol, PPP.*

Дело в том, что одна из основных функций протокола HDLC – это восстановление искаженных и утерянных кадров. Действительно, применение протокола HDLC обеспечивает снижение вероятности искажения бита (BER) с 10^{-3}, что характерно для территориальных аналоговых каналов, до 10^{-9}.

Однако сегодня популярны цифровые каналы, которые и без внешних процедур восстановления кадров обладают высоким качеством (величина BER составляет 10^{-8}-10^{-9}). Для работы по такому каналу восстановительные функции протокола HDLC не нужны. При передаче по аналоговым выделенным каналам современные модемы сами применяют протоколы семейства HDLC (синхронные модемы – HDLC, а асинхронно-синхронные с асинхронным интерфейсом – LAP-M, который также принадлежит семейству HDLC). Поэтому использование HDLC на уровне маршрутизатора или моста становится неоправданным.

Протокол PPP

Этот протокол разработан группой IETF (Internet Engineering Task Force) как часть стека TCP/IP для передачи кадров информации по последовательным глобальным каналам связи взамен устаревшего протокола SLIP (Serial Line IP). Протокол PPP стал фактическим стандартом для глобальных линий связи при соединении удаленных клиентов с серверами и для образования соединений между маршрутизаторами в корпоративной сети. При разработке протокола PPP за основу был взят формат кадров HDLC и дополнен собственными полями. Поля протокола PPP вложены в поле данных кадра HDLC. Позже были разработаны стандарты, использующие вложение кадра PPP в кадры frame relay и других протоколов глобальных сетей.

Основное отличие PPP от других протоколов канального уровня состоит в том, что он добивается согласованной работы различных устройств с помощью переговорной процедуры, во время которой передаются различные параметры, такие как качество линии, протокол аутентификации и инкапсулируемые протоколы сетевого уровня. Переговорная процедура происходит во время установления соединения.

Протокол PPP основан на четырех принципах: переговорное принятие параметров соединения, многопротокольная поддержка, расширяемость протокола, независимость от глобальных служб.

Переговорное принятие параметров соединения. В корпоративной сети конечные системы часто отличаются размерами буферов для временного хранения пакетов, ограничениями на размер пакета, списком поддерживаемых протоколов сетевого уровня. Физическая линия, связывающая конечные устройства, может варьироваться от низкоскоростной аналоговой линии до высокоскоростной цифровой линии с различными уровнями качества обслуживания.

Чтобы справиться со всеми возможными ситуациями, в протоколе PPP имеется набор стандартных

установок, действующих по умолчанию и учитывающих все стандартные конфигурации. При установлении соединения два взаимодействующих устройства для нахождения взаимопонимания пытаются сначала использовать эти установки. Каждый конечный узел описывает свои возможности и требования. Затем на основании этой информации принимаются параметры соединения, устраивающие обе стороны, в которые входят форматы инкапсуляции данных, размеры пакетов, качество линии и процедура аутентификации.

Протокол, в соответствии с которым принимаются параметры соединения, называется *протоколом управления связью* (*Link Control Protocol, LCP*). Протокол, который позволяет конечным узлам договориться о том, какие сетевые протоколы будут передаваться в установленном соединении, называется *протоколом управления сетевым уровнем* (*Network Control Protocol, NCP*). Внутри одного PPP-соединения могут передаваться потоки данных различных сетевых протоколов.

Одним из важных параметров PPP-соединения является режим аутентификации. Для целей аутентификации PPP предлагает по умолчанию протокол PAP (Password Authentication Protocol), передающий пароль по линии связи в открытом виде, или протокол CHAP (Challenge Handshake Authentication Protocol), не передающий пароль по линии связи и поэтому обеспечивающий большую безопасность сети. Пользователям также разрешается добавлять и новые алгоритмы аутентификации. Дисциплина выбора алгоритмов компрессии заголовка и данных аналогична.

Многопротокольная поддержка – способность протокола PPP поддерживать несколько протоколов сетевого уровня – обусловила распространение PPP как стандарта де-факто. В отличие от протокола SLIP, который может переносить только IP-пакеты, или LAP-B, который может переносить только пакеты X.25, PPP работает со многими протоколами сетевого уровня, включая IP, Novell IPX, AppleTalk, DECnet, XNS, Banyan VINES и OSI, а также протоколами канального уровня локальной сети. Каждый протокол сетевого уровня конфигурируется отдельно с помощью соответствующего протокола NCP. Под конфигурированием понимается, во-первых, констатация того факта, что данный протокол будет использоваться в текущей сессии PPP, а во-вторых, переговорное утверждение некоторых параметров протокола. Больше всего параметров устанавливается для протокола IP – IP-адрес узла, IP-адрес серверов DNS, использование компрессии заголовка IP-пакета и т. д. Протоколы конфигурирования параметров соответствующего протокола верхнего уровня называются по имени этого протокола с добавлением аббревиатуры CP (Control Protocol), например протокол IPCP, IPXCP и т. п.

Расширяемость протокола. Под расширяемостью понимается как возможность включения новых протоколов в стек PPP, так и возможность использования собственных протоколов пользователей вместо рекомендуемых в PPP по умолчанию. Это позволяет наилучшим образом настроить PPP для каждой конкретной ситуации.

Независимость от глобальных служб. Начальная версия PPP работала только с кадрами HDLC. Теперь в стек PPP добавлены спецификации, позволяющие использовать PPP в любой технологии глобальных сетей, например ISDN, frame relay, X.25, Sonet и HDLC.

Переговорная процедура протоколов LCP и NCP может и не завершиться соглашением о каком-нибудь параметре. Если, например, один узел предлагает в качестве MTU значение 1000 байт, а другой отвергает это предложение и в свою очередь предлагает значение 1500 байт, которое отвергается первым узлом, то по истечении тайм-аута переговорная процедура может закончиться безрезультатно.

Возникает вопрос – каким образом два устройства, ведущих переговоры по протоколу PPP, узнают о тех параметрах, которые они предлагают своему партнеру? Обычно у реализации протокола PPP есть некоторый набор параметров по умолчанию, которые и используются в переговорах. Тем не менее каждое устройство (и программа, реализующая протокол PPP в операционной системе компьютера) позволяет администратору изменить параметры по умолчанию, а также задать параметры, которые не входят в стандартный набор. Например, IP-адрес для удаленного узла отсутствует в параметрах по умолчанию, но администратор может задать его для сервера удаленного доступа, после чего сервер будет предлагать его удаленному узлу.

Хотя протокол PPP и работает с кадром HDLC, но в нем отсутствуют процедуры контроля кадров и управления потоком протокола HDLC. Поэтому в PPP используется только один тип кадра HDLC – ненумерованный информационный. В поле управления такого кадра всегда содержится величина 03. Для исправления очень редких ошибок, возникающих в канале, необходимы протоколы верхних уровней - TCP, SPX, NetBUEI, NCP и т. п.

Одной из возможностей протокола PPP является использование нескольких физических линий для образования одного логического канала, так называемый транкинг каналов. Эту возможность реализует

дополнительный протокол, который носит название MLPPP (Multi Link PPP). Многие производители поддерживают такое свойство в своих маршрутизаторах и серверах удаленного доступа фирменным способом. Использование стандартного способа всегда лучше, так как он гарантирует совместимость оборудования разных производителей.

Общий логический канал может состоять из каналов разной физической природы. Например, один канал может быть образован в телефонной сети, а другой может являться виртуальным коммутируемым каналов сети frame relay.

6.2.4. Использование выделенных линий для построения корпоративной сети

Для связи двух локальных сетей по арендуемому или собственному выделенному каналу обычно используются мосты или маршрутизаторы. Эти устройства нужны для того, чтобы по выделенному каналу пересылались не все кадры, циркулирующие в каждой локальной сети, а только те, которые предназначены для другой локальной сети.

Схема установки моста или маршрутизатора в этом случае однотипна (рис. 6.14). Сначала необходимо решить проблему физического сопряжения выходного порта моста или маршрутизатора с аппаратурой передачи данных, то есть DCE, подключаемой непосредственно к абонентскому окончанию линии. Если канал аналоговый, то это интерфейс с модемом, а если цифровой – то с устройством DSU/CSU. Интерфейс определяется требованиями DCE – это может быть RS-232C для низкоскоростных линий или же RS-449 или V.35 для высокоскоростных каналов типа T1/E1. Для канала T3/E3 потребуется наличие интерфейса HSSI.

Некоторые устройства имеют программно настраиваемые последовательные интерфейсы, которые могут работать и как RS-449/V.11, и как RS-449/V.10, и как V.35.

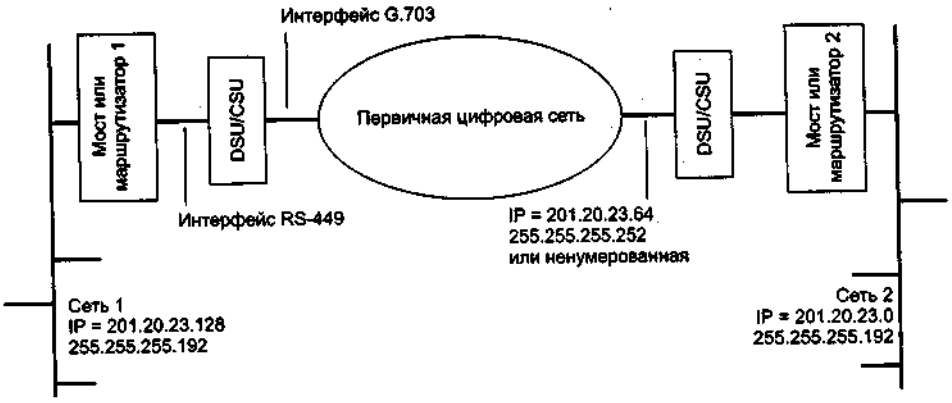


Рис. 6.14. Соединение сетей с помощью выделенного канала

На рис. 6.14 выбрано в качестве примера соединение через цифровой канал E1, поэтому мост/маршрутизатор использует для подключения к каналу устройство DSU/ CSU с внутренним интерфейсом RS-449 и внешним интерфейсом G.703. Часто крупные маршрутизаторы имеют модули со встроенным интерфейсом G.703, тогда необходимость в устройстве DSU/CSU отпадает. Если же выделенный канал был бы аналоговым, то в качестве DCE был бы необходим модем, поддерживающий режим работы по выделенной линии, причем кроме других различных критериев (скорость, контроль ошибок, компрессия) необходимо учитывать возможность модема работать по предоставленному абонентскому окончанию: 4-проводному или 2-проводному.

После решения проблем физического уровня удаленные мосты готовы к работе. После включения каждый мост начинает передавать все кадры из своей локальной сети в выделенный канал и одновременно (так как практически все выделенные каналы дуплексные) принимать кадры из выделенного канала. На основании проходящего трафика каждый мост строит адресную таблицу и начинает передавать в выделенный канал кадры только тем станциям, которые действительно находятся в другой сети, а также широковещательные кадры и кадры с неизвестными MAC-адресами. Современные удаленные мосты при пересылке кадров локальдых сетей упаковывают их в кадры протокола PPP. Переговорная процедура, которую ведут мосты при установлении PPP-соединения, сводится в основном к выбору параметров канального уровня с помощью протокола LPC, а также к взаимной аутентификации (если такая процедура задана в параметрах протокола PPP обоих мостов).

Маршрутизатор после подключения к выделенной линии и локальной сети необходимо конфигурировать. На рис. 6.14 IP-маршрутизаторы связаны по выделенному каналу. Конфигурирование маршрутизаторов в этом случае подобно конфигурированию в локальных сетях. Каждая локальная сеть получает свой IP-адрес с соответствующей маской. Выделенный канал также является отдельной IP-сетью, поэтому можно ему также дать некоторый IP-адрес из диапазона адресов, которым распоряжается администратор корпоративной сети (в данном случае выделенному каналу присвоен адрес сети, состоящей из 2-х узлов, что определяется маской 255.255.255.252). Можно выделенному каналу и не присваивать IP-адрес – такой интерфейс маршрутизатора называется *ненумерованным* (*unnumbered*). Маршрутизатор будет нормально работать в обоих случаях. Как и в локальной сети, маршрутизаторам не нужно вручную задавать аппаратные адреса своих непосредственных соседей, так как отсылая пакеты протокола маршрутизации (RIP или OSPF) по выделенному каналу, маршрутизаторы будут их получать без проблем. Протокол ARP на выделенном канале не используется, так как аппаратные адреса на выделенном канале не имеют практического смысла (в кадре PPP есть два адреса – кадр от DCE или от DTE, но маршрутизатор всегда будет получать кадр от DCE).

Как и в локальных сетях, важной характеристикой удаленных мостов/маршрутизаторов является скорость фильтрации и скорость маршрутизации пакетов, которые часто ограничиваются не внутренними возможностями устройства, а скоростью передачи данных по линии. Для устойчивой работы сети скорость маршрутизации устройства должна быть выше, чем средняя скорость межсетевого трафика. При объединении сетей с помощью выделенного канала рекомендуется сначала выяснить характер межсетевого трафика – его среднее значение и пульсацию. Для хорошей передачи пульсаций пропускная способность канала должна быть большей или равной величине пульсаций трафика. Но такой подход приводит к очень нерациональной загрузке канала, так как при коэффициенте пульсаций 50:1 в среднем будет использоваться только 1/50 пропускной способности канала. Поэтому чаще при выборе канала ориентируются на среднее значение межсетевого трафика. Правда, при этом пульсация будет создавать очередь кадров во внутреннем буфере моста или маршрутизатора, так как канал не может передавать данные с такой высокой скоростью, но очередь обязательно рассосется за конечное время, если среднее значение интенсивности межсетевого трафика меньше средней пропускной способности канала.

Для преодоления ограничений на скорость линии, а также для уменьшения части локального трафика, передаваемого по глобальной линии, в удаленных мостах и маршрутизаторах, работающих на глобальные каналы, используются специальные приемы, отсутствующие в локальных устройствах. Эти приемы не входят в стандарты протоколов, но они реализованы практически во всех устройствах, обслуживающих низкоскоростные каналы, особенно каналы со скоростями в диапазоне от 9600 бит/с до 64 Кбит/с.

К таким приемам относятся технологии сжатия пакетов, спуфинга и сегментации пакетов.

Сжатие пакетов (*компрессия*). Некоторые производители, используя собственные алгоритмы, обеспечивают коэффициент сжатия до 8:1 Стандартные алгоритмы сжатия, применяемые в модемах, устройствах DSU/CSU, самих мостах и маршрутизаторах, обеспечивают коэффициент сжатия до 4:1. После сжатия данных для передачи требуется существенно меньшая скорость канала.

Спуфинг (*spoofing*). Эта технология позволяет значительно повысить пропускную способность линий, объединяющих локальные сети, работающие по протоколам с большим количеством широковещательных рассылок. Во многих стеках протоколов для локальных сетей широковещательные рассылки обеспечивают решение задач поиска ресурсов сети. «Спуфинг» означает надувательство, мистификацию. Главной идеей технологии спуфинга является имитация передачи пакета по глобальной сети. Спуфинг используется не только на выделенных каналах, но и на коммутируемых, а также всегда, когда пропускная способность глобальной сети оказывается на границе некоторого минимального уровня.

Рассмотрим технику спуфинга на примере передачи между удаленными сетями пакетов SAP (Service Advertising Protocol – протокол объявления служб) серверами ОС NetWare. Эти пакеты каждый сервер генерирует каждую минуту, чтобы все клиенты сети могли составить правильное представление об имеющихся в сети разделяемых ресурсах – файловых службах, службах печати и т. п. SAP-пакеты распространяются в IPX-пакетах с широковещательным сетевым адресом (ограниченное широковещание). Маршрутизаторы не должны передавать такие пакеты из сети в сеть, но для SAP-пакетов сделано исключение – маршрутизатор, поддерживающий IPX, распространяет его на все порты, кроме того, на который этот пакет поступил (техника, подобная технике split horizon). Это делается для

того, чтобы клиенты работали в одинаковых условиях независимо от сети, в которой они находятся. Удаленные мосты передают SAP-пакеты «по долгу службы», так как они имеют широковещательные MAC-адреса.

Таким образом, по выделенной линии может проходить достаточно большое количество SAP-пакетов, которое зависит от количества серверов в каждой из локальных сетей, а также количества служб, о которых объявляет каждый сервер. Если эти пакеты посылаются каким-либо сервером, но не доходят до клиентов, то клиенты не могут воспользоваться службами этого сервера.

Если маршрутизаторы или мосты, объединяющие сети, поддерживают технику спуфинга, то они передают по выделенному каналу не каждый SAP-пакет, а например, только каждый пятый. Интенсивность служебного трафика в выделенном канале при этом уменьшается. Но для того, чтобы клиенты не теряли из списка ресурсов удаленной сети серверы, маршрутизатор/мост имитирует приход этих пакетов по выделенному каналу, посылая SAP-пакеты от своего имени каждую минуту, как это и положено по протоколу. При этом маршрутизатор/мост посылает несколько раз копию реального SAP-пакета, получаемого раз в 5 минут по выделенному каналу. Такую процедуру маршрутизатор/мост осуществляет для каждого сервера удаленной сети, генерирующего SAP-пакеты.

Существует несколько различных реализаций техники спуфинга: посылка оригинальных пакетов в глобальный канал происходит по времени или по количеству принятых пакетов, при изменениях в содержимом пакетов. Последний способ достаточно логичен, так как сервер обычно каждый раз повторяет содержимое своего объявления – изменения в составе служб происходят редко. Поэтому, как в алгоритмах маршрутизации типа «изменение связей» достаточно передавать только измененные пакеты, так и для подтверждения нормальной работы достаточно периодически пересылать даже неизмененный пакет (в качестве сообщения HELLO).

Существует достаточно много протоколов которые пользуются широковещательными рассылками, и пограничный маршрутизатор/мост должен их все учитывать. Только ОС Unix весьма редко работает но этому способу, так как ее основной коммуникационный стек TCP/IP проектировался для низкоскоростных глобальных линий связи. А такие ОС, как NetWare, Windows NT, OS/2, разрабатывались в основном в расчете на локальные сети, поэтому пропускную способность каналов связи не экономили.

В ОС NetWare существуют три основных типа широковещательных межсетевых сообщений – кроме сообщений SAP, необходимо также передавать сообщения протокола маршрутизации RIP, который программные маршрутизаторы, работающие на серверах NetWare, поддерживают по умолчанию, а также специальные сообщения watchdogs (называемые также keep alive), которыми обмениваются сервер и клиент, установившие логическое соединение. Сообщения watchdogs используются в том случае, когда временно в рамках данной логической сессии пользовательские данные не передаются. Чтобы поддержать соединение, клиент каждые 5 минут посылает такие сообщения серверу, говоря, что он «жив». Если сервер не получает таких сообщений в течение 15 минут, то сеанс с данным клиентом прекращается. В интерфейсе NetBIOS (а его используют в качестве программного интерфейса приложения во многих ОС) порождается служебный трафик разрешения имен – запросы NameQuery посылаются (также широковещательным способом) каждые 20 минут, если зарегистрированное ранее имя не проявило себя в течение этого периода времени.

Для реализации анализа технология спуфинга требует пакетов сетевого уровня и выше. Поэтому для мостов реализация спуфинга – не такое обычное дело, как для маршрутизаторов. Мосты, поддерживающие спуфинг, не строят таблицы маршрутизации и не продвигают пакеты на основе сетевых адресов, но разбор заголовков и содержимого пакетов верхних уровней делают. Такие интеллектуальные удаленные мосты выпускает, например, компания Gandalf, хотя недорогие маршрутизаторы постепенно вытесняют мосты и в этой области.

Сегментация пакетов – позволяет разделять большие передаваемые пакеты и передавать их сразу через две телефонные линии. Хотя это и не делает телефонные каналы более эффективными, но все же увеличивает скорость обмена данными почти вдвое.

Выводы

- Выделенные каналы широко используются для образования глобальных связей между удаленными локальными сетями.
- Выделенные каналы делятся на аналоговые и цифровые в зависимости от аппаратуры

длительной коммутации. В аналоговых каналах используются FDM-коммутаторы, а в цифровых – TDM. Ненагруженные каналы не проходят через мультиплексоры и коммутаторы и используются чаще всего как абонентские окончания для доступа к глобальным сетям.

- Аналоговые каналы делятся на несколько типов: в зависимости от полосы пропускания – на каналы тональной частоты (3100 Гц) и широкополосные каналы (48 кГц), в зависимости от типа окончания – на каналы с 4-проводным окончанием и каналы с 2-проводным окончанием.
- Для передачи компьютерных данных по аналоговым каналам используются модемы – устройства, относящиеся к типу DCE. Модемы для работы на выделенных каналах бывают следующих типов;
    - асинхронные, асинхронно-синхронные и синхронные модемы;
    - модемы для 4- и 2-проводных окончаний;
    - модемы, работающие только в полудуплексном режиме, и дуплексные модемы;
    - модемы, поддерживающие протоколы коррекции ошибок;
    - широкополосные модемы и модемы для канала тональной частоты.
- Широкополосные модемы работают только по 4-проводным окончаниям в дуплексном синхронном режиме. Многие модели модемов для тонального канала могут работать в различных режимах, совмещая, например, поддержку асинхронного и синхронного режимов работы, 4- и 2-проводные окончания. Стандарт V.34+ является наиболее гибким и скоростным стандартом для модемов тонального канала, он поддерживает как выделенные, так и коммутируемые 2-проводные окончания.
- Цифровые выделенные каналы образуются первичными сетями двух поколений технологии - PDH и SONET/SDH. Эти технологии существуют в двух вариантах – североамериканском и европейском. Последний является также международным, соответствующим рекомендациям ITU-T. Два варианта технологий PDH несовместимы.
- В цифровых первичных сетях используется иерархия скоростей каналов, с помощью которой строятся магистральные каналы и каналы доступа. Технология PDH поддерживает следующие уровни иерархии каналов: абонентский канал 64 Кбит/с (DS-0), каналы T1/E1 (DS-1), каналы T2/E2 (DS-2) (редко сдаваемые в аренду) и каналы T3/E3 (DS-3). Скорость DS-4 определена в стандартах ITU-T, но на практике не используется.
- Технология PDH разрабатывалась как асинхронная, поэтому кадры различных скоростей разделяются специальными битами синхронизации. В этом причина основного недостатка каналов этой технологии – для получения доступа к данным одного низкоскоростного абонентского канала необходимо произвести полное демультиплексирование высокоскоростного канала, например E3, а затем снова выполнить мультиплексирование 480 абонентских каналов в канал E3. Кроме того, технология PDH не обеспечивает автоматической реакции первичной сети на отказ канала или порта.
- Технология SONET/SDH ориентируется на использование волоконно-оптических кабелей. Эта технология также включает два варианта – североамериканский (SONET) и европейско-международный (SDH), но в данном случае они являются совместимыми.
- Технология SONET/SDH продолжает иерархию скоростей каналов PDH - до 10 Гбит/с. Технология основана на полной синхронизации между каналами и устройствами сети, которая обеспечивается наличием центрального пункта распределения синхронизирующих импульсов для всей сети.
- Каналы иерархии PDH являются входными каналами для сетей технологии SONET/SDH, которая переносит ее по своим магистральным каналам.
- Синхронная передача кадров различного уровня иерархии позволяет получить доступ к данным низкоскоростного пользовательского канала, не выполняя полного демультиплексирования высокоскоростного потока. Техника указателей позволяет определить начало пользовательских подкадров внутри синхронного кадра и считать их или добавить «на лету». Эта техника называется техникой «вставки и удаления» (add and drop) пользовательских данных.
- Сети SONET/SDH обладают встроенной отказоустойчивостью за счет избыточности своих кадров и способности мультиплексоров выполнять реконфигурирование путей следования данных. Основной отказоустойчивой конфигурацией является конфигурация двойных

волоконно-оптических колец.

- Внутренние протоколы SONET/SDH обеспечивают мониторинг и управление первичной сетью, в том числе удаленное создание постоянных соединений между абонентами сети.
- Первичные сети SONET/SDH являются основой для большинства телекоммуникационных сетей: телефонных, компьютерных, телексных.
- Для передачи компьютерных данных по выделенным каналам любой природы применяется несколько протоколов канального уровня: SLIP, HDLC и PPP. Протокол PPP в наибольшей степени подходит для современных выделенных каналов, аппаратура которых самостоятельно решает задачу надежной передачи данных. Протокол PPP обеспечивает согласование многих важных параметров канального и сетевого уровня при установлении соединения между узлами.
- Для объединения локальных сетей с помощью выделенных каналов применяются такие DTE, как маршрутизаторы и удаленные мосты. В канале с низкой пропускной способностью маршуртизаторы и мосты используют спуфинг, компрессию и сегментацию данных.

6.3. Глобальные связи на основе сетей с коммутацией каналов

Выделенные линии представляют собой наиболее надежное средство соединения локальных сетей через глобальные каналы связи, так как вся пропускная способность такой линии всегда находится в распоряжении взаимодействующих сетей. Однако это и наиболее дорогой вид глобальных связей – при наличии N удаленных локальных сетей, которые интенсивно обмениваются данными друг с другом, нужно иметь Nx(N-l)/2 выделенных линий. Для снижения стоимости глобального транспорта применяют динамически коммутируемые каналы, стоимость которых разделяется между многими абонентами этих каналов.

Наиболее дешевыми оказываются услуги телефонных сетей, так как их коммутаторы оплачиваются большим количеством абонентов, пользующихся телефонными услугами, а не только абонентами, которые объединяют свои локальные сети.

Телефонные сети делятся на аналоговые и цифровые в зависимости от способа мультиплексирования абонентских и магистральных каналов. Более точно, цифровыми называются сети, в которых на абонентских окончаниях информация представлена в цифровом виде в которых используются цифровые методы мультиплексирования и коммутации, а аналоговыми – сети, которые принимают данные от абонентов аналоговой формы, то есть от классических аналоговых телефонных аппаратов, а мультиплексирование и коммутацию осуществляют как аналоговыми методами, так и цифровыми. В последние годы происходил достаточно интенсивный процесс замены коммутаторов телефонных сетей на цифровые коммутаторы, которые работают на основе технологии TDM. Однако такая сеть по-прежнему останется аналоговой телефонной сетью, даже если все коммутаторы будут работать по технологии TDM, обрабатывая данные в цифровой форме, если абонентские окончания у нее останутся аналоговыми, а аналого-цифровое преобразование выполняется на ближней к абоненту АТС сети. Новая технология модемов V.90 смогла использовать факт существования большого количества сетей, в которых основная часть коммутаторов являются цифровыми.

К телефонным сетям с цифровыми абонентскими окончаниями относятся так называемые службы Switched 56 (коммутируемые каналы 56 Кбит/с) и цифровые сети с интегральными услугами ISDN (Intergrated Services Digital Network). Службы Switched 56 появились в ряде западных стран в результате предоставления конечным абонентам цифрового окончания, совместимого со стандартами линий T1. Эта технология не стала международным стандартом, и сегодня она вытеснена технологией ISDN, которая такой статус имеет.

Сети ISDN рассчитаны не только на передачу голоса, но и компьютерных данных, в том числе и с помощью коммутации пакетов, за счет чего они получили название сетей с интегральными услугами. Однако основным режимом работы сетей ISDN остается режим коммутации каналов, а служба коммутации пакетов обладает слишком низкой по современным меркам скоростью – обычно до 9600 бит/с. Поэтому технология ISDN будет рассмотрена в данном разделе, посвященном сетям с коммутацией каналов. Новое поколение сетей с интеграцией услуг, названное B-ISDN (от broadband – широкополосные), основано уже целиком на технике коммутации пакетов (точнее, ячеек технологии ATM), поэтому эта технология будет рассмотрена в разделе, посвященном сетям с коммутацией пакетов.

Пока географическая распространенность аналоговых сетей значительно превосходит распространенность цифровых, особенно в нашей стране, но это отставание с каждым годом сокращается.

6.3.1. Аналоговые телефонные сети

Организация аналоговых телефонных сетей

Наиболее популярными коммутируемыми каналами являются каналы, создаваемые обычными аналоговыми телефонными сетями. В англоязычной литературе их иногда называют POTS (Plain Old Telephone Service), – что-то вроде «старая добрая телефонная служба», хотя, конечно, название PSTN (Public Switched Telephone Network) – «публичная коммутируемая телефонная сеть» является более официальным. К сожалению, эти сети малопригодны для построения магистралей корпоративных сетей. Со средней пропускной способностью 9600 бит/с коммутируемые аналоговые линии, оснащенные модемами, подходят только для пользователя с минимальными требованиями к времени реакции системы. Максимальная на сегодня пропускная способность в 56 Кбит/с достигается только в том случае, если все коммутаторы в сети на пути следования данных являются цифровыми, да и то такая скорость обеспечивается только в направлении «сеть – пользователь».

Чаще всего такие линии используются для индивидуального удаленного доступа к сети или же как резервные линии связи небольших офисов с центральным отделением предприятия. Доступ по телефонной сети имеет англоязычное название «dial-up access». Тем не менее при недостатке средств коммутируемые аналоговые линии обеспечивают связь локальных сетей между собой. Это выгодный режим соединения, если количество передаваемых данных невелико и данные не требуют частого обновления. В этом случае две сети могут соединяться по аналоговой телефонной сети, например, раз в сутки, передавать в течение нескольких минут данные, а затем разрывать соединение. При повременной оплате телефонного соединения такой режим оказывается эффективным. Обычно к нему прибегают для передачи сводок работы предприятия за день, точнее тех частей сводок, которые имеют небольшие объемы (чаще всего – это числовые показатели, без графики).

Ниже перечислены основные характеристики аналоговых телефонных сетей.

- При вызове пользователи получают прямое соединение через коммутаторы в сети. Прямое соединение эквивалентно паре проводов с полосой пропускания от 300 до 3400 Гц. Абонентское окончание 2-проводное.
- Вызов абонента может осуществляться двумя способами: с помощью импульсного набора с частотой 10 Гц или тонового набора с частотой 10 Гц. При импульсном наборе длительность набора зависит от того, какие цифры образуют номер – например, цифра 0 передается десятью последовательными импульсами, цифра 9 – девятью и т. д. При тоновом наборе любая цифра передается подачей в сеть двух синусоидальных сигналов разной частоты в течение 50 мс (сопровождаемых паузой 50 мс). Поэтому набор номера тоновым способом в среднем в 5 раз быстрее, чем импульсный (к сожалению, в нашей стране импульсный набор пока остается основным способом набора во всех городах).
- Коммутаторы сети не позволяют обеспечить промежуточное хранение данных. Поскольку запоминающие устройства в коммутаторах отсутствуют, возможен отказ в соединении при занятости абонента или при исчерпании коммутатором своих возможностей по соединению входных и выходных каналов (занятость АТС).
- Для передачи дискретных данных по аналоговым коммутируемым сетям используются модемы, поддерживающие процедуру вызова абонента.
- Пропускная способность коммутируемого аналогового канала заранее неизвестна, так как модемы устанавливают соединение на скорости, подходящей для реального качества канала. Так как качество коммутируемых каналов меняется в течение сеанса связи, то модемы изменяют скорость передачи данных динамически.

В телефонных коммутаторах аналоговых телефонных сетей могут использоваться два принципа коммутации – аналоговый, основанный на частотном разделении канала (FDM), и цифровой, основанный на разделении канала во времени (TDM).

Системы, работающие по методу частотного уплотнения, подразделяются на электромеханические и программно-управляемые электронные. Электромеханические системы (например, шаговые искатели)

управляются по проводным цепям и приводятся в действие электродвигателями или шаговыми искателями. В электромеханических системах логика маршрутизации встроена в аппаратуру. В программно-управляемых коммутаторах, логика коммутации реализуется программным обеспечением, а сама коммутация выполняется электронным способом.

Электромеханические коммутаторы, естественно, создают значительные помехи в коммутируемых каналах. Кроме того, дополнительные помехи создает сам способ коммутации уплотненных каналов на основе FDM. Это объясняется тем, что коммутировать уплотненные в общий канал сигналы отдельных абонентов невозможно. Перед операцией коммутации всегда нужно провести полное демультиплексирование сигналов абонентских каналов, то есть превратить сигнал высокочастотной несущей (который находится в диапазоне от 60 до 108 кГц для уплотненного канала первого уровня, состоящего из 12 абонентских каналов) в голосовой сигнал со спектром от 300 до 3400 Гц. Только затем такие каналы можно коммутировать с помощью шаговых искателей или электронных ключей. После коммутации абонентские каналы снова уплотняются в высокочастотный канал, но каждый входной канал теперь уже накладывается на несущую другой порядковой частоты, что и соответствует операции коммутации (напомним, что при TDM-коммутации в уплотненном кадре меняется порядок следования байт).

Операция демультиплексирования высокочастотной несущей, а затем повторное наложение сигналов на высокочастотные несущие создает значительные помехи (треск и свист в телефонной трубке), которые существенно снижают качество коммутируемых каналов по сравнению с выделенными аналоговыми. Понятно, что наличие электромеханических элементов только усугубляет картину, а старые АТС с шаговыми искателями еще эксплуатируются (в Москве только совсем недавно была демонтирована АТС 231, которая работала с 30-х годов и была, естественно, электромеханической).

Переход на цифровые методы коммутации существенно повышает качество коммутируемых каналов даже при том, что сигнал от абонента поступает в ближайшую АТС в аналоговой форме, а значит, подвергается на «последней миле» воздействию помех, которые уже невозможно отфильтровать.

Модемы для работы на коммутируемых аналоговых линиях

Для передачи данных по аналоговым коммутируемым телефонным каналам используются модемы, которые:

- поддерживают процедуру автовызова абонента;
- работают по 2-проводному окончанию, так как в телефонных сетях для коммутируемых каналов предусмотрено именно это окончание.

Чаще всего сегодня для коммутируемых каналов используются те же модели модемов, что и для выделенных, так как последние стандарты определяют два режима работы – по выделенным каналам и по коммутируемым. Естественно, такие комбинированные модели дороже моделей, поддерживающих только один режим работы – по коммутируемым каналам.

Для передачи данных по коммутируемым каналам CCITT разработал ряд основных стандартов, определяющих скорость и метод кодирования сигналов.

Стандарты первой группы являются основными и состоят из следующих спецификаций:

- V.21 – дуплексная асинхронная передача данных на скорости 300 бит/с;
- V.22 – дуплексная асинхронная/синхронная передача данных на скорости 1,2 Кбит/с;
- V.22 bis -дуплексная асинхронная/синхронная передача данных на скоростях 1,2 и 2,4 Кбит/с;
- V.26 ter – дуплексная асинхронная/синхронная передача данных на скоростях 1,2 и 2,4 Кбит/с;
- V.32 – дуплексная асинхронная/синхронная передача данных на скоростях 4,8 и 9,6 Кбит/с;
- V.32 bis – дуплексная асинхронная/синхронная передача на скорости до 14,4 Кбит/с;
- V.34 – дуплексная передача на скорости до 28,8 Кбит/с;
- V.34+ – дуплексная передача на скорости до 33,6 Кбит/с.

На практике сегодня в основном применяют модемы, поддерживающие стандарт V.34+, которые могут адаптироваться к качеству линии.

Для реализации функций автовызова современные модемы поддерживают несколько способов. При работе с модемом по асинхронному интерфейсу обычно используется система команд, предложенная компанией Hayes для своей модели Smartmodem в начале 80-х годов. Каждая команда состоит из набора обычных символов, передаваемых модему в старт-стопном режиме. Например, для указания набора номера в импульсном режиме необходимо послать модему команду ATDP. Это можно сделать даже

вручную, если модем подключен к обычному алфавитно-цифровому терминалу через интерфейс RS-232C.

Для синхронных интерфейсов между модемом и DTE используются два стандарта автонабора номера: V.25 и V.25bis. Стандарт V.25 требует, чтобы, помимо основного интерфейса для передачи данных, модем соединялся с DTE отдельным интерфейсом V.25/RS-366 на специальном 25-контактном разъеме. В стандарте V.25 bis для передачи команд автовызова предусмотрен тот же разъем, что и в основном интерфейсе, то есть RS-232C. Интерфейсы V.25 и V.25 bis могут работать не только в синхронном режиме с DTE, но и в асинхронном, но в основном характерны для синхронных интерфейсов, так как в асинхронном режиме для автовызова чаще используются Hayes-команды.

Для модемов, работающих с DTE по асинхронному интерфейсу, комитет CCITT разработал протокол коррекции ошибок V.42. До его принятия в модемах, работающих по асинхронному интерфейсу, коррекция ошибок обычно выполнялась по протоколам фирмы Microcom, еще одного лидера в области модемных технологий. Эта компания реализовала в своих модемах несколько различных процедур коррекции ошибок, назвав их протоколами MNP (Microcom Networking Protocol) классов 2-4.

В стандарте V.42 основным является другой протокол – протокол LAP-M (Link Access Protocol for Modems). Однако стандарт V.42 поддерживает и процедуры MNP 2-4, поэтому модемы, соответствующие рекомендации V.42, позволяют устанавливать свободную от ошибок связь с любым модемом, поддерживающим этот стандарт, а также с любым MNP-совместимым модемом. Протокол LAP-M принадлежит семейству HDLC и в основном работает так же, как и другие протоколы этого семейства – с установлением соединения, кадрированием данных, нумерацией кадров и восстановлением кадров с поддержкой метода скользящего окна. Основное отличие от других протоколов этого семейства – наличие кадров XID и BREAK. С помощью кадров XID (eXchange Identification) модемы при установлении соединения могут договориться о некоторых параметрах протокола, например о максимальном размере поля данных кадра, о величине тайм-аута при ожидании квитанции, о размере окна и т.п. Эта процедура, напоминает переговорные процедуры протокола PPP. Команда BREAK (BRK) служит для уведомления модема-напарника о том, что поток данных временно приостанавливается. При асинхронном интерфейсе с DTE такая ситуация может возникнуть. Команда BREAK посылается в ненумерованном кадре, она не влияет на нумерацию потока кадров сеанса связи. После возобновления поступления данных модем возобновляет и отправку кадров, как если бы паузы в работе не было.

Почти все современные модемы при работе по асинхронному интерфейсу поддерживают стандарты сжатия данных CCITT V.42bis и MNP-5 (обычно с коэффициентом 1:4, некоторые модели – до 1:8). Сжатие данных увеличивает пропускную способность линии связи. Передающий модем автоматически сжимает данные, а принимающий их восстанавливает. Модем, поддерживающий протокол сжатия, всегда пытается установить связь со сжатием данных, но если второй модем этот протокол не поддерживает, то и первый модем перейдет на обычную связь без сжатия.

При работе модемов по синхронному интерфейсу наиболее популярным является протокол компрессии SDC (Synchronous Data Compression) компании Motorola.

Новый модемный стандарт V.90 является технологией, направленной на обеспечение недорогого и быстрого способа доступа пользователей к сетям поставщиков услуг. Этот стандарт обеспечивает асимметричный обмен данными: со скоростью 56 Кбит/с из сети и со скоростью 30-40 Кбит/с в сеть. Стандарт совместим со стандартом V.34+.

Основная идея технологии асимметричных модемов состоит в следующем. В современных телефонных сетях часто единственным аналоговым звеном в соединении с сервером удаленного доступа является телефонная пара, связывающая модем компьютера с коммутатором телефонной станции. Этот канал оптимизирован для передачи речевых сигналов: максимальная скорость передачи данных определяется из условия предельно допустимого соотношения между шумами физической линии передачи и погрешностью дискретизации звукового сигнала при его оцифровывании. Эта величина задается стандартом V.34+ и равна 33,6 Кбит/с.

Однако все выше приведенные соображения справедливы только для одного направления передачи данных – от аналогового модема к телефонной станции. Именно на этом участке выполняется аналого-цифровое преобразование, которое вносит погрешность квантования. Эта погрешность добавляется к другим помехам линии и ограничивает скорость передачи 33,6 Кбит/с. Обратное же цифро-аналоговое преобразование не вносит дополнительного шума, что делает возможным увеличение скорости передачи от телефонной станции к модему пользователя до 56 Кбит/с.

Достоинством новой технологии является то, что для ее внедрения не требуется вносить какие-либо изменения в оборудование телефонной станции – нужно лишь изменить программу в цифровых модемах, установленных в стойках у поставщика услуг, а также загрузить в пользовательский модем новую программу либо заменить микросхему памяти в зависимости от модели и производителя.

Технологии асимметричных модемов рассчитаны на то, что сервер удаленного доступа поставщика услуг корпоративной или публичной сети с коммутацией пакетов подключен к какой-либо АТС телефонной сети по цифровому интерфейсу, например BRI ISDN, или же по выделенному каналу T1/E1. Так что цифровой поток данных, идущий от сервера, постоянно пересылается сетью в цифровой форме и только на абонентском окончании преобразуется в аналоговую форму. Если же сервер удаленного доступа подключен к телефонной сети по обычному аналоговому окончанию, то даже наличие модема V.90 у сервера не спасет положение – данные будут подвергаться аналого-цифровому преобразованию, и их максимальная скорость не сможет превысить 33,6 Кбит/с. При подключении же модемов V.90 к телефонной сети с обеих сторон обычным способом, то есть через аналоговые окончания, они работают как модемы V.34+. Такая же картина будет наблюдаться в случае, если в телефонной сети на пути трафика встретится аналоговый коммутатор.

6.3.2. Служба коммутируемых цифровых каналов Switched 56

Если все коммутаторы телефонной сети работают по технологии цифровой коммутации TDM, то кажется, что перевод абонентского окончания на передачу данных в цифровой форме – не такая уж сложная вещь. И, имея сеть цифровых телефонных коммутаторов, нетрудно сделать ее полностью цифровой. Однако это не так. Передача данных со скоростью 64 Кбит/с в дуплексном режиме требует либо прокладки между жилыми домами и АТС новых кабелей, либо специальных усилителей-регенераторов на обоих концах абонентского окончания, то есть в том числе и в квартирах. Оба способа связаны с большими затратами труда и материальных средств, так как аналоговые телефонные сети вполне довольствуются тем медным проводом, который в больших количествах уже проложен между АТС и домами того района города, который данная АТС обслуживает и который заканчивается пассивной телефонной розеткой, а не усилителем-регенератором.

Поэтому массовый переход на полностью цифровые телефонные сети связан с большими капиталовложениями и требует значительного времени для его осуществления, что и показала жизнь.

Однако для некоторых особо требовательных абонентов, которые согласны заплатить за повышения качества и скорости коммутируемых каналов, телефонные компании уже достаточно давно предлагают цифровые коммутируемые службы. Обычно такими абонентами являются корпоративные абоненты, которым нужен быстрый и качественный доступ к корпоративной информации.

Одной из первых служб такого рода стала служба Switched 56, предлагаемая различными телекоммуникационными компаниями в США, Англии и некоторых других странах. Технология этой службы (которая в разных странах имеет разное название, например, в Англии – Kilostream) основана на 4-проводном окончании каналов T1. Абонент для подключения к сети должен установить у себя соответствующее оборудование, представляющее собой DSU/CSU со встроенным блоком автовызова. Использование 8-го бита для передачи номера вызываемого абонента, а также для других служебных целей ограничивает скорость передачи данных до 56 Кбит/с. Типичная схема функционирования службы Switched 56 показана на рис. 6.15.

Абонентами обычно являются компьютеры или локальные сети, подключаемые к сети с помощью маршрутизатора или удаленного моста. Местные станции соединяются с некоторой центральной станцией, которая коммутирует цифровые потоки T1/E1. Сеть является полностью цифровой и поддерживает различные скорости передачи данных – от 2400 бит/с до 56 Кбит/с. Абоненты службы Switched 56; подключаются также к общей публичной телефонной сети, однако соединения со скоростью 56 Кбит/с возможны только в том случае, когда оба абонента пользуются этой службой.

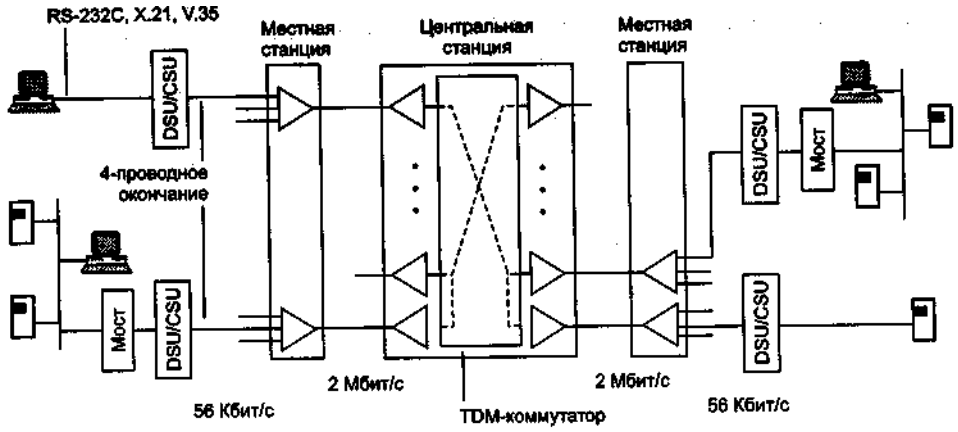


Рис. 6.15. Функционирование службы Switched 56

Стандарты службы Switched 56 разные в разных компаниях и разных странах. Сегодня этот вид службы вытесняется сетями ISDN, стандарты которых являются международными, хотя в этой области также имеются проблемы совместимости сетей разных стран.

6.3.3. ISDN – сети с интегральными услугами

Цели и история создания технологии ISDN

ISDN (*Integrated Services Digital Network – цифровые сети с интегральными услугами*) относятся к сетям, в которых основным режимом коммутации является режим коммутации каналов, а данные обрабатываются в цифровой форме. Идеи перехода телефонных сетей общего пользования на полностью цифровую обработку данных, при которой конечный абонент передает данные непосредственно в цифровой форме, высказывались давно. Сначала предполагалось, что абоненты этой сети будут передавать только голосовые сообщения. Такие сети получили название IDN – Integrated Digital Network. Термин «интегрированная сеть» относился к интеграции цифровой обработки информации сетью с цифровой передачей голоса абонентом. Идея такой сети была высказана еще в 1959 году. Затем было решено, что такая сеть должна предоставлять своим абонентам не только возможность поговорить между собой, но и воспользоваться другими услугами – в первую очередь передачей компьютерных данных. Кроме того, сеть должна была поддерживать для абонентов разнообразные услуги прикладного уровня – факсимильную связь, телетекс (передачу данных между двумя терминалами), видеотекс (получение хранящихся в сети данных на свой терминал), голосовую почту и ряд других. Предпосылки для создания такого рода сетей сложились к середине 70-х годов. К этому времени уже широко применялись цифровые каналы T1 для передачи данных в цифровой форме между АТС, а первый мощный цифровой коммутатор телефонных каналов 4ESS был выпущен компанией Western Electric в 1976 году.

В результате работ, проводимых по стандартизации интегральных сетей в CCITT, в 1980 году появился стандарт G.705, в котором излагались общие идеи такой сети. Конкретные спецификации сети ISDN появились в 1984 году в виде серии рекомендаций I. Этот набор спецификаций был неполным и не подходил для построения законченной сети. К тому же в некоторых случаях он допускал неоднозначность толкования или был противоречивым. В результате, хотя оборудование ISDN и начало появляться примерно с середины 80-х годов, оно часто было несовместимым, особенно если производилось в разных странах. В 1988 году рекомендации серии I были пересмотрены и приобрели гораздо более детальный и законченный вид, хотя некоторые неоднозначности сохранились. В 1992 и 1993 годах стандарты ISDN были еще раз пересмотрены и дополнены. Процесс стандартизации этой технологии продолжается.

Внедрение сетей ISDN началось достаточно давно – с конца 80-х годов, однако высокая техническая сложность пользовательского интерфейса, отсутствие единых стандартов на многие жизненно важные функции, а также необходимость крупных капиталовложений для переоборудования телефонных АТС и каналов связи привели к тому, что инкубационный период затянулся на многие годы, и сейчас, когда прошло уже более десяти лет, распространенность сетей ISDN оставляет желать лучшего. Кроме того, в разных странах судьба ISDN складывалась по-разному. Наиболее давно в национальном масштабе эти

сети работают в таких странах, как Германия и Франция. Тем не менее доля абонентов ISDN даже в этих странах составляет немногим более 5 % от общего числа абонентов телефонной сети. В США процесс внедрения сетей ISDN намного отстал от Европы, поэтому сетевая индустрия только недавно заметила наличие такого рода сетей. Если судить о тех или иных типах глобальных сетей по коммуникационному оборудованию для корпоративных сетей, то может сложиться ложное впечатление, что технология ISDN появилась где-то в 1994 - 1995 годах, так как именно в эти годы начали появляться маршрутизаторы с поддержкой интерфейса ISDN. Это обстоятельство просто отражает тот факт, что именно в эти годы сеть ISDN стала достаточно распространенной в США – стране, компании которой являются лидерами в производстве сетевого оборудования для корпоративных сетей.

Архитектура сети ISDN предусматривает несколько видов служб (рис. 6.16):

- 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 работой сети.

Как видно из приведенного списка, транспортные службы сетей ISDN действительно покрывают очень широкий спектр услуг, включая популярные услуги frame relay. Кроме того, большое внимание уделено средствам контроля сети, которые позволяют маршрутизировать вызовы для установления соединения с абонентом сети, а также осуществлять мониторинг и управление сетью. Управляемость сети обеспечивается интеллектуальностью коммутаторов и конечных узлов сети, поддерживаюших стек протоколов, в том числе и специальных протоколов управления.

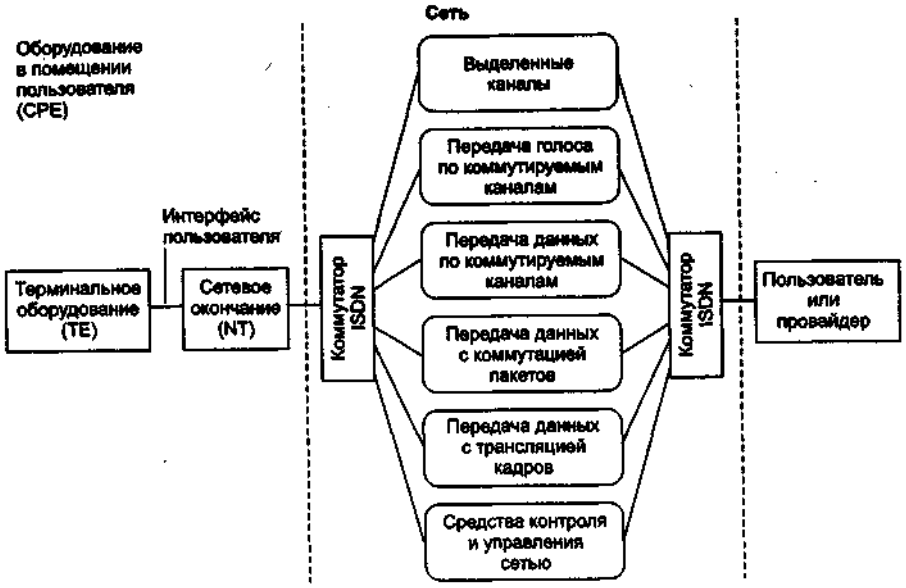


Рис. 6.16. Службы ISDN

Стандарты ISDN описывают также ряд услуг прикладного уровня: факсимильную связь на скорости 64 Кбит/с, телексную связь на скорости 9600 бит/с, видео-текс на скорости 9600 бит/с и некоторые другие

На практике не все сети ISDN поддерживают все стандартные службы. Служба frame relay хотя и была разработана в рамках сети ISDN, однако реализуется, как правило, с помощью отдельной сети коммутаторов кадров, не пересекающейся с сетью коммутаторов ISDN.

Базовой скоростью сети ISDN является скорость канала DS-0, то есть 64 Кбит/с. Эта скорость ориентируется на самый простой метод кодирования голоса – ИКМ, хотя дифференциальное кодирование и позволяет передавать голос с тем же качеством на скорости 32 или 16 Кбит/с.

Пользовательские интерфейсы ISDN

Одним из базовых принципов ISDN является предоставление пользователю стандартного

интерфейса, с помощью которого пользователь может запрашивать у сети разнообразные услуги. Этот интерфейс образуется между двумя типами оборудования, устанавливаемого в помещении пользователя (Customer Premises Equipment, CPE): терминальным оборудованием пользователя TE (компьютер с соответствующим адаптером, маршрутизатор, телефонный аппарат) и сетевым окончанием NT, которое представляет собой устройство, завершающее канал связи с ближайшим коммутатором ISDN.

Пользовательский интерфейс основан на каналах трех типов:

- B – со скоростью передачи данных 64 Кбит/с;
- D – со скоростью передачи данных 16 или 64 Кбит/с;
- H – со скоростью передачи данных 384 Кбит/с (H0), 1536 Кбит/с (H11) или 1920 Кбит/с (H12)

Каналы типа B обеспечивают передачу пользовательских данных (оцифрованного голоса, компьютерных данных или смеси голоса и данных) и с более низкими скоростями, чем 64 Кбит/с. Разделение данных выполняется с помощью техники TDM. Разделением канала B на подканалы в этом случае должно заниматься пользовательское оборудование, сеть ISDN всегда коммутирует целые каналы типа B. Каналы типа B могут соединять пользователей с помощью техники коммутации каналов друг с другом, а также образовывать так называемые полупостоянные (semipermanent) соединения, которые эквиваленты соединениям службы выделенных каналов. Канал типа B может также подключать пользователя к коммутатору сети X.25.

Канал типа D выполняет две основные функции. Первой и основной является передача адресной информации, на основе которой осуществляется коммутация каналов типа B в коммутаторах сети. Второй функцией является поддержание услуг низкоскоростной сети с коммутацией пакетов для пользовательских данных. Обычно эта услуга выполняется сетью в то время, когда каналы типа D свободны от выполнения основной функции.

Каналы типа H предоставляют пользователям возможности высокоскоростной передачи данных. На них могут работать службы высокоскоростной передачи факсов, видеоинформации, качественного воспроизведения звука.

Пользовательский интерфейс ISDN представляет собой набор каналов определенного типа и с определенными скоростями.

Сеть ISDN поддерживает два типа пользовательского интерфейса – начальный (Basic Rate Interface, BRI) и основной (Primay Rate Interface, PRI).

Начальный интерфейс BRI предоставляет пользователю два канала по 64 Кбит/с для передачи данных (каналы типа B) и один канал с пропускной способностью 16 Кбит/с для передачи управляющей информации (канал типа D). Все каналы работают в полнодуплексном режиме. В результете суммарная скорость интерфейса BRI для пользовательских данных составляет 144 Кбит/с по каждому направлению, а с учетом служебной информации – 192 Кбит/с. Различные каналы пользовательского интерфейса разделяют один и тот же физический двухпроводный кабель по технологии TDM, то есть являются логическими каналами, а не физическими. Данные по интерфейсу BRI передаются кадрами, состоящими из 48 бит. Каждый кадр содержит по 2 байта каждого из B каналов, а также 4 бита канала D. Передача кадра длится 250 мс, что обеспечивает скорость данных 64 Кбит/с для каналов B и 16 Кбит/с для канала D. Кроме бит данных кадр содержит служебные биты для обеспечения синхронизации кадров, а также обеспечения нулевой постоянной составляющей электрического сигнала.

Интефейс BRI может поддерживать не только схему 2B+D, но и B+D и просто D (когда пользователь направляет в сеть только пакетизированные данные).

Начальный интерфейс стандартизован в рекомендации I.430.

Основной интерфейс PRI предназначен для пользователей с повышенными требованиями к пропускной способности сети. Интерфейс PRI поддерживает либо схему 30B+D, либо схему 23B+D. В обеих схемах канал D обеспечивает скорость 64 Кбит/с. Первый вариант предназначен для Европы, второй – для Северной Америки и Японии. Ввиду большой популярности скорости цифровых каналов 2,048 Мбит/с в Европе и скорости 1,544 Мбит/с в остальных регионах, привести стандарт на интерфейс PRI к общему варианту не удалось.

Возможны варианты интерфейса PRI с меньшим количеством каналов типа B, например 20B+D. Каналы типа B могут объединяться в один логический высокоскоростной канал с общей скоростью до 1920 Кбит/с. При установке у пользователя нескольких интерфейсов PRI все они могут иметь один канал типа D, при этом количество B каналов в том интерфейсе, который не имеет канала D, может увеличиваться до 24 или 31.

Основной интерфейс может быть основан на каналах типа H. При этом общая пропускная способность интерфейса все равно не должна превышать 2,048 или 1,544 Мбит/с. Для каналов HO возможны интерфейсы 3HO+D для американского варианта и 5HO+D для европейского. Для каналов H1 возможен интерфейс, состоящий только из одного канала H11 (1,536 Мбит/с) для американского варианта или одного канала H12 (1,920 Мбит/с) и одного канала D для европейского варианта.

Кадры интерфейса PRI имеют структуру кадров DS-1 для каналов T1 или E1.

Основной интерфейс PRI стандартизован в рекомендации 1.431.

Подключение пользовательского оборудования к сети ISDN

Подключение пользовательского оборудования к сети ISDN осуществляется в соответствии со схемой подключения, разработанной CCITT (рис. 6.17). Оборудование делится на функциональные группы, и в зависимости от группы различается несколько справочных точек (reference points) соединения разных групп оборудования между собой.

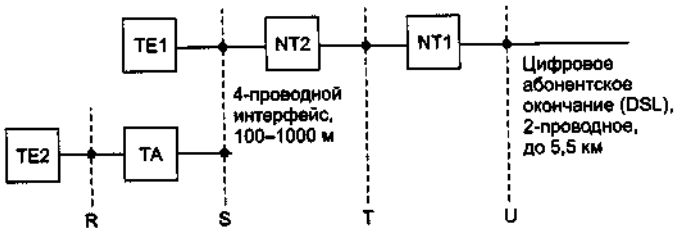


Рис. 6.17. Подключение пользовательского оборудования ISDN

Устройства функциональной группы. NT1 (*Network Termination 1*) образуют *цифровое абонентское окончание* (*Digital Suscriber Line, DSL*) на кабеле, соединяющем пользовательское оборудование с сетью ISDN. Фактически NT1 представляет собой устройство типа CSU, которое работает на физическом уровне и образует дуплексный канал с соответствующим устройством CSU, установленном на территории оператора сети ISDN. Справочная точка U соответствует точке подключения устройства NT1 к сети. Устройство NT1 может принадлежать оператору сети (хотя всегда устанавливается в помещении пользователя), а может принадлежать и пользователю. В Европе принято считать устройство NT1 частью оборудования сети, поэтому пользовательское оборудование (например, маршрутизатор с интерфейсом ISDN) выпускается без встроенного устройства NT1. В Северной Америке принято считать устройство NT1 принадлежностью пользовательского оборудования, поэтому для этого применения оборудование часто выпускается со встроенным устройством NT1.

Если пользователь подключен через интерфейс BRI, то цифровое абонентское окончание выполнено по 2-проводной схеме (как и обычное окончание аналоговой телефонной сети). Для организации дуплексного режима используется технология одновременной выдачи передатчиками потенциального кода 2B1Q с эхо-подавлением и вычитанием своего сигнала из суммарного. Максимальная длина абонентского окончания в этом случае составляет 5,5 км.

При использовании интерфейса PRI цифровое абонентское окончание выполняется по схеме канала T1 или E1, то есть является 4-проводным с максимальной длиной около 1800 м.

Устройства функциональной группы NT2 (*Network Termination 2*) представляют собой устройства канального или сетевого уровня, которые выполняют функции концентрации пользовательских интерфейсов и их мультиплексирование. Например, к этому типу оборудования относятся: офисная АТС (PBX), коммутирующая несколько интерфейсов BRI, маршрутизатор, работающий в режиме коммутации пакетов (например, по каналу D), простой мультиплексор TDM, который мультиплексирует несколько низкоскоростных каналов в один канал типа B. Точка подключения оборудования типа NT2 к устройству NT1 называется справочной точкой типа T. Наличие этого типа оборудования не является обязательным в отличие от NT1.

Устройства функциональной группы TE1 (*Terminal Equipment 1*) относятся к устройствам, которые поддерживают интерфейс пользователя BRI или PRI. Справочная точка S соответствует точке подключения отдельного терминального оборудования, поддерживающего один из интерфейсов пользователя ISDN. Таким оборудованием может быть цифровой телефон или факс-аппарат. Так как оборудование типа NT2 может отсутствовать, то справочные точки S и T объединяются и обозначаются как S/T.

Устройства функциональной группы TE2 (*Terminal Equipment 2*) представляют собой устройства, которые не поддерживают интерфейс BRI или PRI. Таким устройством может быть компьютер, маршрутизатор с последовательными интерфейсами, не относящимися к ISDN, например RS-232C, X.21 или V.35. Для подключения такого устройства к сети ISDN необходимо использовать *терминальный адаптер* (*Terminal Adaptor, TA*). Для компьютеров терминальные адаптеры выпускаются в формате сетевых адаптеров – как встраиваемая карта.

Физически интерфейс в точке S/T представляет собой 4-проводную линию. Так как кабель между устройствами TE1 или TA и сетевым окончанием NT1 или NT2 обычно имеет небольшую длину, то разработчики стандартов ISDN решили не усложнять оборудование, так как организация дуплексного режима на 4-проводной линии намного легче, чем на 2-проводной. Для интерфейса BRI в качестве метода кодирования выбран биполярный AMI, причем логическая единица кодируется нулевым потенциалом, а логический ноль – чередованием потенциалов противоположной полярности. Для интерфейса PRI используются другие коды, те же, что и для интерфейсов T1 и E1, то есть соответственно B8ZS и HDB3.

Физическая длина интерфейса PRI колеблется от 100 до 1000 м в зависимости от схемы подключения устройств (рис. 6.18).

Дело в том, что при небольшом количестве терминалов (TE1 или TE2+TA) разрешается не использовать местную офисную АТС, а подключать до 8 устройств к одному устройству типа NT1 (или NT2 без коммутационных возможностей) с помощью схемы монтажного ИЛИ (подключение напоминает подключение станций к коаксиальному кабелю Ethernet, но только в 4-проводном варианте), При подключении одного устройства TE (через терминальные резисторы R, согласующие параметры линии) к сетевому окончанию NT (см. рис. 6.18, *а*) длина кабеля может достигать 1000 м. При подключении нескольких устройств к пассивному кабелю (см. рис. 6.18, *б*) максимальная длина кабеля сокращается до 100-200 м. Правда, если эти устройства сосредоточены на дальнем конце кабеля (расстояние между ними не превышает 25-50 м), то длина кабеля может быть увеличена до 500 м (см. рис. 6.18, *в*). И наконец, существуют специальные многопортовые устройства NT1, которые обеспечивают звездообразное подключение до 8 устройств, при этом длина кабеля увеличивается до 1000 м (см. рис. 6.18, *г*).

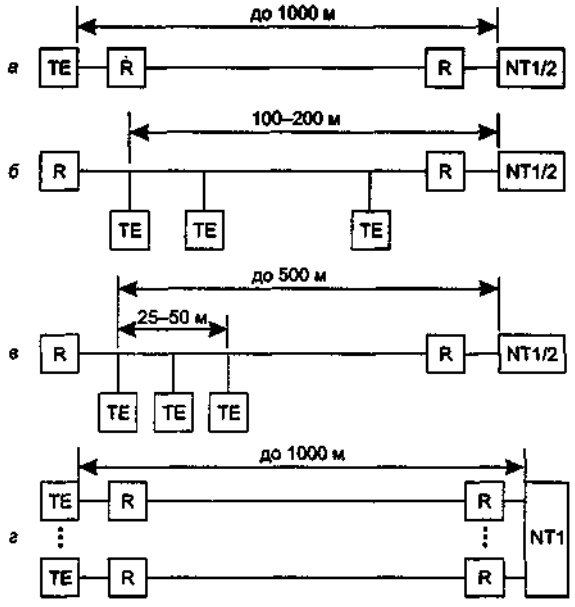


Рис. 6.18. Многоточечное подключение терминалов к сетевому окончанию

Адресация в сетях ISDN

Технология ISDN разрабатывалась как основа всемирной телекоммуникационной сети, позволяющей связывать как телефонных абонентов, так и абонентов других глобальных сетей – компьютерных, телексных. Поэтому при разработке схемы адресации узлов ISDN необходимо было, во-первых, сделать эту схему достаточно емкой для всемирный адресации, а во-вторых, совместимой со схемами адресации других сетей, чтобы абоненты этих сетей, в случае соединения своих сетей через сеть ISDN, могли бы

пользоваться привычными форматами адресов. Разработчики стека TCP/IP пошли по пути введения собственной системы адресации, независимой от систем адресации объединяемых сетей. Разработчики технологии ISDN пошли по другому пути – они решили добиться использования в адресе ISDN адресов объединяемых сетей.

Основное назначение ISDN – это передача телефонного трафика. Поэтому за основу адреса ISDN был взят формат международного телефонного плана номеров, описанный в стандарте ITU-T E.163. Однако этот формат был расширен для поддержки большего числа абонентов и для использования в нем адресов других сетей, например X.25. Стандарт адресации в сетях ISDN получил номер E.164.

Формат E. 163 предусматривает до 12 десятичных цифр в номере, а формат адреса ISDN в стандарте E.164 расширен до 55 десятичных цифр. В сетях ISDN различают *номер абонента и адрес абонента*. Номер абонента соответствует точке T подключения всего пользовательского оборудования к сети. Например, вся офисная АТС может идентифицироваться одним номером ISDN. Номер ISDN состоит из 15 десятичных цифр и делится, как и телефонный номер по стандарту E.163, на поле «Код страны» (от 1 до 3 цифр), поле «Код города» и поле «Номер абонента». Адрес ISDN включает номер плюс до 40 цифр подадреса. Подадрес используется для нумерации терминальных устройств за пользовательским интерфейсом, то есть подключенных к точке S. Например, если на предприятии имеется офисная АТС, то ей можно присвоит один номер, например 7-095-640-20-00, а для вызова абонента, имеющего подадрес 134, внешний абонент должен набрать номер 7-095-640-20-00-134.

При вызове абонентов из сети, не относящейся к ISDN, их адрес может непосредственно заменять адрес ISDN. Например, адрес абонента сети X.25, в которой используется система адресации по стандарту X.121, может быть помещен целиком в поле адреса ISDN, но для указания, что это адрес стандарта X.121, ему должно предшествовать поле префикса, в которое помещается код стандарта адресации, в данном случае стандарта X.121. Коммутаторы сети ISDN могут обработать этот адрес корректно и установить) связь с нужным абонентом сети X.25 через сеть ISDN – либо коммутируя канал типа B с коммутатором X.25, либо передавая данные по каналу типа D в режиме коммутации пакетов. Префикс описывается стандартом ISO 7498.

Стандарт ISO 7498 определяет достаточно сложный формат адреса, причем основой схемы адресации являются первые два поля. Поле AFI (Athority and Format Identifier) задает значения всех остальных полей адреса и формат этих полей. Значением поля AFI является один из 6 типов поддоменов глобального домена адресации:

- четыре типа доменов соответствуют четырем типам публичных телекоммуникационных сетей – сетей с коммутацией пакетов, телексных сетей, публичных телефонных сетей и сетей ISDN;
- пятый тип домена – это географический домен, который назначается каждой стране (в одной стране может быть несколько географических доменов);
- шестой тип домена – это домен организационного типа, в который входят международные организации, например ООН или ATM Forum.

За полем AFI идет поле *IDI* (*Initial Domain Identifier*) – *поле начального идентификатора домена*, а за ним располагается дополнительное поле DSP (Domain Specific Part), которое может нести дополнительные цифры номера абонента, если разрядности поля IDI не хватает.

Определены следующие значения AFI:

- Международные сети с коммутацией пакетов со структурой адресов в стандарте X.121-36, если адрес задается только десятичными цифрами, и 37, если адрес состоит из произвольных двоичных значений. При этом поле, IDI имеет формат в 14 десятичных цифр, а поле DSP может содержать еще 24 цифры.
- Международные сети ISDN со структурой адресов в стандарте E.164-44, если адрес задается только десятичными цифрами, и 45, если адрес состоит из произвольных двоичных значений. При этом поле IDI имеет формат в 15 десятичных цифр, а поле DSP может содержать еще 40 цифр.
- Международные телефонные сети PSTN со структурой адресов в стандарте E.163 - 42, если адрес задается только десятичными цифрами, и 43, если адрес состоит из произвольных двоичных значений. При этом поле IDI имеет формат в 12 десятичных цифр, а поле DSP может содержать еще 26 цифр.
- Международные географические домены со структурой адресов в стандарте ISO DCC (Digital Country Codes) – 38, если адрес задается только десятичными цифрами, и 39, если адрес

состоит из произвольных двоичных значений. При этом поле IDI имеет формат в 3 десятичных цифры (код страны), а поле DSP может содержать еще 35 цифр.

- Домен международных организаций. Для него однобайтовое поле IDI содержит код международной организации, от которой зависит формат поля DSP.

Для первых четырех доменов адрес абонента помещается непосредственно в поле IDI. Для пятого и шестого типов доменов IDI содержит только код страны или код организации, которая контролирует структуру и нумерацию части DSP.

Еще одним способом вызова абонентов из других сетей является указание в адресе ISDN двух адресов: адреса ISDN пограничного устройства, например, соединяющего сеть ISDN с сетью X.25, и адреса узла в сети X.25. Адреса должны разделяться специальным разделителем. Два адреса используются за два этапа – сначала сеть ISDN устанавливает соединение типа коммутируемого канала с пограничным устройством, присоединенным к сети ISDN, а затем передает ему вторую часть адреса, чтобы это устройство осуществило соединение с требуемым абонентом.

Стек протоколов и структура сети ISDN

В сети ISDN (рис. 6.19) существуют два стека протоколов: стек каналов типа D и стек каналов типа B.

Каналы типа D образуют достаточно традиционную сеть с коммутацией пакетов. Прообразом этой сети послужила технология сетей X.25. Для сети каналов D определены три уровня протоколов: физический протокол определяется стандартом I.430/431, канальный протокол LAP-D определяется стандартом Q.921, а на сетевом уровне может использоваться протокол Q.931, с помощью которого выполняется маршрутизация вызова абонента службы с коммутацией каналов, пли же протокол X.25 – в этом случае в кадры протокола LAP-D вкладываются пакеты X.25 и коммутаторы ISDN выполняют роль коммутаторов X.25.

Сеть каналов типа D внутри сети ISDN служит транспортным уровнем для так называемой *системы сигнализации номер 7* (*Signal System Number 7, SS7*). Система SS7 была разработана для целей внутреннего мониторинга и управления коммутаторами телефонной сети общего назначения. Эта система применяется и в сети ISDN. Служба SS7 относится к прикладному уровню модели OSI. Конечному пользователю ее услуги недоступны, так как сообщениями SS7 коммутаторы сети обмениваются только между собой.

Каналы типа B образуют сеть е коммутацией цифровых каналов. В терминах модели OSI на каналах, типа B в коммутаторах сети ISDN определен только протокол физического уровня протокол I.430/431. Коммутация каналов типа B происходит по указаниям, полученным по каналу D. Когда пакеты протокола Q.931 маршрутизируются коммутатором, то при этом происходит одновременная коммутация очередной части составного канала от исходного абонента к конечному.

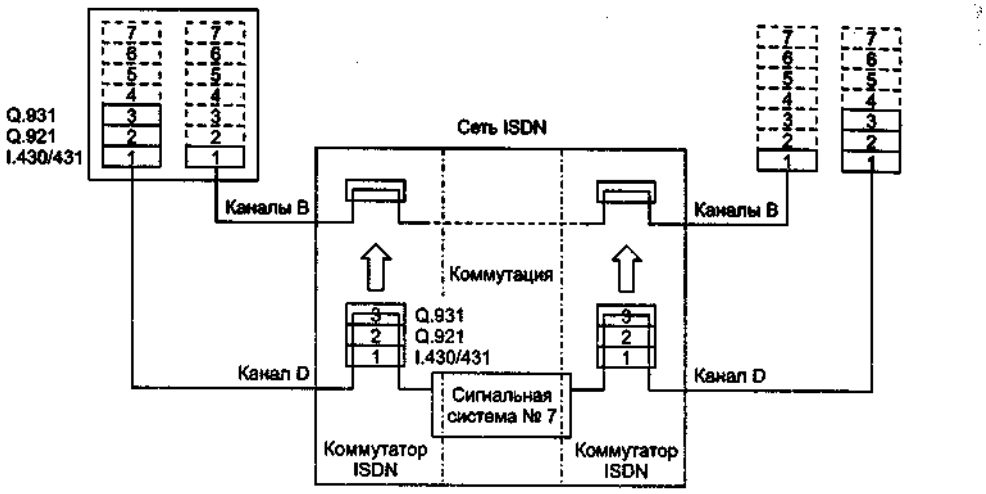


Рис. 6.19. Структура сети ISDN

Протокол LAP-D принадлежит семейству HDLC и обладает всеми родовыми чертами этого семейства, но отличается некоторыми особенностями. Адрес кадра LAP-D состоит из двух байт – один байт определяет код службы, которой пересылаются вложенные в кадр пакеты, а второй используется

для адресации одного из терминалов, если у пользователя к сетевому окончанию NT1 подключено несколько терминалов. Терминальное устройство может поддерживать разные службы – службу установления соединения по протоколу Q.931, службу коммутации пакетов X.25, службу мониторинга сети и т. п. Протокол LAP-D обеспечивает два режима работы: с установлением соединения (единственный режим работы протокола LLC2) и без установления соединения. Последний режим используется, например, для управления и мониторинга сети.

Протокол Q.931 переносит в своих пакетах адрес ISDN вызываемого абонента, на основании которого и происходит настройка коммутаторов на поддержку составного канала типа B.

Использование служб ISDN в корпоративных сетях

Несмотря на большие отличия от аналоговых телефонных сетей, сети ISDN сегодня используются в основном так же, как аналоговые телефонные сети, то есть как сети с коммутацией каналов, но только более скоростные: интерфейс BRI дает возможность установить дуплексный режим обмена со скоростью 128 Кбит/с (логическое объединение двух каналов типа B), а интерфейс PRI – 2,048 Мбит/с. Кроме того, качество цифровых каналов гораздо выше, чем аналоговых, а это значит, что процент искаженных кадров будет гораздо ниже и полезная скорость обмена данными существенно выше.

Обычно интерфейс BRI используется в коммуникационном оборудовании для подключения отдельных компьютеров или небольших локальных сетей, а интерфейс PRI – в маршрутизаторах, рассчитанных на сети средних размеров.

Что же касается объединения компьютерных сетей для поддержки службы с коммутацией пакетов, то здесь сети ISDN предоставляют не очень большие возможности.

На каналах типа B режим коммутации пакетов поддерживается следующим образом – либо с помощью постоянного соединения с коммутатором сети X.25, либо с помощью коммутируемого соединения с этим же коммутатором. То есть каналы типа B в сетях ISDN являются только транзитными для доступа к «настоящей» сети X.25. Собственно, это сводится к первому случаю использования сети ISDN – только как сети с коммутацией каналов.

Развитие технологии трансляции кадров на каналах типа B – технологии frame relay – привело к тому, что сети frame relay стали самостоятельным видом сетей со своей инфраструктурой каналов и коммутаторов. Поэтому эта технология рассматривается ниже в разделе, посвященном сетям с коммутацией пакетов.

Остается служба коммутации пакетов, доступная по каналу D. Так как после передачи адресной информации канал D остается свободным, по нему можно реализовать передачу компьютерных пакетов X.25, поскольку протокол LAP-D позволяет это делать. Чаще всего сеть ISDN используется не как замена сети X.25, а как разветвленная сеть доступа к менее географически распространенной и узкоспециализированной сети X.25 (рис. 6.20). Такая услуга обычно называется «доступ к сети X.25 через канал типа D». Скорость доступа к сети X.25 по каналу типа D обычно не превышает 9600 бит/с.

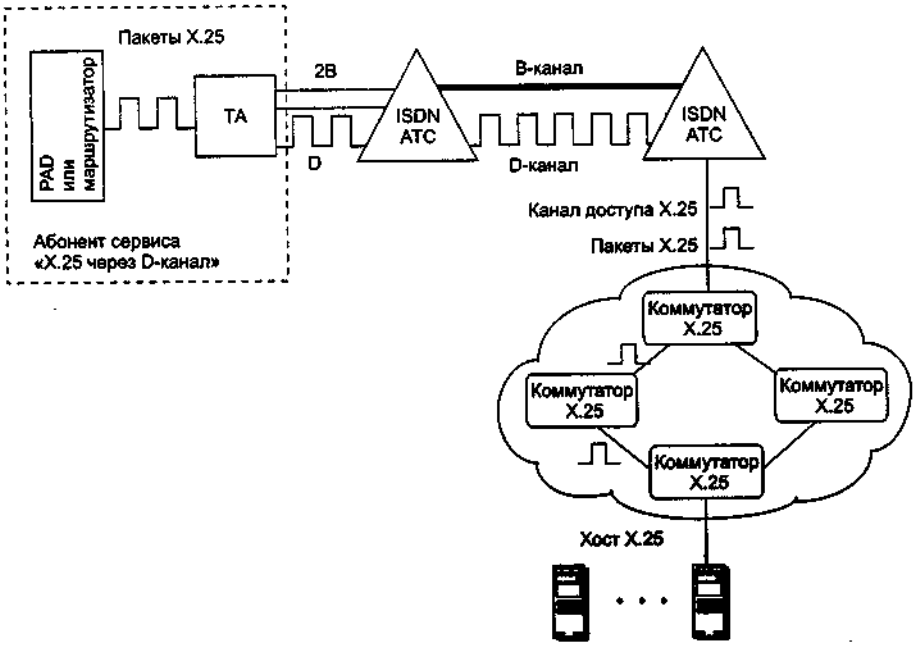


Рис. 6.20. Доступ к сети X.25 через канал типа D сети ISDN

Сети ISDN не рассматриваются разработчиками корпоративных сетей как хорошее средство для создания магистрали сети. Основная причина – отсутствие скоростной службы коммутации пакетов и невысокие скорости каналов, предоставляемых конечным пользователям. Для целей же подключения мобильных и домашних пользователей, небольших филиалов и образования резервных каналов связи сети ISDN сейчас используются очень широко, естественно там, где они существуют. Производители коммуникационного оборудования выпускают широкий спектр продуктов для подключения локальных сетей к ISDN – терминальных адаптеров, удаленных мостов и офисных маршрутизаторов невысокой стоимости.

Выводы

- Сети с коммутацией каналов используются в корпоративных сетях в основном для удаленного доступа многочисленных домашних пользователей и гораздо реже – для соединения локальных сетей.
- Отличительными особенностями всех сетей с коммутацией каналов являются: работа в режиме установления соединений, возможность блокировки вызова конечным абонентом или промежуточным коммутатором, необходимость использования на обоих концах сети устройств, поддерживающих одну и ту же скорость передачи данных, так как этот вид сетей не выполняет промежуточную буферизацию данных.
- Сети с коммутацией каналов делятся на аналоговые и цифровые. Аналоговые сети могут использовать аналоговую (FDM) и цифровую (TDM) коммутацию, но в них всегда абонент подключен по аналоговому 2-проводному окончанию. В цифровых сетях мультиплексирование и коммутация всегда выполняются по способу коммутации TDM, а абоненты всегда подключаются по цифровому абонентскому окончанию (DSL).
- Аналоговые сети обеспечивают вызов посредством импульсного или тонового набора номера с частотой 10 Гц, причем тоновый набор примерно в 5 раз быстрее импульсного.
- Аналоговые сети используют электромеханические коммутаторы, создающие большие помехи, и электронные программно-управляемые коммутаторы. При работе электронного коммутатора в режиме частотного уплотнения (FDM) создаются дополнительные помехи при демультиплексировании и мультиплексировании абонентских каналов.
- Модемы для работы по коммутируемым аналоговым телефонным каналам должны поддерживать функцию автовызова удаленного абонента. При асинхронном интерфейсе модем использует для этого команды Hayes-совместимых модемов, а при синхронном интерфейсе – стандарт V.25 или V.25 bis.
- Основные стандарты модемов для коммутируемых каналов тональной частоты – это

стандарты V.34+, V.90, V.42 и V.42 bis. Стандарт V.34+ является общим стандартом для работы по выделенным и коммутируемым каналам при 2-проводном окончании. Стандарт V.42 определяет протокол коррекции ошибок LAP-M из семейства HDLC, а стандарт VC.42 bis – метод компрессии данных при асинхронном интерфейсе. В синхронном интерфейсе для коррекции ошибок используется протокол HDLC, а для компрессии – фирменный протокол SDC компании Motorola.

- Стандарт V.90 полезен в том случае, когда между модемом пользователя и сервером удаленною доступа поставщика услуг все АТС обеспечивают цифровые методы коммутации, а сервер подключен по цифровому абонентскому окончанию. В этом случае скорость передачи данных от сервера к пользователю повышается до 56 Кбит/с за счет отсутствия аналогово-цифрового преобразования на этом направлении.
- Цифровые сети с коммутацией каналов представлены двумя технологиями: Switched 56 и ISDN.
- Switched 56 – это переходная технология, которая основана на предоставлении пользователю 4-проводного цифрового абонентского окончания T1/E1, но со скоростью 56 Кбит/с. Коммутаторы такой сети работают с использованием цифровой коммутации. Технология Switched 56 обеспечивает соединение компьютеров и локальных сетей со скоростью 56 Кбит/с.
- Цифровые сети с интегрированными услугами – ISDN – разработаны для объединения в одной сети различных транспортных и прикладных служб. ISDN предоставляет своим абонентам услуги выделенных каналов, коммутируемых каналов, а также коммутации пакетов и кадров (frame relay).
- Интерфейс UNI предоставляется пользователям ISDN в двух видах – BRI и PRI. Интерфейс BRI предназначен для массового пользователя и построен по схеме 2B+D. Интерфейс PRI имеет две разновидности – североамериканскую 23B+D и европейскую 30B+D.
- Каналы типа D образуют сеть с коммутацией пакетов, выполняющую двоякую роль в сети ISDN: во-первых, передачу запроса на установление коммутируемого канала типа B с другим абонентом сети, во-вторых, обмен пакетами X.25 с абонентами сети ISDN или внешней сети X.25, соединенной с сетью ISDN.
- Цифровое абонентское окончание DSL сети ISDN для интерфейса BRI представляет собой 2-проводной кабель с максимальной длиной 5,5 км.
- Построение глобальных связей на основе сетей ISDN в корпоративной сети ограничено в основном организацией удаленного доступа и объединением небольших локальных сетей на основании службы коммутации каналов. Служба коммутации пакетов по каналу типа D реализуется редко - это связано с его невысокой скоростью, которая обычно составляет не более 9600 бит/с. Поэтому сети ISDN используются так же, как и аналоговые телефонные сети, но только как более скоростные и надежные.

6.4. Компьютерные глобальные сети с коммутацией пакетов

В предыдущих разделах рассматривалось построение глобальных связей в корпоративной сети на основе выделенных или коммутируемых каналов. Собственно, основные новые проблемы были сосредоточены при этом на физическом и канальном уровнях, так как поверх протоколов этих уровней, специфических для глобального канала, работали те, же сетевые протоколы IP или IXP, которые использовались и для объединения локальных сетей.

Однако для глобальных сетей с коммутацией пакетов, таких как X.25, frame relay или ATM, характерна оригинальная техника маршрутизации пакетов (здесь термин «пакет» используется как родовой для обозначения пакетов X.25, кадров frame relay и ячеек ATM). Эта техника основана на понятии «виртуальный канал» и обеспечивает эффективную передачу долговременных устойчивых потоков данных.

6.4.1. Принцип коммутации пакетов с использованием техники виртуальных каналов

Техника виртуальных каналов, используемая во всех территориальных сетях с коммутацией пакетов,

кроме TCP/IP, состоит в следующем.

Прежде чем пакет будет передан через сеть, необходимо установить *виртуальное соединение* между абонентами сети – терминалами, маршрутизаторами или компьютерами. Существуют два типа виртуальных соединений – *коммутируемый виртуальный канал* (*Switched Virtual Circuit, SVC*) и *постоянный виртуальный канал* (*Permanent Virtual Circuit, PVC*). При создании коммутируемого виртуального канала коммутаторы сети настраиваются на передачу пакетов динамически, по запросу абонента, а создание постоянного виртуального канала происходит заранее, причем коммутаторы настраиваются вручную администратором сети, возможно, с привлечением централизованной системы управления сетью.

Смысл создания виртуального канала состоит в том, что маршрутизация пакетов между коммутаторами сети на основании таблиц маршрутизации происходит только один раз – при создании виртуального канала (имеется в виду создание коммутируемого виртуального канала, поскольку создание постоянного виртуального канала осуществляется вручную и не требует передачи пакетов по сети). После создания виртуального канала передача пакетов коммутаторами происходит на основании так называемых *номеров* или *идентификаторов виртуальных каналов* (*Virtual Channel Identifier, VCI*). Каждому виртуальному каналу присваивается значение VCI на этапе создания виртуального канала, причем это значение имеет не глобальный характер, как адрес абонента, а локальный – каждый коммутатор самостоятельно нумерует новый виртуальный канал. Кроме нумерации виртуального канала, каждый коммутатор при создании этого канала автоматически настраивает так называемые *таблицы коммутации портов* – эти таблицы описывают, на какой порт нужно передать пришедший пакет, если он имеет определенный номер VCI. Так что после прокладки виртуального канала через сеть коммутаторы больше не используют для пакетов этого соединения таблицу маршрутизации, а продвигают пакеты на основании номеров VCI небольшой разрядности. Сами таблицы коммутации портов также включают обычно меньше записей, чем таблицы маршрутизации, так как хранят данные только о действующих на данный момент соединениях, проходящих через данный порт.

Работа сети по маршрутизации пакетов ускоряется за счет двух факторов. Первый состоит в том, что решение о продвижении пакета принимается быстрее из-за меньшего размера таблицы коммутации. Вторым фактором является уменьшение доли служебной информации в пакетах. Адреса конечных узлов в глобальных сетях обычно имеют достаточно большую длину – 14-15 десятичных цифр, которые занимают до 8 байт (в технологии ATM – 20 байт) в служебном поле пакета. Номер же виртуального канала обычно занимает 10-12 бит, так что накладные расходы на адресную часть существенно сокращаются, а значит, полезная скорость передачи данных возрастает.

Режим PVC является особенностью технологии маршрутизации пакетов в глобальных сетях, в сетях TCP/IP такого режима работы нет. Работа в режиме PVC является наиболее эффективной по критерию производительности сети. Половину работы по маршрутизации пакетов администратор сети уже выполнил, поэтому коммутаторы быстро занимаются продвижением кадров на основе готовых таблиц коммутации портов. Постоянный виртуальный канал подобен выделенному каналу в том, что не требуется устанавливать соединение или разъединение. Обмен пакетами по PVC может происходить в любой момент времени. Отличие PVC в сетях X.25 от выделенной линии типа 64 Кбит/с состоит в том, что пользователь не имеет никаких гарантий относительно действительной пропускной способности PVC. Использование PVC обычно намного дешевле, чем аренда выделенной линии, так как пользователь делит пропускную способность сети с другими пользователями.

Режим продвижения пакетов на основе готовой таблицы коммутации портов обычно называют не маршрутизацией, а коммутацией и относят не к третьему, а ко второму (канальному) уровню стека протоколов.

Принцип маршрутизации пакетов на основе виртуальных каналов поясняется на рис. 6.21. При установлении соединения между конечными узлами используется специальный тип пакета – запрос на установление соединения (обычно называемый Call Request), который содержит многоразрядный (в примере семиразрядный) адрес узла назначения.

Пусть конечный узел с адресом 1581120 начинает устанавливать виртуальное соединение с узлом с адресом 1581130. Одновременно с адресом назначения в пакете Call Request указывается и номер виртуального соединения VCI. Этот номер имеет локальное значение для порта компьютера, через который устанавливается соединение. Через один порт можно установить достаточно большое количество виртуальных соединений, поэтому программное обеспечение протокола глобальной сети в компьютере просто выбирает свободный в данный момент для данного порта номер. Если через порт

уже проложено 3 виртуальных соединения, то для нового соединения будет выбран номер 4, по которому всегда можно будет отличить пакеты данного соединения от пакетов других соединений, приходящих на этот порт.

Далее пакет типа Call Request с адресом назначения 1581130, номером VCI 4 и адресом источника 1581120 отправляется в порт 1 коммутатора К1 сети. Адрес назначения используется для маршрутизации пакета на основании таблиц маршрутизации, аналогичных таблицам маршрутизации протокола IP, но с более простой структурой каждой записи. Запись состоит из адреса назначения и номера порта, на который нужно переслать пакет. Адрес следующего коммутатора не нужен, так как все связи между коммутаторами являются связями типа «точка-точка», множественных соединений между портами нет. Стандарты глобальных сетей обычно не описывают какой-либо протокол обмена маршрутной информацией, подобный RIP или OSPF, позволяющий коммутаторам сети автоматически строить таблицы маршрутизации. Поэтому в таких сетях администратор обычно вручную составляет подобную таблицу, указывая для обеспечения отказоустойчивости основной и резервный пути для каждого адреса назначения. Исключением являются сети ATM, для которых разработан протокол маршрутизации PNNI, основанный на алгоритме состояния связей.

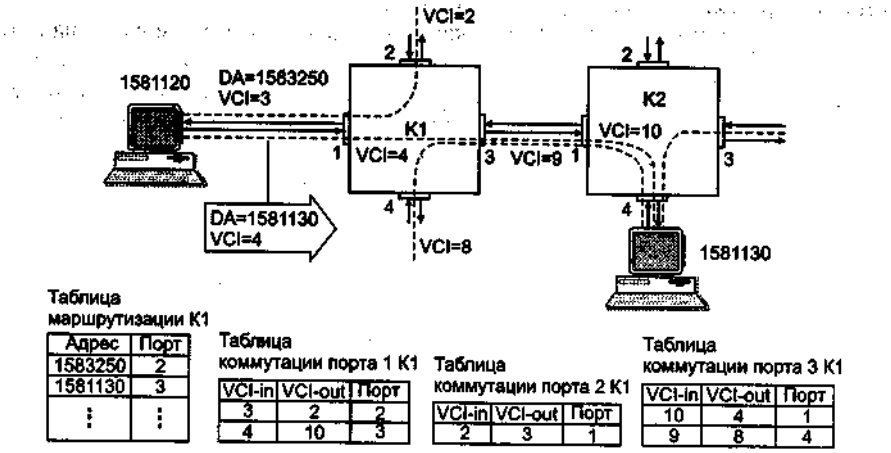


Рис. 6.21. Коммутация в сетях с виртуальными соединениями

В приведенном примере в соответствии с таблицей маршрутизации оказалось необходимым передать пакет Call Request с порта 1 на порт 3. Одновременно с передачей пакета маршрутизатор изменяет номер виртуального соединения пакета – он присваивает пакету первый свободный номер виртуального канала для выходного порта данного коммутатора. Каждый конечный узел и каждый коммутатор ведет свой список занятых и свободных номеров виртуальных соединений для всех своих портов. Изменение номера виртуального канала делается для того, чтобы при продвижении пакетов в обратном направлении (а виртуальные каналы обычно работают в дуплексном режиме), можно было отличить пакеты данного виртуального канала от пакетов других виртуальных каналов, уже проложенных через порт 3. В примере через порт 3 уже проходит несколько виртуальных каналов, причем самый старший занятый номер – это номер 9. Поэтому коммутатор меняет номер прокладываемого виртуального канала с 4 на 10.

Кроме таблицы маршрутизации для каждого порта составляется таблица коммутации. В таблице коммутации входного порта 1 маршрутизатор отмечает, что в дальнейшем пакеты, прибывшие на этот порт с номером VCI равным 4 должны передаваться на порт 3, причем номер виртуального канала должен быть изменен на 10. Одновременно делается и соответствующая запись в таблице коммутации порта 3 – пакеты, пришедшие по виртуальному каналу 10 в обратном направлении нужно передавать на порт с номером 1, меняя номер виртуального канала на 4. Таким образом, при получении пакетов в обратном направлении компьютер-отправитель получает пакеты с тем же номером VCI, с которым он отправлял их в сеть.

В результате действия такой схемы пакеты данных уже не несут длинные адреса конечных узлов, а имеют в служебном поле только номер виртуального канала, на основании которого и производится маршрутизация всех пакетов, кроме пакета запроса на установление соединения. В сети прокладывается виртуальный канал, который не изменяется в течение всего времени существования соединения. Его номер меняется от коммутатора к коммутатору, но для конечных узлов он остается постоянным.

За уменьшение служебного заголовка приходится платить невозможностью баланса трафика внутри

виртуального соединения. При отказе какого-либо канала соединение приходится также устанавливать заново.

По существу, техника виртуальных каналов позволяет реализовать два режима продвижения пакетов – стандартный режим маршрутизации пакета на основании адреса назначения и режим коммутации пакетов на основании номера виртуального канала. Эти режимы применяются поэтапно, причем первый этап состоит в маршрутизации всего одного пакета – запроса на установление соединения.

Техника виртуальных каналов имеет свои достоинства и недостатки по сравнению с техникой IP- или IPX-маршрутизации. Маршрутизация каждого пакета без предварительного установления соединения (ни IP, ни IPX не работают с установлением соединения) эффективна для кратковременных потоков данных. Кроме того, возможно распараллеливание трафика для повышения производительности сети при наличии параллельных путей в сети. Быстрее отрабатывается отказ маршрутизатора или канала связи, так как последующие пакеты просто пойдут по новому пути (здесь, правда, нужно учесть время установления новой конфигурации в таблицах маршрутизации). При использовании виртуальных каналов очень эффективно передаются через сеть долговременные потоки, но для кратковременных этот режим не очень подходит, так как на установление соединения обычно уходит много времени – даже коммутаторы технологии ATM, работающие на очень высоких скоростях, тратят на установление соединения по 5-10 мс каждый. Из-за этого обстоятельства компания Ipsilon разработала несколько лет назад технологию IP-switching, которая вводила в сети ATM, работающие по описанному принципу виртуальных каналов, режим передачи ячеек без предварительного установления соединения. Эта технология действительно ускоряла передачу через сеть кратковременных потоков IP-пакетов, поэтому она стала достаточно популярной, хотя и не приобрела статус стандарта. В главе 5 были рассмотрены методы ускорения маршрутизации трафика IP в локальных сетях. Особенностью всех подобных методов является ускорение передачи долговременных потоков пакетов. Технология IP-switching делает то же самое, но для кратковременных потоков, что хорошо отражает рассмотренные особенности каждого метода маршрутизации – маршрутизации на индивидуальной основе или на основе потоков пакетов, для которых прокладывается виртуальный канал.

6.4.2. Сети X.25

Назначение и структура сетей X.25

Сети X.25 являются на сегодняшний день самыми распространенными сетями с коммутацией пакетов, используемыми для построения корпоративных сетей. Основная причина такой ситуации состоит в том, что долгое время сети X.25 были единственными доступными сетями с коммутацией пакетов коммерческого типа, в которых давались гарантии коэффициента готовности сети. Сеть Internet также имеет долгую историю существования, но как коммерческая сеть она начала эксплуатироваться совсем недавно, поэтому для корпоративных пользователей выбора не было. Кроме того, сети X.25 хорошо работают на ненадёжных линиях благодаря протоколам с установлением соединения и коррекцией ошибок на двух уровнях – канальном и сетевом.

Стандарт X.25 «Интерфейс между оконечным оборудованием данных и аппаратурой передачи данных для терминалов, работающих в пакетном режиме в сетях передачи данных общего пользования» был разработан комитетом CCITT в 1974 году и пересматривался несколько раз. Стандарт наилучшим образом подходит для передачи трафика низкой интенсивности, характерного для терминалов, и в меньшей степени соответствует более высоким требованиям трафика локальных сетей. Как видно из названия, стандарт не описывает внутреннее устройство сети X.25, а только определяет пользовательский интерфейс с сетью. Взаимодействие двух сетей X.25 определяет стандарт X.75.

Технология сетей X.25 имеет несколько существенных признаков, отличающих ее от других технологий.

- Наличие в структуре сети специального устройства – *PAD* (*Packet Assembler Disassembler*), предназначенного для выполнения операции сборки нескольких низкоскоростных потоков байт от алфавитно-цифровых терминалов в пакеты, передаваемые по сети и направляемые компьютерам для обработки. Эти устройства имеют также русскоязычное название «*Сборщик-разборщик пакетов*», *СРП*.
- Наличие трехуровневого стека протоколов с использованием на канальном и сетевом уровнях протоколов с установлением соединения, управляющих потоками данных и исправляющих

ошибки.

- Ориентация на однородные стеки транспортных протоколов во всех узлах сети – сетевой уровень рассчитан на работу только с одним протоколом канального уровня и не может подобно протоколу IP объединять разнородные сети.

Сеть X.25 состоит из коммутаторов (Switches, S), называемых также *центрами коммутации пакетов* (*ЦКП*), расположенных в различных географических точках и соединенных высокоскоростными выделенными каналами (рис. 6.22). Выделенные каналы могут быть как цифровыми, так и аналоговыми.

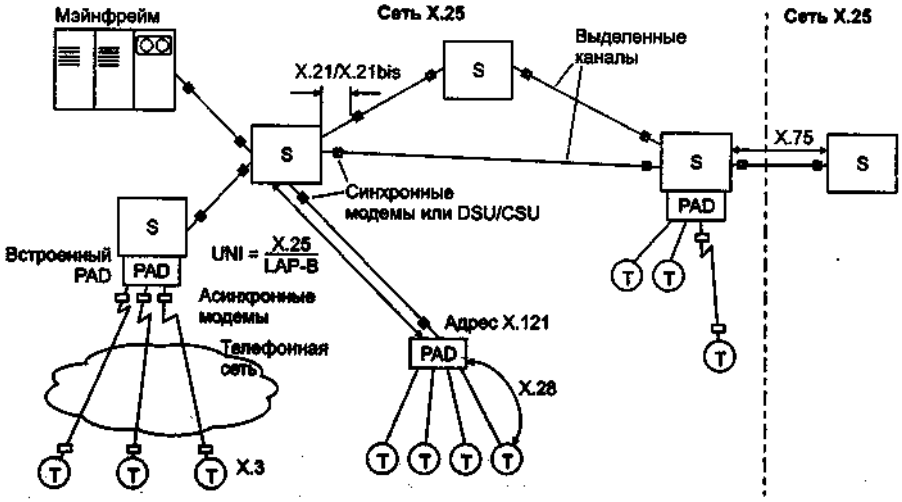


Рис. 6.22. Структура сети X.25

Асинхронные старт-стопные терминалы подключаются к сети через устройства PAD. Они могут быть встроенными или удаленными. Встроенный PAD обычно расположен в стойке коммутатора. Терминалы получают доступ ко встроенному устройству PAD по телефонной сети с помощью модемов с асинхронным интерфейсом. Встроенный PAD также подключается к телефонной сети с помощью нескольких модемов с асинхронным интерфейсом. Удаленный PAD представляет собой небольшое автономное устройство, подключенное к коммутатору через выделенный канал связи X.25. К удаленному устройству PAD терминалы подключаются по асинхронному интерфейсу, обычно для этой цели используется интерфейс RS-232C. Один PAD обычно обеспечивает доступ для 8, 16 или 24 асинхронных терминалов.

К основным функциям PAD, определенных стандартом X.3, относятся:

- сборка символов, полученных от асинхронных терминалов, в пакеты;
- разборка полей данных в пакетах и вывод данных на асинхронные терминалы;
- управление процедурами установления соединения и разъединения по сети X.25 с нужным компьютером;
- передача символов, включающих старт-стопные сигналы и биты проверки на четность, по требованию асинхронного терминала;
- продвижение пакетов при наличии соответствующих условий, таких как заполнение пакета, истечение времени ожидания и др.

Терминалы не имеют конечных адресов сети X.25. Адрес присваивается порту PAD, который подключен к коммутатору пакетов X.25 с помощью выделенного канала.

Несмотря на то что задача подключения «неинтеллектуальных» терминалов к удаленным компьютерам возникает сейчас достаточно редко, функции PAD все еще остаются востребованными. Устройства PAD часто используются для подключения к сетям X.25 кассовых терминалов и банкоматов, имеющих асинхронный интерфейс RS-232.

Стандарт X.28 определяет параметры терминала, а также протокол взаимодействия терминала с устройством PAD. При работе на терминале пользователь сначала проводит некоторый текстовый диалог с устройством PAD, используя стандартный набор символьных команд. PAD может работать с терминалом в двух режимах: управляющем и передачи данных. В управляющем режиме пользователь с помощью команд может указать адрес компьютера, с которым нужно установить соединение по сети X.25, а также установить некоторые параметры работы PAD, например выбрать специальный символ

для обозначения команды немедленной отправки пакета, установить режим эхо-ответов символов, набираемых на клавиатуре, от устройства PAD (при этом дисплей не будет отображать символы, набираемые на клавиатуре до тех пор, пока они не вернутся от PAD – это обычный локальный режим работы терминала с компьютером). При наборе комбинации клавиш Ctrl+P PAD переходит в режим передачи данных и воспринимает все последующие символы как данные, которые нужно передать в пакете X.25 узлу назначения. .

В сущности, протоколы X.3 и X.28 определяют протокол эмуляции терминала, подобный протоколу telnet стека TCP/IP. Пользователь с помощью устройства PAD устанавливает соединение с нужным компьютером, а затем может вести уже диалог с операционный системой этого компьютера (в режиме передачи данных устройством PAD), запуская нужные программы и просматривая результаты их работы на своем экране, как и при локальном подключении терминала к компьютеру.

Компьютеры и локальные сети обычно подключаются к сети X.25 непосредственно через адаптер X.25 или маршрутизатор, поддерживающий на своих интерфейсах протоколы X.25. Для управления устройствами PAD в сети существует протокол X.29, с помощью которого узел сети может управлять и конфигурировать PAD удаленно, по сети. При необходимости передачи данных компьютеры, подключенные к сети X.25 непосредственно, услугами PAD не пользуются, а самостоятельно устанавливают виртуальные каналы в сети и передают по ним данные в пакетах X.25.

Адресация в сетях X.25

Если сеть X.25 не связана с внешним миром, то она может использовать адрес любой длины (в пределах формата поля адреса) и давать адресам произвольные значения. Максимальная длина поля адреса в пакете X.25 составляет 16 байт.

Рекомендация X.121 CCITT определяет международную систему нумерации адресов для сетей передачи данных общего пользования. Если сеть X.25 хочет обмениваться данными с другими сетями X.25, то в ней нужно придерживаться адресации стандарта X.121.

Адреса X.121 (называемые также *International Data Numbers, IDN*) имеют разную длину, которая может доходить до 14 десятичных знаков. Первые четыре цифры IDN называют *кодом идентификации сети* (*Data Network Identification Code, DNIC*). DNIC поделен на две части; первая часть (3 цифры) определяет страну, в которой находится сеть, а вторая – номер сети X.25 в данной стране. Таким образом, внутри каждой страны можно организовать только 10 сетей X.25. Если же требуется перенумеровать больше, чем 10 сетей для одной страны, проблема решается тем, что одной стране дается несколько кодов. Например, Россия имела до 1995 года один код – 250, а в 1995 году ей был выделен еще один код – 251. Остальные цифры называются *номером национального терминала* (*National Terminal Numbe, NTN*). Эти цифры позволяют идентифицировать определенный DTE в сети X.25.

Международные сети X.25 могут также использовать международный стандарт нумерации абонентов ISO 7498, описанный выше.

По стандарту ISO 7498 для нумерации сетей X.25 к адресу в формате X.121 добавляется только один байт префикса, несущий код 36 (использование в адресе только кодов десятичных цифр) или 37 (использование произвольных двоичных комбинаций). Этот код позволяет универсальным коммутаторам, например коммутаторам сети ISDN, поддерживающим также и коммутацию пакетов X.25, автоматически распознавать тип адреса и правильно выполнять маршрутизацию запроса на установление соединения.

Стек протоколов сети X.25

Стандарты сетей X.25 описывают 3 уровня протоколов (рис. 6.23).

- На физическом уровне определены синхронные интерфейсы X.21 и X.21 bis к оборудованию передачи данных – либо DSU/CSU,если выделенный канал является цифровым, либо к синхронному модему, если канал выделенный.
- На канальном уровне используется подмножество протокола HDLC, обеспечивающее возможность автоматической передачи в случае возникновения ошибок в линии. Предусмотрен выбор из двух процедур доступа к каналу: LAP или LAP-B.
- На сетевом уровне определен протокол X.25/3 обмена пакетами между оконечным

оборудованием и сетью передачи данных.

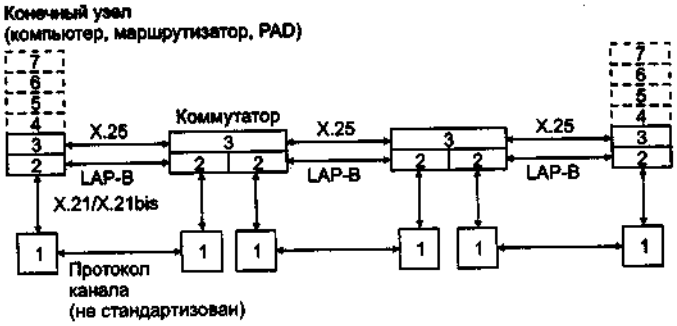


Рис. 6.23. Стек протоколов сети X.25

Транспортный уровень может быть реализован в конечных узлах, но он стандартом не определяется.

Протокол физического уровня канала связи не оговорен, и это дает возможность использовать каналы разных стандартов.

На канальном уровне обычно используется протокол LAP-B. Этот протокол обеспечивает сбалансированный режим работы, то есть оба узла, участвующих в соединении, равноправны. По протоколу LAP-B устанавливается соединение между пользовательским оборудованием DTE (компьютером, IP- или IPX-маршрутизатором) и коммутатором сети. Хотя стандарт это и не оговаривает, но по протоколу LAP-B возможно также установление соединения на канальном уровне внутри сети между непосредственно связанными коммутаторами. Протокол LAP-B почти во всех отношениях идентичен протоколу LLC2, описанному в главе 3, кроме адресации. Кадр LAP-B содержит одно однобайтовое адресное поле (а не два – DSAP и SSAP), в котором указывается не адрес службы верхнего уровня, а направление передачи кадра – 0x01 для направления команд от DTE к DCE (в сеть) или ответов от DCE к DTE (из сети) и 0x03 для направления ответов от DTE к DCE или команд от DCE к DTE. Поддерживается как нормальный режим (с максимальным окном в 8 кадров и однобайтовым полем управления), так и расширенный режим (с максимальным окном в 128 кадров и двухбайтовым полем управления).

Сетевой уровень X25/3 (в стандарте он назван не сетевым, а пакетным уровнем) реализуется с использованием 14 различных типов пакетов, по назначению аналогичных типам кадров протокола LAP-B. Так как надежную передачу данных обеспечивает протокол LAP-B, протокол X.25/3 выполняет функции маршрутизации пакетов, установления и разрыва виртуального канала между конечными абонентами сети и управления потоком пакетов.

После установления соединения на канальном уровне конечный узел должен установить виртуальное соединение с другим конечным узлом сети. Для этого он в кадрах LAP-B посылает пакет Call Request протокола X.25. Формат пакета Call Request показан на рис. 6.24.

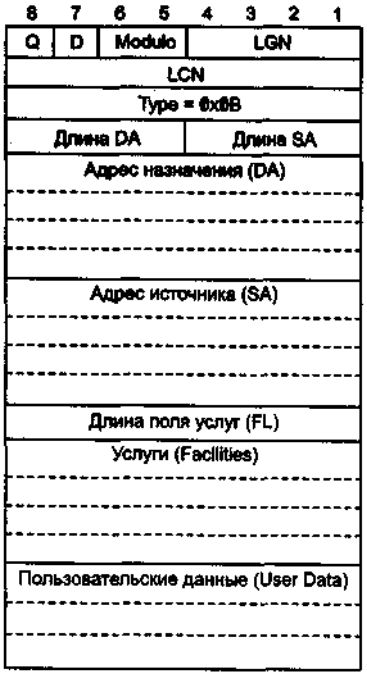


Рис. 6.24. Формат пакета Call Request

Поля, расположенные в первых трех байтах заголовка пакета, используются во всех типах кадров протокола X.25. Признаки Q и D и Modulo расположены в старшей части первого байта заголовка. Признак Q предназначен для распознавания на сетевом уровне типа информации в поле данных пакета. При получении пакета информация, расположенная в поле данных, а также значение бита Q передается верхним уровням пользовательского стека протоколов (непосредственно транспортному уровню этого стека). Значение Q=l означает управляющую пользовательскую информацию, а Q=0 – данные. Признак D означает подтверждение приема пакета узлом назначения. Обычный механизм подтверждения принятия пакетов с помощью квитанций имеет для протокола X.25 только локальный смысл – прием пакета подтверждает ближайший коммутатор сети, через который конечный узел запросил и установил виртуальное соединение. Если же узел-источник запросил подтверждение приема конечным узлом, то это подтверждение индицируется установкой бита D (delivery confirmation) в пакетах, идущих от узла назначения.

Признак Modulo говорит о том, по какому модулю – 8 или 128 – ведется нумерация пакетов. Значение 10 означает модуль 128, а 01 – модуль 8.

Поле *Номер логической группы* (*Lodical Group Number, LGN*) содержит значение *номера логической группы виртуального канала*. Каналы образуют логические группы по функциональному признаку, например:

- постоянный виртуальный канал;
- коммутируемый виртуальный канал только для входящих сообщений (симплексный);
- коммутируемый виртуальный канал только для исходящих сообщений (симплексный);
- коммутируемый дуплексный виртуальный канал.

Максимальное количество логических групп – 12, хотя в конкретной сети допустимо и меньшее количество.

Поле *Номер логического канала* (*Logjcal Channel Number, LCN*) содержит номер виртуального канала, назначаемый узлом-источником (для коммутируемых виртуальных каналов) или администратором сети (для постоянных виртуальных каналов). Максимальное количество виртуальных каналов, проходящих через один порт, равно 256.

Поле *Тип* (*Type*) указывает тип пакета. Например, для пакета Call Request отведено значение типа, равное OxOB. Младший бит этого поля определяет, является ли пакет управляющим (бит равен 1) или пакетом данных (бит равен 0). Значение OxOB содержит 1 в младшем бите, поэтому это управляющий пакет, а остальные биты в этом случае определяют подтип пакета. В пакете данных остальные биты поля *Type* используются для переноса номеров квитанций N(S) и N(R).

Следующие два поля определяют длину адресов назначения и источника (DA и SA) в пакете. Запрос на установление виртуального канала указывает оба адреса. Первый адрес нужен для маршрутизации

пакета Call Request, а второй – для принятия решения узлом назначения о возможности установления виртуального соединения с данным узлом-источником. Если узел назначения решает принять запрос, то он должен отправить пакет Call Accepted – «Запрос принят», в котором также указать оба адреса, поменяв их, естественно, местами. Адреса могут иметь произвольный формат или же соответствовать требованиям стандарта X.121 или ISO 7498.

Сами адреса назначения и источника занимают отведенное им количество байт в следующих двух полях.

Поля *Длина поля услуг* (*Facilities length*) и *Услуги* (*Facilities*) нужны для согласования дополнительных услуг, которые оказывает сеть абоненту. Например, услуга «Идентификатор пользователя сети» позволяет задать идентификатор пользователя (отличный от его сетевого адреса), на основании которого могут оплачиваться счета за пользование сетью. Пользователь с помощью услуги «Согласование параметров управления потоком» может попросить сеть использовать нестандартные значения параметров протокола – размера окна, максимального размера поля данных пакета и т. п. Протокол X.25 допускает следующие максимальные значения длины поля данных: 16,32,64,128, 256, 512 и 1024 байт. Предпочтительной является длина 128 байт.

Пакет Call Request принимается коммутатором сети и маршрутизируется на основании таблицы маршрутизации, прокладывая при этом виртуальный канал. Начальное значение номера виртуального канала задает пользователь в этом пакете в поле LCN (аналог поля VCI, упоминавшегося при объяснении принципа установления виртуальных каналов). Протокол маршрутизации для сетей X.25 не определен.

Для сокращения размера адресных таблиц в коммутаторах в сетях X.25 реализуется принцип агрегирования адресов. Все терминалы, имеющие общий префикс в адресе, подключаются при этом к общему входному коммутатору подсети, соответствующей значению префикса. Например, если путь ко всем терминалам, имеющим адреса с префиксом 250 720, пролегает через общий коммутатор К1, то в таблице маршрутизации коммутаторов, через которые проходит путь к коммутатору К1, помещается единственная запись – 250 720, которая соответствует как конечному узлу 250 720 11, так и конечному узлу 250 720 26. Маски в коммутаторах не используются, а младшие разряды адреса, которые не нужны при маршрутизации, просто опускаются.

После установления виртуального канала конечные узлы обмениваются пакетами другого формата – формата пакетов данных (пакет Data). Этот формат похож на описанный формат пакета Call Request – первые три байта в нем имеют те же поля, а адресные поля и поля услуг отсутствуют. Пакет данных не имеет поля, которое бы определяло тип переносимых в пакете данных, то есть поля, аналогичного полю Protocol в IP-пакете. Для устранения этого недостатка первый байт в поле данных всегда интерпретируется как признак типа данных.

Коммутаторы (ЦКП) сетей X.25 представляют собой гораздо более простые и дешевые устройства по сравнению с маршрутизаторами сетей TCP/IP. Это объясняется тем, что они не поддерживают процедур обмена маршрутной информацией и нахождения оптимальных маршрутов, а также не выполняют преобразований форматов кадров канальных протоколов. По принципу работы они ближе к коммутаторам локальных сетей, чем к маршрутизаторам. Однако работа, которую выполняют коммутаторы X.25 над пришедшими кадрами, включает больше этапов, чем при продвижении кадров коммутаторами локальных сетей. Коммутатор X.25 должен принять кадр LAP-B и ответить на него другим кадром LAP-B, в котором подтвердить получение кадра с конкретным номером. При утере или искажении кадра коммутатор должен организовать повторную передачу кадра. Если же с кадром LAP-B все в порядке, то коммутатор должен извлечь пакет X.25, на основании номера виртуального канала определить выходной порт, а затем сформировать новый кадр LAP-B для дальнейшего продвижения пакета. Коммутаторы локальных сетей такой работой не занимаются и просто передают кадр в том виде, в котором он пришел, на выходной порт.

В результате производительность коммутаторов X.25 оказывается обычно невысокой – несколько тысяч пакетов в секунду. Для низкоскоростных каналов доступа, которыми много лет пользовались абоненты этой сети (1200-9600 бит/с), такой производительности коммутаторов хватало для работы сети.

Гарантий пропускной способности сеть X.25 не дает. Максимум, что может сделать сеть, – это приоритезировать трафик отдельных виртуальных каналов. Приоритет канала указывается в запросе на установление соединения в поле услуг.

Протоколы сетей X.25 были специально разработаны для низкоскоростных линий с высоким уровнем

помех. Именно такие линии составляют пока большую часть телекоммуникационной структуры нашей страны, поэтому сети X.25 будут по-прежнему еще долго являться наиболее рациональным выбором для многих регионов.

6.4.3. Сети Frame Relay

Назначение и общая характеристика

Сети frame relay – сравнительно новые сети, которые гораздо лучше подходят для передачи пульсирующего трафика локальных сетей по сравнению с сетями X.25, правда, это преимущество проявляется только тогда, когда каналы связи приближаются по качеству к каналам локальных сетей, а для глобальных каналов такое качество обычно достижимо только при использовании волоконно-оптических кабелей.

Преимущество сетей frame relay заключается в их низкой протокольной избыточности и дейтаграммном режиме работы, что обеспечивает высокую пропускную способность и небольшие задержки кадров. Надежную передачу кадров технология frame relay не обеспечивает. Сети frame relay специально разрабатывались как общественные сети для соединения частных локальных сетей. Они обеспечивают скорость передачи данных до 2 Мбит/с.

Особенностью технологии frame relay является гарантированная поддержка основных показателей качества транспортного обслуживания локальных сетей – средней скорости передачи данных по виртуальному каналу при допустимых пульсациях трафика. Кроме технологии frame relay гарантии качества обслуживания на сегодня может предоставить только технология ATM, в то время как остальные технологии предоставляют требуемое качество обслуживания только в режиме «с максимальными усилиями» (best effort), то есть без гарантий.

Технология frame relay в сетях ISDN стандартизована как служба. В рекомендациях I.122, вышедших в свет в 1988 году, эта служба входила в число дополнительных служб пакетного режима, но затем уже при пересмотре рекомендаций в 1992-93 гг. она была названа службой frame relay и вошла в число служб режима передачи кадров наряду со службой frame switching. Служба frame switching работает в режиме гарантированной доставки кадров с регулированием потока. На практике поставщики телекоммуникационных услуг предлагают только службу frame relay.

Технология frame relay сразу привлекла большое внимание ведущих телекоммуникационных компаний и организаций по стандартизации. В ее становлении и стандартизации помимо CCITT (ITU-T) активное участие принимают Frame Relay Forum и комитет T1S1 института ANSI.

Некоммерческую организацию Frame Relay Forum образовали в 1990 году компании Cisco Systems, StrataCom (сегодня – подразделение Cisco Systems), Northern Telecom и Digital Equipment Corporation для развития и конкретизации стандартов CCITT и ANSI. Спецификации Frame Relay Forum носят название FRF и имеют порядковые номера. Спецификации FRF часто стандартизуют те аспекты технологии frame relay, которые еще не нашли свое отражение в стандартах ITU-T и ANSI. Например, спецификация FRF.11 определяет режим передачи голоса по сетям frame relay.

Консорциум Frame Relay Forum разработал спецификацию, отвечающую требованиям базового протокола frame relay, разработанного T1S1 и CCITT. Однако консорциум расширил базовый протокол, включив дополнительные возможности по управлению сетью со стороны пользователя, что очень важно при использовании сетей frame relay в сложных составных корпоративных сетях. Эти дополнения к frame relay называют обобщенно *Local Management Interface* (*LMI*) – *локальный интерфейс управления*.

Стандарты ITU-T обычно отличаются высоким уровнем сложности и наличием многих возможностей, которые достаточно трудно воплотить на практике. Спецификации Frame Relay Forum упрощают некоторые аспекты стандартов ITU-T или отбрасывают некоторые возможности. Так, технология frame switching не нашла своего отражения в спецификациях FRF, а процедуры создания коммутируемых виртуальных каналов появились в спецификациях FRF позже, чем в стандартах ITU-T, и оказались более простыми.

Стандарты frame relay, как ITU-T/ANSI, так и Frame Relay Forum, определяют два типа виртуальных каналов - постоянные (PVC) и коммутируемые (SVC). Это соответствует потребностям пользователей, так как для соединений, по которым трафик передается почти всегда, больше подходят постоянные каналы, а для соединений, которые нужны только на несколько часов в месяц, больше подходят

коммутируемые каналы.

Однако производители оборудования frame relay и поставщики услуг сетей frame relay начали с поддержки только постоянных виртуальных каналов. Это, естественно, является большим упрощением технологии. Тем не менее в последние годы оборудование, поддерживающее коммутируемые виртуальные каналы, появилось, и появились поставщики, предлагающие такую услугу.

Стек протоколов frame relay

Технология frame relay использует для передачи данных технику виртуальных соединений, аналогичную той, которая применялась в сетях X.25, однако стек протоколов frame relay передает кадры (при установленном виртуальном соединении) по протоколам только физического и канального уровней, в то время как в сетях X.25 и после установления соединения пользовательские данные передаются протоколом 3-го уровня.

Кроме того, протокол канального уровня LAP-F в сетях frame relay имеет два режима работы - основной (core) и управляющий (control). В основном режиме, который фактически практикуется в сегодняшних сетях frame relay, кадры передаются без преобразования и контроля, как и в коммутаторах локальных сетей. За счет этого сети frame relay обладают весьма высокой производительностью, так как кадры в коммутаторах не подвергаются преобразованию, а сеть не передает квитанции подтверждения между коммутаторами на каждый пользовательский кадр, как это происходит в сети X.25. Пульсации трафика передаются сетью frame relay достаточно быстро и без больших задержек.

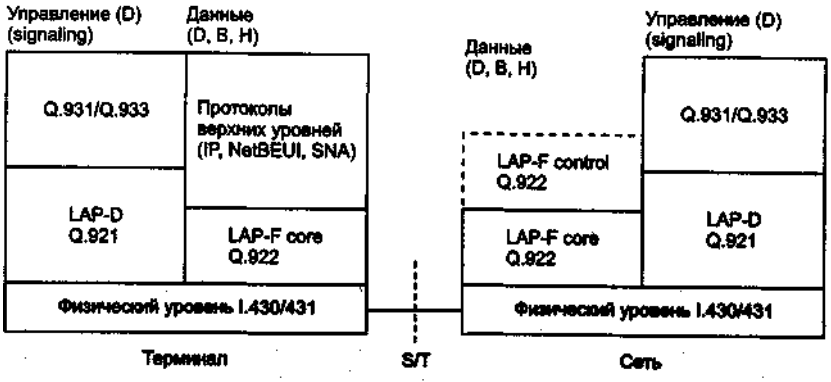


Рис. 6.25. Стек протоколов frame relay

При таком подходе уменьшаются накладные расходы при передаче пакетов локальных сетей, так как они вкладываются сразу в кадры канального уровня, а не в пакеты сетевого уровня, как это происходит в сетях X.25.

Структура стека (рис. 6.25) хорошо отражает происхождение технологии frame relay в недрах технологии ISDN, так как сети frame relay заимствуют многое из стека протоколов ISDN, особенно в процедурах установления коммутируемого виртуального канала.

Основу технологии составляет протокол LAP-F core, который является весьма упрощенной версией протокола LAP-D. Протокол LAP-F (стандарт Q.922 ITU-T) работает на любых каналах сети ISDN, а также на каналах типа T1/E1. Терминальное оборудование посылает в сеть кадры LAP-F в любой момент времени, считая что виртуальный канал в сети коммутаторов уже проложен. При использовании PVC оборудованию frame relay нужно поддерживать только протокол LAP-F core.

Протокол LAP-F contol является необязательной надстройкой над LAP-F core, которая выполняет функции контроля доставки кадров и управления потоком. С помощью протокола LAP-F control сетью реализуется служба frame switching.

Для установки коммутируемых виртуальных каналов стандарт ITU-T предлагает канал D пользовательского интерфейса. На нем по-прежнему работает знакомый протокол LAP-D, который используется для надежной передачи кадров в сетях ISDN. Поверх этого протокола работает протокол Q.931 или протокол Q.933 (который является упрощением и модификацией протокола Q.931 ISDN), устанавливающий виртуальное соединение на основе адресов конечных абонентов (в стандарте E.164 или ISO 7498), а также номера виртуального соединения, который в технологии frame relay носит название Data Link Connection Identifier – DLCI.

После того как коммутируемый виртуальный канал в сети frame relay установлен посредством протоколов LAP-D и Q.931/933, кадры могут транслироваться по протоколу LAP-F, который коммутирует их с помощью таблиц коммутации портов, в которых используются локальные значения DLCI. Протокол LAP-F core выполняет не все функции канального уровня по сравнению с протоколом LAP-D, поэтому ITU-T изображает его на пол-уровня ниже, чем протокол LAP-D, оставляя место для функций надежной передачи пакетов протоколу LAP-F control.

Из-за того, что технология frame relay заканчивается на канальном уровне, она хорошо согласуется с идеей инкапсуляции пакетов единого сетевого протокола, например IP, в кадры канального уровня любых сетей, составляющих интерсеть. Процедуры взаимодействия протоколов сетевого уровня с технологией frame relay стандартизованы, например, принята спецификация RFC 1490, определяющая методы инкапсуляции в трафик frame relay трафика сетевых протоколов и протоколов канального уровня локальных сетей и SNA.

Другой особенностью технологии frame relay является отказ от коррекции обнаруженных в кадрах искажений. Протокол frame relay подразумевает, что конечные узлы будут обнаруживать и корректировать ошибки за счет работы протоколов транспортного или более высоких уровней. Это требует некоторой степени интеллектуальности от конечного оборудования, что по большей части справедливо для современных локальных сетей. В этом отношении технология frame relay близка к технологиям локальных сетей, таким как Ethernet, Token Riag и FDDI, которые тоже только отбрасывают искаженные кадры, но сами не занимаются их повторной передачей.

Структуа кадра протокола LAP-F приведена на рис. 6.26.

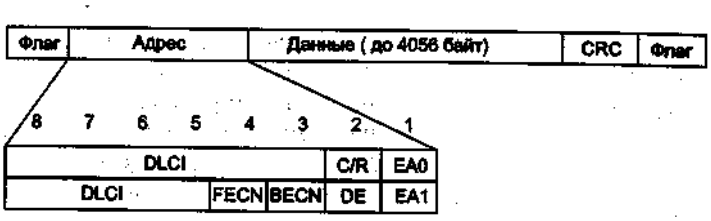


Рис. 6.26. Формат кадра LAP-F

За основу взят формат кадра HDLC, но поле адреса существенно изменило свой формат, а поле управления вообще отсутствует.

Поле номера виртуального соединения (Data Link Connection Identifier, DLCI) состоит из 10 битов, что позволяет использовать до 1024 виртуальных соединений. Поле DLCI может занимать и большее число разрядов – этим управляют признаки EAO и EA1 (Extended Address – расширенный адрес). Если бит в этом признаке установлен в ноль, то признак называется EAO и означает, что в следующем байте имеется продолжение поля адреса, а если бит признака равен 1, то поле называется EA1 и индицирует окончание поля адреса.

Десятиразрядный формат DLCI является основным, но при использовании трех байт для адресации поле DLCI имеет длину 16 бит, а при использовании четырех байт – 23 бита.

Стандарты frame relay (ANSI, ITU-T) распределяют адреса DLCI между пользователями и сетью следующим образом:

- 0 – используется для виртуального канала локального управления (LMI);
- 1-15 – зарезервированы для дальнейшего применения;
- 16-991 – используются абонентами для нумерации PVC и SVC;
- 992-1007 – используются сетевой транспортной службой для внутрисетевых соединений;
- 1008-1022 – зарезервированы для дальнейшего применения;
- 1023 – используются для управления канальным уровнем.

Таким образом, в любом интерфейсе frame relay для оконечных устройств пользователя отводится 976 адресов DLCI.

Поле данных может иметь размер до 4056 байт.

Поле C/R имеет обычный для протокола семейства HDLC смысл – это признак «команда-ответ».

Поля DE, FECN и BECN используются протоколом для управлением трафиком и поддержания заданного качества обслуживания виртуального канала.

ПРИМЕЧАНИЕ Способность технологии frame relay гарантировать некоторые параметры качества

обслуживания (QoS) является ключевой. Именно поэтому данная технология получила широкое распространение и считается одной из самых перспективных технологий глобальных сетей.

Поддержка качества обслуживания

Технология frame relay благодаря особому подходу гарантированно обеспечивает основные параметры качества транспортного обслуживания, необходимые при объединении локальных сетей.

Вместо приоритезации трафика используется процедура заказа качества обслуживания при установлении соединения, отсутствующая в сетях X.25 и пробивающая себе дорогу в сетях TCP/IP в форме экспериментального протокола RSVP, который пока не поддерживается поставщиками услуг Internet. В технологии frame relay заказ и поддержание качества обслуживания встроен в технологию.

Для каждого виртуального соединения определяется несколько параметров, влияющих на качество обслуживания.

- *CIR* (*Committed Information Rate*) – согласованная информационная скорость, с которой сеть будет передавать данные пользователя.
- *Bc* (*Committed Burst Size*) – согласованный объем пульсации, то есть максимальное количество байтов, которое сеть будет передавать от этого пользователя за интервал времени T.
- *Be* (*Excess Burst Size*) – дополнительный объем пульсации, то есть максимальное количество байтов, которое сеть будет пытаться передать сверх установленного значения Bc за интервал времени T.

Если эти величины определены, то время T определяется формулой: T = Bc/CIR. Можно задать значения CIR и T, тогда производной величиной станет величина всплеска трафика Bc.

Соотношение между параметрами CIR, Be, Be и T иллюстрирует рис. 6.27.

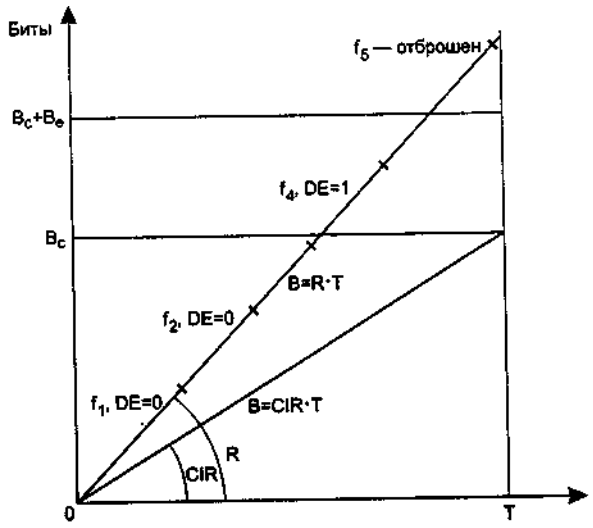


Рис. 6.27. Реакция сети на поведение пользователя: R – скорость канала доступа; $f_1 - f_4$ – кадры

Гарантий по задержкам передачи кадров технология frame relay не дает, оставляя эту услугу сетям ATM.

Основным параметром, по которому абонент и сеть заключают соглашение при установлении виртуального соединения, является согласованная скорость передачи данных. Для постоянных виртуальных каналов это соглашение является частью контракта на пользование услугами сети. При установлении коммутируемого виртуального канала соглашение о качестве обслуживания заключается автоматически с помощью протокола Q.931/933 – требуемые параметры CIR, Bc и Be передаются в пакете запроса на установление соединения.

Так как скорость передачи данных измеряется на каком-то интервале времени, то интервал T и является таким контрольным интервалом, на котором проверяются условия соглашения. В общем случае пользователь не должен за этот интервал передать в сеть данные со средней скоростью,

превосходящей CIR. Если же он нарушает соглашение, то сеть не только не гарантирует доставку кадра, но помечает этот кадр признаком DE (Discard Eligibility), равным 1, то есть как кадр, подлежащий удалению. Однако кадры, отмеченные таким признаком, удаляются из сети только в том случае, если коммутаторы сети испытывают перегрузки. Если же перегрузок нет, то кадры с признаком DE=1 доставляются адресату.

Такое щадящее поведение сети соответствует случаю, когда общее количество данных, переданных пользователем в сеть за период T, не превышает объема Bc+Be. Если же этот порог превышен, то кадр не помечается признаком DE, а немедленно удаляется из сети.

На рис. 6.27 изображен случай, когда за интервал времени T в сеть по виртуальному каналу поступило 5 кадров. Средняя скорость поступления информации в сеть составила на этом интервале R бит/с, и она оказалась выше CIR. Кадры fl, f2 и f3 доставили в сеть данные, суммарный объем которых не превысил порог Bc, поэтому эти кадры ушли дальше транзитом с признаком DE=0. Данные кадра f4, прибавленные к данным кадров fl, f2 и f3, уже превысили порог Bc, но еще не превысили порога Bc+Be, поэтому кадр f4 также ушел дальше, но уже с признаком DE=1. Данные кадра f5, прибавленные к данным предыдущих кадров, превысили порог Bc+Be, поэтому этот кадр был удален из сети.

Для контроля соглашения о параметрах качества обслуживания все коммутаторы сети frame relay выполняют так называемый алгоритм «дырявого ведра» (Leaky Bucket). Алгоритм использует счетчик C поступивших от пользователя байт. Каждые T секунд этот счетчик уменьшается на величину Bc (или же сбрасывается в 0, если значение счетчика меньше, чем Bc). Все кадры, данные которых не увеличили значение счетчика свыше порога Bc, пропускаются в сеть со значением признака DE=0. Кадры, данные которых привели к значению счетчика, большему Bc, но меньшему Bc+Be, также передаются в сеть, но с признаком DE=1. И наконец, кадры, которые привели к значению счетчика, большему Bc+Be, отбрасываются коммутатором.

Пользователь может договориться о включении не всех параметров качества обслуживания на данном виртуальном канале, а только некоторых.

Например, можно использовать только параметры CIR и Bc. Этот вариант дает более качественное обслуживание, так как кадры никогда не отбрасываются коммутатором сразу. Коммутатор только помечает кадры, которые превышают порог Bc за время T, признаком DE=1. Если сеть не сталкивается с перегрузками, то кадры такого канала всегда доходят до конечного узла, даже если пользователь постоянно нарушает договор с сетью.

Популярен еще один вид заказа на качество обслуживания, при котором оговаривается только порог Be, а скорость CIR полагается равной нулю. Все кадры такого канала сразу же отмечаются признаком DE=1, но отправляются в сеть, а при превышении порога Be они отбрасываются. Контрольный интервал времени T в этом случае вычисляется как Be/R, где R – скорость доступа канала.

На рис. 6.28 приведен пример сети frame relay с пятью удаленными региональными отделениями корпорации. Обычно доступ к сети осуществляется каналами с большей чем CIR пропускной способностью. Но при этом пользователь платит не за пропускную способность канала, а за заказанные величины CIR, Bc и Be. Так, при использовании в качестве канала доступа канала T1 и заказа службы со скоростью CIR, равной 128 Кбит/с, пользователь будет платить только за скорость 128 Кбит/с, а скорость канала T1 в 1,544 Мбит/с будет влиять на верхнюю границу возможной пульсации Bc+Be.

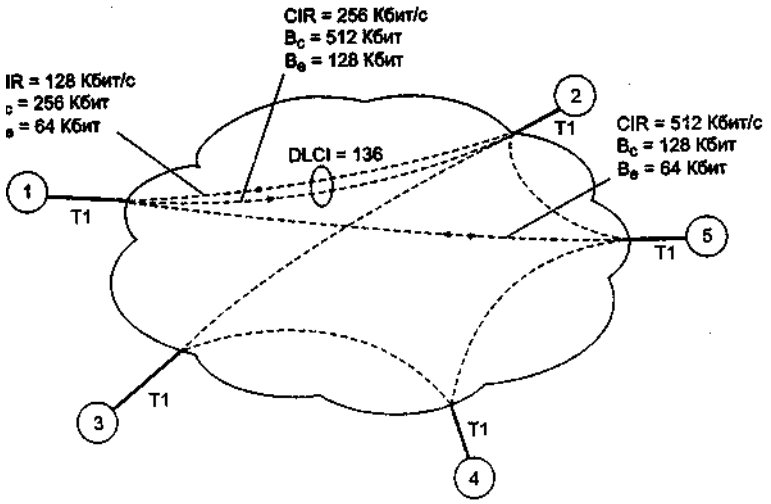


Рис. 6.28. Пример использования сети frame relay

Параметры качества обслуживания могут быть различными для разных направлений виртуального канала. Так, на рис. 6.28 абонент 1 соединен с абонентом 2 виртуальным каналом с DLCI=136. При направлении от абонента 1 к абоненту 2 канал имеет среднюю скорость 128 Кбит/с с пульсациями Bc=256 Кбит (интервал T составил 1 с) и Be=64 Кбит. А при передаче кадров в обратном направлении средняя скорость уже может достигать значения 256 Кбит/с с пульсациями Bc=512 Кбит и Be=128 Кбит.

Механизм заказа средней пропускной способности и максимальной пульсации является основным механизмом управления потоками кадров в сетях frame relay. Соглашения должны заключаться таким образом, чтобы сумма средних скоростей виртуальных каналов не превосходила возможностей портов коммутаторов. При заказе постоянных каналов за это отвечает администратор, а при установлении коммутируемых виртуальных каналов – программное обеспечение коммутаторов. При правильно взятых на себя обязательствах сеть борется с перегрузками путем удаления кадров с признаком DE-1 и кадров, превысивших порог Bc+Be.

Тем не менее в технологии frame relay определен еще и дополнительный (необязательный) механизм управления кадрами. Это механизм оповещения конечных пользователей о том, что в коммутаторах сети возникли перегрузки (переполнение необработанными кадрами). Бит FECN (Forward Explicit Congestion Bit) кадра извещает об этом принимающую сторону. На основании значения этого бита принимающая сторона должна с помощью протоколов более высоких уровней (TCP/IP, SPX и т.п.) известить передающую сторону о том, что та должна снизить интенсивность отправки пакетов и сеть.

Бит BECN (Backward Explicit Congestion Bit) извещает о переполнении в сети передающую сторону и является рекомендацией немедленно снизить темп передачи. Бит BECN обычно отрабатывается на уровне устройств доступа к сети frame relay – маршрутизаторов, мультиплексоров и устройств CSU/DSU. Протокол frame relay не требует от устройств, получивших кадры с установленными битами FECN и BECN, немедленного прекращения передачи кадров в данном направлении, как того требуют кадры RNR сетей X.25. Эти биты должны служить указанием для протоколов более высоких уровней (TCP, SPX, NCP и т. п.) о снижении темпа передачи пакетов. Так как регулирование потока инициируется в разных протоколах по-разному – как принимающей стороной, так и передающей, – то разработчики протоколов frame relay учли оба направления снабжения предупреждающей информацией о переполнении сети.

В общем случае биты FECN и BECN могут игнорироваться. Но обычно устройства доступа к сети frame relay (Frame Relay Access Device, FRAD) отрабатывают по крайней мере признак BECN.

При создании коммутируемого виртуального канала параметры качества обслуживания передаются в сеть с помощью протокола Q.931. Этот протокол устанавливает виртуальное соединение с помощью нескольких служебных пакетов.

Абонент сети frame relay, который хочет установить коммутируемое виртуальное соединение с другим абонентом, должен передать в сеть по каналу D сообщение SETUP, которое имеет несколько параметров, в том числе:

- DLCI;
- адрес назначения (в формате E. 164, X. 121 или ISO 7498);
- максимальный размер кадра в данном виртуальном соединении;

- запрашиваемое значение CIR для двух направлений;
- запрашиваемое значение Bc для двух направлений;
- запрашиваемое значение Be для двух направлений.

Коммутатор, с которым соединен пользователь, сразу же передает пользователю пакет CALL PROCEEDING – вызов обрабатывается. Затем он анализирует параметры, указанные в пакете, и если коммутатор может их удовлетворить (располагая, естественно, информацией о том, какие виртуальные каналы на каждом порту он уже поддерживает), то пересылает сообщение SETUP следующему коммутатору. Следующий коммутатор выбирается по таблице маршрутизации. Протокол автоматического составления таблиц маршрутизации для технологии frame relay не определен, поэтому может использоваться фирменный протокол производителя оборудования или же ручное составление таблицы. Если все коммутаторы на пути к конечному узлу согласны принять запрос, то пакет SETUP передается в конечном счете вызываемому абоненту. Вызываемый абонент немедленно передает в сеть пакет CALL PROCEEDING и начинает обрабатывать запрос. Если запрос принимается, то вызываемый абонент передает в сеть новый пакет – CONNECT, который проходит в обратном порядке по виртуальному пути. Все коммутаторы должны отметить, что данный виртуальный канал принят вызываемым абонентом. При поступлении сообщения CONNECT вызывающему абоненту он должен передать в сеть пакет CONNECT ACKNOWLEDGE.

Сеть также должна передать вызываемому абоненту пакет CONNECT ACKNOWLEDGE, и на этом соединение считается установленным. По виртуальному каналу могут передаваться данные.

Использование сетей frame relay

Услуги frame relay обычно предоставляются теми же операторами, которые эксплуатируют сети X.25. Большая часть производителей выпускает сейчас коммутаторы, которые могут работать как по протоколам X.25, так и по протоколам frame relay.

Технология frame relay начинает занимать в территориальных сетях с коммутацией пакетов ту же нишу, которую заняла в локальных сетях технология Ethernet. Их роднит то, что они предоставляют только быстрые базовые транспортные услуги, доставляя кадры в узел назначения без гарантий, дейтаграммным способом. Однако если кадры теряются, то сеть frame realay, как и сеть Ethernet, не предпринимает никаких усилий для их восстановления. Отсюда следует простой вывод – полезная пропускная способность прикладных протоколов при работе через сети frame relay будет зависеть от качества каналов и методов восстановления пакетов на уровнях стека, расположенного над протоколом frame relay. Если каналы качественные, то кадры будут теряться и искажаться редко, так что скорость восстановления пакетов протоколом TCP или NCP будет вполне приемлема. Если же кадры искажаются и теряются часто, то полезная пропускная способность в сети frame relay может упасть в десятки раз, как это происходит в сетях Ethernet при плохом состоянии кабельной системы.

Поэтому сети frame relay следует применять только при наличии на магистральных каналах волоконно-оптических кабелей высокого качества. Каналы доступа могут быть и на витой паре, как это разрешает интерфейс G.703 или абонентское окончание ISDN. Используемая на каналах доступа аппаратура передачи данных должна обеспечить приемлемый уровень искажения данных – не ниже 10^{-6}.

На величины задержек сеть frame relay гарантий не дает, и это основная причина, которая сдерживает применение этих сетей для передачи голоса. Передача видеоизображения тормозится и другим отличием сетей frame realy от ATM – низкой скоростью доступа в 2 Мбит/с, что для передачи видео часто недостаточно.

Тем не менее многие производители оборудования для сетей frame relay поддерживают передачу голоса. Поддержка устройствами доступа заключается в присвоении кадрам, переносящим замеры голоса, приоритетов. Магистральные коммутаторы frame relay должны обслуживать такие кадры в первую очередь. Кроме того, желательно, чтобы сеть frame relay, передающая кадры с замерами голоса, была недогруженной. При этом в коммутаторах не возникают очереди кадров, и средние задержки в очередях близки к нулевым.

Необходимо также соблюдение еще одного условия для качественной передачи голоса – передавать замеры голоса необходимо в кадрах небольших размеров, иначе на качество будут влиять задержки упаковки замеров в кадр, так называемые задержки пакетизации, которые более подробно рассматриваются в разделе, посвященном технологии ATM.

Для стандартизации механизмов качественной передачи голоса через сеть frame relay выпущена спецификация FRF.11. Однако в ней решены еще не все проблемы передачи голоса, поэтому работа в этом направлении продолжается.

Ввиду преобладания в коммерческих сетях frame relay услуг постоянных коммутируемых каналов и гарантированной пропускной способности, эти сети предоставляют услуги, очень похожие на услуги дробных выделенных линий T1/E1, но только за существенно меньшую плату.

При использовании PVC сеть frame relay хорошо подходит для объединения локальных сетей с помощью мостов, так как в этом случае от моста не нужна поддержка механизма установления виртуального канала, что требует некоторого программного «интеллекта». Мост может отправлять кадры протокола Ethernet или FDDI непосредственно в кадрах LAP-F или же может использовать поверх протокола LAP-F протокол PPP. Стандарт Internet RFC 1490 определяет формат заголовка SNAP для случая передачи через сеть frame relay непосредственно кадров канального уровня.

Чаще доступ к сетям frame relay реализуют не удаленные мосты, а маршрутизаторы, которые в случае поддержки на последовательных портах протокола frame relay как основного называют устройствами доступа FRAD (хотя и мост, и любое устройство, которое поддерживает протоколы UNI frame relay, относятся к классу FRAD).

Так как сети frame relay передают кадры с небольшими задержками, с их помощью часто передают трафик сетей SNA, особенно в том случае, когда они используют такие чувствительные к задержкам протоколы, как SDLC (фирменный протокол канального уровня компании IBM).

Виртуальные каналы в качестве основы построения корпоративной сети имеют один недостаток – при большом количестве точек доступа и смешанном характере связей необходимо большое количество виртуальных каналов, каждый из которых оплачивается отдельно. В сетях с маршрутизацией отдельных пакетов, таких как TCP/IP, абонент платит только за количество точек доступа, а не за количество связей между ними.

6.4.4. Технология ATM

Гетерогенность – неотъемлемое качество любой крупной вычислительной сети, и на согласование разнородных компонентов системные интеграторы и администраторы тратят большую часть своего времени. Поэтому любое средство, сулящее перспективу уменьшения неоднородности сети, привлекает пристальный интерес сетевых специалистов. Технология *асинхронного режима передачи* (*Asynchronous Transfer Mode, ATM*) разработана как единый универсальный транспорт для нового поколения сетей с интеграцией услуг, которые называются широкополосными сетями ISDN (Broadband-ISDN, B-ISDN).

По планам разработчиков единообразие, обеспечиваемое ATM, будет состоять в том, что одна транспортная технология сможет обеспечить несколько перечисленных ниже возможностей.

- Передачу в рамках одной транспортной системы компьютерного и мультимедийного (голос, видео) трафика, чувствительного к задержкам, причем для каждого вида трафика качество обслуживания будет соответствовать его потребностям.
- Иерархию скоростей передачи данных, от десятков мегабит до нескольких гигабит в секунду с гарантированной пропускной способностью для ответственных приложений.
- Общие транспортные протоколы для локальных и глобальных сетей.
- Сохранение имеющейся инфрасфуктуры физических каналов или физических протоколов: T1/E1, T3/E3, SDH STM-n, FBDI.
- Взаимодействие с унаследованными протоколами локальных и глобальных сетей: IP, SNA, Ethernet, ISDN.

Главная идея технологии асинхронного режима передачи была высказана достаточно давно – этот термин ввела лаборатория Bell Labs еще в 1968 году. Основной разрабатываемой технологией тогда была технология TDM с синхронными методами коммутации, основанными на порядковом номере байта в объединенном кадре. Главный недостаток технологии TDM, которую также называют технологией синхронной передачи STM (Synchronous Transfer Mode), заключается в невозможности перераспределять пропускную способность объединенного канала между подканалами. В те периоды времени, когда по подканалу не передаются пользовательские данные, объединенный канал все равно передает байты этого подканала, заполненные нулями.

Попытки загрузить периоды простоя подканалов приводят к необходимости введения заголовка для данных каждого подканала. В промежуточной технологии STDM (Statistical TDM), которая позволяет

заполнять периоды простоя передачей пульсаций трафика других подканалов, действительно вводятся заголовки, содержащие номер подканала. Данные при этом оформляются в пакеты, похожие по структуре на пакеты компьютерных сетей. Наличие адреса у каждого пакета позволяет передавать его асинхронно, так как местоположение его относительно данных других подканалов уже не является его адресом. Асинхронные пакеты одного подканала вставляются в свободные тайм-слоты другого подканала, но не смешиваются с данными этого подканала, так как имеют собственный адрес.

Технология ATM совмещает в себе подходы двух технологий – коммутации пакетов и коммутации каналов. От первой она взяла на вооружение передачу данных в виде адресуемых пакетов, а от второй – использование пакетов небольшого фиксированного размера, в результате чего задержки в сети становятся более предсказуемыми. С помощью техники виртуальных каналов, предварительного заказа параметров качества обслуживания канала и приоритетного обслуживания виртуальных каналов с разным качеством обслуживания удается добиться передачи в одной сети разных типов трафика без дискриминации. Хотя сети ISDN также разрабатывались для передачи различных видов трафика в рамках одной сети, голосовой трафик явно был для разработчиков более приоритетным. Технология ATM с самого начала разрабатывалась как технология, способная обслуживать все виды трафика в соответствии с их требованиями.

Службы верхних уровней сети B-ISDN должны быть примерно такими же, что и у сети ISDN – это передача факсов, распространение телевизионного изображения, голосовая почта, электронная почта, различные интерактивные службы, например проведение видеоконференций. Высокие скорости технологии ATM создают гораздо больше возможностей для служб верхнего уровня, которые не могли быть реализованы сетями ISDN – например, для передачи цветного телевизионного изображения необходима полоса пропускания в районе 30 Мбит/с. Технология ISDN такую скорость поддержать не может, а для ATM она не составляет больших проблем.

Разработку стандартов ATM осуществляет группа организаций под названием ATM Forum под эгидой специального комитета IEEE, а также комитеты ITU-T и ANSI. ATM – это очень сложная технология, требующая стандартизации в самых различных аспектах, поэтому, хотя основное ядро стандартов было принято в 1993 году, работа по стандартизации активно продолжается. Оптимизм внушает тот факт, что в ATM Forum принимают участие практически все заинтересованные стороны – производители телекоммуникационного оборудования, производители оборудования локальных сетей, операторы телекоммуникационных сетей и сетевые интеграторы. До широкого распространения технологии ATM по оценкам специалистов должно пройти еще 5-10 лет. Такой прогноз связан не только с отсутствием полного набора принятых стандартов, но и с невозможностью быстрой замены уже установленного дорогого оборудования, которое хотя и не так хорошо, как хотелось бы, но все же справляется со своими обязанностями. Кроме того, многое еще нужно сделать в области стандартизации взаимодействия ATM с существующими сетями, как компьютерными, так и телефонными.

Основные принципы технологии ATM

Сеть ATM имеет классическую структуру крупной территориальной сети – конечные станции соединяются индивидуальными каналами с коммутаторами нижнего уровня, которые в свою очередь соединяются с коммутаторами более высоких уровней. Коммутаторы ATM пользуются 20-байтными адресами конечных узлов для маршрутизации трафика на основе техники виртуальных каналов. Для частных сетей ATM определен протокол маршрутизации PNNI (Private NNI), с помощью которого коммутаторы могут строить таблицы маршрутизации автоматически. В публичных сетях ATM таблицы маршрутизации могут строиться администраторами вручную, как и в сетях X.25, или могут поддерживаться протоколом PNNI.

Коммутация пакетов происходит на основе идентификатора виртуального канала (Virtual Channel Identifier, VCI), который назначается соединению при его установлении и уничтожается при разрыве соединения. Адрес конечного узла ATM, на основе которого прокладывается виртуальный канал, имеет иерархическую структуру, подобную номеру в телефонной сети, и использует префиксы, соответствующие кодам стран, городов, сетям поставщиков услуг и т. п., что упрощает маршрутизацию запросов установления соединения, как и при использовании агрегированных IP-адресов в соответствии с техникой CIDR.

Виртуальные соединения могут быть постоянными (Permanent Virtual Circuit, PVC) и

коммутируемыми (Switched Virtual Circuit, SVC). Для ускорения коммутации в больших сетях используется понятие виртуального пути – Virtual Path, который объединяет виртуальные каналы, имеющие в сети ATM общий маршрут между исходным и конечным узлами или общую часть маршрута между некоторыми двумя коммутаторами сети. Идентификатор виртуального пути (Virtual Path Identifier, VPI) является старшей частью локального адреса и представляет собой общий префикс для некоторого количества различных виртуальных каналов. Таким образом, идея агрегирования адресов в технологии ATM применена на двух уровнях – на уровне адресов конечных узлов (работает на стадии установления виртуального канала) и на уровне номеров виртуальных каналов (работает при передаче данных по имеющемуся виртуальному каналу).

Соединения конечной станции ATM с коммутатором нижнего уровня определяются стандартом UNI (User Network Interface). Спецификация UNI определяет структуру пакета, адресацию станций, обмен управляющей информацией, уровни протокола ATM, способы установления виртуального канала и способы управления трафиком. В настоящее время принята версия UNI 4.0, но наиболее распространенной версией, поддерживаемой производителями оборудования, является версия UNI 3.1.

Стандарт ATM не вводит свои спецификации на реализацию физического уровня. Здесь он основывается на технологии SDH/SONET, принимая ее иерархию скоростей. В соответствии с этим начальная скорость доступа пользователя сети – это скорость OC-3 155 ,Мбит/с. Организация ATM Forum определила для ATM не все иерархии скоростей SDH, а только скорости OC-3 и OC-12 (622 Мбит/с). На скорости 155 Мбит/с можно использовать не только волоконно-оптический кабель, но и неэкранированную витую пару категории 5. На скорости 622 Мбит/с допустим только волоконно-оптический кабель, причем как SMF, так и MMF.

Имеются и другие физические интерфейсы к сетям ATM, отличные от SDH/ SONET. К ним относятся интерфейсы T1/E1 и T3/E3, распространенные в глобальных сетях, и интерфейсы локальных сетей – интерфейс с кодировкой 4B/5B со скоростью 100 Мбит/с (FDDI) и интерфейс со скоростью 25 Мбит/с, предложенный компанией IBM и утвержденный ATM Forum. Кроме того, для скорости 155,52 Мбит/с определен так называемый «cell-based» физический уровень, то есть уровень, основанный на ячейках, а не на кадрах SDH/SONET. Этот вариант физического уровня не использует кадры SDH/SONET, а отправляет по каналу связи непосредственно ячейки формата ATM, что сокращает накладные расходы на служебные данные, но несколько усложняет задачу синхронизации приемника с передатчиком на уровне ячеек.

Все перечисленные выше характеристики технологии ATM не свидетельствуют о том, что это некая «особенная» технология, а скорее представляют ее как типичную технологию глобальных сетей, основанную на технике виртуальных каналов. Особенности же технологии ATM лежат в области качественного обслуживания разнородного трафика и объясняются стремлением решить задачу совмещения в одних и тех же каналах связи и в одном и том же коммуникационном оборудовании компьютерного и мультимедийного трафика таким образом, чтобы каждый тип трафика получил требуемый уровень обслуживания и не рассматривался как «второстепенный».

Трафик вычислительных сетей имеет ярко выраженный асинхронный и пульсирующий характер. Компьютер посылает пакеты в сеть в случайные моменты времени, по мере возникновения в этом необходимости. При этом интенсивность посылки пакетов в сеть и их размер могут изменяться в широких пределах – например, коэффициент пульсаций трафика (отношения максимальной мгновенной интенсивности трафика к его средней интенсивности) протоколов без установления соединений может доходить до 200, а протоколов с установлением соединений – до 20. Чувствительность компьютерного трафика к потерям данных высокая, так как без утраченных данных обойтись нельзя и их необходимо восстановить за счет повторной передачи.

Мультимедийный трафик, передающий, например, голос или изображение, характеризуется низким коэффициентом пульсаций, высокой чувствительностью к задержкам передачи данных (отражающихся на качестве воспроизводимого непрерывного сигнала) и низкой чувствительностью к потерям данных (из-за инерционности физических процессов потерю отдельных замеров голоса или кадров изображения можно компенсировать сглаживанием на основе предыдущих и последующих значений).

Сложность совмещения компьютерного и мультимедийного трафика с диаметрально противоположными характеристиками хорошо видна на рис. 6.29.

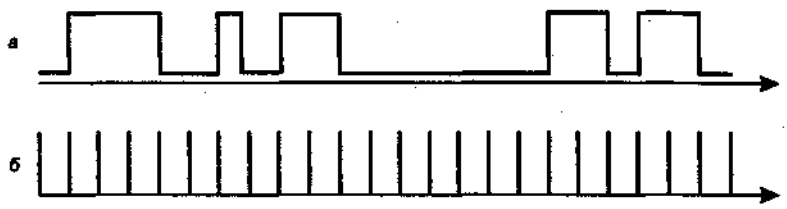


Рис. 6.29. Два типа трафика: *а* – компьютерный; *б* – мультимедийный

На возможности совмещения этих двух видов трафика большое влияние оказывает размер компьютерных пакетов. Если размер пакета может меняться в широком диапазоне (например, от 29 до 4500 байт, как в технологии FDDI), то даже при придании голосовым пакетам высшего приоритета обслуживания в коммутаторах время ожидания компьютерного пакета может оказаться недопустимо высоким. Например, пакет в 4500 байт будет передаваться в выходной порт на скорости 2 Мбит/с (максимальная скорость работы порта коммутатора frame relay) 18 мс. При совмещении трафика за это время необходимо через этот же порт передать 144 замера голоса. Прерывать передачу пакета в сетях нежелательно, так как при распределенном характере сети накладные расходы на оповещение соседнего коммутатора о прерывании пакета, а потом – о возобновлении передачи пакета с прерванного места оказываются слишком большими.

Подход, реализованный в технологии ATM, состоит в передаче любого вида трафика – компьютерного, телефонного или видео – пакетами фиксированной и очень маленькой длины в 53 байта. Пакеты ATM называют ячейками – cell. Поле данных ячейки занимает 48 байт, а заголовок – 5 байт.

Чтобы пакеты содержали адрес узла назначения и в то же время процент служебной информации не превышал размер поля данных пакета, в технологии ATM применен стандартный для глобальных вычислительных сетей прием – передача ячеек в соответствии с техникой виртуальных каналов с длиной номера виртуального канала в 24 бит, что вполне достаточно для обслуживания большого количества виртуальных соединений каждым портом коммутатора глобальной (может быть всемирной) сети ATM.

Размер ячейки ATM является результатом компромисса между телефонистами и компьютерщиками – первые настаивали на размере поля данных в 32 байта, а вторые – в 64 байта.

Чем меньше пакет, тем легче имитировать услуги каналов с постоянной битовой скоростью, которая характерна для телефонных сетей. Ясно, что при отказе от жестко синхронизированных временных слотов для каждого канала идеальной синхронности добиться будет невозможно, однако чем меньше размер пакета, тем легче этого достичь.

Для пакета, состоящего из 53 байт, при скорости в 155 Мбит/с время передачи кадра на выходной порт составляет менее 3 мкс. Так что эта задержка не очень существенна для трафика, пакеты которого должны передаваться каждые 125 мкс.

Однако на выбор размера ячейки большее влияние оказала не величина ожидания передачи ячейки, а задержка пакетизации. *Задержка пакетизации* – это время, в течение которого первый замер голоса ждет момента окончательного формирования пакета и отправки его по сети. При размере поля данных в 48 байт одна ячейка ATM обычно переносит 48 замеров голоса, которые делаются с интервалом в 125 мкс. Поэтому первый замер должен ждать примерно 6 мс, прежде чем ячейка будет отправлена по сети. Именно по этой причине телефонисты боролись за уменьшения размера ячейки, так как 6 мс – это задержка, близкая к пределу, за которым начинаются нарушения качества передачи голоса. При выборе размера ячейки в 32 байта задержка пакетизации составила бы 4 мс, что гарантировало бы более качественную передачу голоса. А стремление компьютерных специалистов увеличить поле данных до 64 байт вполне понятно – при этом повышается полезная скорость передачи данных. Избыточность служебных данных при использовании 48-байтного поля данных составляет 10 %, а при использовании 32-байтного поля данных она сразу повышается до 16 %.

Выбор для передачи данных любого типа небольшой ячейки фиксированного размера еще не решает задачу совмещения разнородного трафика в одной сети, а только создает предпосылки для ее решения. Для полного решения этой задачи технология ATM привлекает и развивает идеи *заказа пропускной способности* и *качества обслуживания*, реализованные в технологии frame relay. Но если сеть frame relay изначально была предназначена для передачи только пульсирующего компьютерного трафика (в связи с этим для сетей frame relay так трудно дается стандартизация передачи голоса), то разработчики

технологии ATM проанализировали всевозможные образцы трафика, создаваемые различными приложениями, и выделили 4 основных класса трафика, для которых разработали различные механизмы резервирования и поддержания требуемого качества обслуживания.

Класс трафика (называемый также классом услуг – service class) качественно характеризует требуемые услуги по передаче данных через сеть ATM. Если приложение указывает сети, что требуется, например, передача голосового трафика, то из этого становится ясно, что особенно важными для пользователя будут такие показатели качества обслуживания, как задержки и вариации задержек ячеек, существенно влияющие на качество переданной информации – голоса или изображения, а потеря отдельной ячейки с несколькими замерами не так уж важна, так как, например, воспроизводящее голос устройство может аппроксимировать недостающие замеры и качество пострадает не слишком. Требования к синхронности передаваемых данных очень важны для многих приложений – не только голоса, но и видеоизображения, и наличие этих требований стало первым критерием для деления трафика на классы.

Другим важным параметром трафика, существенно влияющим на способ его передачи через сеть, является величина его пульсаций. Разработчики технологии ATM решили выделить два различных типа трафика в отношении этого параметра – трафик с постоянной битовой скоростью (Constant Bit Rate, CBR) и трафик с переменной битовой скоростью (Variable Bit Rate, VBR).

К разным классам были отнесены трафики, порождаемые приложениями, использующими для обмена сообщениями протоколы с установлением соединений и без установления соединений. В первом случае данные передаются самим приложением достаточно надежно, как это обычно делают протоколы с установлением соединения, поэтому от сети ATM высокой надежности передачи не требуется. А во втором случае приложение работает без установления соединения и восстановлением потерянных и искаженных данных не занимается, что предъявляет повышенные требования к надежности передачи ячеек сетью ATM.

В результате было определено пять классов трафика, отличающихся следующими качественными характеристиками:

- наличием или отсутствием пульсации трафика, то есть трафики CBR или VBR;
- требованием к синхронизации данных между передающей и принимающей сторонами;
- типом протокола, передающего свои данные через сеть ATM, – с установлением соединения или без установления соединения (только для случая передачи компьютерных данных).

Основные характеристики классов трафика ATM приведены в табл. 6.4.

Таблица 6.4. Классы трафика ATM

Класс трафика	Характеристика
A	Постоянная битовая скорость – Constant Bit Rate, CBR. Требуются временные соотношения между передаваемыми и принимаемыми данными. С установлением соединения. Примеры: голосовой трафик, трафик телевизионного изображения
B	Переменная битовая скорость – Variable Bit Rate, VBR. Требуются временные соотношения между передаваемыми и принимаемыми данными. С установлением соединения. Примеры: компрессированный голос, компрессированное видеоизображение
C	Переменная битовая скорость – Variable Bit Rate, VBR. Не требуются временные соотношения между передаваемыми и принимаемыми данными. С установлением соединения. Примеры: трафик компьютерных сетей, в которых конечные узлы работают по протоколам с установлением соединений: frame relay, X.25, LLC2, TCP

D	Переменная битовая скорость – Variable Bit Rate, VBR. Не требуются временные соотношения между передаваемыми и принимаемыми данными. Без установления соединения. Примеры: трафик компьютерных сетей, в которых конечные узлы работают по протоколам без установления соединений (IP, Ethernet DNS, SNMP)
X	Тип трафика и его параметры определяются пользователем

Очевидно, что только качественных характеристик, задаваемых классом трафика, для описания требуемых услуг оказывается недостаточно. В технологии ATM для каждого класса трафика определен набор количественных параметров, которые приложение должно задать. Например, для трафика класса A необходимо указать постоянную скорость, с которой приложение будет посылать данные в сеть, а для трафика класса B – максимально возможную скорость, среднюю скорость и максимально возможную пульсацию. Для голосового трафика можно не только указать на важность синхронизации между передатчиком и приемником, но и количественно задать верхние границы задержки и вариации задержки ячеек.

В технологии ATM поддерживается следующий набор основных количественных параметров:

- Peak Cell Rate (PCR) – максимальная скорость передачи данных;
- Sustained Cell Rate (SCR) – средняя скорость передачи данных;
- Minimum Cell Rate (MCR) – минимальная скорость передачи данных;
- Maximum Burst Size (MBS) – максимальный размер пульсации;
- Cell Loss Ratio (CLR) – доля потерянных ячеек;
- Cell Transfer Delay (CTD) - задержка передачи ячеек;
- Cell Delay Variation (CDV) - вариация задержки ячеек.

Параметры скорости измеряются в ячейках в секунду, максимальный размер пульсации – в ячейках, а временные параметры – в секундах. Максимальный размер пульсации задает количество ячеек, которое приложение может передать с максимальной скоростью PCR, если задана средняя скорость. Доля потерянных ячеек является отношением потерянных ячеек к общему количеству отправленных ячеек по данному виртуальному соединению. Так как виртуальные соединения являются дуплексными, то для каждого направления соединения могут быть заданы разные значения параметров.

В технологии ATM принят не совсем традиционный подход к трактовке термина «качество обслуживания» - QoS. Обычно качество обслуживания трафика характеризуется параметрами пропускной способности (здесь это RCR, SCR, MCR, MBS), параметрами задержек пакетов (CTD и CDV), а также параметрами надежности передачи пакетов (CLR). В ATM характеристики пропускной способности называют *параметрами трафика* и не включают их в число параметров качества обслуживания QoS, хотя по существу они таковыми являются. Параметрами QoS в ATM являются только параметры CTD, CDV и CLR. Сеть старается обеспечить такой уровень услуг, чтобы поддерживались требуемые значения и параметров трафика, и задержек ячеек, и доли потерянных ячеек.

Соглашение между приложением и сетью ATM называется трафик-контрактом. Основным его отличием от соглашений, применяемых в сетях frame relay, является выбор одного из нескольких определенных классов трафика, для которого наряду с параметрами пропускной способности трафика могут указываться параметры задержек ячеек, а также параметр надежности доставки ячеек. В сети frame relay класс трафика один, и он характеризуется только параметрами пропускной способности.

Необходимо подчеркнуть, что задание только параметров трафика (вместе с параметрами QoS) часто не полностью характеризует требуемую услугу, поэтому задание класса трафика полезно для уточнения нужного характера обслуживания данного соединения сетью.

В некоторых случаях специфика приложения такова, что ее трафик не может быть отнесен к одному из четырех стандартных классов. Поэтому для этого случая введен еще один класс X, который не имеет никаких дополнительных описаний, а полностью определяется теми количественными параметрами трафика и QoS, которые оговариваются в трафик-контракте.

Если для приложения не критично поддержание параметров пропускной способности и QoS, то оно

может отказаться от задания этих параметров, указав признак «Best Effort» в запросе на установление соединения. Такой тип трафика получил название трафика с неопределенной битовой скоростью – Unspecified Bit Rate, UBR.

После заключения трафик-контракта, который относится к определенному виртуальному соединению, в сети ATM работает несколько протоколов и служб, обеспечивающих нужное качество обнаруживания. Для трафика UBR сеть выделяет ресурсы «по возможности», то есть те, которые в данный момент свободны от использования виртуальными соединениями, заказавшими определенные параметры качества обслуживания.

Технология ATM изначально разрабатывалась для поддержки как постоянных, так и коммутируемых виртуальных каналов (в отличие от технологии frame relay, долгое время не поддерживающей коммутируемые виртуальные каналы). Автоматическое заключение трафик-контракта при установлении коммутируемого виртуального соединения представляет собой весьма непростую задачу, так как коммутаторам ATM необходимо определить, смогут ли они в дальнейшем обеспечить передачу трафика данного виртуального канала наряду с трафиком других виртуальных каналов таким образом, чтобы выполнялись требования качества обслуживания каждого канала.

Стек протоколов ATM

Стек протоколов ATM показан на рис 6.30, а распределение протоколов по конечным узлам и коммутаторам ATM – на рис. 6.31.

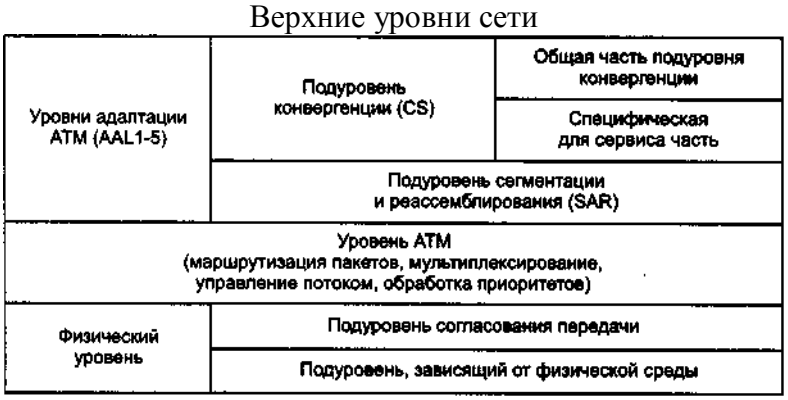


Рис. 6.30. Структура стека протоколов ATM

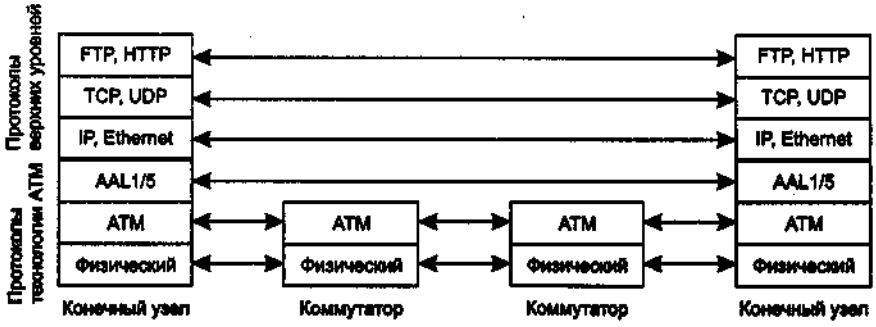


Рис. 6.31. Распределение протоколов по узлам и коммутаторам сети ATM

Стек протоколов ATM соответствует нижним уровням семиуровневой модели ISO/OSI и включает уровень адаптации ATM, собственно уровень ATM и физический уровень. Прямого соответствия между уровнями протоколов технологии ATM и уровнями модели OSI нет.

Уровень адаптации AAL

Уровень адаптации (ATM Adaptation Layer, AAL) представляет собой набор протоколов AAL1-AAL5, которые преобразуют сообщения протоколов верхних уровней сети ATM в ячейки ATM нужного формата. Функции этих уровней достаточно условно соответствуют функциям транспортного уровня

модели OSI, например функциям протоколов TCP или UDP. Протоколы AAL при передаче пользовательского трафика работают только в конечных узлах сети (см. рис. 6.31), как и транспортные протоколы большинства технологий.

Каждый протокол уровня AAL обрабатывает пользовательский трафик определенного класса. На начальных этапах стандартизации каждому классу трафика соответствовал свой протокол AAL, который принимал в конечном узле пакеты от протокола верхнего уровня и заказывал с помощью соответствующего протокола нужные параметры трафика и качества обслуживания для данного виртуального канала. При развитии стандартов ATM такое однозначное соответствие между классами трафика и протоколами уровня AAL исчезло, и сегодня разрешается использовать для одного и того же класса трафика различные протоколы уровня AAL.

Уровень адаптации состоит из нескольких подуровней. Нижний подуровень AAL называется подуровнем сегментации и реассемблирования (Segmentation And Reassembly, SAR). Эта часть не зависит от типа протокола AAL (и, соответственно, от класса передаваемого трафика) и занимается разбиением (сегментацией) сообщения, принимаемого AAL от протокола верхнего уровня, на ячейки ATM, снабжением их соответствующим заголовком и передачей уровню ATM для отправки в сеть.

Верхний подуровень AAL называется подуровнем конвергенции – Convergence Sublayer, CS. Этот подуровень зависит от класса передаваемого трафика. Протокол подуровня конвергенции решает такие задачи, как, например, обеспечение временной синхронизации между передающим и принимающим узлами (для трафика, требующего такой синхронизации), контролем и возможным восстановлением битовых ошибок в пользовательской информации, контролем целостности передаваемого пакета компьютерного протокола (X.25, frame relay).

Протоколы AAL для выполнения своей работы используют служебную информацию, размещаемую в заголовках уровня AAL. После приема ячеек, пришедших по виртуальному каналу, подуровень SAR протокола AAL собирает посланное по сети исходное сообщение (которое в общем случае было разбито на несколько ячеек ATM) с помощью заголовков AAL, которые для коммутаторов ATM являются прозрачными, так как помещаются в 48-битном поле данных ячейки, как и полагается протоколу более высокого уровня. После сборки исходного сообщения протокол AAL проверяет служебные поля заголовка и концевика кадра AAL и на их основании принимает решение о корректности полученной информации.

Ни один из протоколов AAL при передаче пользовательских данных конечных узлов не занимается восстановлением потерянных или искаженных данных. Максимум, что делает протокол AAL, – это уведомляет конечный узел о таком событии. Так сделано для ускорения работы коммутаторов сети ATM в расчете на то, что случаи потерь или искажения данных будут редкими. Восстановление потерянных данных (или игнорирование этого события) отводится протоколам верхних уровней, не входящим в стек протоколов технологии ATM.

Протокол AAL1 обычно обслуживает трафик класса A с постоянной битовой скоростью (Constant Bit Rate, CBR), который характерен, например, для цифрового видео и цифровой речи и чувствителен к временным задержкам. Этот трафик передается в сетях ATM таким образом, чтобы эмулировать обычные выделенные цифровые линии. Заголовок AAL1 занимает в поле данных ячейки ATM 1 или 2 байта, оставляя для передачи пользовательских данных соответственно 47 или 46 байт. В заголовке один байт отводится для нумерации ячеек, чтобы приемная сторона могла судить о том, все ли посланные ячейки дошли до нее или нет. При отправке голосового трафика временная отметка каждого замера известна, так как они следуют друг за другом с интервалом в 125 мкс, поэтому при потере ячейки можно скорректировать временную привязку байт следующей ячейки, сдвинув ее на 125x46 мкс. Потеря нескольких байт замеров голоса не так страшна, так как на приемной стороне воспроизводящее оборудование сглаживает сигнал. В задачи протокола AAL1 входит сглаживание неравномерности поступления ячеек данных в узел назначения.

Протокол AAL2 был разработан для передачи трафика класса B, но при развитии стандартов он был исключен из стека протоколов ATM, и сегодня трафик класса B передается с помощью протокола AAL1, AAL3/4 или AAL5.

Протокол AAL3/4 обрабатывает пульсирующий трафик – обычно характерный для трафика локальных сетей – с переменной битовой скоростью (Variable Bit Rate, VBR). Этот трафик обрабатывается так, чтобы не допустить потерь ячеек, но ячейки могут задерживаться коммутатором. Протокол AAL3/4 выполняет сложную процедуру контроля ошибок при передаче ячеек, нумеруя каждую составляющую часть исходного сообщения и снабжая каждую ячейку контрольной суммой.

Правда, при искажениях или потерях ячеек уровень не занимается их восстановлением, а просто отбрасывает все сообщение – то есть все оставшиеся ячейки, так как для компьютерного трафика или компрессированного голоса потеря части данных является фатальной ошибкой. Протокол AAL3/4 образовался в результате слияния протоколов AAL3 и AAL4, которые обеспечивали поддержку трафика компьютерных сетей соответственно с установлением соединения и без установления соединения. Однако ввиду большой близости используемых форматов служебных заголовков и логики работы протоколы AAL3 и AAL4 были впоследствии объединены.

Протокол AAL5 является упрощенным вариантом протокола AAL4 и работает быстрее, так как вычисляет контрольную сумму не для каждой ячейки сообщения, а для всего исходного сообщения в целом и помещает ее в последнюю ячейку сообщения. Первоначально протокол AAL5 разрабатывался для передачи кадров сетей frame relay, но теперь он чаще всего используется для передачи любого компьютерного трафика. Протокол AAL5 может поддерживать различные параметры качества обслуживания, кроме тех, которые связаны с синхронизацией передающей и принимающей сторон. Поэтому он обычно используется для поддержки всех классов трафика, относящегося к передаче компьютерных данных, то есть классов C и D. Некоторые производители оборудования с помощью протокола AAL5 обслуживают трафик CBR, оставляя задачу синхронизации трафика протоколам верхнего уровня.

Протокол AAL5 работает не только в конечных узлах, но и в коммутаторах сети ATM. Однако там он выполняет служебные функции, не связанные с передачей пользовательских данных. В коммутаторах ATM протокол AAL5 поддерживает служебные протоколы более высоких уровней, занимающиеся установлением коммутируемых виртуальных соединений.

Существует определенный интерфейс между приложением, которому требуется передать трафик через сеть ATM, и уровнем адаптации AAL. С помощью этого интерфейса приложение (протокол компьютерной сети, модуль оцифровывания голоса) заказывает требуемую услугу, определяя тип трафика, его параметры, а также параметры QoS. Технология ATM допускает два варианта определения параметров QoS: первый – непосредственное задание их каждым приложением, второй – назначение их по умолчанию в зависимости от типа трафика. Последний способ упрощает задачу разработчика приложения, так как в этом случае выбор максимальных значений задержки доставки ячеек и вариации задержек перекладывается на плечи администратора сети.

Самостоятельно обеспечить требуемые параметры трафика и QoS протоколы AAL не могут. Для выполнения соглашений трафик-контракта требуется согласованная работа коммутаторов сети вдоль всего виртуального соединения. Эта работа выполняется протоколом ATM, обеспечивающим передачу ячеек различных виртуальных соединений с заданным уровнем качества обслуживания.

Протокол ATM

Протокол ATM занимает в стеке протоколов ATM примерно то же место, что протокол IP в стеке TCP/IP или протокол LAP-F в стеке протоколов технологии frame relay. Протокол ATM занимается передачей ячеек через коммутаторы при установленном и настроенном виртуальном соединении, то есть на основании готовых таблиц коммутации портов. Протокол ATM выполняет коммутацию по номеру виртуального соединения, который в технологии ATM разбит на две части – *идентификатор виртуального пути* (*Virtual Path Identifier, VPI*) и *идентификатор виртуального канала* (*Virtual Channel Identifier, VCI*). Кроме этой основной задачи протокол ATM выполняет ряд функций по контролю за соблюдением трафик-контракта со стороны пользователя сети, маркировке ячеек-нарушителей, отбрасыванию ячеек-нарушителей при перегрузке сети, а также управлению потоком ячеек для повышения производительности сети (естественно, при соблюдении условий трафик-контракта для всех виртуальных соединений).

Протокол ATM работает с ячейками следующего формата, представленного на рис. 6.32.

Поле *Управление потоком* (*Generic Flow Control*) используется только при взаимодействии конечного узла и первого коммутатора сети. В настоящее время его точные функции не определены.

Поля *Идентификатор виртуального пути* (*Vitual Path Identifier, VPI*) и *Идентификатор виртуального канала* (*Vitual Channel Identifier, VCI*) занимают соответственно 1 и 2 байта. Эти поля задают номер виртуального соединения, разделенный на старшую (VPI) и младшую (VCI) части.

Поле *Идентификатор типа данных* (*Payload Type Identifier, PTI*) состоит из 3-х бит и задает тип данных, переносимых ячейкой, – пользовательские или управляющие (например, управляющие

установлением виртуального соединения). Кроме того, один бит этого поля используется для указания перегрузки в сети – он называется Explicit Congestion Forward Identifier, EFCI – и играет ту же роль, что бит FECN в технологии frame relay, то есть передает информацию о перегрузке по направлению потока данных.

Поле *Приоритет потери кадра* (*CellLoss Priority, CLP*) играет в данной технологии ту же роль, что и поле DE в технологии frame relay – в нем коммутаторы ATM отмечают ячейки, которые нарушают соглашения о параметрах качества обслуживания, чтобы удалить их при перегрузках сети. Таким образом, ячейки с CLP=0 являются для сети высокоприоритетными, а ячейки с CLP=1 – низкоприоритетными.

Поле *Управление ошибками в заголовке* (*Header Error Control, HEC*) содержит контрольную сумму, вычисленную для заголовка ячейки. Контрольная сумма вычисляется с помощью техники корректирующих кодов Хэмминга, поэтому она позволяет не только обнаруживать ошибки, но и исправлять все одиночные ошибки, а также некоторые двойные. Поле HEC обеспечивает не только обнаружение и исправление ошибок в заголовке, но и нахождение границы начала кадра в потоке байтов кадров SDH, которые являются предпочтительным физическим уровнем технологии ATM, или же в потоке бит физического уровня, основанного на ячейках. Указателей, позволяющих в поле данных кадра STS-n (STM-n) технологии SONET/SDH обнаруживать границы ячеек ATM (подобных тем указателям, которые используются для определения, например, границ виртуальных контейнеров подканалов T1/E1), не существует. Поэтому коммутатор ATM вычисляет контрольную сумму для последовательности из 5 байт, находящихся в поле данных кадра STM-n, и, если вычисленная контрольная сумма говорит о корректности заголовка ячейки ATM, первый байт становится границей ячейки. Если же это не так, то происходит сдвиг на один байт и операция продолжается. Таким образом, технология ATM выделяет асинхронный поток ячеек ATM в синхронных кадрах SDH или потоке бит физического уровня, основанного на ячейках.

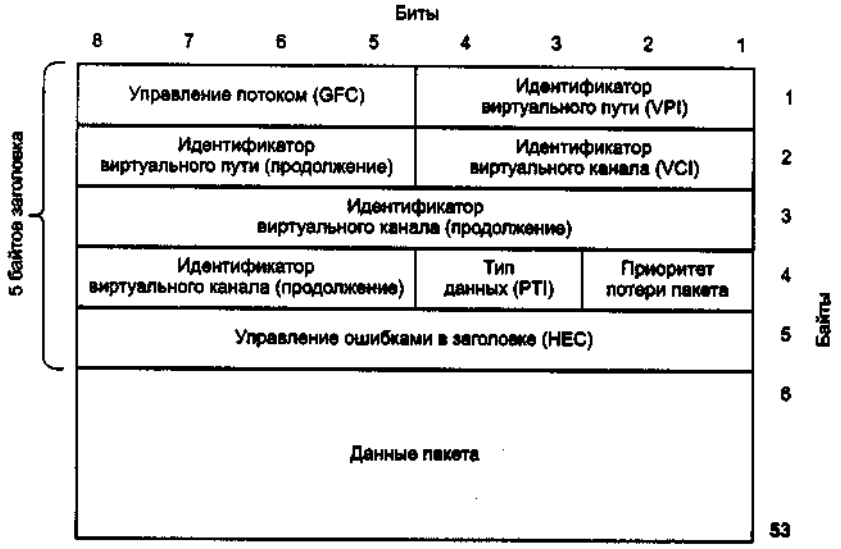


Рис. 6.32. Формат ячейки ATM

Рассмотрим методы коммутации ячеек ATM на основе пары чисел VPI/VCI. Коммутаторы ATM могут работать в двух режимах – коммутации виртуального пути и коммутации виртуального канала. В первом режиме коммутатор выполняет продвижение ячейки только на основании значения поля VPI, а значение поля VCI он игнорирует. Обычно так работают магистральные коммутаторы территориальных сетей. Они доставляют ячейки из одной сети пользователя в другую на основании только старшей части номера виртуального канала, что соответствует идее агрегирования адресов. В результате один виртуальный путь соответствует целому набору виртуальных каналов, коммутируемых как единое целое.

После доставки ячейки в локальную сеть ATM ее коммутаторы начинают коммутировать ячейки с учетом как VPI, так и VCI, но при этом им хватает для коммутации только младшей части номера виртуального соединения, так что фактически они работают с VCI, оставляя VPI без изменения. Последний режим называется режимом коммутации виртуального канала.

Для создания коммутируемого виртуального канала в технологии ATM используются протоколы, не

показанные на рис. 6.30. Подход здесь аналогичен подходу в сети ISDN – для установления соединения разработан отдельный протокол Q.2931, который весьма условно можно отнести к сетевому уровню. Этот протокол во многом похож на протоколы Q.931 и Q.933 (даже номером), но в него внесены, естественно, изменения, связанные с наличием нескольких классов трафика и дополнительных параметров качества обслуживания. Протокол Q.2931 опирается на достаточно сложный протокол канального уровня SSCOP, который обеспечивает надежную передачу пакетов Q.2931 в своих кадрах. В свою очередь, протокол SSCOP работает поверх протокола AAL5, который необходим для разбиения кадров SSCOP на ячейки ATM и сборки этих ячеек в кадры при доставке кадра SSCOP в коммутатор назначения.

ПРИМЕЧАНИЕ Протокол Q.2931 появился в стеке протоколов технологии ATM после принятия версии интерфейса UNI 3.1, а до этого в версии UNI 3.0 вместо него использовался протокол Q.93B. Из-за несовместимости протоколов Q.2931 и Q.93B версии пользовательского интерфейса UNI 3.0 и UNI 3.1 также несовместимы. Версия UNI 4.0 обратно совмеаима с UNI 3.1, так как основана на тех же служебных протоколах, что и версия UNI 3.1.

Виртуальные соединения, образованные с помощью протокола Q.2931, бывают симплексными (однонаправленными) и дуплексными.

Протокол Q.2931 позволяет также устанавливать виртуальные соединения типа «один-к-одному» (point-to-point) и «один-ко-многим» (point-to-multipoint). Первый случай поддерживается во всех технологиях, основанных на виртуальных каналах, а второй характерен для технологии ATM и является аналогом мультивещания, но с одним ведущим вещающим узлом. При установлении соединения «один-ко-многим» ведущим считается узел, который является инициатором этого соединения. Сначала этот узел устанавливает виртуальное соединение всего с одним узлом, а затем добавляет к соединению с помощью специального вызова по одному новому члену. Ведущий узел становится вершиной дерева соединения, а остальные узлы – листьями этого дерева. Сообщения, которые посылает ведущий узел, принимают все листья соединения, но сообщения, которые посылает какой-либо лист (если соединение дуплексное), принимает только ведущий узел.

Пакеты протокола Q.2931, предназначенные для установления коммутируемого виртуального канала, имеют те же названия и назначение, что и пакеты протокола Q.933, рассмотренные выше при изучении технологии frame relay, но структура их полей, естественно, другая.

Адресом конечного узла в коммутаторах ATM является 20-байтный адрес. Этот адрес может иметь различный формат, описываемый стандартом ISO 7498. При работе в публичных сетях используется адрес стандарта E.164, при этом 1 байт составляет AFI, 8 байт занимает IDI – основная часть адреса E.164 (15 цифр телефонного номера), а остальные 11 байт части DSP (Domain Specific Part) распределяются следующим образом.

- 4 байта занимает поле старшей части DSP – High-Order Domain Spesifie Part (HO-DSP), имеющее гибкий формат и, в сущности, представляющее собой номер сети ATM, который может делиться на части для агрегированной маршрутизации по протоколу PNNI, подобной той, которая используется в технике CIDR для сетей IP.
- 6 байт занимает поле идентификатора конечной системы – End System Identifier (ESI), которое имеет смысл MAC-адреса узла ATM, причем формат его также соответствует формату MAC-адресов IEEE.
- 1 байт составляет поле селектора, которое не используется при установлении виртуального канала, а имеет для узла локальное назначение.

При работе в частных сетях ATM обычно применяется формат адреса, соответствующий домену международных организаций, причем в качестве международной организации выступает ATM Forum. В этом случае поле IDI занимает 2 байта, которые содержат код ATM Forum, данный ISO, а структура остальной части DSP соответствует описанной выше за исключением того, что поле HO-DSP занимает не 4, а 10 байт.

Адрес ESI присваивается конечному узлу на предприятии-изготовителе в соответствии с правилами IEEE, то есть 3 первых байта содержат код предприятия, а остальные три байта – порядковый номер, за уникальность которого отвечает данное предприятие.

Конечный узел при подключении к коммутатору ATM выполняет так называемую процедуру

регистрации. При этом конечный узел сообщает коммутатору свой ESI-адрес, а коммутатор сообщает конечному узлу старшую часть адреса, то есть номер сети, в которой работает узел.

Кроме адресной части пакет CALL SETUP протокола Q.2931, с помощью которого конечный узел запрашивает установление виртуального соединения, включает также части, описывающие параметры трафика и требования QoS. При поступлении такого пакета коммутатор должен проанализировать эти параметры и решить, достаточно ли у него свободных ресурсов производительности для обслуживания нового виртуального соединения. Если да, то новое виртуальное соединение принимается и коммутатор передает пакет CALL SETUP дальше в соответствии с адресом назначения и таблицей маршрутизации, а если нет, то запрос отвергается.

Категории услуг протокола ATM и управление трафиком

Для поддержания требуемого качества обслуживания различных виртуальных соединений и рационального использования ресурсов в сети на уровне протокола ATM реализовано несколько служб, предоставляющих услуги различных категорий (service categories) по обслуживанию пользовательского трафика. Эти службы являются внутренними службами сети ATM, они предназначены для поддержания пользовательского трафика различных классов совместно с протоколами AAL. Но в отличие от протоколов AAL, которые работают в конечных узлах сети, данные службы распределены по всем коммутаторам сета. Услуги этих служб разбиты на категории, которые в общем соответствуют классам трафика поступающим на вход уровня AAL конечного узла. Услуги уровня ATM заказываются конечным узлом через интерфейс UNI с помощью протокола Q.2931 при установлении виртуального соединения. Как и при обращении к уровню AAL, при заказе услуги необходимо указать категорию услуги, а также параметры трафика и параметры QoS. Эти параметры берутся из аналогичных параметров уровня AAL или же определяются по умолчанию в зависимости от категории услуги.

Всего на уровне протокола ATM определено пять категорий услуг, которые поддерживаются одноименными службами:

- CBR – услуги для трафика с постоянной битовой скоростью;
- rtVBR – услуги для трафика с переменной битовой скоростью, требующего соблюдения средней скорости передачи данных и синхронизации источника и приемника;
- nrtVBR – услуги для трафика с переменной битовой скоростью, требующего соблюдения средней скорости передачи данных и не требующего синхронизации источника и приемника;
- ABR – услуги для трафика с переменной битовой скоростью, требующего соблюдения некоторой минимальной скорости передачи данных и не требующего синхронизации источника и приемника;
- UBR – услуги для трафика, не предъявляющего требований к скорости передачи данных и синхронизации источника и приемника.

Названия большинства категорий услуг совпадают с названием типов пользовательского трафика, для обслуживания которого они разработаны, но необходимо понимать, что сами службы уровня ATM и их услуги – это внутренние механизмы сети ATM, которые экранируются от приложения уровнем AAL.

Услуги категории CBR предназначены для поддержания трафика синхронных приложений – голосового, эмуляции цифровых выделенных каналов и т. п. Когда приложение устанавливает соединение категории CBR, оно заказывает пиковую скорость трафика ячеек PCR, являющуюся максимальной скоростью, которую может поддерживать соединение без риска потерять ячейку, а также параметры QoS: величины максимальной задержки ячеек CTD, вариации задержки ячеек CDV и максимальной доли потерянных ячеек CLR.

Затем данные передаются по этому соединению с запрошенной скоростью – не с большей и, в большинстве случаев, не меньшей, хотя уменьшение скорости приложением возможно, например, при передаче компрессированного голоса с помощью услуги категории CBR. Любые ячейки, передаваемые станцией с большей скоростью, контролируются первым коммутатором сети и помечаются признаком CLP=1. При перегрузках сети они могут просто отбрасываться сетью. Ячейки, которые запаздывают и не укладываются в интервал, оговоренный параметром вариации задержки CDV, также считаются мало значащими для приложения и отмечаются признаком низкого приоритета CLP=1.

Для соединений CBR нет ограничений на некоторую дискретность заказа скорости PCR, как, например, в каналах T1/E1, где скорость должна быть кратна 64 Кбит/с.

По сравнению со службой CBR, службы VBR требуют более сложной процедуры заказа соединения между сетью и приложением. В дополнение к пиковой скорости PCR приложение VBR заказывает еще и два других параметра: длительно поддерживаемую скорость - SCR, которая представляет собой среднюю скорость передачи данных, разрешенную приложению, а также максимальный размер пульсации – MBS. Максимальный размер пульсации измеряется в количестве ячеек ATM. Пользователь может превышать скорость вплоть до величины PCR, но только на короткие периоды времени, в течение которых передается объем данных, не превышающий MBS. Этот период времени называется Burst Tolerance, BT – терпимость к пульсации. Сеть вычисляет этот период как производный от трех заданных значений PCR, SCR и MBS.

Если скорость PCR наблюдается в течение периода времени, большего чем BT, то ячейки помечаются как нарушители – устанавливается признак CLP=1.

Для услуг категории rtVBR задаются и контролируются те же параметры QoS, что и для услуг категории CBR, а услуги категории nrtVBR ограничиваются поддержанием параметров трафика. Сеть также поддерживает для обеих категорий услуг VBR определенный максимальный уровень доли потерянных ячеек CLR, который либо задается явно при установлении соединения, либо назначается по умолчанию в зависимости от класса трафика.

Для контроля параметров трафика и QoS в технологии ATM применяется так называемый обобщенный алгоритм контроля скорости ячеек – Generic Cell Rate Algorithm, который может проверять соблюдение пользователем и сетью таких параметров, как PCR, CDV, SCR, BT, CTD и CDV. Он работает по модифицированному алгоритму «дырявого ведра», применяемому в технологии frame relay.

Для многих приложений, которые могут быть чрезвычайно «взрывными» в отношении интенсивности трафика, невозможно точно предсказать параметры трафика, оговариваемые при установлении соединения. Например, обработка транзакций или трафик двух взаимодействующих локальных сетей непредсказуемы по своей природе – изменения интенсивности трафика слишком велики, чтобы заключить с сетью какое-либо разумное соглашение.

В отличие от CBR и обеих служб VBR, служба UBR не поддерживает ни параметры трафика, ни параметры качества обслуживания. Служба UBR предлагает только доставку «по возможности» без каких-либо гарантий. Разработанная специально для обеспечения возможности превышения полосы пропускания, служба UBR представляет собой частичное решение для тех непредсказуемых «взрывных» приложений, которые не готовы согласиться с фиксацией параметров трафика.

Главными недостатками услуг UBR являются отсутствие управления потоком данных и неспособность принимать во внимание другие типы трафика. Несмотря на перегрузку сети, соединения UBR будут продолжать передачу данных. Коммутаторы сети могут буферизовать некоторые ячейки поступающего трафика, но в некоторый момент буферы переполняются, и ячейки теряются. А так как для соединений UBR не оговаривается никаких параметров трафика и QoS, то их ячейки отбрасываются в первую очередь.

Служба ABR подобно службе UBR предоставляет возможность превышения полосы пропускания, но благодаря технике управления трафиком при перегрузке сети она дает некоторые гарантии сохранности ячеек. ABR – это первый тип служб уровня ATM, который действительно обеспечивает надежный транспорт для пульсирующего трафика за счет того, что может находить неиспользуемые интервалы в общем трафике сети и заполнять их своими ячейками, если другим категориям служб эти интервалы не нужны.

Как и в службах CBR и VBR, при установлении соединения категории ABR оговаривается значение пиковой скорости PCR. Однако соглашение о пределах изменения задержки передачи ячеек или о параметрах пульсации не заключается. Вместо этого сеть и конечный узел заключают соглашение о требуемой минимальной скорости передачи MCR. Это гарантирует приложению, работающему в конечном узле, небольшую пропускную способность, обычно минимально необходимую для того, чтобы приложение работало. Конечный узел соглашается не передавать данные со скоростью, выше пиковой, то есть PCR, а сеть соглашается всегда обеспечивать минимальную скорость передачи ячеек MCR.

Если при установлении соединения ABR не задаются значения максимальной и минимальной скорости, то по умолчанию считается, что PCR совпадает со скоростью линии доступа станции к сети, а MCR считается равной нулю.

Трафик соединения категории ABR получает гарантированное качество услуг в отношении доли потерянных ячеек и пропускной способности. Что касается задержек передачи ячеек, то хотя сеть и

старается свести их к минимуму, но гарантий по этому параметру не дает. Следовательно, служба ABR не предназначена для приложений реального времени, а предназначена для приложений, в которых поток данных не очень чувствителен к задержкам в передаче.

При передаче трафика CBR, VBR и UBR явное управление перегрузками в сети отсутствует. Вместо этого используется механизм отбрасывания ячеек-нарушителей, а узлы, пользующиеся услугами CBR и VBR, стараются не нарушать условия контракта под угрозой потери ячеек, поэтому они обычно не пользуются дополнительной пропускной способностью, даже если она в данный момент доступна в сети.

Служба ABR позволяет воспользоваться резервами пропускной способности сети, так как сообщает конечному узлу о наличии в данный момент избыточной пропускной способности с помощью механизма обратной связи. Этот же механизм может помочь службе ABR снизить скорость передачи данных конечным узлом в сеть (вплоть до минимального значения MCR), если сеть испытывает перегрузку. Узел, пользующийся услугами ABR, должен периодически посылать в сеть наряду с ячейками данных специальные служебные ячейки управления ресурсами – Resource Management, RM. Ячейки RM, которые узел отправляет вдоль потока данных, называются прямыми ячейками RM – Forward Recource Management (FRM), а ячейки, которые идут в обратном по отношению к потоку данных направлении, называются обратными ячейками RM – Backward Recource Management (BRM).

Существует несколько петель обратной связи. Самая простая петля обратной связи – между конечными станциями. При ее наличии коммутатор сети извещает конечную станцию о перегрузке с помощью специального флага в поле прямого управления перегрузками (флаг EFCI) ячейки данных, переносимой протоколом ATM. Затем конечная станция посылает через сеть сообщение, содержащееся в специальной ячейке управления BRM исходной станции, говоря ей о необходимости уменьшить скорость посылки ячеек в сеть.

В этом способе конечная станция несет основную ответственность за управление потоком, а коммутаторы играют пассивную роль в петле обратной связи, только уведомляя станцию-отправитель о перегрузке.

Такой простой способ имеет несколько очевидных недостатков. Конечная станция не узнает из сообщения BRM, на какую величину нужно уменьшить скорость передачи данных в сеть. Поэтому она просто понизит скорость до минимальной величины MCR, хотя, возможно, это и не обязательно. Кроме того, при большой протяженности сети коммутаторы должны продолжать буферизовать данные все время, пока уведомление о перегрузке будет путешествовать по сети, а для глобальных сетей это время может быть достаточно большим, и буферы могут переполниться, так что требуемый эффект достигнут не будет.

Разработаны и более сложные схемы управления потоком, в которых коммутаторы играют более активную роль, а узел-отправитель узнает более точно о возможной в данный момент скорости отправки данных в сеть.

В первой схеме узел-источник посылает в ячейке FRM явное значение скорости передачи данных в сеть, которую он хотел бы поддерживать в данное время. Каждый коммутатор, через который проходит по виртуальному пути это сообщение, может уменьшить запрашиваемую скорость до некоторой величины, которую он может поддерживать в соответствии с имеющимися у него свободными ресурсами (или оставить запрашиваемую скорость без изменения). Узел назначения, получив ячейку FRM, превращает ее в ячейку BRM и отправляет в обратном направлении, причем он тоже может уменьшить запрашиваемую скорость. Получив ответ в ячейке BRM, узел-источник точно узнает, какая скорость отправки ячеек в сеть для него в данный момент доступна.

Во второй схеме каждый коммутатор сети может работать как узел-источник и узел назначения. Как узел-источник он может сам генерировать ячейки FRM и отправлять их по имеющимся виртуальным каналам. Как узел назначения он может отправлять на основе получаемых ячеек FRM ячейки BRM в обратном направлении. Такая схема является более быстродействующей и полезной в протяженных территориальных сетях.

Как видно из описания, служба ABR предназначена не только для прямого поддержания требований к обслуживанию конкретного виртуального соединения, но и для более рационального распределения ресурсов сети между ее абонентами, что в конечном итоге также приводит к повышению качества обслуживания всех абонентов сети.

Коммутаторы сети ATM используют различные механизмы для поддержания требуемого качества услуг. Кроме описанных в стандартах ITU-T и ATM Forum механизмов заключения соглашения на

основе параметров трафика и параметров QoS, а затем отбрасывания ячеек, не удовлетворяющих условиям соглашения, практически все производители оборудования ATM реализуют в своих коммутаторах несколько очередей ячеек, обслуживаемых с различными приоритетами.

Стратегия приоритетного обслуживания трафика основана на категориях услуг каждого виртуального соединения. До принятия спецификации ABR в большинстве коммутаторов ATM была реализована простая одноуровневая схема обслуживания, которая давала трафику CBR первый приоритет, трафику VBR второй, а трафику UBR – третий. При такой схеме комбинация CBR и VBR может потенциально заморозить трафик, обслуживаемый другим классом служб. Такая схема не будет правильно работать с трафиком ABR, так как не обеспечит его требования к минимальной скорости передачи ячеек. Для обеспечения этого требования должна быть выделена некоторая гарантированная полоса пропускания.

Чтобы поддерживать службу ABR, коммутаторы ATM должны реализовать двухуровневую схему обслуживания, которая бы удовлетворяла требованиям CBR, VBR и ABR. Но этой схеме коммутатор предоставляет некоторую часть своей пропускной способности каждому классу служб. Трафик CBR получает часть пропускной способности, необходимую для поддержания пиковой скорости PCR, трафик VBR получает часть пропускной способности, необходимую для поддержания средней скорости SCR, а трафик ABR получает часть пропускной способности, достаточную для обеспечения требования минимальной скорости ячеек MCR. Это гарантирует, что каждое соединение может работать без потерь ячеек и не будет доставлять ячейки ABR за счет трафика CBR или VBR. На втором уровне этого алгоритма трафик CBR и VBR может забрать всю оставшуюся пропускную способность сети, если это необходимо, так как соединения ABR уже получили свою минимальную пропускную способность, которая им гарантировалась.

Передача трафика IP через сети ATM

Технология ATM привлекает к себе общее внимание, так как претендует на роль всеобщего и очень гибкого транспорта, на основе которого строятся другие сети. И хотя технология ATM может использоваться непосредственно для транспортировки сообщений протоколов прикладного уровня, пока она чаще переносит пакеты других протоколов канального и сетевого уровней (Ethernet, IP, IPX, frame relay, X.25), сосуществуя с ними, а не полностью заменяя. Поэтому протоколы и спецификации, которые определяют способы взаимодействия технологии ATM с другими технологиями, очень важны для современных сетей. А так как протокол IP является на сегодня основным протоколом построения составных сетей, то стандарты работы IP через сети ATM являются стандартами, определяющими взаимодействие двух наиболее популярных технологий сегодняшнего дня.

Протокол Classical IP (RFC 1577) является первым (по времени появления) протоколом, определившим способ работы интерсети IP в том случае, когда одна из промежуточных сетей работает по технологии ATM. Из-за классической концепции подсетей протокол и получил свое название – Classical.

Одной из основных задач, решаемых протоколом Classical IP, является традиционная для IP-сетей задача – поиск локального адреса следующего маршрутизатора или конечного узла по его IP-адресу, то есть задача, возлагаемая в локальных сетях на протокол ARP. Поскольку сеть ATM не поддерживает широковещательность, традиционный для локальных сетей способ широковещательных ARP-запросов здесь не работает. Технология ATM, конечно, не единственная технология, в которой возникает такая проблема, – для обозначения таких технологий даже ввели специальный термин – «Нешироковещательные сети с множественным доступом» (Non-Broadcast networks with Multiple Access, NBMA). К сетям NBMA относятся, в частности, сети X.25 и frame relay.

В общем случае для нешироковещательных сетей стандарты TCP/IP определяют только ручной способ построения ARP-таблиц, однако для технологии ATM делается исключение – для нее разработана процедура автоматического отображения IP-адресов на локальные адреса. Такой особый подход к технологии ATM объясняется следующими причинами. Сети NBMA (в том числе X.25 и frame relay) используются, как правило, как транзитные глобальные сети, к которым подключается ограниченное число маршрутизаторов, а для небольшого числа маршрутизаторов можно задать ARP-таблицу вручную. Технология ATM отличается тем, что она применяется для построения не только глобальных, но и локальных сетей. В последнем случае размерность ARP-таблицы, которая должна содержать записи и о пограничных маршрутизаторах, и о множестве конечных узлов, может быть очень

большой. К тому же, для крупной локальной сети характерно постоянное изменение состава узлов, а значит, часто возникает необходимость в корректировке таблиц. Все это делает ручной вариант решения задачи отображения адресов для сетей ATM мало пригодным.

В соответствии со спецификацией Classical IP одна сеть ATM может быть представлена в виде нескольких IP-подсетей, так называемых логических подсетей (Logical IP Subnet, LIS) (рис. 6.33). Все узлы одной LIS имеют общий адрес сети. Как и в классической IP-сети, весь трафик между подсетями обязательно проходит через маршрутизатор, хотя и существует принципиальная возможность передавать его непосредственно через коммутаторы ATM, на которых построена сеть ATM. Маршрутизатор имеет интерфейсы во всех LIS, на которые разбита сеть ATM.

ПРИМЕЧАНИЕ Подход спецификации Classical IP к подсетям напоминает технику виртуальных локальных сетей VLAN -там также вводятся ограничения на имеющуюся возможность связи через коммутаторы для узлов, принадлежащих разным VLAN.

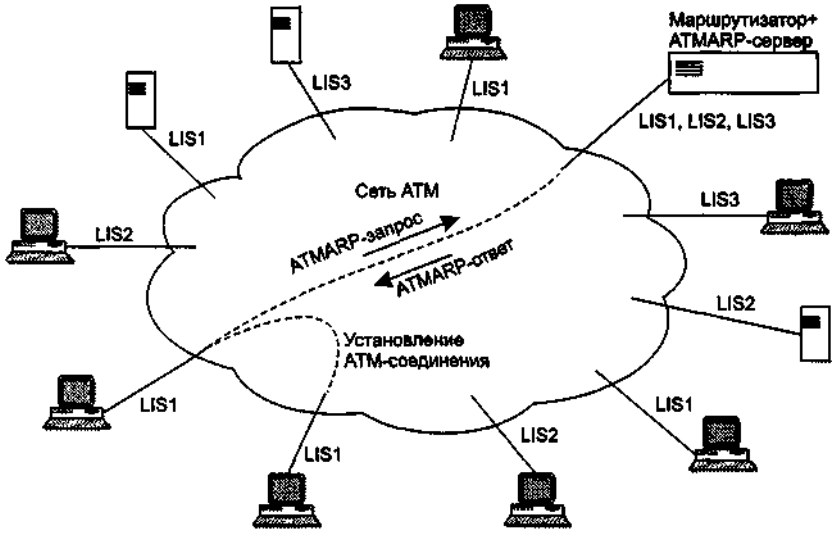


Рис. 6.33. Логические IP-подсети в сети ATM

В отличие от классических подсетей маршрутизатор может быть подключен к сети ATM одним физическим интерфейсом, которому присваивается несколько IP-адресов в соответствии с количеством LIS в сети.

Решение о введении логических подсетей связано с необходимостью обеспечения традиционного разделения большой сети ATM на независимые части, связность которых контролируется маршрутизаторами, как к этому привыкли сетевые интеграторы и администраторы. Решение имеет и очевидный недостаток – маршрутизатор должен быть достаточно производительным для передачи высокоскоростного трафика ATM между логическими подсетями, в противном случае он станет узким местом сети. В связи с повышенными требованиями по производительности, предъявляемыми сетями ATM к маршрутизаторам, многие ведущие производители разрабатывают или уже разработали модели маршрутизаторов с общей производительностью в несколько десятков миллионов пакетов в секунду.

Все конечные узлы конфигурируются традиционным образом – для них задается их собственный IP-адрес, маска и IP-адрес маршрутизатора по умолчанию. Кроме того, задается еще один дополнительный параметр – адрес ATM (или номер VPI/VCI для случая использования постоянного виртуального канала, то есть PVC) так называемого сервера ATMARP. Введение центрального сервера, который поддерживает общую базу данных для всех узлов сети, – это типичный прием для работы через нешироковещательную сеть. Этот прием используется во многих протоколах, в частности в протоколе LAN Emulation, рассматриваемом далее.

Каждый узел использует адрес ATM сервера ATMARP, чтобы выполнить обычный запрос ARP. Этот запрос имеет формат, очень близкий к формату запроса протокола ARP из стека TCP/IP. Длина аппаратного адреса в нем определена в 20 байт, что соответствует длине адреса ATM. В каждой логической подсети имеется свой сервер ATMARP, так как узел может обращаться без посредничества маршрутизатора только к узлам своей подсети. Обычно роль сервера ATMARP выполняет маршрутизатор, имеющий интерфейсы во всех логических подсетях.

При поступлении первого запроса ARP от конечного узла сервер сначала направляет ему встречный

инверсный запрос ATMARP, чтобы выяснить IP- и ATM-адреса этого узла. Этим способом выполняется регистрация каждого узла в сервере ATMARP, и сервер получает возможность автоматически строить базу данных соответствия IP- и ATM-адресов. Затем сервер пытается выполнить запрос ATMARP узла путем просмотра своей базы. Если искомый узел уже зарегистрировался в ней и он принадлежит той же логической подсети, что и запрашивающий узел, то сервер отправляет в качестве ответа запрашиваемый адрес. В противном случае дается негативный ответ (такой тип ответа в обычном широковещательном варианте протокола ARP не предусматривается).

Конечный узел, получив ответ ARP, узнает ATM-адрес своего соседа по логической подсети и устанавливает с ним коммутируемое виртуальное соединение. Если же он запрашивал ATM-адрес маршрутизатора по умолчанию, то он устанавливает с ним соединение, чтобы передать IP-пакет в другую сеть.

Для передачи IP-пакетов через сеть ATM спецификация Classical IP определяет использование протокола уровня адаптации AAL5, при этом спецификация ничего не говорит ни о параметрах трафика и качества обслуживания, ни о требуемой категории услуг CBR, rtVBR, nrtVBR или UBR.

Сосуществование ATM с традиционными технологиями локальных сетей

Технология ATM разрабатывалась сначала как «вещь в себе», без учета того факта, что в существующие технологии сделаны большие вложения и поэтому никто не станет сразу отказываться от установленного и работающего оборудования, даже если появляется новое, более совершенное. Это обстоятельство оказалось не столь важным для территориальных сетей, которые в случае необходимости могли предоставить свои оптоволоконные каналы для построения сетей ATM. Учитывая, что стоимость высокоскоростных оптоволоконных каналов, проложенных на большие расстояния, часто превышает стоимость остального сетевого оборудования, переход на новую технологию ATM, связанный с заменой коммутаторов, во многих случаях оказывался экономически оправданным.

Для локальных сетей, в которых замена коммутаторов и сетевых адаптеров равнозначна созданию новой сети» переход на технологию ATM мог быть вызван только весьма серьезными причинами. Гораздо привлекательнее полной замены существующей локальной сети новой сетью ATM выглядела возможность «постепенного» внедрения технологии ATM в существующую на предприятии сеть. При таком подходе фрагменты сети, работающие по новой технологии ATM, могли бы мирно сосуществовать с другими частями сети, построенными на основе традиционных технологий, таких как Ethernet или FDDI, улучшая характеристики сети там, где это нужно, и оставляя сети рабочих групп или отделов в прежнем виде. Применение маршрутизаторов IP, реализующих протокол Classical IP, решает эту проблему, но такое решение не всегда устраивает предприятия, пользующиеся услугами локальных сетей, так как, во-первых, требуется обязательная поддержка протокола IP во всех узлах локальных сетей, а во-вторых, требуется установка некоторого количества маршрутизаторов, что также не всегда приемлемо. Отчетливо ощущалась необходимость способа согласования технологии ATM с технологиями локальных сетей без привлечения сетевого уровня.

В ответ на такую потребность ATM Forum разработал спецификацию, называемую LAN emulation, LANE (то есть эмуляция локальных сетей), которая призвана обеспечить совместимость традиционных протоколов и оборудования локальных сетей с технологией ATM. Эта спецификация обеспечивает совместную работу этих технологий на канальном уровне. При таком подходе коммутаторы ATM работают в качестве высокоскоростных коммутаторов магистрали локальной сети, обеспечивая не только скорость, но и гибкость соединений коммутаторов ATM между собой, поддерживающих произвольную топологию связей, а не только древовидные структуры.

Спецификация LANE определяет способ преобразования кадров и адресов MAC-уровня традиционных технологий локальных сетей в ячейки и коммутируемые виртуальные соединения SVC технологии ATM, а также способ обратного преобразования. Всю работу по преобразованию протоколов выполняют специальные компоненты, встраиваемые в обычные коммутаторы локальных сетей, поэтому ни коммутаторы ATM, ни рабочие станции локальных сетей не замечают того, что они работают с чуждыми им технологиями. Такая прозрачность была одной из главных целей разработчиков спецификации LANE.

Так как эта спецификация определяет только канальный уровень взаимодействия, то с помощью коммутаторов ATM и компонентов эмуляции LAN можно образовать только виртуальные сети,

называемые здесь эмулируемыми сетями, а для их соединения нужно использовать обычные маршрутизаторы.

Рассмотрим основные идеи спецификации на примере сети, изображенной на рис. 6.34.

Основными элементами, реализующими спецификацию, являются программные компоненты LEC (LAN Emulation Client) и LES (LAN Emulation Server). Клиент LEC выполняет роль пограничного элемента, работающего между сетью ATM и станциями некоторой локальной сети. На каждую присоединенную к сети ATM локальную сеть приходится один клиент LEC.

Сервер LES ведет общую таблицу соответствия MAC-адресов станций локальных сетей и ATM-адресов пограничных устройств с установленными на них компонентами LEC, к которым присоединены локальные сети, содержащие эти станции. Таким образом, для каждой присоединенной локальной сети сервер LES хранит один ATM-адрес пограничного устройства LEC и несколько MAC-адресов станций, входящих в эту сеть. Клиентские части LEC динамически регистрируют в сервере LES MAC-адреса каждой станции, заново подключаемой к присоединенной локальной сети.

Программные компоненты LEC и LES могут быть реализованы в любых устройствах – коммутаторах, маршрутизаторах или рабочих станциях ATM.

Когда элемент LEC хочет послать пакет через сеть ATM станции другой локальной сети, также присоединенной к сети ATM, он посылает запрос на установление соответствия между MAC-адресом и ATM-адресом серверу LES. Сервер LES отвечает на запрос, указывая ATM-адрес пограничного устройства LEC, к которому присоединена сеть, содержащая станцию назначения. Зная ATM-адрес, устройство LEC исходной сети самостоятельно устанавливает виртуальное соединение SVC через сеть ATM обычным способом, описанным в спецификации UNI. После установления связи кадры MAC локальной сети преобразуются в ячейки ATM каждым элементом LEC с помощью стандартных функций сборки-разборки пакетов (функции SAR) стека ATM.

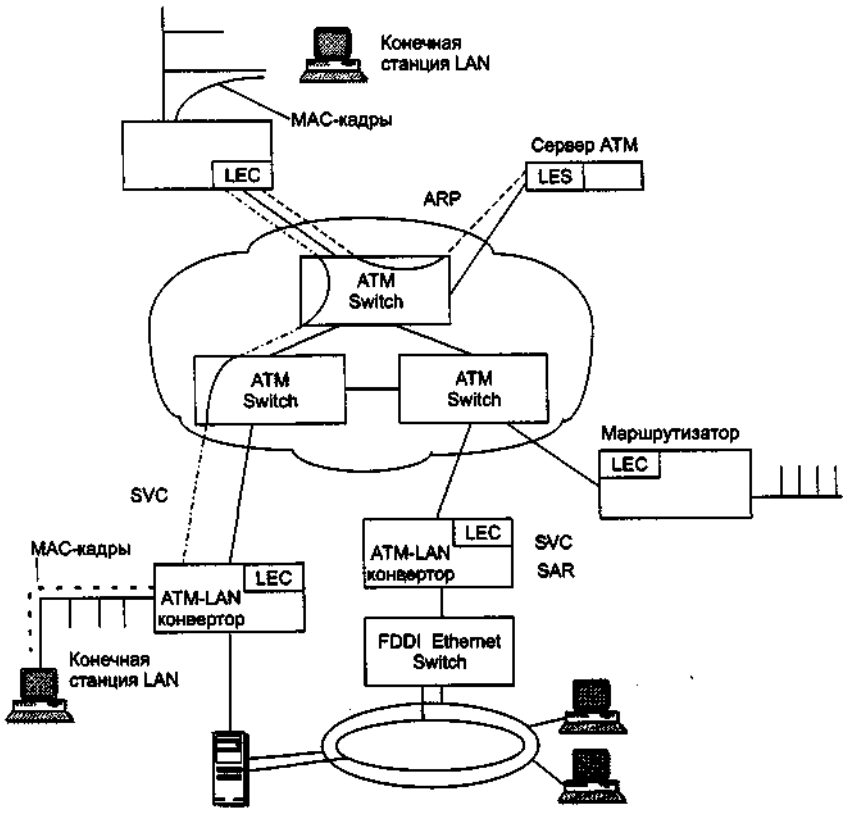


Рис. 6.34. Принципы работы технологии LAN emulation

В спецификации LANE также определен сервер для эмуляции в сети ATM широковещательных пакетов локалыших сетей, а также пакетов с неизвестными адресами, так называемый сервер BUS (Broadcast and Unknown Server). Этот сервер распространяет такие пакеты во все пограничные коммутароры, присоединившие свои сети к эмулируемой сети.

В рассмотренном примере все пограничные коммутаторы образуют одну эмулируемую сеть. Если же необходимо образовать несколько эмулируемых сетей, не взаимодействующих прямо между собой, то для каждой такой сети необходимо активизировать собственные серверы LES и BUS, а в пограничных

коммутаторах активизировать по одному элементу LEC для каждой эмулируемой сети. Для хранения информации о количестве активизированных эмулируемых сетей, а также ATM-адресах соответствующих серверов LES и BUS вводится еще один сервер – сервер конфигурации LECS (LAN Emulation Configuration Server).

Спецификация LANE существует сегодня в двух версиях. Вторая версия ликвидировала некоторые недостатки первой, связанные с отсутствием механизма резервирования серверов LES и BUS в нескольких коммутаторах, что необходимо для надежной работы крупной сети, а также добавила поддержку разных классов трафика.

На основе технологии LANE работает новая спецификация ATM Forum – Multiprotocol Over ATM, MPOA. Эта спецификация ATM определяет эффективную передачу трафика сетевых протоколов – IP, IPX, DECnet и т. п. через сеть ATM. По назначению она близка к спецификации Classical IP, однако решает гораздо больше задач. Технология MPOA позволяет пограничным коммутаторам 3-го уровня, поддерживающим какой-либо сетевой протокол, но не строящим таблицы маршрутизации, находить кратчайший путь через сеть ATM. MPOA использует для этого серверный подход, аналогичный тому, что применен в LANE. Сервер MPOA регистрирует адреса (например, IP-адреса) сетей, обслуживаемых пограничными коммутаторами 3-го уровня, а затем по запросу предоставляет их клиентам MPOA, встроенным в эти коммутаторы. С помощью технологии MPOA маршрутизаторы или коммутаторы 3-го уровня могут объединять эмулируемые сети, образованные на основе спецификации LANE.

Использование технологии ATM

Технология ATM расширяет свое присутствие в локальных и глобальных сетях не очень быстро, но неуклонно. В последнее время наблюдается устойчивый ежегодный прирост числа сетей, выполненных по этой технологии, в 20-30 %.

В локальных сетях технология ATM применяется обычно на магистралях, где хорошо проявляются такие ее качества, как масштабируемая скорость (выпускаемые сегодня корпоративные коммутаторы ATM поддерживают на своих портах скорости 155 и 622 Мбит/с), качество обслуживания (для этого нужны приложения, которые умеют запрашивать нужный класс обслуживания), петлевидные связи (которые позволяют повысить пропускную способность и обеспечить резервирование каналов связи). Петлевидные связи поддерживаются в силу того, что ATM – это технология с маршрутизацией пакетов, запрашивающих установление соединений, а значит, таблица маршрутизации может эти связи учесть – либо за счет ручного труда администратора, либо за счет протокола маршрутизации PNNI.

Основной соперник технологии ATM в локальных сетях – технология Gigabit Ethernet. Она превосходит ATM в скорости передачи данных – 1000 Мбит/с по сравнению с 622 Мбит/с, а также в затратах на единицу скорости. Там, где коммутаторы ATM используются только как высокоскоростные устройства, а возможности поддержки разных типов трафика игнорируются, технологию ATM, очевидно, заменит технология Gigabit Ethernet. Там же, где качество обслуживания действительно важно (видеоконференции, трансляция телевизионных передач и т, п.), технология ATM останется. Для объединения настольных компьютеров технология ATM, вероятно, еще долго не будет использоваться, так как здесь очень серьезную конкуренцию ей составляет технология Fast Ethernet.

В глобальных сетях ATM применяется там, где сеть frame relay не справляется с большими объемами трафика, и там, где нужно обеспечить низкий уровень задержек, необходимый для передачи информации реального времени.

Сегодня основной потребитель территориальных коммутаторов ATM – это Internet. Коммутаторы ATM используются как гибкая среда коммутации виртуальных каналов между IP-маршрутизаторами, которые передают свой трафик в ячейках ATM. Сети ATM оказались более выгодной средой соединения IP-маршрутизаторов, чем выделенные каналы SDH, так как виртуальный канал ATM может динамически перераспределять свою пропускную способность между пульсирующим трафиком клиентов IP-сетей. Примером магистральной сети ATM крупного поставщика услуг может служить сеть компании UUNET – одного из ведущих поставщиков услуг Internet Северной Америки (рис. 6.35).

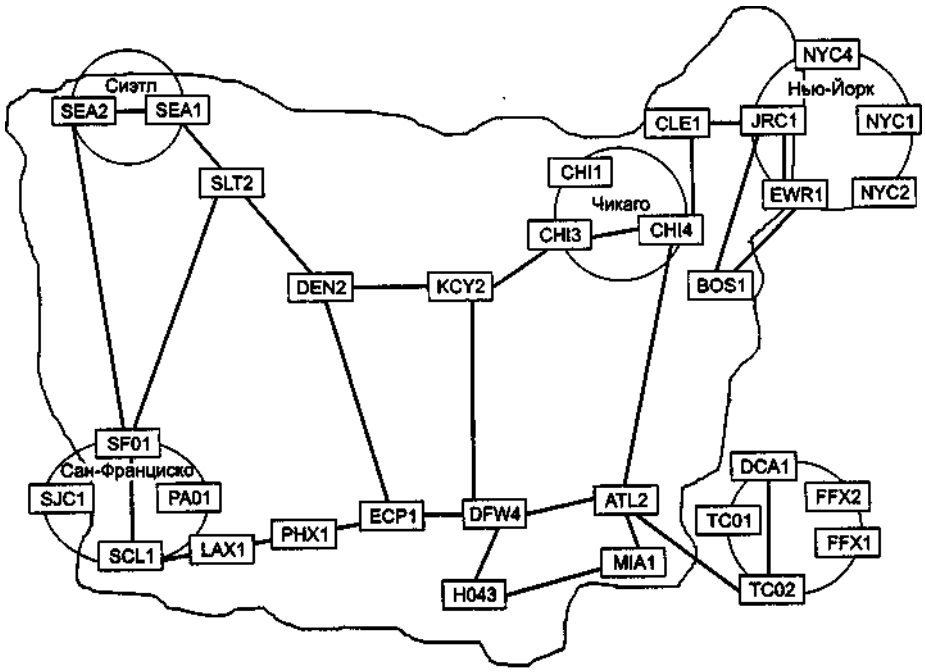


Рис. 6.35. Магистральная сеть ATM компании UUNET

Сегодня по данным исследовательской компании Distributed Networking Associates около 85 % всего трафика, переносимого в мире сетями ATM, составляет трафик компьютерных сетей (наибольшая доля приходится на трафик IP – 32 %).

Хотя технология ATM разрабатывалась для одновременной передачи данных компьютерных и телефонных сетей, передача голоса по каналам CBR для сетей ATM составляет всего 5 % от общего трафика, а передача видеоинформации –10 %. Телефонные компании пока предпочитают передавать свой трафик непосредственно по каналам SDH, не довольствуясь гарантиями качества обслуживания ATM. Кроме того, технология ATM пока имеет недостаточно стандартов для плавного включения в существующие телефонные сети, хотя работы в этом направлении идут.

Что же касается совместимости ATM с технологиями компьютерных сетей, то разработанные в этой области стандарты вполне работоспособны и удовлетворяют пользователей и сетевых интеграторов.

Выводы

- К технологиям глобальных сетей с коммутацией пакетов относятся сети X.25, frame relay, SMDS, ATM и TCP/IP. Все эти сети, кроме сетей TCP/IP, используют маршрутизацию пакетов, основанную на виртуальных каналах между конечными узлами сети.
- Сети TCP/IP занимают особое положение среди технологий глобальных сетей, так как они выполняют роль технологии объединения сетей любых типов, в том числе и сетей всех остальных глобальных технологий. Таким образом, сети TCP/ IP относятся к более высокоуровневым технологиям, чем технологии собственно глобальных сетей.
- Техника виртуальных каналов заключается в разделении операций маршрутизации и коммутации пакетов. Первый пакет таких сетей содержит адрес вызываемого абонента и прокладывает виртуальный путь в сети, настраивая промежуточные коммутаторы. Остальные пакеты проходят по виртуальному каналу в режиме коммутации на основании номера виртуального канала, который является локальным адресом для каждого порта каждого коммутатора.
- Техника виртуальных каналов имеет преимущества и недостатки по сравнению с техникой маршрутизации каждого пакета, характерной для сетей IP или IPX. Преимуществами являются: ускоренная коммутация пакетов по номеру виртуального канала, а также сокращение адресной части пакета, а значит, и избыточности заголовка. К недостаткам следует отнести невозможность распараллеливания потока данных между двумя абонентами по параллельным путям, а также неэффективность установления виртуального пути для кратковременных потоков данных.
- Сети X.25 относятся к одной из наиболее старых и отработанных технологий глобальных

сетей. Трехуровневый стек протоколов сетей X.25 хорошо работает на ненадежных зашумленных каналах связи, исправляя ошибки и управляя потоком данных на канальном и пакетном уровнях.

- Сети X.25 поддерживают групповое подключение к сети простых алфавитно-цифровых терминалов за счет включения в сеть специальных устройств PAD, каждое из которых представляет собой особый вид терминального сервера.
- На надежных волоконно-оптических каналах технология X.25 становится избыточной и неэффективной, так как значительная часть работы ее протоколов ведется «вхолостую».
- Сети frame relay работают на основе весьма упрощенной, по сравнению с сетями X.25, технологией, которая передает кадры только по протоколу канального уровня – протоколу LAP-F. Кадры при передаче через коммутатор не подвергаются преобразованиям, из-за чего технология и получила свое название.
- Важной особенностью технологии frame relay является концепция резервирования пропускной способности при прокладке в сети виртуального канала. Сети frame relay создавались специально для передачи пульсирующего компьютерного трафика, поэтому при резервировании пропускной способности указывается средняя скорость трафика CIR и согласованный объем пульсаций Bc.
- Сеть frame relay гарантирует поддержку заказанных параметров качества обслуживания за счет предварительного расчета возможностей каждого коммутатора, а также отбрасывания кадров, которые нарушают соглашение о трафике, то есть посылаются в сеть слишком интенсивно.
- Большинство первых сетей frame relay поддерживали только службу постоянных виртуальных каналов, а служба коммутируемых виртуальных каналов стала применяться на практике только недавно.
- Технология ATM является дальнейшим развитием идей предварительного резервирования пропускной способности виртуального канала, реализованных в технологии frame relay.
- Технология ATM поддерживает основные типы трафика, существующие у абонентов разного типа: трафик с постоянной битовой скоростью CBR, характерный для телефонных сетей и сетей передачи изображения, трафик с переменной битовой скоростью VBR, характерный для компьютерных сетей, а также для передачи компрессированного голоса и изображения.
- Для каждого типа трафика пользователь может заказать у сети значения нескольких параметров качества обслуживания – максимальной битовой скорости PCR, средней битовой скорости SCR, максимальной пульсации MBS, а также контроля временных соотношений между передатчиком и приемником, важных для трафика, чувствительного к задержкам.
- Технология ATM сама не определяет новые стандарты для физического уровня, а пользуется существующими. Основным стандартом для ATM является физический уровень каналов технологий SONET/SDH и PDH.
- Ввиду того что ATM поддерживает все основные существующие типы трафика, она выбрана в качестве транспортной основы широкополосных цифровых сетей с интеграцией услуг – сетей B-ISDN, которые должны заменить сети ISDN.

6.5. Удаленный доступ

Если магистральные связи между локальными сетями всегда строятся путем соединения локальных сетей с территориальным транспортом через маршрутизаторы, то для организации удаленного доступа могут использоваться различные схемы и продукты. Продукты удаленного доступа могут существенно отличаться реализованными в них функциями, а значит, и возможностями при решении конкретной практической задачи.

4.5.1. Основные схемы глобальных связей при удаленном доступе

Удаленный доступ – очень широкое понятие, которое включает в себя различные типы и варианты взаимодействия компьютеров, сетей и приложений. Если рассматривать все многочисленные схемы взаимодействия, которые обычно относят к, удаленному доступу, то всем им присуще *использование*

глобальных каналов или глобальных сетей при взаимодействии. Кроме того, для удаленного доступа, как правило, характерна *несимметричность взаимодействия*, когда, с одной стороны, имеется центральная крупная сеть или центральный компьютер, а с другой – отдельный удаленный терминал, компьютер или небольшая сеть, которые хотят получить доступ к информационным ресурсам центральной сети. Количество удаленных от центральной сети узлов и сетей, требующих этот доступ, постоянно растет, поэтому современные средства удаленного доступа рассчитаны на поддержку большого количества удаленных клиентов.

Типы взаимодействующих систем

На рис. 6.36 приведены основные схемы удаленного доступа, отличающиеся типом взаимодействующих систем:

- терминал–компьютер – (1);
- компьютер–компьютер – (2);
- компьютер–сеть – (3);
- сеть–сеть – (4).

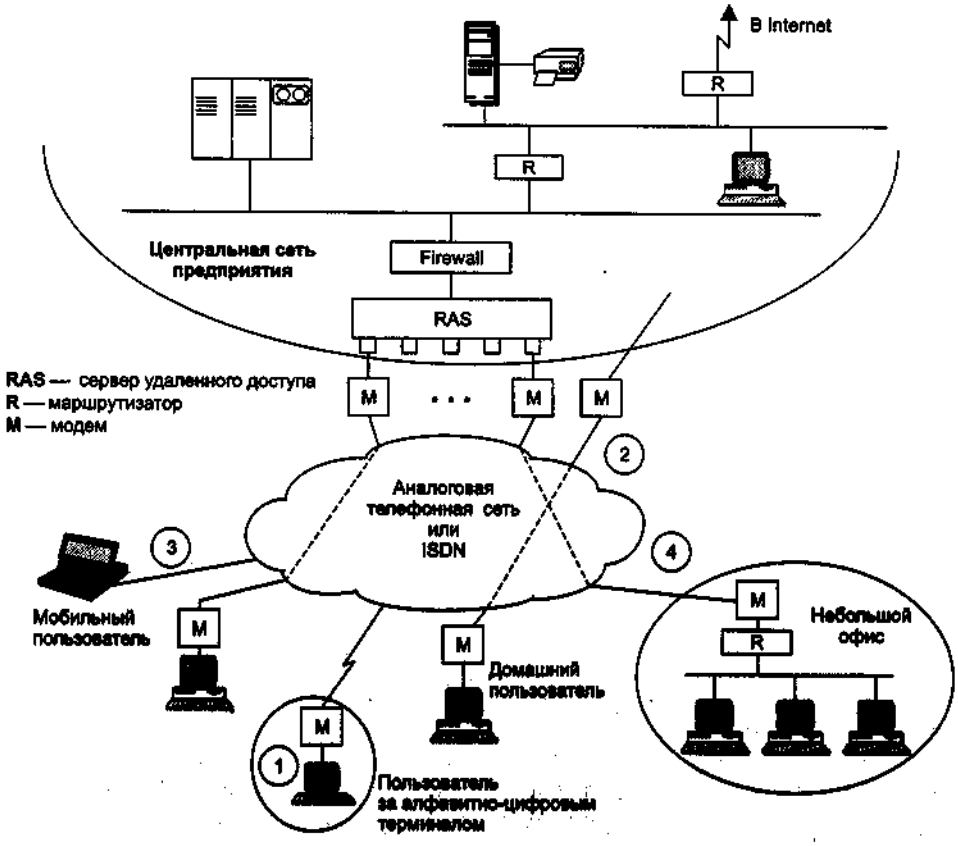


Рис. 6.36. Общая схема удаленного доступа

Первые три вида удаленного доступа часто объединяют понятием индивидуального доступа, а схемы доступа сеть - сеть иногда делят на два класса – ROBO и SOHO. Класс ROBO (Regional Office/Branch Office) соответствует случаю подключения к центральной сети сетей средних размеров – сетей региональных подразделений предприятия, а классу SOHO (Small Office/Home Office) соответствует случай удаленного доступа сетей небольших офисов и домашних сетей.

Типы поддерживаемых служб

Схемы удаленного доступа могут отличаться также и типом служб, которые поддерживаются для удаленного клиента. Наиболее часто используется удаленный доступ к файлам, базам данных, принтерам в том же стиле, к которому пользователь привык при работе в локальной сети. Такой режим называется *режимом удаленного узла* (*remote node*). Иногда при удаленном доступе реализуется обмен с центральной сетью сообщениями *электронной почты*, с помощью которого можно в автоматическом

режиме получить запрашиваемые корпоративные данные, например из базы данных.

Особое место среди всех видов удаленного доступа к компьютеру занимает способ, при котором пользователь получает возможность удаленно работать с компьютером таким же способом, как если бы он управлял им с помощью локально подключенного терминала. В этом режиме он может запускать на выполнение программы на удаленном компьютере и видеть результаты из выполнения. При этом принято подразделять такой способ доступа на *терминальный доступ и удаленное управление*. Если у удаленного пользователя в распоряжении имеется только неинтеллектуальный алфавитно-цифровой терминал (вариант 1 на рис. 6.36) или же он запускает на своем персональном компьютере программу эмуляции такого терминала (например, Term90 из утилит Nortom Commander или же программу Terminal из утилит Windows 3.1), то такой режим работы называют терминальным доступом. Для владельца алфавитно-цифрового терминала, например VT-100, этот вид удаленного доступа является единственно возможным. Доступ к мэйнфрейму IBM, работающему под управлением операционной системы MVS, с помощью доступа через удаленный или встроенный PAD, который затем работает с мэйнфреймом через сеть X.25, также является примером терминального доступа. Отличительной особенностью терминального доступа является то, что операционные системы на компьютере, к которому получают доступ пользователи, рассчитаны на многотерминальный режим работы, поэтому главное здесь – отличная от стандартного варианта схема подключения терминала, ориентированная на глобальные сети.

При удаленном управлении пользователь запускает на своем компьютере программу, которая эмулирует ему на экране сеанс работы с операционной системой – DOS, Windows, OS/2, – которая не поддерживает многотерминальный режим работы. Программа эмуляции экрана через глобальные каналы взаимодействует с дополнительным программным обеспечением, работающим под управлением соответствующей операционной системы на удаленном компьютере. Пользователь, как и при терминальном доступе, также получает полное управление удаленным компьютером, при этом он видит на экране графический интерфейс привычной ему операционной системы, в качестве которой чаще всего выступает Windows. Результат получается практически тот же, но за счет нестандартного дополнительного программного обеспечения на удаленном компьютере.

Типы используемых глобальных служб

Схема организации удаленного доступа во многом определяется теми глобальными транспортными службами, которые доступны в точках нахождения многочисленных клиентов удаленного доступа. Кроме степени распространенности необходимо учитывать и стоимость глобальной службы. С учетом этих двух обстоятельств наиболее часто для организации удаленного доступа используется служба телефонных сетей – аналоговых (Plain Old Telephone Service – POTS) и, если это возможно, ISDN. Только эти сети пока могут обеспечить дешевый доступ практически из любого географического пункта. Правда, для нашей страны это справедливо только для аналоговых телефонных сетей, службы же ISDN доступны только в крупных городах и то фрагментарно. В то же время для большинства стран Западной Европы, Японии, Южной Кореи, а также США и Канады получение услуг ISDN для небольшого офиса или домашнего пользователя – это реальность сегодняшнего дня, и этим объясняется большое количество продуктов для организации удаленного доступа, ориентированных на службу ISDN. Хотя количество установленных абонентских окончаний ISDN даже в развитых странах пока в процентном отношении и невелико по отношению к общему числу абонентов телефонной сети, но главную роль здесь играет то, что при заказе такого окончания абонент получает его в течение нескольких недель.

Служба выделенных каналов экономически оправдана только при подключении небольшого числа крупных подразделений предприятия, а для отдельных пользователей ее использование – слишком большая роскошь.

Служба сетей с коммутацией пакетов, таких как X.25 или frame relay, из-за своей стоимости также малопригодна для индивидуальных пользователей. Кроме того, точки доступа к этим сетям далеко не так распространены, как точки доступа к телефонной сети, имеющиеся почти в каждой квартире, не говоря уже о небольших офисах. Прямые подключения к сетям X.25 или frame relay целесообразны для организации равноправных связей сетей или же для подключения сетей класса ROBO, так как такого рода сетей у предприятия обычно немного, а связь с центральной сетью им нужна постоянно. Для индивидуальных пользователей проблема подключения к сети X.25 решается доступом по телефонной

сети через устройство PAD, оснащенное модемным пулом, если такой доступ оправдан экономически.

Экономические аспекты удаленного доступа должны учитывать способ его оплаты и интенсивность использования, которое обычно оценивается количеством часов загруженности глобальных каналов в месяц. Необходимо иметь в виду, что практически все транспортные службы удаленного доступа, связанные с коммутируемыми каналами, оплачиваются повременно, а в транспортных службах постоянных каналов схема оплаты помесячная, не зависящая от загрузки канала. Например, доступ через аналоговую телефонную сеть или ISDN оплачивается повременно, а доступ через выделенный канал 64 Кбит/с или постоянный канал 64 Кбит/с в сети frame relay оплачивается фиксированной месячной суммой.

Поэтому обычно сначала определяется, какое количество часов в месяц будет тот или иной удаленный пользователь работать с центральной локальной сетью удаленно. Затем на основании тарифов оплаты телекоммуникационных услуг находится тот вид услуги, который более экономичен для данного количества часов месячной работы. Обычно при количестве часов до 20-40 более; выгодными являются аналоговые телефонные сети и ISDN. Для пользователей; которым требуется большее чем 40 количество часов доступа в месяц, например 60-80, может оказаться более выходным воспользоваться выделенным каналом frame relay. Необходимо отметить, что коммутируемые услуги в сетях, основанных на технике виртуальных каналов, обычно оплачиваются также по временной схеме. Так оплачиваются услуги коммутируемых виртуальных каналов ATM и только появляющаяся аналогичная служба сетей frame relay.

6.5.2. Доступ компьютер - сеть

В связи с широким использованием на предприятиях локальных сетей наиболее часто встречающийся вид удаленного доступа – это доступ не к отдельному компьютеру, а к сети в целом. Для этой цели в центральной сети предприятия устанавливается специальная система – сервер удаленного доступа (Remote Access Server, RAS), который выполняет большой спектр функций по обслуживанию многочисленных удаленных клиентов. Задачи сервера удаленного доступа, который часто называют также коммуникационным сервером, зависят от схемы удаленного доступа.

Очевидно, что для экономии модемов можно не ставить на каждый компьютер центральной сети отдельный модем, а организовать общий *пул модемов* и сделать его разделяемым ресурсом как для звонков из локальной сети, так и для звонков извне. Действительно, если каждому пользователю выделить персональный модем (и персональную линию связи), то, как правило, большую часть времени он будет простаивать, поэтому гораздо эффективнее использовать то число модемов (и линий), которое реально необходимо.

Разделяемый для пользователей локальный пул модемов создается с помощью так называемого *коммуникационного сервера* (*Communication Server*). Коммуникационный сервер – это обычный компьютер или специализированное устройство, предоставляющее пользователям локальной сети прозрачный доступ к последовательным портам ввода/вывода, к которым подключены разделяемые модемы. Пользователь, подключившийся по локальной сети к коммуникационному серверу, может работать с одним из подключенных к нему модемов точно так же, как если бы этот модем был подключен непосредственно к компьютеру пользователя. Таким образом, коммуникационный сервер обслуживает пользователей локальной сети, делая локальные модемы разделяемыми ресурсами. Говорят, что коммуникационный сервер поддерживает режим dial-out – режим, который позволяет пользователям локальной сети устанавливать по своей инициативе связь через телефонную сеть с каким-либо удаленным компьютером.

Сервер удаленного доступа (*Remote Access Server, RAS*) обслуживает не локальных, а удаленных пользователей, предоставляя им доступ к ресурсам локальной сети – файлам, принтерам и т. п. – извне. Сервер удаленного доступа поддерживает режим dial-in – режим, который позволяет пользователю, работающему на удаленном компьютере, устанавливать связь с локальной сетью *по его инициативе*. Именно это является основной задачей систем удаленного доступа. С этой точки зрения удаленный доступ можно определить как эффективный способ разделения ресурсов централизованных серверов между удаленными клиентами.

Часто коммуникационный сервер и сервер удаленного доступа являются одним и тем же продуктом, выполненным либо в качестве дополнительного программного обеспечения в среде какой-либо популярной ОС, либо в качестве отдельного устройства. За таким комбинированным продуктом обычно

закрепляется название сервера удаленного доступа. Примерами программных серверов удаленного доступа являются сервер Microsoft RAS, работающий в составе ОС Windows NT, и сервер NetWare Connect, работающий в среде ОС NetWare.

Однако если режим dial-in поддерживают все серверы удаленного доступа по определению, то режим dial-out является факультативным и реализуется не всегда.

Режимы dial-in и dial-out только говорят о том, кто является инициатором установления соединения - удаленный пользователь или пользователь локальной сети.

В зависимости от потребностей пользователей и возможностей программно-аппаратного обеспечения удаленный доступ может осуществляться в соответствии с различными схемами: удаленный узел, удаленное управление и взаимодействие с помощью электронной почты.

Удаленный узел

Одним из вариантов удаленного доступа типа компьютер - сеть является режим *удаленного узла* (*remote node*). Программное обеспечение удаленного узла на клиентской машине позволяет последовательному порту и модему (или терминальному адаптеру ISDN) стать медленным узлом удаленной локальной сети, взаимодействующим обычным способом с сетевыми операционными системами при разделении их ресурсов. В локальной сети должен быть установлен сервер удаленного доступа, поддерживающий режим удаленного узла. Это означает, что сервер должен поддерживать один из протоколов канального уровня, используемых на глобальном канале. Протокол канального уровня необходим для связи удаленного компьютера с центральной локальной сетью. Так как чаще всего этот канал является коммутируемым каналом телефонной сети или ISDN, то сервер удаленного доступа должен поддерживать протоколы PPP и SLIP, используемые на этих каналах. В сети X.25 или frame relay сервер удаленного доступа должен поддерживать протоколы этих сетей, то есть протоколы LAP-B и X.25/3 для первого случая и LAP-F для второго (если сеть frame relay поддерживает только постоянные виртуальные каналы). При получении по глобальному каналу кадров соответствующего протокола, сервер, работающий в режиме удаленного узла, извлекает из кадра, например, PPP, пакеты тех общих протоколов сетевого уровня, по которым работают удаленный компьютер и компьютеры локальной сети. Такими протоколами могут быть протоколы IP, IPX или немаршрутизируемый протокол NetBEUI. Далее вступают в работу протоколы верхних уровней, и пользователь получает такой же доступ, как если бы его компьютер находился непосредственно в локальной сети, но с небольшим исключением – скорость обмена его компьютера с остальными компьютерами удаленной сети зависит от пропускной способности глобального канала связи.

Клиенты, работающие в режиме удаленного узла, могут логически войти в сеть таким же образом, как если бы они были локальными пользователями, отображать сетевые диски и даже загружать программы через удаленную связь. Но удаленная загрузка больших программ неразумна, так как самый скоростной модем 33,6 Кбит/с работает со скоростью, составляющей только 3 % от скорости сегмента Ethernet, и программа, которая в локальной сети загружается за 30 с, будет загружаться по удаленной связи в течение 15-20 минут. Поэтому в режиме удаленного узла локальные копии программ, как правило, эффективнее.

Другая проблема связана со способом работы сетевых операционных систем. Серверы часто рассылают широковещательные сообщения всем узлам сети для проверки подключенных и работающих клиентов. Такие широковещательные рассылки могут заблокировать удаленный доступ, если они не фильтруются перед отправкой по удаленным связям. Поэтому перед приобретением любого продукта необходимо проверить по его описаниям, может ли он работать в режиме удаленного доступа.

Компьютер, использующий режим удаленного узла, наиболее эффективно работает с системами клиент-сервер, так как в этом случае трафик по глобальному каналу обычно передается не очень интенсивный – запрос клиента обрабатывается на сервере, а по глобальному каналу передается только ответ. В режиме клиент-сервер работают многие корпоративные СУБД (например, Oracle, Informix, Microsoft SQL Server), а также приложения, ориентированные на эту архитектуру. Многие административные утилиты современных операционных систем поддерживают такой режим, например User Manager for Domains ОС Windows NT.

Серверы, работающие в режиме удаленного узла, выполняют свои функции различным образом.

Первый вариант – это реализация в сервере удаленного узла функционального эквивалента маршрутизатора с WAN-портами для асинхронных модемов, ISDN-линий или асинхронного доступа к

PAD X.25. Этот вариант универсален, так как обеспечивает доступ как отдельных компьютеров, так и локальных сетей. Однако данный вариант при подключении отдельного компьютера избыточен, поскольку требует выделения отдельного номера сети каждому подключившемуся к сети пользователю.

Второй вариант основан на работе сервера удаленного узла в режиме шлюза. Если удаленные клиенты и локальная сеть работают на протоколе IP, то всем удаленным компьютерам присваивается один и тот же номер IP-сети, совпадающий с номером локальной сети, к которой они получают доступ. В этом случае сервер выполняет функции посредника по протоколу ARP (говорят, что он поддерживает режим proxy ARP), отвечая компьютерам локальной сети своим MAC-адресом на запросы о IP-адресах, принадлежащих удаленным подключившимся узлам. Для протокола NetBIOS работа сервера в режиме шлюза – это единственно возможный режим работы, так как этот протокол не может маршрутизироваться.

В сервере удаленного узла могут быть реализованы оба варианта работы, которые выбираются в зависимости от типа клиента (компьютер или сеть), а также протокола.

Операционные системы Mac OS, OS/2, Windows 95 и Windows NT Workstation включают в стандартную поставку клиентскую часть программного обеспечения удаленного узла. В настоящее время имеется явная тенденция использования клиентами удаленного узла протокола PPP. В результате достигается совместимость клиентских и серверных частей систем различных производителей, работающих в режиме удаленного узла.

Удаленное управление и терминальный доступ

Другим распространенным вариантом удаленного доступа являются две разновидности практически одного и того же режима - *удаленное управление* (*remote control*) и *терминальный доступ* (*terminal access*). При этом способе удаленный компьютер становится, в сущности, виртуальным терминалом компьютера-хоста, который может быть, а может и не быть подключен к сети. Этот вариант позволяет запустить любое приложение на компьютере-хосте, а также получить доступ к любым данным этого хоста. Если компьютер-хост подключен к сети, то и удаленные его пользователи становятся полноправными членами сети, действуя как пользователи компьютера-хоста.

Выше уже было сказано, что отличия удаленного управления от терминального доступа только в том, что при удаленном управлении пользователь связывается с операционной системой, не рассчитанной на поддержку многотерминального режима (MS-DOS, Windows 3.1, Windows 95/98, Windows NT, OS/2 Warp), а терминальный доступ осуществляется к операционным системам, для которых многотерминальный режим является основным (Unix, IBM VM, IMB OS-400, VAX VMS).

Удаленное управление или терминальный доступ нужны тогда, когда удаленный пользователь работает с приложениями, не оптимизированными для работы в сети, например с традиционными СУБД персональных компьютеров типа dBase, Paradox или Access. Иначе, когда такое приложение находится на одном компьютере, а файлы баз данных – на другом, в сети создается чрезмерно интенсивный трафик.

Централизованная схема удаленного управления требует установки в локальной сети предприятия специального программного продукта – сервера удаленного управления, например сервера WinFrame компании Citrix. На клиентских удаленных компьютерах также нужно установить дополнительное программное обеспечение – клиента удаленного управления.

Протоколы, используемые программами удаленного управления для передачи информации об обновлении экрана, нажатиях клавиш и перемещениях мыши, являются нестандартными – поэтому нужно устанавливать серверную и клиентские части удаленного управления от одного производителя. Например, пользователи программного клиента удаленного доступа Norton pcAnywhere не смогут дозвониться до хоста, работающего под управлением программ ReachOut, LapLink for Windows, Carbon Copy, Remotely Possible или Close-Up.

При терминальном доступе также желательно установить в центральной сети специальный продукт – терминальный сервер. Можно обойтись и без него, но тогда на каждый компьютер, к которому нужно подключиться в режиме удаленного терминала, нужно ставить модем и выделять ему отдельный телефонный номер. Терминальный сервер принимает запросы на связь с определенным компьютером и передает по локальной сети коды нажатия клавиш и символы, подлежащие отображению на экране пользовательского терминала. Для взаимодействия по локальной сети с многотерминальными ОС терминальный сервер использует стандартные протоколы эмуляции терминала, например telnet для

Unix, DEC LAT для VAX VMS.

Почта

Почта является еще одним видом удаленного доступа. Почтовые шлюзы, доступные по коммутируемым телефонным линиям, и клиентское почтовое обеспечение удаленного доступа могут быть достаточными для удовлетворения потребностей многих обычных пользователей. Такие почтовые шлюзы позволяют удаленным пользователям или даже удаленным офисам звонить в почтовую систему центрального отделения, обмениваться входящими и исходящими сообщениями и файлами, а затем отключаться.

Продукты, предназначенные для этих целей, варьируются от клиентских программ для одного пользователя, таких как cc:mail Mobile фирмы Lotus, до полномасштабных шлюзов, которые организуют почтовый обмен между удаленными серверами и корпоративной локальной сетью (например, Exchange компании Microsoft).

Почтовые шлюзы могут быть полезны в случае, когда количество данных, которыми обмениваются удаленные пользователи с центральным офисом, не очень большое. Из-за того, что среднее время сессии пользователь - шлюз сравнительно невелико, шлюз центральной сети не должен поддерживать большое количество телефонных линий. Обычно почтовое соединение легко устанавливается, а стоимость программного обеспечения шлюза незначительна.

Шлюзы работают в автоматическом режиме без вмешательства человека. Если в удаленном офисе работают один или два сотрудника и им не нужен доступ к корпоративным данным в реальном масштабе времени, то почтовый шлюз может быть хорошим решением. Некоторые приложения автоматически принимают запросы в виде писем электронной почты, а затем посылают в таком же виде ответы. Так, например, работают многие СУБД.

Не только почта, но и другие приложения, написанные для локальной вычислительной сети, могут иметь специфические программные модули, предназначенные для удаленных соединений. Такие программы устанавливают соединения между собой с помощью нестандартных протоколов и часто увеличивают эффективность соединения за счет специальных приемов, например путем передачи только обновлений между удаленным компьютером и хостом. Примером продуктов этого класса являются программные системы коллективной работы.

6.5.3. Удаленный доступ через промежуточную сеть

Общая схема двухступенчатого доступа

Раньше удаленный международный или междугородный доступ отдельных пользователей всегда реализовывался по схеме, основанной на использовании международной или междугородной телефонной связи. Публичные территориальные сети с коммутацией пакетов (в основном – сети X.25) не были так распространены, чтобы, находясь в любом городе, посланный в командировку сотрудник мог получить доступ к этой сети, а через нее – к маршрутизатору или серверу удаленного доступа своего предприятия.

Поэтому удаленные пользователи непосредственно звонили по телефонной сети на сервер удаленного доступа своего предприятия, не считаясь с затратами на международные или междугородные переговоры.

Однако сегодня очень часто служба международной сети с коммутацией пакетов имеется во многих городах, и чаще всего это служба Internet. По мере развития услуг сетей frame relay возможно, что и эта технология получит такое же массовое распространение. Поэтому стала возможной двухступенчатая связь удаленного пользователя со своей корпоративной сетью – сначала выполняется доступ по городской телефонной сети к местному поставщику услуг Internet, а затем через Internet пользователь соединяется со своей корпоративной сетью.

Такой вариант может значительно удешевить доступ по сравнению с непосредственным подключением через междугородные АТС.

Обычно экономия происходит за счет перехода от междугородных (или международных) звонков к местным Если поставщик услуг сети с коммутацией пакетов поддерживает доступ по коммутируемым телефонным сетям, то непосредственный доступ к серверу, установленному в центральной сети,

находящейся в другом городе, заменяется звонком на сервер удаленного доступа местного поставщика услуг (рис. 6.37).

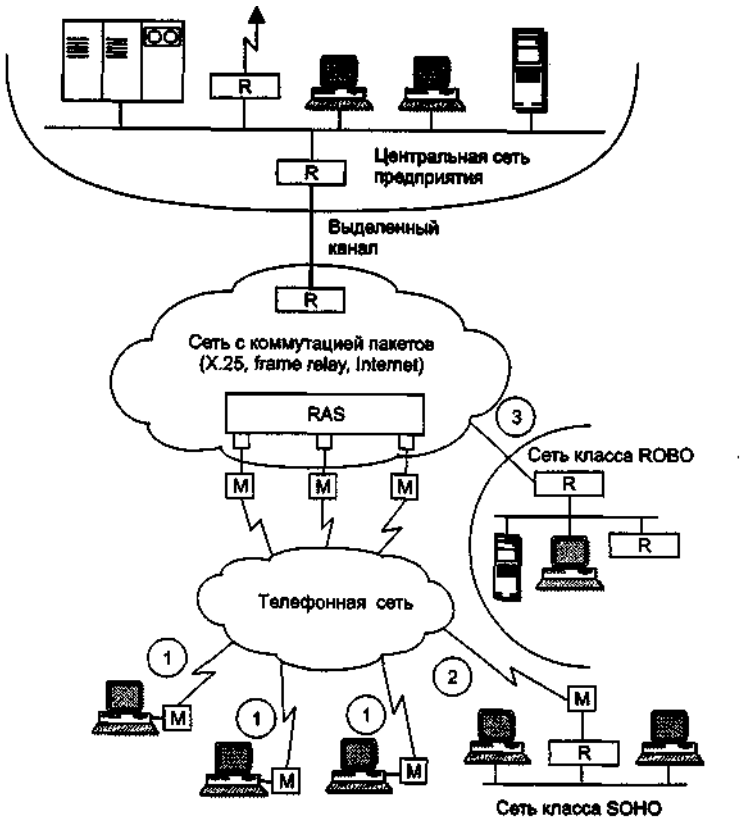


Рис. 6.37. Подключение удаленных пользователей через промежуточную публичную сеть с коммутацией пакетов

Центральная сеть предприятия, используя выделенный канал, обычно непосредственно подключается к той же сети с коммутацией пакетов, что и удаленные пользователи в других городах.

Стандартизация клиентов удаленного доступа на основе протоколов PPP и SLIP упрощает проблемы обслуживания разнородных пользователей одним поставщиком услуг при использовании Internet в качестве промежуточной сети. Для сетей X.25 протоколы взаимодействия сети офиса с сетью поставщика услуг также вполне определены, хотя иногда наблюдаются случаи различной настройки одного и того же протокола в оборудовании и программном обеспечении клиента и поставщика услуг.

Выгода от Internet в качестве промежуточного транспорта оказывается особенно ощутимой, так как расценки поставщиков услуг Internet намного ниже, чем расценки поставщиков услуг сетей X.25. Это обстоятельство является не последней причиной бурного распространения технологии intranet, использующей транспортные и информационные службы Internet для внутрикорпоративных нужд.

Ввиду большой популярности Internet в качестве инструмента для доступа к корпоративной сети для этой двухступенчатой схемы разработано много протоколов и средств, которые создают виртуальный туннель между пользователем и точкой входа в корпоративную сеть – маршрутизатором или сервером удаленного доступа. Этот туннель решает две задачи. Во-первых, передачу через IP-сеть, которой является Internet, чужеродного для нее трафика – протоколов IPX, NetBEUI, непосредственно Ethernet и т. п. Во-вторых, туннель создает защищенный канал, данные в котором шифруются.

Промежуточная телефонная сеть делает доступ через Internet к корпоративной сети весьма медленным. В последнее время появилось несколько решений, позволяющих пользователю получить весьма быстрый доступ к Internet через существующие инфраструктуры абонентских окончаний телефонных сетей и сетей кабельного телевидения.

Технологии ускоренного доступа к Internet через абонентские окончания телефонных и кабельных сетей

Сегодня многие телекоммуникационные компании разных стран мира начали активно внедрять различные варианты цифровых абонентских линий (DSL). В последнее время наибольшее внимание специалистов привлекла технология асимметричной цифровой абонентской линии (Asymmetric Digital

Subscriber Line, ADSL), но помимо нее пользователям предложены также службы симметричной цифровой абонентской линии (SDSL), цифровой абонентской линии с переменной скоростью (Rate Adaptive DSL, RADSL) и сверхбыстрой цифровой абонентской линии (Very high-speed DSL, VDSL).

Цифровые абонентские окончания появились достаточно давно – впервые их ввели первичные сети каналов T1/E1. Цифровое абонентское окончание High-speed DSL (HDSL) работает по 4-проводной линии со скоростью до 1,544 или 2,048 Мбит/с. Цифровое абонентское окончание сети ISDN работает по 2-проводному окончанию со скоростью 128 Кбит/с.

Однако сегодня пользователям хотелось бы получить доступ к Internet (и через Internet к своим корпоративным сетям) с помощью стандартного 2-проводного телефонного окончания, установив при этом на своем домашнем компьютере какое-нибудь устройство типа модема. Перечисленные выше технологии позволяют это сделать с помощью специальных модемов.

Эти технологии рассчитаны на высокоскоростную передачу данных на коротком отрезке витой пары, соединяющей абонента с ближайшей телефонной АТС, то есть на решение проблемы «последней мили», отделяющей потребителя от поставщика услуг. В то время как обычные модемы (V.34, V.34+) рассчитаны на работу с полосой пропускания в 3100 Гц через сеть с произвольным количеством коммутаторов, модемы* DSL могут получить в свое распоряжение полосу порядка 1 МГц – эта величина зависит от длины кабеля до АТС и сечения используемых проводов. Отличия условий работы модемов* DSL от обычных модемов показаны на рис. 6.38 на примере ADSL-модемов.

ADSL-модемы, подключаемые к обоим концам короткой линии между абонентом и АТС, образуют три канала: быстрый канал передачи данных из сети в компьютер, менее быстрый дуплексный канал передачи данных из компьютера в сеть и простой канал телефонной связи, по которому передаются обычные телефонные разговоры. Передача данных в канале «сеть-абонент» происходит со скоростью от 1,5 до 6 Мбит/с, в канале «абонент-сеть» – со скоростью от 16 Кбит/с до 1 Мбит/с. В обоих случаях конкретная величина скорости передачи зависит от длины и качества линии. Асимметричный характер скорости передачи данных вводится специально, так как удаленный пользователь Internet или корпоративной сети обычно загружает данные из сети в свой компьютер, а в обратном направлении идут либо квитанции, либо поток данных существенно меньшей скорости. Для получения асимметрии скорости полоса пропускания абонентского окончания делится между каналами также асимметрично.

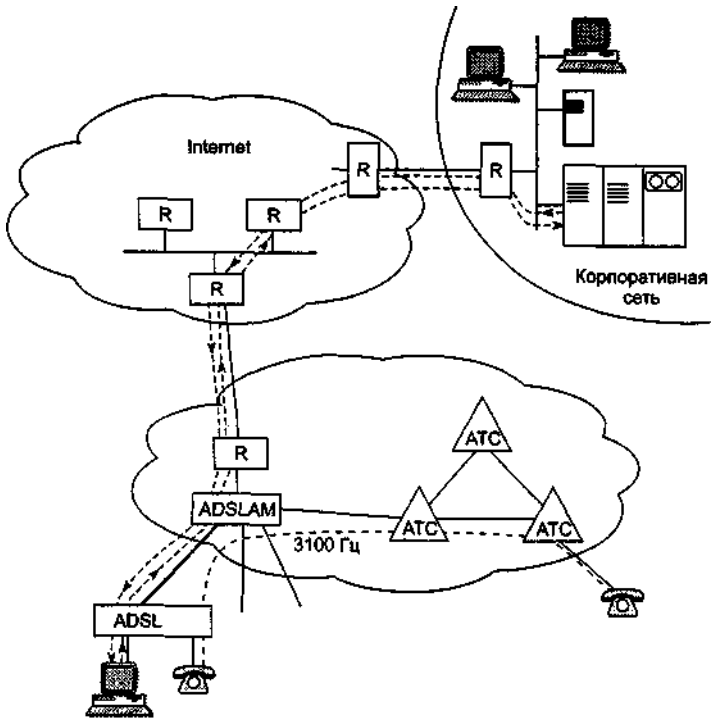


Рис. 6.38. Отличия условий работы ADSL-модемов от обычных модемов

На дальнем конце абонентского окончания должен располагаться так называемый мультиплексор доступа ADSL – ADSLAM. Этот мультиплексор выделяет подканалы из общего канала и отправляет голосовой подканал в 3100 Гц на АТС, а высокоскоростные каналы данных направляет на маршрутизатор, который должен находиться рядом с ADSLAM.

Одно из главных преимуществ технологии ADSL по сравнению с аналоговыми модемами и протоколами ISDN и HDSL – то, что поддержка голоса никак не отражается на параллельной передаче данных по двум быстрым каналам. Причина подобного эффекта состоит в том, что ADSL основана на принципах разделения частот, благодаря чему голосовой канал надежно отделяется от двух других каналов передачи данных. Такой метод передачи гарантирует надежную работу канала POTS даже при нарушении питания ADSL-модема. Никакие конкурирующие системы передачи данных не обеспечивают работу обычного телефонного канала столь же надежно. Хотя технологии ISDN и HDSL поддерживают режим обычной телефонной связи, для ее установления они требуют организации специального канала с пропускной способностью 64 Кбит/с.

Маршрутизатор, расположенный в здании АТС, должен соединяться выделенным высокоскоростным каналом с другим маршрутизатором Internet (или другой сети с коммутацией пакетов). Если центральная сеть предприятия подключена к Internet через выделенный высокоскоростной канал, то все удаленные пользователи, у которых установлены модемы ADSL, получают высокоскоростной доступ к сети своего предприятия на тех же телефонных каналах, которые всегда соединяли их с городской АТС.

Широкое распространение технологий *DSL должно сопровождаться некоторой перестройкой работы поставщиков услуг Internet и поставщиков услуг телефонных сетей, так как их оборудование должно теперь работать совместно.

Стандарт на ADSL-модемы уже принят. Правда, он узаконил только один из реализованных в этой технологии видов кодирования – DMT, в то время как более дешевое CAP-кодирование, используемое некоторыми разработчиками этой технологии, пока не является стандартным. Модемы *DSL являются частным случаем модемов, работающих на коротких ненагруженных линиях. Еще до появления технологий ADSL и ей подобных, модемы short range или short haul применялись для связи не очень удаленных между собой сетей и компьютеров. Этот класс модемов включал как очень простые устройства, так называемые драйверы линий, которые не модулировали сигнал, а просто являлись усилителями, так и сложные модемы, способные работать со скоростью до 2,048 Мбит/с (например, модемы компании RAD Data Communications).

Кроме абонентских окончаний телефонных сетей в последнее время для скоростного доступа к Internet стали применять абонентские окончания кабельного телевидения. Для этих целей уже разработан специальный вид модемов – кабельные модемы. В кабельных модемах используется имеющийся коаксиальный 75-омный телевизионный кабель для передачи данных из сети в компьютер со скоростью до 30 Мбит/с, а из компьютера в сеть – со скоростью до 10 Мбит/с. При этом качество передаваемых сигналов очень высокое.

Высокоскоростные абонентские окончания создают для поставщиков услуг Internet дополнительную проблему – им необходимо иметь очень скоростные каналы доступа к остальной части Internet, так как 10 абонентов с трафиком по 8 Мбит/с создают общий трафик в 80 Мбит/с, который качественно можно передать только с помощью технологий SONET/SDH или ATM. Ведущие поставщики услуг Internet, например UUCP, такие каналы уже имеют.

Выводы

- Удаленный доступ характеризуется использованием глобальных транспортных служб, несимметричностью взаимодействия и большим количеством удаленных пользователей.
- При удаленном доступе в основном используются аналоговые телефонные сети и ISDN – ввиду их распространенности и невысокого уровня оплаты при соединениях небольшой длительности.
- Удаленные пользователи подключаются к специальному устройству центральной сети - серверу удаленного доступа (RAS), которое работает в режиме маршрутизатора или шлюза в зависимости от протоколов, используемых удаленным пользователем.
- Наиболее универсальным режимом удаленного доступа является режим удаленного узла, при котором компьютер пользователя является узлом локальной сети предприятия со всеми его возможностями, но только подключенным к сети через низкоскоростной канал по протоколу PPP.
- Связь с центральной локальной сетью по инициативе удаленного пользователя называется режимом dial-in (основной режим), а по инициативе пользователя центральной сети – dial-out.
- Режимы терминального доступа и удаленного управления позволяют удаленному

пользователю подключиться к компьютеру центральной сети в режиме, имитирующем работу локального терминала. Этот режим очень экономно расходует полосу пропускания глобального канала и рекомендуется для тех случаев, когда необходим низкоскоростной канал – 4800 или 9600 бит/с.

- Для удаленного доступа может использоваться режим электронной почты, который автоматически поддерживается многими приложениями, в том числе СУБД, для получения запросов и отправки ответов.
- Для экономичного удаленного доступа в последнее время часто используется двухступенчатая схема доступа, в которой на первом этапе удаленный пользователь подключается через местную телефонную сеть к местному поставщику услуг Internet, а через Internet выполняется второй этап подключения – к центральной сети, расположенной в другом городе или другой стране.
- Для скоростного доступа к Internet через инфраструктуру абонентских окончаний телефонных аналоговых сетей или сетей кабельного телевидения разработаны новые технологии цифрового абонентского окончания – технологии *DSL, из которых наибольший интерес представляет технология асимметричного доступа ADSL.

Вопросы и упражнения

1. Чем отличаются модемы от устройств DSU/CSU?
2. Предприятие решило создать собственную глобальную сеть. Какой тип глобальных связей будет наиболее эффективен, если предприятию необходимо соединить локальную сеть в штаб-квартире с тремя локальными сетями региональных подразделений, расположенных в разных городах? Средняя интенсивность трафика между сетями подразделений и центральной сетью оценивается диапазоном значений от 500 Кбит/с до 1 Мбит/с.
3. Вы убедились, что модем устойчиво работает на выделенном 2-проводном канале как в асинхронном, так и в синхронном режимах. Какой режим вы предпочтете?
4. К устройству какого уровня в терминах модели OSI можно отнести современный модем?
5. Можно ли использовать обычное абонентское окончание телефонной аналоговой сети, имеющееся в офисе, для подключения к каналу E1?
6. Каким видом услуг цифровых сетей можно воспользоваться, если необходимо соединить две локальные сети, находящиеся в разных городах, причем интенсивность межсетевого трафика составляет от 100 до 180 Кбит/с?
7. Сколько каналов T1 можно передать в одном канале STS-1?
8. Может ли сеть X.25 работать без устройств PAD?
9. Какие устройства необходимо применить для подключения мэйнфрейма, имеющего только интерфейсы RS-232C, к локальной сети Ethernet, если известно, что сетевые адаптеры Ethernet для этого мэйнфрейма не выпускаются?
10. Каким образом пользователь может подключиться к встроенному устройству PAD через телефонную сеть, если он работает за терминалом, который не поддерживает процедуры вызова абонента через телефонную сеть автоматически?
11. Какую услугу ISDN целесообразно использовать, если к этой сети подключены с помощью терминальных адаптеров два персональных компьютера и им нужно постоянно обмениваться данными со скоростью 2400 бит/с с пульсациями до 9600 бит/с, причем величины задержек пакетов не являются критичными?
12. Какую услугу ISDN целесообразно использовать, если к этой сети подключены с помощью маршрутизаторов две локальные сети, причем межсетевой трафик имеет интенсивность от 100 до 512 Кбит/с в течение длительного периода времени?
13. Сравните количество кадров, которое порождает обмен двумя сообщениями TCP (посылка данных и получение квитанции) между двумя конечными хостами, соединенными одним промежуточным коммутатором для случаев, когда этот коммутатор является коммутатором X.25 и когда этот коммутатор является коммутатором frame relay?
14. В каком случае процент дошедших кадров через сеть frame relay до конечного узла будет выше: когда услуга заказана на основании параметров CIR, Bc и Be или когда услуга заказана на основании только параметров CIR и Bc (подразумевается, что значения параметров CIR и Bc в

обоих случаях совпадают). Сеть frame relay недогружена, а узел-источник отправляет данные со скоростью, часто значительно превышающей CIR?

15. Если у вашего предприятия появилась необходимость соединить многочисленные сети филиалов с центральной сетью и между собой, но в распоряжении имеются только выделенные аналоговые каналы с установленными синхронными модемами 19,2 Кбит/с, то какую технологию из следующих вы выберете: X.25, frame relay или ATM? Обоснуйте факторы, которые повлияют на ваше решение.
16. Для какой из категории услуг сеть ATM явно управляет потоком данных? Почему для других категорий услуг управление потоком данных не используется?
17. Вы хотите вручную настроить постоянный виртуальный канал в двух корпоративных сетях ATM, соединенных публичной сетью ATM. Вы не хотите, чтобы ваши номера VCI зависели от номеров виртуальных каналов, используемых администратором в публичной сети ATM. Какой вид коммутации вы закажете у поставщика услуг публичной сети ATM?
18. Вы купили модем V.90 и связываетесь по телефонной сети со своим знакомым, который также использует модем V.90. Вы уверены, что все АТС на пути между вами и вашим знакомым работают в цифровом режиме. На какой скорости вы получите соединение со своим знакомым?
19. В каких случаях выгоднее использовать для удаленного доступа: сеть ISDN с интерфейсом B+D, выделенный цифровой канал 64 Кбит/с, постоянный виртуальный канал frame relay с CIR=64 Кбит/с?
20. Вы соединили две локальные сети удаленным мостом, работающим через постоянный виртуальный канал в сети frame relay. Сессия протокола NetBEUI между компьютерами разных сетей часто разрывается, в то же время в том случае, когда компьютеры принадлежат одной локальной сети, их взаимодействие протекает без проблем. В чем может быть причина такой ситуации?

ГЛАВА 7. СРЕДСТВА АНАЛИЗА И УПРАВЛЕНИЯ СЕТЯМИ

Любая сложная вычислительная сеть требует дополнительных специальных средств управления помимо тех, которые имеются в стандартных сетевых операционных системах. Это связано с большим количеством разнообразного коммуникационного оборудования, работа которого критична для выполнения сетью своих основных функций. Распределенный характер крупной корпоративной сети делает невозможным поддержание ее работы без централизованной системы управления, которая в автоматическом режиме собирает информацию о состоянии каждого концентратора, коммутатора, мультиплексора и маршрутизатора и предоставляет эту информацию оператору сети. Обычно система управления работает в автоматизированном режиме, выполняя наиболее простые действия по управлению сетью автоматически, а сложные решения предоставляя принимать человеку на основе подготовленной системой информации. Система управления должна быть интегрированной. Это означает, что функции управления разнородными устройствами должны служить общей цели обслуживания конечных пользователей сети с заданным качеством.

Сами системы управления представляют собой сложные программно-аппаратные комплексы, поэтому существует граница целесообразности применения системы управления – она зависит от сложности сети, разнообразия применяемого коммуникационного оборудования и степени его распределенности по территории. В небольшой сети можно применять отдельные программы управления наиболее сложными устройствами, например коммутатором, поддерживающим технику VLAN. Обычно каждое устройство, которое требует достаточно сложного конфигурирования, производитель сопровождает автономной программой конфигурирования и управления. Однако при росте сети может возникнуть проблема объединения разрозненных программ управления устройствами в единую систему управления, и для решения этой проблемы придется, возможно, отказаться от этих программ и заменить их интегрированной системой управления.

7.1. Функции и архитектура систем управления сетями

7.1.1. Функциональные группы задач управления

Системы управления корпоративными сетями существуют не очень давно. Одной из первых систем

такого назначения, получившей широкое распространение, был программный продукт SunNet Manager, выпущенный в 1989 году компанией SunSoft. SunNet Manager был ориентирован на управление коммуникационным оборудованием и контроль трафика сети. Именно эти функции имеют чаще всего в виду, когда говорят о системе управления сетью. Кроме систем управления сетями существуют и системы управления другими элементами корпоративной сети: системы управления ОС, СУБД, корпоративными приложениями. Применяются также системы управления телекоммуникационными сетями: телефонными, а также первичными сетями технологий PDH и SDH.

Независимо от объекта управления, желательно, чтобы система управления выполняла ряд функций, которые определены международными стандартами, обобщающими опыт применения систем управления в различных областях. Существуют рекомендации ITU-T X.700 и близкий к ним стандарт ISO 7498-4, которые делят задачи системы управления на пять функциональных групп:

- управление конфигурацией сети и именованием;
- обработка ошибок;
- анализ производительности и надежности;
- управление безопасностью;
- учет работы сети.

Рассмотрим задачи этих функциональных областей управления применительно к системам управления сетями.

Управление конфигурацией сети и именованием (*Configuration Management*). Эти задачи заключаются в конфигурировании параметров как *элементов сети* (*Network Element, NE*), так и сети в целом. Для элементов сети, таких как маршрутизаторы, мультиплексоры и т. п., с помощью этой группы задач определяются сетевые адреса, идентификаторы (имена), географическое положение и пр.

Для сети в целом управление конфигурацией обычно начинается с построения карты сети, то есть отображении реальных связей между элементами сети и изменении связей между элементами сети – образование новых физических или логических каналов, изменение таблиц коммутации и маршрутизации.

Управление конфигурацией (как и другие задачи системы управления) могут выполняться в автоматическом, ручном или полуавтоматическом режимах. Например, карта сети может составляться автоматически, на основании зондирования реальной сети пакетами-исследователями, а может быть введена оператором системы управления вручную. Чаще всего применяются полуавтоматические методы, когда автоматически полученную карту оператор подправляет вручную. Методы автоматического построения топологической карты, как правило, являются фирменными разработками.

Более сложной задачей является настройка коммутаторов и маршрутизаторов на поддержку маршрутов и виртуальных путей между пользователями сети. Согласованная ручная настройка таблиц маршрутизации при полном или частичном отказе от использования протокола маршрутизации (а в некоторых глобальных сетях, например X.25, такого протокола просто не существует) представляет собой сложную задачу. Многие системы управления сетью общего назначения ее не выполняют, но существуют специализированные системы конкретных производителей, например система NetSys компании Cisco Systems, которая решает ее для маршрутизаторов этой же компании.

Обработка ошибок (*Fault Management*). Эта группа задач включает выявление, определение и устранение последствий сбоев и отказов в работе сети. На этом уровне выполняется не только регистрация сообщений об ошибках, но и их фильтрация, маршрутизация и анализ на основе некоторой корреляционной модели. Фильтрация позволяет выделить из весьма интенсивного потока сообщений об ошибках, который обычно наблюдается в большой сети, только важные сообщения, маршрутизация обеспечивает их доставку нужному элементу системы управления, а корреляционный анализ позволяет найти причину, породившую поток взаимосвязанных сообщений (например, обрыв кабеля может быть причиной большого количества сообщений о недоступности сетей и серверов).

Устранение ошибок может быть как автоматическим, так и полуавтоматическим. В первом случае система непосредственно управляет оборудованием или программными комплексами и обходит отказавший элемент за счет резервных каналов и т. п. В полуавтоматическом режиме основные решения и действия по устранению неисправности выполняют люди, а система управления только помогает в организации этого процесса – оформляет квитанции на выполнение работ и отслеживает их поэтапное выполнение (подобно системам групповой работы).

В этой группе задач иногда выделяют подгруппу задач управления проблемами, подразумевая под проблемой сложную ситуацию, требующую для разрешения обязательного привлечения специалистов

по обслуживанию сети.

Анализ производительности и надежности (*Performance Management*). Задачи этой группы связаны с оценкой на основе накопленной статистической информации таких параметров, как время реакции системы, пропускная способность реального или виртуального канала связи между двумя конечными абонентами сети, интенсивность трафика в отдельных сегментах и каналах сети, вероятность искажения данных при их передаче через сеть, а также коэффициент готовности сети или ее определенной транспортной службы. Функции анализа производительности и надежности сети нужны как для оперативного управления сетью, так и для планирования развития сети.

Результаты анализа производительности и надежности позволяют контролировать *соглашение об уровне обслуживания* (*Service Level Agreement, SLA*), заключаемое между пользователем сети и ее администраторами (или компанией, продающей услуги). Обычно в SLA оговариваются такие параметры надежности, как коэффициент готовности службы в течение года и месяца, максимальное время устранения отказа, а также параметры производительности, например, средняя и максимальная пропускная способности при соединении двух точек подключения пользовательского оборудования, время реакции сети (если информационная служба, для которой определяется время реакции, поддерживается внутри сети), максимальная задержка пакетов при передаче через сеть (если сеть используется только как транзитный транспорт). Без средств анализа производительности и надежности поставщик услуг публичной сети или отдел информационных технологий предприятия не сможет ни проконтролировать, ни тем более обеспечить нужный уровень обслуживания для конечных пользователей сети.

Управление безопасностью (*Security Management*). Задачи этой группы включают в себя контроль доступа к ресурсам сети (данным и оборудованию) и сохранение целостности данных при их хранении и передаче через сеть. Базовыми элементами управления безопасностью являются процедуры аутентификации пользователей, назначение и проверка прав доступа к ресурсам сети, распределение и поддержка ключей шифрования, управления полномочиями и т. п. Часто функции этой группы не включаются в системы управления сетями, а реализуются либо в виде специальных продуктов (например, системы аутентификации и авторизации Kerberos, различных защитных экранов, систем шифрования данных), либо входят в состав операционных систем и системных приложений.

Учет работы сети (*Accounting Management*). Задачи этой группы занимаются регистрацией времени использования различных ресурсов сети – устройств, каналов и транспортных служб. Эти задачи имеют дело с такими понятиями, как время использования службы и плата за ресурсы – billing. Ввиду специфического характера оплаты услуг у различных поставщиков и различными формами соглашения об уровне услуг, эта группа функций обычно не включается в коммерческие системы и платформы управления типа HP OpenView, а реализуется в заказных системах, разрабатываемых для конкретного заказчика.

Модель управления OSI не делает различий между управляемыми объектами – каналами, сегментами локальных сетей, мостами, коммутаторами и маршрутизаторами, модемами и мультиплексорами, аппаратным и программным обеспечением компьютеров, СУБД. Все эти объекты управления входят в общее понятие «система», и управляемая система взаимодействует с управляющей системой по открытым протоколам OSI.

Однако на практике деление систем управления по типам управляемых объектов широко распространено. Ставшими классическими системы управления сетями, такие как SunNet Manager, HP OpenView или Cabletron Spectrum, управляют только коммуникационными объектами корпоративных сетей, то есть концентраторами и коммутаторами локальных сетей, а также маршрутизаторами и удаленными мостами, как устройствами доступа к глобальным сетям. Оборудованием территориальных сетей обычно управляют системы производителей телекоммуникационного оборудования, такие как RADView компании RAD Data Communications, MainStreetXpress 46020 компании Newbridge и т. п.

Рассмотрим, как преломляются общие функциональные задачи системы управления, определенные в стандартах X.700/ISO 7498-4, в задачи такого конкретного класса систем управления, как системы управления компьютерами и их системным и прикладным программным обеспечением. Их называют *системами управления системой* (*System Management System*).

Обычно система управления системой выполняет следующие функции.

- *Учет используемых аппаратных и программных средств* (*Configuration Management*). Система автоматически собирает информацию об установленных в сети компьютерах и создает записи в специальной базе данных об аппаратных и программных ресурсах. После этого

администратор может быстро выяснить, какими ресурсами он располагает и где тот или иной ресурс находится, например, узнать о том, на каких компьютерах нужно обновить драйверы принтеров, какие компьютеры обладают достаточным количеством памяти, дискового пространства и т. п.

- *Распределение и установка программного обеспечения* (*Configuration Management*). После завершения обследования администратор может создать пакеты рассылки нового программного обеспечения, которое нужно инсталлировать на всех компьютерах сети или на какой-либо группе компьютеров. В большой сети, где проявляются преимущества системы управления, такой способ инсталляции может существенно уменьшить трудоемкость этой процедуры. Система может также позволять централизованно устанавливать и администрировать приложения, которые запускаются с файловых серверов, а также дать возможность конечным пользователям запускать такие приложения с любой рабочей станции сети.
- *Удаленный анализ производительности и возникающих проблем* (*Fault Managemen*t *and Performance Management*). Эта группа функций позволяет удаленно измерять наиболее важные параметры компьютера, операционной системы, СУБД и т. д. (например, коэффициент использования процессора, интенсивность страничных прерываний, коэффициент использования физической памяти, интенсивность выполнения транзакций). Для разрешения проблем эта группа функций может давать администратору возможность брать на себя удаленное управление компьютером в режиме эмуляции графического интерфейса популярных операционных систем. База данных системы управления обычно хранит детальную информацию о конфигурации всех компьютеров в сети для того, чтобы можно было выполнять удаленный анализ возникающих проблем.

Примерами систем управления системами являются Microsoft System Management Server (SMS), CA Unicenter, HP Operationscenter и многие другие.

Как видно из описания функций системы управления системами, они повторяют функции системы управления сетью, но только для других объектов. Действительно, функция учета используемых аппаратных и программных средств соответствует функции построения карты сети, функция распределения и установки программного обеспечения – функции управления конфигурацией коммутаторов и маршрутизаторов, а функция анализа производительности и возникающих проблем – функции производительности.

Эта близость функций систем управления сетями и систем управления системами позволила разработчикам стандартов OSI не делать различия между ними и разрабатывать общие стандарты управления.

На практике уже несколько лет также заметна отчетливая тенденция интеграции систем управления сетями и системами в единые интегрированные продукты управления корпоративными сетями, например CA Unicenter TNG или TME-10 IBM/Tivoli. Наблюдается также интеграция систем управления телекоммуникационными сетями с системами управления корпоративными сетями.

7.1.2. Многоуровневое представление задач управления

Кроме описанного выше разделения задач управления на несколько функциональных групп, полезно разделять задачи управления на уровни в соответствии с иерархической организацией корпоративной сети. Корпоративная сеть строится иерархически, отражая иерархию самого предприятия и его задач. Нижний уровень сети составляют элементы сети – отдельные компьютеры, коммуникационные устройства, каналы передачи данных. На следующем уровне иерархии эти элементы образуют сети разного масштаба – сеть рабочей группы, сеть отдела, сеть отделения и, наконец, сеть предприятия в целом.

Для построения интегрированной системы управления разнородными элементами сети естественно применить многоуровневый иерархический подход. Это, в принципе, стандартный подход для построения большой системы любого типа и назначения – от государства до автомобильного завода. Применительно к системам управления сетями наиболее проработанным и эффективным для создания многоуровневой иерархической системы является стандарт Telecommunication Management Network (TMN), разработанный совместными усилиями ITU-T, ISO, ANSI и ETSI. Хотя этот стандарт и предназначался изначально для телекоммуникационных сетей, но ориентация на использование общих

принципов делает его полезным для построения любой крупной интегрированной системы управления сетями. Стандарты TMN состоят из большого количества рекомендаций ITU-T (и стандартов других организаций), но основные принципы модели TMN описаны в рекомендации M.3010.

На каждом уровне иерархии модели TMN решаются задачи одних и тех же пяти функциональных групп, рассмотренных выше (то есть управления конфигурацией, производительностью, ошибками, безопасностью и учетом), однако на каждом уровне эти задачи имеют свою специфику. Чем выше уровень управления, тем более общий и агрегированный характер приобретает собираемая о сети информация, а сугубо технический характер собираемых данных начинает по мере повышения уровня меняться на производственный, финансовый и коммерческий.

Модель TMN упрощенно можно представить в виде двухмерной диаграммы (рис. 7.1).

Нижний уровень – *уровень элементов сети* (*Network Element layer, NE*) – состоит из отдельных устройств сети: каналов, усилителей, оконечной аппаратуры, мультиплексоров, коммутаторов и т. п. Элементы могут содержать встроенные средства для поддержки управления – датчики, интерфейсы управления, а могут и представлять вещь в себе, требующую для связи с системой управления разработки специального оборудования – *устройств связи с объектом, УСО*. Современные технологии обычно имеют встроенные функции управления, которые позволяют выполнять хотя бы минимальные операции по контролю за состоянием устройства и за передаваемым устройством трафиком. Подобные функции встроены в технологии FDDI, ISDN, frame relay, SDH. В этом случае устройство всегда можно охватить системой управления, даже если оно не имеет специального блока управления, так как протокол технологии обязывает устройство поддерживать некоторые функции управления. Устройства, которые работают по протоколам, не имеющим встроенных функций контроля и управления, снабжаются отдельным блоком управления, который поддерживает один из двух наиболее распространенных протоколов управления – SNMP или CMIP. Эти протоколы относятся к прикладному уровню модели OSI.

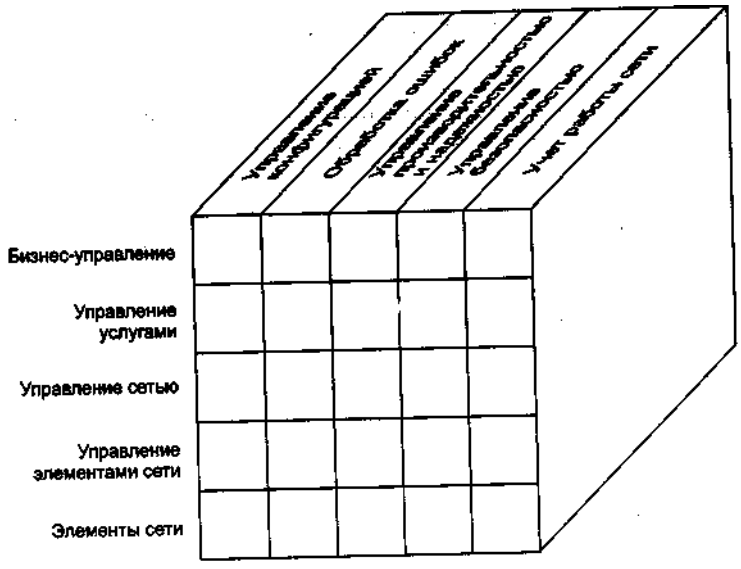


Рис. 7.1. Многоуровневое представление задач управления сетью

Следующий уровень – *уровень управления элементами сети* (*network element management layer*) – представляет собой элементарные системы управления. Элементарные системы управления автономно управляют отдельными элементами сети – контролируют канал связи SDH, управляют коммутатором или мультиплексором. Уровень управления элементами изолирует верхние слои системы управления от деталей и особенностей управления конкретным оборудованием. Этот уровень ответственен за моделирование поведения оборудования и функциональных ресурсов нижележащей сети. Атрибуты этих моделей позволяют управлять различными аспектами поведения управляемых ресурсов. Обычно элементарные системы управления разрабатываются и поставляются производителями оборудования. Примерами таких систем могут служить системы управления CiscoView от Cisco Systems, Optivity от Bay Networks, RAD View от RAD Data Communications и т. д.

Выше лежит *уровень управления сетью* (*Network management layer*). Этот уровень координирует работу элементарных систем управления, позволяя контролировать конфигурацию составных каналов,

согласовывать работу транспортных подсетей разных технологий и т. п. С помощью этого уровня сеть начинает работать как единое целое, передавая данные между своими абонентами.

Следующий уровень – *уровень управления услугами* (*Service management layer*) – занимается контролем и управлением за транспортными и информационными услугами, которые предоставляются конечным пользователям сети. В задачу этого уровня входит подготовка сети к предоставлению определенной услуги, ее активизация, обработка вызовов клиентов. Формирование услуги (service provisioning) заключается в фиксации в базе данных значений параметров услуги, например, требуемой средней пропускной способности, максимальных величин задержек пакетов, коэффициента готовности и т. п. В функции этого уровня входит также выдача уровню управления сетью задания на конфигурирование виртуального или физического канала связи для поддержания услуги. После формирования услуги данный уровень занимается контролем за качеством ее реализации, то есть за соблюдением сетью всех принятых на себя обязательств в отношении производительности и надежности транспортных услуг. Результаты контроля качества обслуживания нужны, в частности, для подсчета оплаты за пользование услугами клиентами сети. Например, в сети frame relay уровень управления услугами следит за заказанными пользователем значениями средней скорости CIR и согласованной пульсации Bc, фиксируя нарушения со стороны пользователя и сети.

Уровень бизнес-управления (*Business management layer*) занимается вопросами долговременного планирования сети с учетом финансовых аспектов деятельности организации, владеющей сетью. На этом уровне помесячно и поквартально подсчитываются доходы от эксплуатации сети и ее отдельных составляющих, учитываются расходы на эксплуатацию и модернизацию сети, принимаются решения о развитии сети с учетом финансовых возможностей. Уровень бизнес-управления обеспечивает для пользователей и поставщиков услуг возможность предоставления дополнительных услуг. Этот уровень является частным случаем уровня автоматизированной системы управления предприятием (АСУП), в то время как все нижележащие уровни соответствуют уровням автоматизированной системы управления технологическими процессами (АСУТП), для такого специфического типа предприятия, как телекоммуникационная или корпоративная сеть. Но если телекоммуникационная сеть действительно чаще всего является основой телекоммуникационной компании, то корпоративную сеть и обслуживающий ее персонал обычно трудно назвать предприятием. Тем не менее на некоторых западных фирмах корпоративная сеть выделена в автономное производственное подразделение со своим бюджетом и со своими финансовыми договорами на обслуживание, которое данное подразделение заключает с основными производственными подразделениями предприятия.

7.1.3. Архитектуры систем управления сетями

Выделение в системах управления типовых групп функций и разбиение этих функций на уровни еще не дает ответа на вопрос, каким же образом устроены системы управления, из каких элементов они состоят и какие архитектуры связей этих элементов используются на практике.

Схема менеджер – агент

В основе любой системы управления сетью лежит элементарная схема взаимодействия агента с менеджером. На основе этой схемы могут быть построены системы практически любой сложности с большим количеством агентов и менеджеров разного типа.

Схема «менеджер - агент» представлена на рис, 7.2.

Агент является посредником между управляемым ресурсом и основной управляющей программой-менеджером. Чтобы один и тот же менеджер мог управлять различными реальными ресурсами, создается некоторая модель управляемого ресурса, которая отражает только те характеристики ресурса, которые нужны для его контроля и управления. Например, модель маршрутизатора обычно включает такие характеристики, как количество портов, их тип, таблицу маршрутизации, количество кадров и пакетов протоколов канального, сетевого и транспортного уровней, прошедших через эти порты.

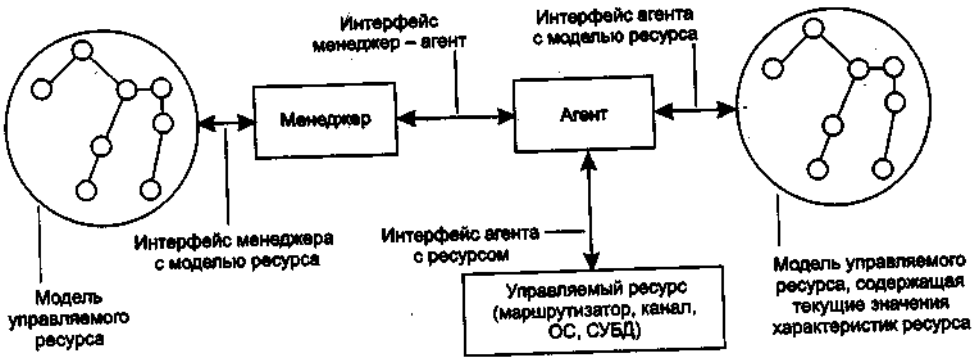


Рис. 7.2. Взаимодействие агента, менеджера и управляемого ресурса

Менеджер получает от агента только те данные, которые описываются моделью ресурса. Агент же является некоторым экраном, освобождающим менеджера от ненужной информации о деталях реализации ресурса. Агент поставляет менеджеру обработанную и представленную в нормализованном виде информацию. На основе этой информации менеджер принимает решения по управлению, а также выполняет дальнейшее обобщение данных о состоянии управляемого ресурса, например, строит зависимость загрузки порта от времени.

Для получения требуемых данных от объекта, а также для выдачи на него управляющих воздействий агент взаимодействует с реальным ресурсом некоторым нестандартным способом. Когда агенты встраиваются в коммуникационное оборудование, то разработчик оборудования предусматривает точки и способы взаимодействия внутренних узлов устройства с агентом. При разработке агента для операционной системы разработчик агента пользуется теми интерфейсами, которые существуют в этой ОС, например интерфейсами ядра, драйверов и приложений. Агент может снабжаться специальными датчиками для получения информации, например датчиками релейных контактов или датчиками температуры.

Менеджер и агент должны располагать одной и той же моделью управляемого ресурса, иначе они не смогут понять друг друга. Однако в использовании этой модели агентом и менеджером имеется существенное различие. Агент наполняет модель управляемого ресурса текущими значениями характеристик данного ресурса, и в связи с этим модель агента называют *базой данных управляющей информации – Management Information Base, MIB*. Менеджер использует модель, чтобы знать о том, чем характеризуется ресурс, какие характеристики он может запросить у агента и какими параметрами можно управлять.

Менеджер взаимодействует с агентами по стандартному протоколу. Этот протокол должен позволять менеджеру запрашивать значения параметров, хранящихся в базе MIB, а также передавать агенту управляющую информацию, на основе которой тот должен управлять устройством. Различают управление *in-band*, то есть по тому же каналу, по которому передаются пользовательские данные, и управление *out-of-band*, то есть вне канала, по которому передаются пользовательские данные. Например, если менеджер взаимодействует с агентом, встроенным в маршрутизатор, по протоколу SNMP, передаваемому по той же локальной сети, что и пользовательские данные, то это будет управление in-band. Если же менеджер контролирует коммутатор первичной сети, работающий по технологии частотного уплотнения FDM, с помощью отдельной сети X.25, к которой подключен агент, то это будет управление out-of-band. Управление по тому же каналу, по которому работает сеть, более экономично, так как не требует создания отдельной инфраструктуры передачи управляющих данных. Однако способ out-of-band более надежен, так как он предоставляет возможность управлять оборудованием сети и тогда, когда какие-то элементы сети вышли из строя и по основным каналам оборудование недоступно. Стандарт многоуровневой системы управления TMN имеет в своем названии слово Network, подчеркивающее, что в общем случае для управления телекоммуникационной сетью создается отдельная управляющая сеть, которая обеспечивает режим out-of-band.

Обычно менеджер работает с несколькими агентами, обрабатывая получаемые от них данные и выдавая на них управляющие воздействия. Агенты могут встраиваться в управляемое оборудование, а могут и работать на отдельном компьютере, связанном с управляемым оборудованием по какому-либо интерфейсу. Менеджер обычно работает на отдельном компьютере, который выполняет также роль консоли управления для оператора или администратора системы.

Модель менеджер - агент лежит в основе таких популярных стандартов управления, как стандарты

Internet на основе протокола SNMP и стандарты управления ISO/OSI на основе протокола CMIP.

Агенты могут отличаться различным уровнем интеллекта – они могут обладать как самым минимальным интеллектом, необходимым для подсчета проходящих через оборудование кадров и пакетов, так и весьма высоким, достаточным для выполнения самостоятельных действий по выполнению последовательности управляющих действий в аварийных ситуациях, построению временных зависимостей, фильтрации аварийных сообщений и т. п.

Структуры распределенных систем управления

В крупной корпоративной сети полностью централизованная система управления, построенная на базе единственного менеджера, вряд ли будет работать хорошо по нескольким причинам. Во-первых, такой вариант не обеспечивает необходимой масштабируемости по производительности, так как единственный менеджер вынужден будет обрабатывать весь поток сообщений от всех агентов, что при нескольких тысячах управляемых объектов потребует очень высокопроизводительной платформы для работы менеджера и перегрузит служебной управляющей информацией каналы передачи данных в той сети, где будет расположен менеджер. Во-вторых, такое решение не обеспечит необходимого уровня надежности, так как при отказе единственного менеджера будет потеряно управление сетью. В-третьих, в большой распределенной сети целесообразно располагать в каждом географическом пункте отдельным оператором или администратором, управляющим своей частью сети, а это удобнее реализовать с помощью отдельных менеджеров для каждого оператора.

Схема «менеджер - агент» позволяет строить достаточно сложные в структурном отношении распределенные системы управления.

Обычно распределенная система управления включает большое количество связок менеджер - агент, которые дополняются рабочими станциями операторов сети, с помощью которых они получают доступ к менеджерам (рис. 7.3).

Каждый агент собирает данные и управляет определенным элементом сети. Менеджеры, иногда также называемые серверами системы управления, собирают данные от своих агентов, обобщают их и хранят в базе данных. Операторы, работающие за рабочими станциями, могут соединиться с любым из менеджеров и с помощью графического интерфейса просмотреть данные об управляемой сети, а также выдать менеджеру некоторые директивы по управлению сетью или ее элементами.

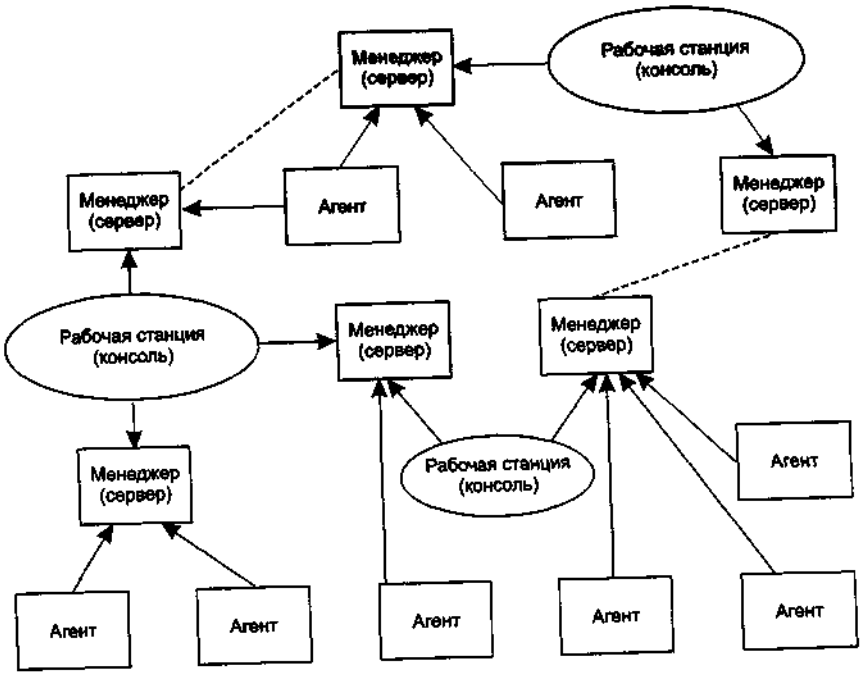


Рис. 7.3. Распределенная система управления на основе нескольких менеджеров и рабочих станций

Наличие нескольких менеджеров позволяет распределить между ними нагрузку по обработке данных управления, обеспечивая масштабируемость системы.

Как правило, связи между агентами и менеджерами носят более упорядоченный характер, чем тот,

который показан на рис. 7.3. Чаще всего используются два подхода к их соединению – одноранговый (рис. 7.4) и иерархический (рис. 7.5).

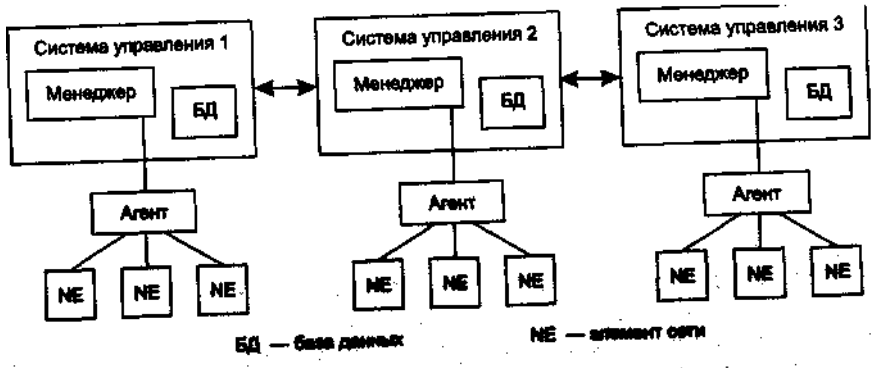


Рис. 7.4. Одноранговые связи между менеджерами

В случае одноранговых связей каждый менеджер управляет своей частью сети на основе информации, получаемой от нижележащих агентов. Центральный менеджер отсутствует. Координация работы менеджеров достигается за счет обмена информацией между базами данных каждого менеджера.

Одноранговое построение системы управления сегодня считается неэффективым и устаревшим. Обычно оно вызвано тем обстоятельством, что элементарные системы управления построены как монолитные системы, которые первоначально не были ориентированы на модульность системы (например, многие системы управления, разработанные производителями оборудования, не поддерживают стандартные интерфейсы для взаимодействия с другими системами управления). Затем эти менеджеры нижнего уровня стали объединяться для создания интегрированной системы управления сетью, но связи между ними оказалось возможным создавать только на уровне обмена между базами данных, что достаточно медленно. Кроме того, в базах данных таких менеджеров накапливается слишком детальная информация об управляемых элементах сети (так как первоначально эти менеджеры разрабатывались как менеджеры нижнего уровня), вследствие чего такая информация малопригодна для координации работы всей сети в целом. Такой подход к построению системы управления называется подходом «снизу вверх».

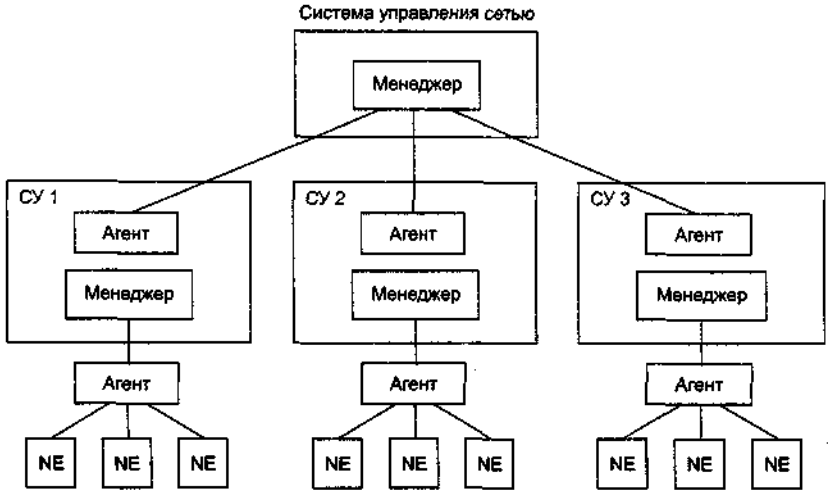


Рис. 7.5. Иерархические связи между менеджерами

Гораздо более гибким является иерархическое построение связей между менеджерами. Каждый менеджер нижнего уровня выполняет также функции агента для менеджера верхнего уровня. Такой агент работает уже с гораздо более укрупненной моделью (MIB) своей части сети, в которой собирается именно та информация, которая нужна менеджеру верхнего уровня для управления сетью в целом. Обычно для разработки моделей сети на разных уровнях проектирование начинают с верхнего уровня, на котором определяется состав информации, требуемой от менеджеров-агентов более низкого уровня,

поэтому такой подход назван подходом «сверху вниз». Он сокращает объемы информации, циркулирующей между уровнями системы управления, и приводит к гораздо более эффективной системе управления.

Модель TMN в наибольшей степени соответствует иерархической архитектуре связей между менеджерами, хотя известны реализации принципов TMN и в одноуровневых архитектурах.

Платформенный подход

При построении систем управления крупными локальными и корпоративными сетями обычно используется платформенный подход, когда индивидуальные программы управления разрабатываются не «с нуля», а используют службы и примитивы, предоставляемые специально разработанным для этих целей программным продуктом – платформой. Примерами платформ для систем управления являются такие известные продукты, как *HP OpenView, SunNet Manager и Sun Soltice, Cabletron Spectrum, IMB/Tivoli TMN10.*

Эти платформы создают общую операционную среду для приложений системы управления точно так же, как универсальные операционные системы, такие как Unix или Windows NT, создают операционную среду для приложений любого типа, таких как MS Word, Oracle и т. п. Платформа обычно включает поддержку протоколов взаимодействия менеджера с агентами – SNMP и реже CMIP, набор базовых средств для построения менеджеров и агентов, а также средства графического интерфейса для создания консоли управления. В набор базовых средств обычно входят функции, необходимые для построения карты сети, средства фильтрации сообщений от агентов, средства ведения базы данных. Набор интерфейсных функций платформы образует интерфейс прикладного программирования (API) системы управления. Пользуясь этим API, разработчики из третьих фирм создают законченные системы управления, которые могут управлять специфическим оборудованием в соответствии с пятью основными группами функций.

Обычно платформа управления поставляется с каким-либо универсальным менеджером, который может выполнять некоторые базовые функции управления без программирования. Чаще всего к этим функциям относятся функции построения карты сети (группа Configuration Management), а также функции отображения состояния управляемых устройств и функции фильтрации сообщений об ошибках (группа Fault Management). Например, одна из наиболее популярных платформ HP OpenView поставляется с менеджером Network Node Manager, который выполняет перечисленные функции.

Чем больше функций выполняет платформа, тем лучше. В том числе и таких, которые нужны для разработки любых аспектов работы приложений, прямо не связанных со спецификой управления. В конце концов, приложения системы управления – это прежде всего приложения, а потом уже приложения системы управления. Поэтому полезны любые средства, предоставляемые платформой, которые ускоряют разработку приложений вообще и распределенных приложений в частности.

Компании, которые производят коммуникационное оборудование, разрабатывают дополнительные менеджеры для популярных платформ, которые выполняют функции управления оборудованием данного производителя более полно. Примерами таких менеджеров могут служить менеджеры системы Optivity компании Bay Networks и менеджеры системы Trancsend компании 3Com, которые могут работать в среде платформ HP OpenView и SunNet Manager.

Выводы

- Желательно, чтобы системы управления сетями выполняли все пять групп функций, определенных стандартами ISO/ITU-T для систем управления объектами любого типа.
- Система управления большой сетью должна иметь многоуровневую иерархическую структуру в соответствии со стандартами Telecommunication Management Network (TMN), позволяющую объединить разрозненные системы управления элементами сети в единую интегрированную систему.
- В основе всех систем управления сетями лежит схема «агент - менеджер». Эта схема использует абстрактную модель управляемого ресурса, называемую базой управляющей информации – Management Information Base, MIB.
- Агент взаимодействует с управляемым ресурсом по нестандартному интерфейсу, а с менеджером – по стандартному протоколу через сеть.

- В больших системах управления используется несколько менеджеров, которые взаимодействуют друг с другом по одной из двух схем – одноранговой и иерархической.
- Иерархическая схема взаимодействия менеджеров соответствует стандартам TMN и является более перспективной.
- При построении систем управления активно используется платформенный подход. Платформа системы управления выполняет для менеджеров роль операционной системы для обычных приложений, так как обеспечивает разработчика менеджеров набором полезных системных вызовов общего для любой системы управления назначения.

7.2. Стандарты систем управления

7.2.1. Стандартизуемые элементы системы управления

При формализации схемы «менеджер - агент» могут быть стандартизованы следующие аспекты ее функционирования:

- протокол взаимодействия агента и менеджера;
- интерфейс «агент - управляемый ресурс»;
- интерфейс «агент - модель управляемого ресурса»;
- интерфейс «менеджер - модель управляемого ресурса»;
- справочная система о наличии и местоположении агентов и менеджеров, упрощающая построение распределенной системы управления;
- язык описания моделей управляемых ресурсов, то есть язык описания MIB;
- схема наследования классов моделей объектов (дерево наследования), которая позволяет строить модели новых объектов на основе моделей более общих объектов, например, модели маршрутизаторов на основе модели обобщенного коммуникационного устройства;
- схема иерархических отношений моделей управляемых объектов (дерево включения), которая позволяет отразить взаимоотношения между отдельными элементами реальной системы, например, принадлежность модулей коммутации определенному коммутатору или отдельных коммутаторов и концентраторов определенной подсети.

Существующие стандарты на системы управления отличаются тем, что в них может быть стандартизованы не все перечисленные выше аспекты схемы «менеджер - агент».

В стандартах систем управления как минимум стандартизуется некоторый способ формального описания моделей управляемых объектов, а также определяется протокол взаимодействия между менеджером и агентом.

Сегодня на практике применяются два семейства стандартов управления сетями – стандарты Internet, построенные на основе протокола SNMP (Simple Network Management Protocol), и международные стандарты ISO/ITU-T, использующие в качестве протокола взаимодействия агентов и менеджеров протокол CMIP (Common Management Information Protocol).

Стандарты систем управления, основанных на протоколе SNMP, формализуют минимум аспектов системы управления, а стандарты ISO/ITU-T – максимум аспектов, как и большинство стандартов, разработанных ITU-T. Традиционно, в локальных и корпоративных сетях применяются в основном системы управления на основе SNMP, а стандарты ISO/ITU-T и протокол CMIP находят применение в телекоммуникационных сетях.

7.2.2. Стандарты систем управления на основе протокола SNMP

Концепции SNMP-управления

В системах управления, построенных на основе протокола SNMP, стандартизуются следующие элементы:

- протокол взаимодействия агента и менеджера;
- язык описания моделей MIB и сообщений SNMP – язык абстрактной синтаксической нотации ASN.1 (стандарт ISO 8824:1987, рекомендации ITU-T X.208);
- несколько конкретных моделей MIB (MIB-I, MIB-II, RMON, RMON 2), имена объектов

которых регистрируются в дереве стандартов ISO.

Все остальное отдается на откуп разработчику системы управления.

Протокол SNMP и тесно связанная с ним концепция SNMP MIB были разработаны для управления маршрутизаторами Internet как временное решение. Но, как это часто бывает со всем временным, простота и эффективность решения обеспечили успех этого протокола, и сегодня он используется при управлении практически любыми видами оборудования и программного обеспечения вычислительных сетей. И хотя в области управления телекоммуникационными сетями наблюдается устойчивая тенденция применения стандартов ITU-T, в которые входит протокол CMIP, и здесь имеется достаточно много примеров успешного использования SNMP-управления. Агенты SNMP встраиваются в аналоговые модемы, модемы ADSL, коммутаторы ATM и т. д.

SNMP – это протокол прикладного уровня, разработанный для стека TCP/IP, хотя имеются его реализации и для других стеков, например IPX/SPX. Протокол SNMP используется для получения от сетевых устройств информации об их статусе, производительности и других характеристиках, которые хранятся в базе данных управляющей информации MIB (Management Information Base). Простота SNMP во многом определяется простотой MIB SNMP, особенно их первых версий MIB I и MIB II. Кроме того, сам протокол SNMP также весьма несложен.

Существуют стандарты, определяющие структуру MIB, в том числе набор типов ее объектов, их имена и допустимые операции над этими объектами (например, «читать»).

Древовидная структура MIB содержит обязательные (стандартные) поддеревья, а также в ней могут находиться частные (private) поддеревья, позволяющие изготовителю интеллектуальных устройств управлять какими-либо специфическими функциями устройства на основе специфических объектов MIB.

Агент в протоколе SNMP – это обрабатывающий элемент, который обеспечивает менеджерам, размещенным на управляющих станциях сети, доступ к значениям переменных MIB и тем самым дает им возможность реализовывать функции по управлению и наблюдению за устройством.

Основные операции по управлению вынесены в менеджер, а агент SNMP выполняет чаще всего пассивную роль, передавая в менеджер по его запросу значения накопленных статистических переменных. При этом устройство работает с минимальными издержками на поддержание управляющего протокола. Оно использует почти всю свою вычислительную мощность для выполнения своих основных функций маршрутизатора, моста или концентратора, а агент занимается сбором статистики и значений переменных состояния устройства и передачей их менеджеру системы управления.

Примитивы протокола SNMP

SNMP – это протокол типа «запрос-ответ», то есть на каждый запрос, поступивший от менеджера, агент должен передать ответ. Особенностью протокола является его чрезвычайная простота – он включает в себя всего несколько команд.

- Команда Get-request используется менеджером для получения от агента значения какого-либо объекта по его имени.
- Команда GetNext- request используется менеджером для извлечения значения следующего объекта (без указания его имени) при последовательном просмотре таблицы объектов.
- С помощью команды Get-response агент SNMP передает менеджеру ответ на команды Get-request или GetNext-request.
- Команда Set используется менеджером для изменения значения какого-либо объекта. С помощью команды Set происходит собственно управление устройством. Агент должен понимать смысл значений объекта, который используется для управления устройством, и на основании этих значений выполнять реальное управляющее воздействие – отключить порт, приписать порт определенной VLAN и т. п. Команда Set пригодна также для установки условия, при выполнении которого агент SNMP должен послать менеджеру соответствующее сообщение. Может быть определена реакция на такие события, как инициализация агента, рестарт агента, обрыв связи, восстановление связи, неверная аутентификация и потеря ближайшего маршрутизатора. Если происходит любое из этих событий, то агент инициализирует прерывание.
- Команда Trap используется агентом для сообщения менеджеру о возникновении особой

ситуации.

- Версия SNMP v.2 добавляет к этому набору команду GetBulk, которая позволяет менеджеру получить несколько значений переменных за один запрос.

Структура SNMP MIB

На сегодня существует несколько стандартов на базы данных управляющей информации для протокола SNMP. Основными являются стандарты MIB-I и MIB-II, а также версия базы данных для удаленного управления RMON MIB. Кроме этого существуют стандарты для специальных устройств MIB конкретного типа (например, MIB для концентраторов или MIB для модемов), а также частные MIB конкретных фирм-производителей оборудования.

Первоначальная спецификация MIB-I определяла только операции чтения значений переменных. Операции изменения или установки значений объекта являются частью спецификаций MIB-II.

Версия MIB-I (RFC 1156) определяет 114 объектов, которые подразделяются на 8 групп.

- System – общие данные об устройстве (например, идентификатор поставщика, время последней инициализации системы).
- Interfaces – параметры сетевых интерфейсов устройства (например, их количество, типы, скорости обмена, максимальный размер пакета).
- Address Translation Table – описание соответствия между сетевыми и физическими адресами (например, по протоколу ARP).
- Internet Protocol – данные, относящиеся к протоколу IP (адреса IP-шлюзов, хостов, статистика о IP-пакетах).
- ICMP – данные, относящиеся к протоколу обмена управляющими сообщениями ICMP.
- TCP – данные, относящиеся к протоколу TCP (например, о TCP-соединениях),
- UDP – данные, относящиеся к протоколу UDP (число переданных, принятых и ошибочных UPD-дейтаграмм).
- EGP – данные, относящиеся к протоколу обмена маршрутной информацией Exterior Gateway Protocol, используемому в Internet (число принятых с ошибками и без ошибок сообщений).

Из этого перечня групп переменных видно, что стандарт MIB-I разрабатывался с жесткой ориентацией на управление маршрутизаторами, поддерживающими протоколы стека TCP/IP.

В версии MIB-II (RFC 1213), принятой в 1992 году, был существенно (до 185) расширен набор стандартных объектов, а число групп увеличилось до 10.

На рис. 7.6 приведен пример древовидной структуры базы объектов MIB-II. На нем показаны две из 10 возможных групп объектов – System (имена объектов начинаются с префикса Sys) и Interfaces (префикс if). Объект SysUpTime содержит значение продолжительности времени работы системы с момента последней перезагрузки, объект SysObjectID – идентификатор устройства (например, маршрутизатора).

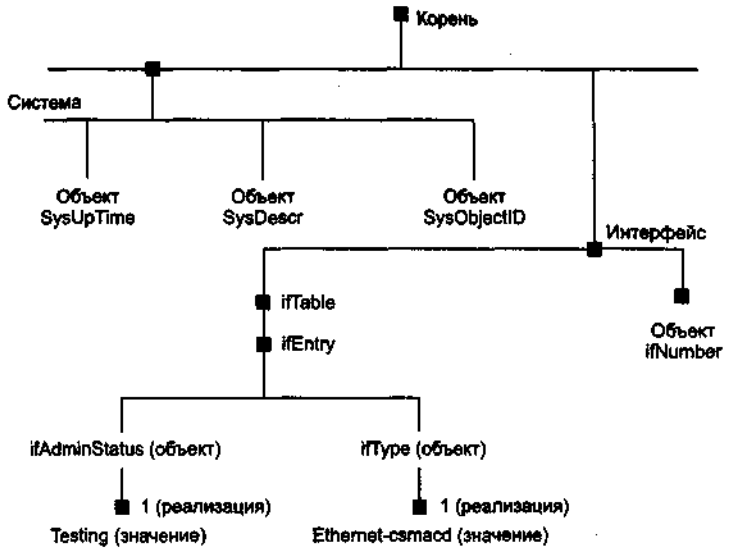


Рис. 7.6. Стандартное дерево MIB-II (фрагмент)

Объект ifNumber определяет количество сетевых интерфейсов устройства, а объект if Entry является вершиной поддерева, описывающего один из конкретных интерфейсов устройства. Входящие в это поддерево объекты ifType и ifAdminStatus определяют соответственно тип и состояние одного из интерфейсов, в данном случае интерфейса Ethernet.

В число объектов, описывающих каждый конкретный интерфейс устройства, включены следующие.

- ifType – тип протокола, который поддерживает интерфейс. Этот объект принимает значения всех стандартных протоколов канального уровня, например rfc877-x25, ethernet-csmacd, iso88023-csmacd, iso88024-tokenBus, iso88025-tokenRing и т. д.
- ifMtu – максимальный размер пакета сетевого уровня, который можно послать через этот интерфейс.
- ifSpeed – пропускная способность интерфейса в битах в секунду (100 для Fast Ethernet).
- ifPhysAddress – физический адрес порта, для Fast Ethernet им будет MAC-адрес.
- ifAdminStatus – желаемый статус порта:
- up – готов передавать пакеты;
- down – не готов передавать пакеты;
- testing – находится в текстовом режиме.
- ifOperStatus – фактический текущий статус порта, имеет те же значения, что и ifAdminStatus.
- ifInOctets – общее количество байт, принятое данным портом, включая служебные, с момента последней инициализации SNMP-агента.
- ifInUcastPkts – количество пакетов с индивидуальным адресом интерфейса, доставленных протоколу верхнего уровня.
- ifInNUcastPkts – количество пакетов с широковещательным или мультивещательным адресом интерфейса, доставленных протоколу верхнего уровня.
- ifInDiscards – количество пакетов, которые были приняты интерфейсом, оказались корректными, но не были доставлены протоколу верхнего уровня, скорее всего из-за переполнения буфера пакетов или же по иной причине.
- ifInErrors – количество пришедших пакетов, которые не были переданы протоколу верхнего уровня из-за обнаружения в них ошибок.

Кроме объектов, описывающих статистику по входным пакетам, имеются аналогичные объекты, но относящиеся к выходным пакетам.

Как видно из описания объектов MIB-II, эта база данных не дает детальной статистики по характерным ошибкам кадров Ethernet, кроме этого, она не отражает изменение характеристик во времени, что часто интересует сетевого администратора.

Эти ограничения были впоследствии сняты новым стандартом на MIB – RMON MIB, который специально ориентирован на сбор детальной статистики по протоколу Ethernet, к тому же с поддержкой такой важной функции, как построение агентом зависимостей статистических характеристик от времени.

Форматы и имена объектов SNMP MIB

Для именования переменных базы MIB и однозначного определения их форматов используется дополнительная спецификация, называемая SMI – Structure of Management Information. Например, спецификация SMI включает в качестве стандартного имя IpAddress и определяет его формат как строку из 4 байт. Другой пример – имя Counter, для которого определен формат в виде целого числа в диапазоне от 0 до 2^{32}-1.

При описании переменных MIB и форматов протокола SNMP спецификация SMI опирается на формальный язык ASN.1, принятый ISO в качестве нотации для описания терминов коммуникационных протоколов (правда, многие коммуникационные протоколы, например IP, PPP или Ethernet, обходятся без этой нотации). Нотация ASN.1 служит для установления однозначного соответствия между терминами, взятыми из стандартов, предназначенных для человеческого использования, и теми данными, которые передаются в коммуникационных протоколах аппаратурой. Достигаемая однозначность очень важна для гетерогенной среды, характерной для корпоративных сетей. Так, вместо того чтобы указать, что некоторая переменная протокола представляет собой целое число, разработчик

протокола, использующий нотацию ASN.1, должен точно определить формат и допустимый диапазон переменной. В результате документация на MIB, написанная с помощью нотации ASN.1, может точно и механически транслироваться в форму кодов, характерных для сообщений протоколов.

Нотация ASN.1 похожа на другие метаязыки, например нормальную Бэкусову форму, используемую при описании языков програмирования,в частности Алгола. Нотация ASN.1 поддерживает базовый набор различных типов данных, таких как целое число, строка и т. п., а также позволяет конструировать из этих базовых типов составные данные – массивы, перечисления, структуры.

Существуют правила трансляции структур данных, описанных на ASN.1, в структуры данных языков программирования, например C++. Соответственно, имеются трансляторы, выполняющие эту работу. Примеры описаний данных с помощью ASN.1 приведены ниже при описании протокольных блоков данных SNMP.

Нотация ASN.1 широко используется при описании многих стандартов OSI, в частности моделей управляемых объектов и структуры сообщений протокола CMIP.

Имена переменных MIB могут быть записаны как в символьном, так и в числовом форматах. Символьный формат используется для представления переменных в текстовых документах и на экране дисплея, а числовые имена – в сообщениях протокола SNMP. Например, символьному имени SysDescr соответствует числовое имя 1, а более точно 1.3.6.1.2.1.1.1.

Составное числовое имя объекта SNMP MIB соответствует полному имени этого объекта в дереве регистрации объектов стандартизации ISO. Разработчики протокола SNMP не стали использовать традиционный для стандартов Internet способ фиксации численных параметров протокола в специальном RFC, называемом «Assigned Numbers» (там описываются, например, численные значения, которые может принимать поле Protocol пакета IP, и т. п.). Вместо этого они зарегистрировали объекты баз MIB SNMP во всемирном дереве регистрации стандартов ISO, показанном на рис, 7.7.

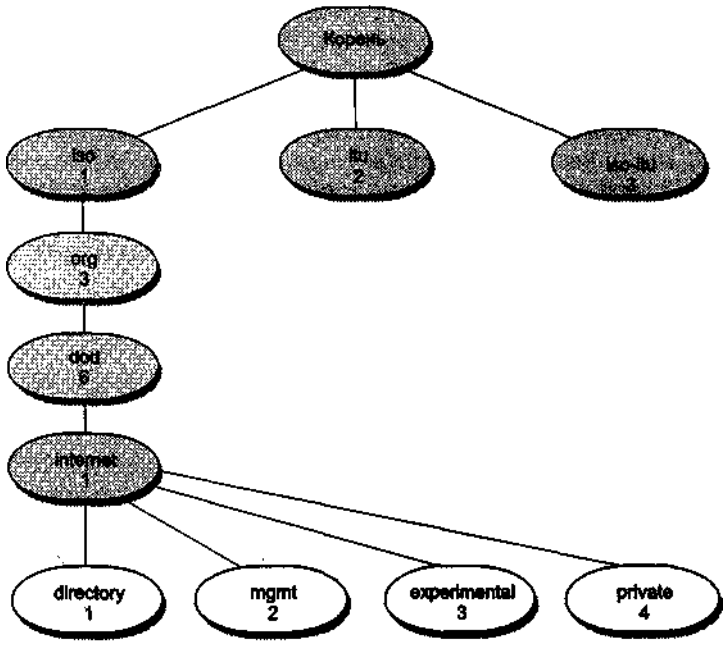


Рис. 7.7. Пространство имен объектов ISO

Как и в любых сложных системах, пространство имен объектов ISO имеет древовидную иерархическую структуру, причем на рис. 7.7 показана только его верхняя часть. От корня этого дерева отходят три ветви, соответствующие стандартам, контролируемым ISO, ITU и совместно ISO-ITU. В свою очередь, организация ISO создала ветвь для стандартов, создаваемых национальными и международными организациями (ветвь org). Стандарты Internet создавались под эгидой Министерства обброны США (Departament of Defence, DoD), поэтому стандарты MIB попали в поддерево dod-internet, а далее, естественно, в группу стандартов управления сетью – ветвь mgmt. Объекты любых стандартов, создаваемых под эгидой ISO, однозначно идентифицируются составными символьными именами, начинающимися от корня этого дерева. В сообщениях протоколов символьные имена не используются, а применяются однозначно соответствующие им составные числовые имена. Каждая ветвь дерева имен объектов нумеруется в дереве целыми числами слева направо, начиная с единицы, и эти числа и заменяют символьные имена. Поэтому полное символьное имя объекта MIB имеет вид:

iso.org.dod.internet.mgmt.mib, а полное числовое имя: 1.3.6.1.2.1.

Группа объектов private (4) зарезервирована за стандартами, создаваемыми частными компаниями, например Cisco, Hewlett-Packard и т. п. Это же дерево регистрации используется для именования классов объектов CMIP и TMN.

Соответственно, каждая группа объектов MIB-I и MIB-II также имеет кроме кратких символьных имен, приведенных выше, полные символьные имена и соответствующие им числовые имена. Например, краткое символьное имя группы System имеет полную форму iso.org.dod.internet.mgmt.mib.system, а ее соответствующее числовое имя – 1.3.6.1.2.1. Часть дерева имен ISO, включающая группы объектов MIB, показана на рис. 7.8.

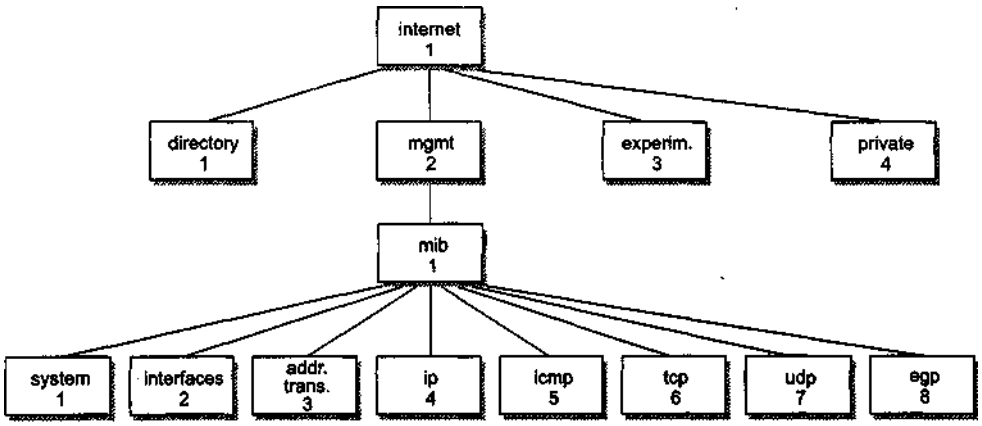


Рис. 7.8. Часть дерева имен ISO, включающая группы объектов MIB-I

Формат сообщений SNMP

Протокол SNMP обслуживает передачу данных между агентами и станцией, управляющей сетью. SNMP использует дейтаграммный транспортный протокол UDP, не обеспечивающий надежной доставки сообщений. Протокол, организующий надежную передачу дейтаграмм на основе соединений TCP, весьма загружает управляемые устройства, которые на момент разработки протокола SNMP были не очень мощные, поэтому от услуг протокола TCP решили отказаться.

SNMP часто рассматривают только как решение для управления сетями TCP/IP. Хотя SNMP чаще всего и работает над UDP (он может также работать и над TCP), он может работать и над транспортными сетевыми протоколами стека OSI – TPO, TP4, CNLS, а также над протоколами MAC-уровня. Растет поддержка протокола SNMP и в других транспортных средах. Например, фирма Novell начала поддерживать протокол SNMP с версии NetWare 3.11, а некоторые производители оборудования (например, Bay Networks) реализуют в своих устройствах передачу сообщений SNMP с помощью как IP, так и IPX.

Сообщения SNMP, в отличие от сообщений многих других коммуникационных протоколов, не имеют заголовков с фиксированными полями. В соответствии с нотацией ASN.1 сообщение SNMP состоит из произвольного количества полей, и каждое поле предваряется описателем его типа и размера.

Любое сообщение SNMP состоит из трех основных частей: версии протокола (version), идентификатора общности (community), используемого для группирования устройств, управляемых определенным менеджером, и области данных, в которой собственно и содержатся описанные выше команды протокола, имена объектов и их значения. Область данных делится на блоки данных протокола (Protocol Data Unit, PDU).

Общий формат сообщения SNMP в нотации ASN.1 выглядит следующим образом:

```
SNMP-Message ::=
 SEQUENCE {
version INTEGER {
 version-1 (0)
}.
```

community
OCTET STRING,
SNMP-PDUs
ANY
}

Область данных может содержать пять различных типов PDU, соответствующих пяти командам протокола SNMP:

SNMP-PDUs ::=
CHOICE {
get-request
GetRequest-PDU,
get-next-request
GetNextRequest-PDU.
get-response
GetResponse-PDU,
set-request
SetRequest-PDU.
trap
Trap-PDU,
}

И наконец, для каждого типа PDU имеется определение его формата. Например, формат блока GetRequest-PDU описан следующим образом:

GetRequest-PDU ::=
IMPLICIT SEQUENCE {
request-id
RequestID.
error-status
ErrorStatus.
error-Index
Errorlndex,
variable-bindings
VarBindList
}

Далее стандарт SNMP определяет соответственно формат переменных блока GetRequest-PDU. Переменная RequestID – это 4-байтовое целое число (используется для установления соответствия ответов запросам), ErrorStatus и ErrorIndex – это однобайтовые целые, которые в запросе должны быть установлены в 0. VarBindList – это список числовых имен объектов, значениями которых интересуется менеджер. В нотации ASN.1 этот список состоит из пар «имя - значение». При запросе значение переменной должно быть установлено в null.

Вот пример сообщения протокола SNMP, которое представляет собой запрос о значении объекта SysDescr (числовое имя 1.3.6.1.2.1.1.1).

30	29	02	01	00			
SEQUENCE	len=41	INTEGER	len=1	vers=0			
04	06	70	75	62	6C	69	63
String	len=6	p	u	b	i	i	c
A0	1C	02	04	05	AE	56	02
getred	len=28	INTEGER	len=4	—	requested ID	—	—
02	01	00	02	01	00		
INTEGER	len=1	Status	INTEGER	len=1	error	index	

30	0E	30	0C	06	08		
SEQUENCE	len=14	SEQUENCE	len=12	objectid	len=8		
2B	06	01	02	01	01	01	00
1,3	6	1	2	1	1	1	0
05	00						
null	len=0						

Как видно из описания, сообщение начинается с кода 30 (все коды шестнадцатеричные), который соответствует ключевому слову SEQUENCE (последовательность). Длина последовательности указывается в следующем байте (41 байт). Далее следует целое число длиной 1 байт – это версия протокола SNMP (в данном случае О, то есть SNMP v.l, а 1 означала бы SNMP v.2). Поле community имеет тип string (строка символов) длиной в 6 байт со значением public. Остальную часть сообщения составляет блок данных GetRequest-PDU. То, что это операция Get-request, говорит код АО (это значение определено в протоколе SNMP, а не в нотации ASN.1), а общая длина блока данных – 28 байт. В соответствии со структурой блока Getrequest-PDU, далее идет идентификатор запроса (он определен как целое 4-байтовое число). Затем в блоке следуют два однобайтовых целых числа статуса и индекса ошибки, которые в запросе установлены в 0. И наконец, завершает сообщение список объектов, состоящий из одной пары – имени 1.3.6.1.2.1.1.1.0 и значения null.

Спецификация RMON MIB

Новейшим добавлением к функциональным возможностям SNMP является спецификация RMON, которая обеспечивает удаленное взаимодействие с базой MIB. До появления RMON протокол SNMP не мог использоваться удаленным образом, он допускал только локальное управление устройствами. База RMON MIB обладает улучшенным набором свойств для удаленного управления, так как содержит агрегированную информацию об устройстве, не требующую передачи по сети больших объемов информации. Объекты RMON MIB включают дополнительные счетчики ошибок в пакетах, более гибкие средства анализа трендов и статистики, более мощные средства фильтрации для захвата и анализа отдельных пакетов, а также более сложные условия установления сигналов предупреждения. Агенты RMON MIB более интеллектуальны по сравнению с агентами MIB-I или MIB-II и выполняют значительную часть работы по обработке информации об устройстве, которую раньше выполняли менеджеры. Эти агенты могут располагаться внутри различных коммуникационных устройств, а также быть выполнены в виде отдельных программных модулей, работающих на универсальных персональных компьютерах и ноутбуках.

Объекту RMON присвоен номер 16 в наборе объектов MIB, а сам объект RMON объединяет 10 групп следующих объектов.

- Statistics – текущие накопленные статистические данные о характеристиках пакетов, количестве коллизий и т. п.
- History – статистические данные, сохраненные через определенные промежутки времени для последующего анализа тенденций их изменений.
- Alarms – пороговые значения статистических показателей, при превышении которых агент RMON посылает сообщение менеджеру.
- Hosts – данные о хостах сети, в том числе и о их MAC-адресах.
- HostTopN – таблица наиболее загруженных хостов сети.
- Traffic Matrix – статистика об интенсивности трафика между каждой парой хостов сети, упорядоченная в виде матрицы.
- Filter – условия фильтрации пакетов.
- Packet Capture – условия захвата пакетов.
- Event – условия регистрации и генерации событий.

Данные группы пронумерованы в указанном порядке, поэтому, например, группа Hosts имеет числовое имя 1.3.6.1.2.1.16.4.

Десятую группу составляют специальные объекты протокола Token Ring.

Всего стандарт RMON MIB определяет около 200 объектов в 10 группах, зафиксированных в двух

документах – RFC 1271 для сетей Ethernet и RFC 1513 для сетей Token Ring.

Отличительной чертой стандарта RMON MIB является его независимость от протокола сетевого уровня (в отличие от стандартов MIB-I и MIB-II, ориентированных на протоколы TCP/IP). Поэтому он удобен для гетерогенных сред, использующих различные протоколы сетевого уровня.

Рассмотрим более подробно группу Statistics, которая определяет, какую информацию о кадрах (называемых в стандарте пакетами) Ethernet может предоставить агент RMON. Группа History основана на объектах группы Statistics, так как ее объекты просто позволяют строить временные ряды для объектов группы Statistics.

В группу Statistics входят наряду с некоторыми другими следующие объекты.

- etherStatsDropEvents – общее число событий, при которых пакеты были проигнорированы агентом из-за недостатка его ресурсов. Сами пакеты при этом не обязательно были потеряны интерфейсом.
- etherStatsOctets – общее число байт (включая ошибочные пакеты), принятых из сети (исключая преамбулу н включая байты контрольной суммы).
- etherStatsPkts – общее число полученных пакетов (включая ошибочные).
- etherStatsBroadcastPkts – общее число хороших пакетов, которые были посланы по широковещательному адресу.
- etherStatsMulticastPkts – общее число хороших пакетов, полученных по мультивещательному адресу.
- etherStatsCRCAlignErrors – общее число полученных пакетов, которые имели длину (исключая преамбулу) между 64 и 1518 байт, не содержали целое число байт (alignment error) или имели неверную контрольную сумму (FCS error).
- etherStatsUndersizePkts – общее число пакетов, которые имели длину меньше, чем 64 байт, но были правильно сформированы.
- etherStatsOversizePkts – общее число полученных пакетов, которые имели длину больше, чем 1518 байт, но были тем не менее правильно сформированы.
- etherStatsFragments – общее число полученных пакетов, которые не состояли из целого числа байт или имели неверную контрольную сумму и имели к тому же длину, меньшую 64 байт.
- etherStatsJabbers – общее число полученных пакетов, которые не состояли из целого числа байт или имели неверную контрольную сумму и имели к тому же длину, большую 1518 байт.
- etherStatsCollisions – наилучшая оценка числа коллизий на данном сегменте Ethernet.
- etherStatsPkts640ctets – общее количество полученных пакетов (включая плохие) размером 64 байт.
- etherStatsPkts65to1270ctets – общее количество полученных пакетов (включая плохие) размером от 65 до 127 байт.
- etherStatsPkts128to2550ctets – общее количество полученных пакетов (включая плохие) размером от 128 до 255 байт.
- etherStatsPkts256to51lOctets – общее количество полученных пакетов (включая плохие) размером от 256 до 511 байт.
- etherStatsPkts512tol0230ctets – общее количество полученных пакетов (включая плохие) размером от 512 до 1023 байт.
- etherStatsPktsl024to15180ctets – общее количество полученных пакетов (включая плохие) размером от 1024 до 1518 байт.

Как видно из описания объектов, с помощью агента RMON, встроенного в повторитель или другое коммуникационное устройство, можно провести очень детальный анализ работы сегмента Ethernet или Fast Ethernet. Сначала можно получить данные о встречающихся в сегменте типах ошибок в кадрах, а затем целесообразно собрать с помощью группы History зависимости интенсивности этих ошибок от времени (в том числе и привязав их ко времени). После анализа временных зависимостей часто уже можно сделать некоторые предварительные выводы об источнике ошибочных кадров и на этом основании сформулировать более тонкие условия захвата кадров со специфическими признаками (задав условия в группе Filter), соответствующими выдвинутой версии. После этого можно провести еще более детальный анализ за счет изучения захваченных кадров, извлекая их из объектов группы Packet Capture.

Позже был принят стандарт RMON 2, который распространяет идеи интеллектуальной RMON MIB на протоколы верхних уровней, выполняя часть работы анализаторов протоколов.

Недостатки протокола SNMP

Протокол SNMP служит основой многих систем управления, хотя имеет несколько принципиальных недостатков, которые перечислены ниже.

- Отсутствие средств взаимноя аутентификации агентов и менеджеров. Единственным средством, которое можно было бы отнести к средствам аутентификации, является использование в сообщениях так называемой «строки сообщества» – «community String». Эта строка передается по сети в открытой форме в сообщении SNMP и служит основой для деления агентов и менеджеров на «сообщества», так что агент взаимодействует только с теми менеджерами, которые указывают в поле community string ту же символьную строку, что и строка, хранящаяся в памяти агента. Это, безусловно, не способ аутентификации, а способ структурирования агентов и менеджеров. Версия SNMP v.2 должна была ликвидировать этот недостаток, но в результате разногласий между разработчиками стандарта новые средства аутентификации хотя и появились в этой версии, но как необязательные.
- Работа через ненадежный протокол UDP (а именно так работает подавляющее большинство реализаций агентов SNMP) приводит к потерям аварийных сообщений (сообщений trap) от агентов к менеджерам, что может привести к некачественному управлению. Исправление ситуации путем перехода на надежный транспортный протокол с установлением соединений чревато потерей связи с огромным количеством встроенных агентов SNMP, имеющихся в установленном в сетях оборудовании. (Протокол CMIP изначально работает поверх надежного транспорта стека OSI и этим недостатком не страдает.)

Разработчики платформ управления стараются преодолеть эти недостатки. Например, в платформе HP OV Telecom DM TMN, являющейся платформой для разработки многоуровневых систем управления в соответствии со стандартами TMN и ISO, работает новая реализация SNMP, организующая надежный обмен сообщениями между агентами и менеджерами за счет самостоятельной организации повторных передач сообщений SNMP при их потерях.

7.2.3. Стандарты управления OSI

Модель сетевого управления OSI – OSI Management Framework – определена в документе ISO/IEC 7498-4: Basic Reference Model, Part 4, Management Framework. Она является развитием общей семиуровневой модели взаимодействия открытых систем для случая, когда одна система управляет другой.

Документ ISO/IEC 7498-4 состоит из следующих основных разделов.

- Термины и общие концепции.
- Модель управления системами.
- Информационная модель.
- Функциональные области управления системами.
- Структура стандартов управления системами.

Функциональные области управления системами уже были рассмотрены в разделе 7.1, как имеющие общее значение для любых систем управления.

Стандарты ISO в области управления использует терминологию, которая частично совпадает с терминологией систем управления SNMP, а частично от нее отличается.

Как показано на рис. 7.9, обмен управляющей информацией с использованием протокола управления (Management Protocol) происходит между субъектами приложений управления системами (Systems Management Application Entities, SMAE). Субъекты SMAE расположены на прикладном уровне семиуровневой модели OSI и являются элементами службы управления. Под субъектом в модели OSI понимается активный в данный момент элемент протокола какого-либо уровня, участвующий во взаимодействии. Примерами SMAE являются агенты и менеджеры систем управления.

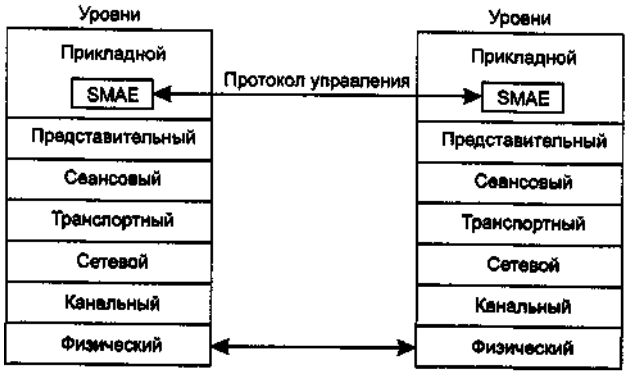


Рис. 7.9. Концепция SMAE

Агенты и менеджеры

Определения функций агентов и менеджеров в стандартах OSI достаточно хорошо согласуются с определениями систем SNMP, за некоторыми исключениями в терминологии. Сообщения, которые агент посылает менеджеру по своей инициативе, называются *уведомлениями – notifications*.

Например, если некоторый элемент сети X отказал, то менеджеру необходимо обновить свою базу данных конфигурации сети. Элемент X, который является для •системы управления *управляемым объектом* (*managed object*), может послать уведомление агенту. Элемент X может находиться в той же управляемой системе, что и агент, или может находиться в другой системе. В свою очередь агент посылает уведомление менеджеру о том, что элемент X отказал. В соответствии с этим уведомлением менеджер обновляет базу данных конфигурации.

ПРИМЕЧАНИЕ В стандартах Internet под объектом понимается отдельный атрибут базы MIB, являющейся моделью управляемого ресурса, а в стандартах ISO объект обозначает всю модель управляемого ресурса.

Менеджер не только собирает и сопоставляет данные, получаемые от агентов, на основе этих данных он может также выполнять административные функции, управляя операциями удаленных агентов.

В стандартах OSI границы между менеджерами и агентами не очень четкие. Субъект SMAE, выполняющий в одном взаимодействии роль менеджера, может в другом взаимодействии выполнять роль агента, и наоборот.

Стандарты OSI не определяют способов взаимодействия агента с управляемыми объектами. Стандарты OSI также не говорят о том, как агент взаимодействует с управляемыми объектами, которые находятся за пределами управляемой системы, то есть объектами, с которыми нужно взаимодействовать через сеть. В таких случаях может потребоваться, например, чтобы один агент запросил данные о некотором объекте от другого агента. Порядок такого рода взаимодействия также не определяется стандартами OSI.

Чтобы менеджер и агент смогли взаимодействовать, каждый должен иметь определенные знания о другом. Эти знания модель OSI называет *контекстом приложения* (*Application Context, AC*). AC описывает элементы прикладного уровня стека OSI, которые используются агентами и менеджерами.

ПРИМЕЧАНИЕ Необходимо отметить, что стандарты управления OSI в значительной степени ориентированы на стек протоколов OSI (именно стек, а не модель OSI), так же как системы управления SNMP ориентированы на роботу со стеком TCP/IP.

Прикладной уровень стека OSI включает несколько вспомогательных служб общего назначения, которые используются прикладными протоколами и пользовательскими приложениями (в том числе и приложениями управления) для автоматизации наиболее часто выполняемых действий. Это не законченные протоколы прикладного уровня, подобные протоколам ftp, telnet или NCP, с помощью которых пользователь сети может выполнить какое-то полезное действие, а вспомогательные системные функции, которые помогают разработчику прикладного протокола или приложения написать его программу компактно и эффективно. На прикладном уровне стека OSI существуют следующие

вспомогательных службы.

- ACSE (Association Control Service Element). Отвечает за установление соединений между приложениями различных систем. Соединение (сессия, сеанс) на прикладном уровне OSI носит название *ассоциации*. Ассоциации бывают индивидуальными и групповыми (shared).
- RTSE (Reliable Transfer Service Element). Занимается поддержкой восстановления диалога, вызванного разрывом нижележащих коммуникационных служб, в рамках ассоциации.
- ROSE (Remote Operations Service Element). Организует выполнение программных функций на удаленных машинах (аналог службы вызова *удаленных процедур, RPC*).

Протокол CMIP, используемый в стандартах OSI для взаимодействия между менеджерами и агентами, а также программные реализации менеджеров и агентов широко пользуются услугами данных вспомогательных служб, в особенности службы ROSE для вызова удаленных процедур.

Управление системами, управление уровнем и операции уровня

Основная модель управления OSI включает: управление системами, управление N-уровнем и операции N-уровня. Это разбиение на три области сделано для того, чтобы учесть все возможные ситуации, возникающие при управлении.

Управление системами имеет дело с управляемыми объектами на всех семи уровнях OSI, включая прикладной уровень. Оно основано на надежной передаче с установлением соединения управляющей информации между конечными системами. Необходимо подчеркнуть, что модель управления OSI не разрешает использования служб без установления соединения.

Управление N-уровнем ограничено управляемыми объектами какого-то определенного уровня семиуровневой модели. Протокол управления использует при этом коммуникационные протоколы нижележащих уровней. Управление N-уровнем полезно, когда нет возможности использовать все семь уровней OSI. В этом случае допускается пользоваться протоколом управления N-уровня, который строго предназначен для данного уровня. Примерами уровневого протокола управления являются протоколы управления для локальных сетей, разработанные институтом IEEE (SMT технологии FDDI), которые ограничены уровнями 1 и 2.

Наконец, *операции N-уровня* сводятся к мониторингу и управлению на основе управляющей информации, содержащейся в коммуникационных протоколах только данного уровня. Например, данные мониторинга сети, содержащиеся в кадрах STM-n технологии SDH, относятся к операциям N-уровня, а именно физического уровня.

Стандарты на управление N-уровнем и операции N-уровня не входят в набор стандартов управления OSI. Стандарты OSI рассматривают только управление системами с помощью полного семиуровневого стека.

Основная модель управления системами подразумевает выполнение управляющих операций и передачу уведомлений между одноранговыми системами, что означает необязательность жесткого распределения ролей на управляющие и управляемые системы. Эта модель облегчает реализацию распределенных аспектов управления. С другой стороны, допускается реализация одноранговых систем как управляющих и управляемых.

Информационная модель управления

Управляемый объект – это представление OSI о ресурсе в целях управления. Ресурс может быть описан как управляемый объект. Конкретный управляемый объект – это экземпляр (instance) некоторого класса управляемых объектов. Модель управления OSI широко использует объектно-ориентированный подход. Класс управляемых объектов – это набор свойств, которые могут быть обязательными или условными. С помощью описания одного класса управляемых объектов, например коммутаторов, можно создать другой класс управляемых объектов, например коммутаторов, поддерживающих технику VLAN, унаследовав все свойства класса коммутаторов, но добавив новые атрибуты.

Для управления ресурсами менеджер и агент должны быть осведомлены о деталях этих ресурсов. Детализация представления управляемых объектов, которые требуются для выполнения функций управления, хранится в репозитории, известном как Management Information Base (MIB). Базы MIB OSI хранят не только описания классов управляемых объектов, но и характеристики сети и ее элементов. Базы MIB содержат характеристики каждой части управляемого оборудования и ресурсов. MIB также

включает описание действий, которые могут выполняться на основе собранных данных или же вызываемые внешними командами. Базы MIB позволяют внешним системам опрашивать, изменять, создавать и удалять управляемые объекты (реальные ресурсы сети при этом, естественно, продолжают работать). Протокол CMIP и локальные интерфейсы управления обеспечивают доступ к этим возможностям.

MIB – это концептуальная модель, и она не имеет никакой связи со способом физического или логического хранения данных в ресурсе. Стандарты не определяют аспекты собственно хранения данных. Протоколы OSI определяют синтаксис информации, хранящейся в MIB, и семантику обмена данными.

Управляющие знания и деревья знаний

Крупная система управления обычно состоит из большого количества агентов и менеджеров. Для организации автоматического взаимодействия между менеджерами и агентами необходимо каким-то образом задать данные, содержащие характеристики агентов и менеджеров. Менеджеру необходимо знать о том, какие агенты работают в системе управления, их имена и сетевые адреса, поддерживаемые ими классы управляемых объектов и т. п. Агенту также необходима аналогичная информация о менеджерах, так как ему нужно отправлять по своей инициативе уведомления и отвечать на запросы менеджеров.

Такие данные называются в модели OSI *разделяемыми управляющими знаниями* (*shared management knowledge*) между менеджером и агентом. (В системах SNMP организация этих данных не стандартизована, и в каждой конкретной системе управления эти данные хранятся в индивидуальной форме).

Разделяемые управляющие знания должны быть известны до установления ассоциации между агентом и менеджером. Обычно они хранятся в каком-либо файле или распределенной базе данных и запрашиваются каждый раз, когда устанавливается ассоциация. Во время установления ассоциации происходит обмен разделяемыми управляющими знаниями.

В OSI стандартизуются различные аспекты организации управляющих знаний и доступа к ним. Следование объектно-ориентированному подходу обусловило использование для хранения этих знаний специальных системных объектов.

Стандарт ISO 10164-16.2 определяет модель объектов управляющих знаний и классы таких объектов. Кроме того, определены функции работы с управляющими знаниями.

Имеются три типа управляющих знаний и, соответственно, три типа объектов, которые описывают эти знания.

- *Знания репертуара* (*Repertoire Knowledge*) описывают'возможности управляемой системы, включающие перечень поддерживаемых классов управляемых объектов, поддерживаемые функции управления и именования. Знания репертуара .помогают менеджеру идентифицировать возможности управляемых систем без доступа к ним.
- *Знания определений* (*Definition Knowledge*) включают формальные описания классов управляемых объектов, категории тестов, классов взаимосвязей и определения управляющей информации, понимаемой управляемой системой.
- *Знания об экземплярах* (*Instance Knowledge*) обеспечивают информацию о конкретных экземплярах управляемых объектов, имеющихся в управляемой системе.

Использование древовидных баз данных для хранения управляющих знаний

В системе управления знания о поддерживаемых классах объектов и о порожденных экземплярах объектов должны храниться в какой-либо форме, удобной для предоставления модулям системы управления доступа к этой информации. Архитектура управления OSI предусматривает несколько схем базы данных об управляемых объектах и их классах. Эти схемы обычно называют деревьями из-за иерархической организации информации. Существуют следующие деревья.

- *Дерево наследования* (*Inheritance Tree*), называемое также деревом регистрации. Описывает отношения между базовыми и производными классами. Подчиненный класс наследует все характеристики суперкласса и дополняет их специфическими расширениями (дополнительными атрибутами, поведениями и действиями). Классы объектов OSI

регистрируются в том же дереве, что и объекты MIB Internet. Дерево наследования может быть глобальным, то есть начинаться с корня, представляющего весь мир, или локальным, имеющим корень, соответствующий верхнему уровню объектов данной организации или сети. Все управляемые объекты OSI должны быть зарегистрированы в глобальном дереве ISO (в котором зарегистрированы объекты MIB-I, MIB-II, RMON MIB стандарта SNMP). Объекты, представляющие международные стандарты, регистрируются в международной ветви дерева, а частные модели, разработанные производителями систем управления, регистрируются в ветвях дерева, начинающихся с ветви private.

- *Дерево включений* (*Containment Tree*). Описывает отношения включения управляемых объектов реальной системы.

ПРИМЕЧАНИЕ Между деревом наследования и деревом включений нет прямой связи. Например, в дереве включений объект «корпоративный концентратор» может включать объекты «интерфейс Ethernet» и «модуль удаленного доступа», которые представляют модели реальных модулей, установленных в слоты корпоративного концентратора. В то же время в дереве наследования класс объектов «интерфейсы Ethernet» подчинен классу объектов «интерфейсы», а класс объектов «модуль удаленного доступа» подчинен классу «коммуникационное оборудование третьего уровня», на основании которого он порожден.

- *Дерево имен* (*naming tree*) определяет способ именования объектов в системе управления. Объекты OSI могут иметь имена нескольких типов: относительное отличительное имя (Relative Distinguished Name, RDN), отличительное имя (Distinguished Name, DN), иногда называемое полным отличительным именем (Full Distinguished Name, FDN), и локальное отличительное имя (Local Distinguished Name, LDN). Эти имена связаны с деревом включений, так как определяют имена объектов относительно включающих их объектов. Относительное имя, RDN, соответствует короткому имени, которое однозначно определяет объект среди множества других объектов, подчиненных тому же родительскому объекту. Например, имя interface_a является RDN-именем, уникально характеризующим объект среди объектов, подчиненных объекту node_a. Полное отличительное имя FDN представляет собой последовательность RDN-имен, начинающуюся в вершине глобального дерева имен, то есть дерева, описывающего некоторую глобальную сеть. Наконец, локальное отличительное имя – это последовательность RDN-имен, но начинающаяся не в глобальном корне, а в корне дерева имен локальной системы управления, отвечающей за часть глобального дерева имен данной сети.

Дерево имен обычно совмещается с деревом включений.

Пример дерева включений показан на рис. 7.10. Экземпляр управляемого объекта класса corp-conc (корпоративный концентратор) имеет имя B1, а также атрибут max-slotes, описывающий максимальное количество слотов данного класса концентраторов, равный в данном случае 14. В этот объект включено ряд других объектов: объекты класса repeator, switch и RAS, которые в свою очередь включают объекты типа interface, описывающие порты модулей концентратора.

Имя класса объекта позволяет обратиться к описанию класса и узнать полный список атрибутов этого класса или ссылку на родительский класс, у которого наследуются все или некоторые атрибуты. Имя экземпляра объекта дает информацию о принадлежности конкретного модуля или интерфейса определенному коммуникационному устройству, например имя B1.E1.P2 определяет второй порт модуля повторителя E1, входящего в состав корпоративного концентратора B1.

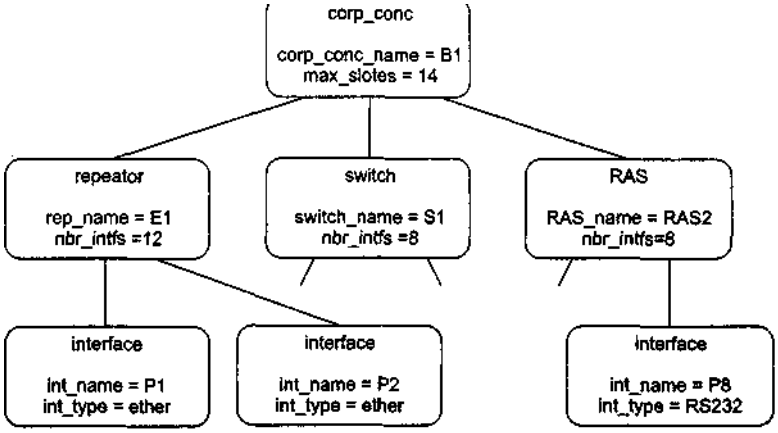


Рис. 7.10. Пример дерево включений

Правила определения управляемых объектов

Классы управляемых объектов OSI должны определяться в соответствии со стандартом GDMO (Guidelines for the Definition of Managed Objects – Правила определения управляемых объектов), являющимся стандартом ISO 10165-4.

В GDMO определяется несколько *шаблонов* (*templates*) – пустых форм, которые заполняются для описания определенного класса управляемых объектов. В шаблоне класса перечисляются комплекты свойств (PACKAGES), которые составляют класс. Шаблон комплекта свойств PACKAGE перечисляет Атрибуты, Группы атрибутов, Действия, Поведение и Уведомления, то есть свойства, сгруппированные для удобства описания класса объектов. Отношения наследования между классами описываются с помощью шаблона Связывание имен.

Атрибуты и Группы атрибутов определяют параметры объекта, которые можно читать и узнавать из них о состоянии объекта. Свойства Действия описывают возможные управляющие воздействия, которые допускается применять к данному объекту – например, мультиплексировать несколько входных потоков в один выходной. Свойство Поведение описывает реакцию объекта на примененное к нему действие. Уведомления составляют набор сообщений, которые генерирует объект по своей инициативе.

Заполненные шаблоны GDMO определяют представление класса и его свойств. Заполнение шаблонов выполняется в соответствии с нотацией ASN.1. В отличие от стандартов SNMP, использующих только подмножество типов данных ASN.1, в GDMO и CMIP применяется полная версия ASN.1.

На основании правил GDMO определено несколько международных стандартов на классы управляемых объектов. Документы Definition of Management Information (DMI, ISO/IEC 10165-2:1991) и Generic Management Information (GMI, ISO/IEC CD 10165-5:1992) являются первыми определениями MIB на основе окончательной версии GDMO. Эти MIB могут рассматриваться как ISO-эквивалент для Internet MIB II, так как они создают основу для построения более специфических MIB. Например, DMI определяет класс объектов, называемый Top, который является верхним суперклассом, – он содержит атрибуты, которые наследуются всеми другими классами управляемых объектов. Определены также классы объектов System и Network, занимающие верхние позиции в дереве наследования, так что любой агент должен понимать их атрибуты.

В 1992 году была завершена работа и над более специфическими классами объектов – объектами сетевого и транспортного уровней (ISO/IEC 10737-1 и ISO/ IEC 10733).

Сегодня многие организации работают над созданием классов объектов на основе GDMO. Это и международные организации по стандартизации – ISO, ITU-T, ANSI, ETSI, X/Open, и организации, разрабатывающие платформы и инструментальные средства для систем управления, такие как SunSoft, Hewlett-Packard, Vertel, ISR Global. Для телекоммуникационных сетей в рамках архитектуры TMN разработан стандарт M.3100, который описывает ряд специфических для телекоммуникационных сетей классов объектов.

Описания классов управляемых объектов OSI регистрируются как в частных ветвях дерева ISO – ветвях компаний Sun, Hewlett-Packard, IBM и т. п., так и в публичных ветвях, контролируемых ISO или другими международными органами стандартизации.

В отсутствие одной регистрирующей организации, такой как IETF Internet, использование классов объектов OSI представляет собой непростую задачу.

Протокол CMIP и услуги CMIS

Доступ к управляющей информации, хранящейся в управляемых объектах, обеспечивается с помощью элемента системы управления, называемого службой CMSIE (Common Management Information Service Element). Служба CMSIE построена в архитектуре распределенного приложения, где часть функций выполняет менеджер, а часть – агент. Взаимодействие между менеджером и агентом осуществляется по протоколу CMIP. Услуги, предоставляемые службой CMSIE, называются услугами CMIS (Common Management Information Services).

Протокол CMIP и услуги CMIS определены в стандартах X.710 и X.711 ITU-T.

Услуги CMIS разделяются на две группы – услуги, инициируемые менеджером (запросы), и услуги, инициируемые агентом (уведомления).

Услуги, инициируемые менеджером, включают следующие операции:

- M-CREATE инструктирует агента о необходимости создать новый экземпляр объекта определенного класса или новый атрибут внутри экземпляра объекта;
- M-DELETE инструктирует агента о необходимости удаления некоторого экземпляра объекта определенного класса или атрибута внутри экземпляра объекта;
- M-GET инструктирует агента о возвращении значения некоторого атрибута определенного экземпляра объекта;
- M-SET инструктирует агента об изменении значения некоторого атрибута определенного экземпляра объекта;
- M-ACTION инструктирует агента о необходимости выполнения определенного действия над одним или несколькими экземплярами объектов.

Агент инициирует только одну операцию:

M-EVENT_REPORT – отправка уведомления менеджеру.

Для реализации своих услуг служба CMISE должна использовать службы прикладного уровня стека OSI – ACSE, ROSE.

Отличие услуг CMIS от аналогичных услуг SNMP состоит в большей гибкости. Если запросы GET и SET протокола SNMP применимы только к одному атрибуту одного объекта, то запросы M-GET, M-SET, M-ACTION и M-DELETE могут применяться к более чем одному объекту. Для этого стандарты CMIP/CMIS вводят такие понятия, как *обзор* (*scoping*), *фильтрация* (*filtering*) и *синхронизация* (*synchronization*).

Обзор

Запрос CMISE может использовать обзор, чтобы опросить одновременно несколько объектов. Вводятся четыре уровня обзора:

- базовый объект, определенный своим отличительным именем FDN;
- объекты, расположенные на n-м уровне подчинения относительно базового (сам базовый объект находится на уровне 0) в дереве включения;
- базовый объект и все объекты, расположенные на подчиненных ему уровнях до n-го (включительно) в дереве включения;
- поддерево – базовый объект и все ему подчиненные в дереве включения.

Фильтрация

Фильтрация заключается в применении булевого выражения к запросу менеджера. Запрос применяется только к тем объектам и их атрибутам, для которых данное булево выражение верно. Булевы выражения могут включать операторы отношения =, >=, <=, >, < или определенные атрибуты. Возможно построение сложных фильтров на основе объединения нескольких фильтров в один составной.

Синхронизация

При выполнении запросов к нескольким объектам используется одна из двух схем синхронизации: атомарная или «по возможности». При атомарной схеме запрос выполняется только в том случае, когда все объекты, попадающие в область действия обзора или фильтра, могут успешно выполнить данный запрос. Синхронизация «по возможности» подразумевает передачу запроса всем объектам, к которым запрос относится. Операция завершается при выполнении запроса любым количеством объектов.

Протокол CMIP представляет собой набор операций, прямо соответствующих услугам CMIS. Таким образом, в протоколе CMIP определены операции M-GET, M-SET, M-CREATE и т. д. Для каждой операции определен формат блока данных, переносимых по сети от менеджера агенту, и наоборот.

Формат протокольных блоков данных CMIP описывается нотацией ASN.1 и имеет гораздо более сложную структуру, чем блоки SNMP. Например, блок данных операции M-GET имеет поля для задания имен атрибутов, значения которых запрашивает менеджер, а также поля задания параметров обзора и фильтрации, определяющих множество экземпляров объектов, на которые будет воздействовать данный запрос. Имеются также поля для задания параметров прав доступа к объекту.

Сравнение протоколов SNMP и CMIP

- Применение протокола SNMP позволяет строить как простые, так и сложные системы управления, а применение протокола CMIP определяет некоторый, достаточно высокий начальный уровень сложности системы управления, так как для его работы необходимо реализовать ряд вспомогательных служб, объектов и баз данных объектов.
- Агенты CMIP выполняют, как правило, более сложные функции, чем агенты SNMP. Из-за этого операции, которые менеджеру можно выполнить над агентом SNMP, носят атомарный характер, что приводит к многочисленным обменам между менеджером и агентом.
- Уведомления (traps) агента SNMP посылаются менеджеру без ожидания подтверждения, что может привести к тому, что важные сетевые проблемы останутся незамеченными, так как соответствующее уведомление окажется потерянным, в то время как уведомления агента CMIP всегда передаются с помощью надежного транспортного протокола и в случае потери будут переданы повторно.
- Решение части проблем SNMP может быть достигнуто за счет применения более интеллектуальных MIB (к которым относится RMON MIB), но для многих устройств и ситуаций таких MIB нет (или нет стандарта, или нет соответствующей MIB в управляемом оборудовании).
- Протокол CMIP рассчитан на интеллектуальных агентов, которые могут по одной простой команде от менеджера выполнить сложную последовательность действий.
- Протокол CMIP существенно лучше масштабируется, так как может воздействовать сразу на несколько объектов, а ответы от агентов проходят через фильтры, которые ограничивают передачу управляющей информации только определенным агентам и менеджерам.

Выводы

- Существуют два популярных семейства стандартов систем управления: стандарты Internet, описывающие системы управления на основе протокола SNMP, и международные стандарты управления открытых систем (OSI), разработанные ISO и ITU-T, опирающиеся на протокол управления CMIP. Семейство стандартов Internet специфицирует минимум аспектов и элементов системы управления, а семейство стандартов ISO/ITU-T – максимум.
- Системы управления SNMP основаны на следующих концепциях, ориентированных на минимальную загрузку управляемых устройств:
- агент выполняет самые простые функции и работает в основном по инициативе менеджера;
- система управления состоит из одного менеджера, который периодически опрашивает всех агентов;
- протокол взаимодействия между агентом и менеджером SNMP опирается на простой ненадежный транспортный протокол UDP (для разгрузки управляемого устройства) и

использует два основных типа команд – get для получения данных от агента и set для передачи управляющих воздействий агенту;

- агент может послать данные менеджеру по своей инициативе с помощью команды trap, но число ситуаций, в которых он применяет эту команду, очень невелико.
- Базы управляющей информации MIB в стандартах Internet состоят из дерева атрибутов, называемых объектами и группами объектов.
- Первые MIB Internet были ориентированы на управление маршрутизаторами: MIB-I – только контроль, MIB-II – контроль и управление. Более поздняя разработка RMON MIB была направлена на создание интеллектуальных агентов, контролирующих нижний уровень, – интерфейсы Ethernet и Token Ring. Имена объектов стандартных MIB Internet зарегистрированы в дереве регистрации имен стандартов ISO.
- Стандарты ISO/ITU-T для представления управляемых устройств используют объектно-ориентированный подход. Определено несколько суперклассов обобщенных управляемых объектов, на основании которых путем наследования свойств должны создаваться более специфические классы объектов.
- Для описания управляемых объектов OSI разработаны правила GDMO, основанные на формах определенной структуры, заполняемых с помощью языка ASN.1.
- Для представления знаний об управляемых объектах, агентах и менеджерах системы управления OSI используется три древовидные базы данных: дерево наследования, которое описывает отношения наследования между классами объектов, дерево включения, которое описывает отношения соподчинения между конкретными элементами системы управления, и дерево имен, которое определяет иерархические имена объектов в системе.
- Протокол CMIP, который является протоколом взаимодействия между агентами и менеджерами системы управления OSI, позволяет с помощью одной команды воздействовать сразу на группу агентов, применив такие опции, как обзор и фильтрация.

7.3. Мониторинг и анализ локальных сетей

Постоянный контроль за работой локальной сети, составляющей основу любой корпоративной сети, необходим для поддержания ее в работоспособном состоянии. Контроль – это необходимый первый этап, который должен выполняться при управлении сетью. Ввиду важности этой функции ее часто отделяют от других функций систем управления и реализуют специальными средствами. Такое разделение функций контроля и собственно управления полезно для небольших и средних сетей, для которых установка интегрированной системы управления экономически нецелесообразна. Использование автономных средств контроля помогает администратору сети выявить проблемные участки и устройства сети, а их отключение или реконфигурацию он может выполнять в этом случае вручную. Процесс контроля работы сети обычно делят на два этапа – мониторинг и анализ. На этапе мониторинга выполняется более простая процедура – процедура сбора первичных данных о работе сети: статистики о количестве циркулирующих в сети кадров и пакетов различных протоколов, состоянии портов концентраторов, коммутаторов и маршрутизаторов и т. п.

Далее выполняется этап анализа, под которым понимается более сложный и интеллектуальный процесс осмысления собранной на этапе мониторинга информации, сопоставления ее с данными, полученными ранее, и выработки предположений о возможных причинах замедленной или ненадежной работы сети.

Задачи мониторинга решаются программными и аппаратными измерителями, тестерами, сетевыми анализаторами, встроенными средствами мониторинга коммуникационных устройств, а также агентами систем управления. Задача анализа требует более активного участия человека и использования таких сложных средств, как экспертные системы, аккумулирующие практический опыт многих сетевых специалистов.

7.3.1. Классификация средств мониторинга и анализа

Все многообразие средств, применяемых для анализа и диагностики вычислительных сетей, можно разделить на несколько крупных классов.

- Агенты систем управления, поддерживающие функции одной из стандартных MIB и поставляющие информацию по протоколу SNMP или CMIP. Для получения данных от агентов обычно требуется наличие системы управления, собирающей данные от агентов в автоматическом режиме.
- Встроенные системы диагностики и управления (Embedded systems). Эти системы выполняются в виде программно-аппаратных модулей, устанавливаемых в коммуникационное оборудование, а также в виде программных модулей, встроенных в операционные системы. Они выполяют функции диагностики и управления только одним устройством, и в этом их основное отличие от централизованных систем управления. Примером средств этого класса может служить модуль управления многосегментным повторителем Ethernet, реализующий функции автосегментации портов при обнаружении неисправностей, приписывания портов внутренним сегментам повторителя и некоторые другие. Как правило, встроенные модули управления «по совместительству» выполняют роль SNMP-агентов, поставляющих данные о состоянии устройства для систем управления.
- Анализаторы протоколов (Protocol analyzers). Представляют собой программные или аппаратно-программные системы, которые ограничиваются в отличие от систем управления лишь функциями мониторинга и анализа трафика в сетях. Хороший анализатор протоколов может захватывать и декодировать пакеты большого количества протоколов, применяемых в сетях, – обычно несколько десятков. Анализаторы протоколов позволяют установить некоторые логические условия для захвата отдельных пакетов и выполняют полное декодирование захваченных пакетов, то есть показывают в удобной для специалиста форме вложенность пакетов протоколов разных уровней друг в друга с расшифровкой содержания отдельных полей каждого пакета.
- Экспертные системы. Этот вид систем аккумулирует знания технических специалистов о выявлении причин аномальной работы сетей и возможных способах приведения сети в работоспособное состояние. Экспертные системы часто реализуются в виде отдельных подсистем различных средств мониторинга и анализа сетей: систем управления сетями, анализаторов протоколов, сетевых анализаторов. Простейшим вариантом экспертной системы является контекстно-зависимая система помощи. Более сложные экспертные системы представляют собой, так называемые базы знаний, обладающие элементами искусственного интеллекта. Примерами таких систем являются экспертные системы, встроенные в систему управления Spectrum компании Cabletron и анализатора протоколов Sniffer компании Network General. Работа экспертных систем состоит в анализе большого числа событий для выдачи пользователю краткого диагноза о причине неисправности сети.
- Оборудование для диагностики и сертификации кабельных систем. Условно это оборудование можно поделить на четыре основные группы: сетевые мониторы, приборы для сертификации кабельных систем, кабельные сканеры и тестеры.
- Сетевые мониторы (называемые также сетевыми анализаторами) предназначены для тестирование кабелей различных категорий. Сетевые мониторы собирают также данные о статистических показателях трафика – средней интенсивности общего трафика сети, средней интенсивности потока пакетов с определенным типом ошибки и т. п. Эти устройства являются наиболее интеллектуальными устройствами из всех четырех групп устройств данного класса, так как работают не только на физическом, но и на канальном, а иногда и на сетевом уровнях.
- Устройства для сертификации кабельных систем выполняют сертификацию в соответствии с требованиями одного из международных стандартов на кабельные системы.
- Кабельные сканеры используются для диагностики медных кабельных систем.
- Тестеры предназначены для проверки кабелей на отсутствие физического разрыва.
- Многофункциональные портативные устройства анализа и диагностики. В связи с развитием технологии больших интегральных схем появилась возможность производства портативных приборов, которые совмещали бы функции нескольких устройств: кабельных сканеров, сетевых мониторов и анализаторов протоколов.

7.3.2. Анализаторы протоколов

Анализатор протоколов представляет собой либо специализированное устройство, либо персональный компьютер, обычно переносной, класса Notebook, оснащенный специальной сетевой картой и соответствующим программным обеспечением. Применяемые сетевая карта и программное обеспечение должны соответствовать технологии сети (Ethernet, Token Ring, FDDI, Fast Ethernet). Анализатор подключается к сети точно так же, как и обычный узел. Отличие состоит в том, что анализатор может принимать все пакеты данных, передаваемые по сети, в то время как обычная станция – только адресованные ей. Для этого сетевой адаптер анализатора протоколов переводится в режим *«беспорядочного» захвата – promiscuous mode.*

Программное обеспечение анализатора состоит из ядра, поддерживающего работу сетевого адаптера и программного обеспечения, декодирующего протокол канального уровня, с которым работает сетевой адаптер, а также наиболее распространенные протоколы верхних уровней, например IP, TCP, ftp, telnet, HTTP, IPX, NCP, NetBEUI, DECnet и т. п. В состав некоторых анализаторов может входить также экспертная система, которая позволяет выдавать пользователю рекомендации о том, какие эксперименты следует проводить в данной ситуации, что могут означать те или иные результаты измерений, как устранить некоторые виды неисправности сети.

Анализаторы протоколов имеют некоторые общие свойства.

- Возможность (кроме захвата пакетов) измерения среднестатистических показателей трафика в сегменте локальной сети, в котором установлен сетевой адаптер анализатора. Обычно измеряется коэффициент использования сегмента, матрицы перекрестного трафика узлов, количество хороших и плохих кадров, прошедших через сегмент.
- Возможность работы с несколькими агентами, поставляющими захваченные пакеты из разных сегментов локальной сети. Эти агенты чаще всего взаимодействуют с анализатором протоколов по собственному протоколу прикладного уровня, отличному от SNMP или CMIP.
- Наличие развитого графического интерфейса, позволяющего представить результаты декодирования пакетов с разной степенью детализации.
- Фильтрация захватываемых и отображаемых пакетов. Условия фильтрации задаются в зависимости от значения адресов назначения и источника, типа протокола или значения определенных полей пакета. Пакет либо игнорируется, либо записывается в буфер захвата. Использование фильтров значительно ускоряет и упрощает анализ, так как исключает захват или просмотр ненужных в данный момент пакетов.
- Использование триггеров. Триггеры – это задаваемые администратором некоторые условия начала и прекращения процесса захвата данных из сети. Такими условиями могут быть: время суток, продолжительность процесса захвата, появление определенных значений в кадрах данных. Триггеры могут использоваться совместно с фильтрами, позволяя более детально и тонко проводить анализ, а также продуктивнее расходовать ограниченный объем буфера захвата.
- Многоканальность. Некоторые анализаторы протоколов позволяют проводить одновременную запись пакетов от нескольких сетевых адаптеров, что удобно для сопоставления процессов, происходящих в разных сегментах сети. Возможности анализа проблем сети на физическом уровне у анализаторов протоколов минимальные, поскольку всю информацию они получают от стандартных сетевых адаптеров. Поэтому они передают и обобщают информацию физического уровня, которую сообщает им сетевой адаптер, а она во многом зависит от типа сетевого адаптера. Некоторые сетевые адаптеры сообщают более детальные данные об ошибках кадров и интенсивности коллизий в сегменте, а некоторые вообще не передают такую информацию верхним уровням протоколов, на которых работает анализатор протоколов.

С распространением серверов Windows NT все более популярным становится анализатор Network Monitor фирмы Microsoft. Он является частью сервера управления системой SMS, а также входит в стандартную поставку Windows NT Server, начиная с версии 4.0 (версия с усеченными функциями). Network Monitor в версии SMS является многоканальным анализатором протоколов, поскольку может получать данные от нескольких агентов Network Monitor Agent, работающих в среде Windows NT Server, однако в каждый момент времени анализатор может работать только с одним агентом, так что сопоставить данные разных каналов с его помощью не удастся. Network Monitor поддерживает фильтры захвата (достаточно простые) и дисплейные фильтры, отображающие нужные кадры после захвата (более сложные). Экспертной системой Network Monitor не располагает.

7.3.3. Сетевые анализаторы

Сетевые анализаторы представляют собой эталонные измерительные приборы для диагностики и сертификации кабелей и кабельных систем. Они могут с высокой точностью измерить все электрические параметры кабельных систем, а также работают на более высоких уровнях стека протоколов. Сетевые анализаторы генерируют синусоидальные сигналы в широком диапазоне частот, что позволяет измерять на приемной паре амплитудно-частотную характеристику и перекрестные наводки, затухание и суммарное затухание. Сетевой анализатор представляет собой лабораторный прибор больших размеров, достаточно сложный в обращении.

Многие производители дополняют сетевые анализаторы функциями статистического анализа трафика – коэффициента использования сегмента, уровня широковещательного трафика, процента ошибочных кадров, а также функциями анализатора протоколов, которые обеспечивают захват пакетов разных протоколов в соответствии с условиями фильтров и декодирование пакетов.

7.3.4. Кабельные сканеры и тестеры

Основное назначение *кабельных сканеров* – измерение электрических и механических параметров кабелей: длины кабеля, параметра NEXT, затухания, импеданса, схемы разводки пар проводников, уровня электрических шумов в кабеле. Точность измерений, произведенных этими устройствами, ниже, чем у сетевых анализаторов, но вполне достаточна для оценки соответствия кабеля стандарту.

Для определения местоположения неисправности кабельной системы (обрыва, короткого замыкания, неправильно установленного разъема и т. д.) используется метод «отраженного импульса» (Time Domain Reflectometry, TDR). Суть этого метода состоит в том, что сканер излучает в кабель короткий электрический импульс и измеряет время задержки до прихода отраженного сигнала. По полярности отраженного импульса определяется характер повреждения кабеля (короткое замыкание или обрыв). В правильно установленном и подключенном кабеле отраженный импульс почти отсутствует.

Точность измерения расстояния зависит от того, насколько точно известна скорость распространения электромагнитных волн в кабеле. В различных кабелях она будет разной. Скорость распространения электромагнитных волн в кабеле (Nominal Velocity of Propagation, NVP) обычно задается в процентах от скорости света в вакууме. Современные сканеры содержат в себе электронную таблицу данных о NVP для всех основных типов кабелей, что дает возможность пользователю устанавливать эти параметры самостоятельно после предварительной калибровки.

Кабельные сканеры – это портативные приборы, которые обслуживающий персонал может постоянно носить с собой.

Кабельные тестеры – наиболее простые и дешевые приборы для диагностики кабеля. Они позволяют определить непрерывность кабеля, однако, в отличие от кабельных сканеров, не дают ответа на вопрос о том, в каком месте произошел сбой.

7.3.5. Многофункциональные портативные приборы мониторинга

В последнее время начали выпускаться многофункциональные портативные приборы, которые объединяют в себе возможности кабельных сканеров, анализаторов протоколов и даже некоторые функции систем управления, сохраняя в то же время такое важное свойство, как портативность. Многофункциональные приборы мониторинга имеют специализированный физический интерфейс, позволяющий выявлять проблемы и тестировать кабели на физическом уровне, который дополняется микропроцессором с программным обеспечением для выполнения высокоуровневых функций.

Рассмотрим типичный набор функций и свойств такого прибора, который оказывается очень полезным для диагностики причин разнообразных неполадок в сети, происходящих на всех уровнях стека протоколов, от физического до прикладного.

Интерфейс пользователя

Прибор обычно предоставляет пользователю удобный и интуитивно понятный интерфейс, основанный на системе меню. Графический интерфейс пользователя реализован на многострочном

жидкокристаллическом дисплее и индикаторах состояния на светодиодах, извещающих пользователя о наиболее общих проблемах наблюдаемых сетей. Имеется обширный файл подсказок оператору с уровневым доступом в соответствии с контекстом. Информация о состоянии сети представляется таким образом, что пользователи любой квалификации могут ее быстро понять.

Функции проверки аппаратуры и кабелей

Многофункциональные приборы сочетают наиболее часто используемые на практике функции кабельных сканеров с рядом новых возможностей тестирования.

Сканирование кабеля

Функция позволяет измерять длину кабеля, расстояние до самого серьезного дефекта и распределение импеданса по длине кабеля. При проверке неэкранированной витой пары могут быть выявлены следующие ошибки: расщепленная пара, обрывы, короткое замыкание и другие виды нарушения соединения.

Для сетей Ethernet на коаксиальном кабеле эти проверки могут быть осуществлены на работающей сети.

Функция определения распределения кабельных жил

Осуществляет проверку правильности подсоединения жил, наличие промежуточных разрывов и перемычек на витых парах. На дисплей выводится перечень связанных между собой контактных групп.

Функция определения карты кабелей

Используется для составления карты основных кабелей и кабелей, ответвляющихся от центрального помещения.

Автоматическая проверка кабеля

В зависимости от конфигурации возможно определить длину, импеданс, схему подключения жил, затухание и параметр NEXT на частоте до 100 МГц.

Автоматическая проверка выполняется для:

- коаксиальных кабелей;
- экранированной витой пары с импедансом 150 Ом;
- неэкранированной витой пары с сопротивлением 100 Ом.

Целостность цепи при проверке постоянным током

Эта функция используется при проверке коаксиальных кабелей для верификации правильности используемых терминаторов и их установки.

Определение номинальной скорости распространения

Функция вычисляет номинальную скорость распространения (Nominal Velocity of Propagation, NVP) по кабелю известной длины и дополнительно сохраняет полученные результаты в файле для определяемого пользователем типа кабеля (User Defined cable type) или стандартного кабеля.

Комплексная автоматическая проверка пары «сетевой адаптер-концентратор»

Этот комплексный тест позволяет последовательно подключить прибор между конечным узлом сети и концентратором. Тест дает возможность автоматически определить местонахождение источника неисправности – кабель, концентратор, сетевой адаптер или программное обеспечение станции.

Автоматическая проверка сетевых адаптеров

Проверяет правильность функционирования вновь установленных или «подозрительных» сетевых адаптеров. Для сетей Ethernet по итогам проверки сообщаются: MAC-адрес, уровень напряжения сигналов (а также присутствие и полярность импульсов Link Test для 10BASE-T). Если сигнал не обнаружен на сетевом адаптере, то тест автоматически сканирует соединительный разъем и кабель для их диагностики.

Функции сбора статистики

Эти функции позволяют в реальном масштабе времени проследить за изменением наиболее важных параметров, характеризующих «здоровье» сегментов сети. Статистика обычно собирается с разной степенью детализации по разным группам.

Сетевая статистика

В этой группе собраны наиболее важные статистические показатели – коэффициент использования сегмента (utilization), уровень коллизий, уровень ошибок и уровень широковещательного трафика. Превышение этими показателями определенных порогов в первую очередь говорят о проблемах в том сегменте сети, к которому подключен многофункциональный прибор.

Статистика ошибочных кадров

Эта функция позволяет отслеживать все типы ошибочных кадров для определенной технологии. Например, для технологии Ethernet характерны следующие типы ошибочных кадров.

- Укороченные кадры (Short frames). Это кадры, имеющие длину, меньше допустимой, то есть меньше 64 байт. Иногда этот тип кадров дифференцируют на два класса – просто короткие кадры (short), у которых имеется корректная контрольная сумма, и «коротышки» (runts), не имеющие корректной контрольной суммы. Наиболее вероятными причинами появления укороченных кадров являются неисправные сетевые адаптеры и их драйверы.
- Удлиненные кадры (Jabbers). Это кадры, имеющие длину, превышающую допустимое значение в 1518 байт с хорошей или плохой контрольной суммой. Удлиненные кадры являются следствием затянувшейся передачи, которая появляется из-за неисправностей сетевых адаптеров.
- Кадры нормальных размеров, но с плохой контрольной суммой (Bad FCS) и кадры с ошибками выравнивания по границе байта. Кадры с неверной контрольной суммой являются следствием множества причин – плохих адаптеров, помех на кабелях, плохих контактов, некорректно работающих портов повторителей, мостов, коммутаторов и маршрутизаторов. Ошибка выравнивания всегда сопровождается ошибкой по контрольной сумме, поэтому некоторые средства анализа трафика не делают между ними различий. Ошибка выравнивания может быть следствием прекращения передачи кадра при распознавании коллизии передающим адаптером.
- Кадры-призраки (ghosts) являются результатом электромагнитных наводок на кабеле. Они воспринимаются сетевыми адаптерами как кадры, не имеющие нормального признака начала кадра – 10101011. Кадры-призраки имеют длину более 72 байт, в противном случае они классифицируются как удаленные коллизии. Количество обнаруженных кадров-призраков в большой степени зависит от точки подключения сетевого анализатора. Причинами их возникновения являются петли заземления и другие проблемы с кабельной системой.

Знание процентного распределения общего количества ошибочных кадров по их типам может многое подсказать администратору о возможных причинах неполадок в сети. Даже небольшой процент ошибочных кадров может привести к значительному снижению полезной пропускной способности сети, если протоколы, восстанавливающие искаженные кадры, работают с большими тайм-аутами ожидания квитанций. Считается, что в нормально работающей сети процент ошибочных кадров не должен превышать 0,01 %, то есть не более 1 ошибочного кадра из 10 000.

Статистика по коллизиям

Эта группа характеристик дает информацию о количестве и видах коллизий, отмеченных на сегменте сети, позволяет определить наличие и местонахождение проблемы. Анализаторы протоколов обычно не могут дать дифференцированной картины распределения общего числа коллизий по их отдельным типам, в то же время знание преобладающего типа коллизий может помочь понять причину плохой работы сети.

Ниже приведены основные типы коллизий сети Ethernet.

- Локальная коллизия (Local Collision). Является результатом одновременной передачи двух или более узлов, принадлежащих к тому сегменту, в котором производятся измерения. Если многофункциональный прибор не генерирует кадры, то в сети на витой паре или волоконно-оптическом кабеле локальные коллизии не фиксируются. Слишком высокий уровень локальных коллизий является следствием проблем с кабельной системой.
- Удаленная коллизия (Remote Collision). Эти коллизии происходят на другой стороне повторителя (по отношению к тому сегменту, в котором установлен измерительный прибор). В сетях, построенных на многопортовых повторителях (10Base-T, 10Base-FL/FB, 100Base-TX/FX/T4, Gigabit Ethernet), все измеряемые коллизии являются удаленными (кроме тех случаев, когда анализатор сам генерирует кадры и может быть виновником коллизии). Не все анализаторы протоколов и средства мониторинга одинаковым образом фиксируют удаленные коллизии. Это происходит из-за того, что некоторые измерительные средства и системы не фиксируют коллизии, происходящие при передаче преамбулы.
- Поздняя коллизия (Late Collision). Это коллизия, которая происходит после передачи первых 64 байт кадра (по протоколу Ethernet коллизия должна обнаруживаться при .передаче первых 64 байт кадра). Результатам поздней коллизии будет кадр, который имеет длину более 64 байт и содержит неверное значение контрольной суммы. .Чаще всего это указывавает на то, что сетевой адаптер, являющийся источником конфликта, оказывается не в состоянии, правильно прослушивать линию и поэтому не может вовремя остановить передачу. Другой причиной поздней коллизии является слишком большая длина кабельной системы или слишком большое количество промежуточных повторителей, приводящее к превышению максимального значения времени двойного оборота сигнала.

Средняя интенсивность коллизий в нормально работающей сети должна быть меньше 5 %. Большие всплески (более 20 %) могут быть индикатором кабельных проблем.

Распределение используемых сетевых протоколов

Эта статистическая группа относится к протоколам сетевого уровня. На дисплее отображается список основных протоколов в убывающем порядке относительно процентного соотношения кадров, содержащих пакеты данного протокола к общему числу кадров в сети.

Основные отправители (Top Sendes)

Функция позволяет отслеживать наиболее активные передающие узлы локальной сети. Прибор можно настроить на фильтрацию по единственному адресу и выявить список основных отправителей кадров для данной станции. Данные отражаются на дисплее в виде диаграммы вместе с перечнем основных отправителей кадров.

Основные получатели (Top Receivers)

Функция позволяет следить за наиболее активными узлами-получателями сети. Информация отображается в виде, аналогичном приведенному выше.

Основные генераторы широковещательного трафика (Top Broadcasters)

Функция выявляет станции сети, которые больше остальных генерируют кадры с широковещательными и групповыми адресами.

Генерирование трафика (Traffic Generation)

Прибор может генерировать трафик для проверки работы сети при повышенной нагрузке. Трафик может генерироваться параллельно с активизированными функциями *Сетевая статистика, Статистика ошибочных кадров и Статистика по коллизиям*.

Пользователь может задать параметры генерируемого трафика, такие как интенсивность и размер кадров. Для тестирования мостов и маршрутизаторов прибор может автоматически создавать заголовки IP- и IPX-пакетов, и все что требуется от оператора – это внести адреса источника и назначения.

В ходе испытаний пользователь может увеличить на ходу размер и частоту следования кадров с помощью клавиш управления курсором. Это особенно ценно при поиске источника проблем производительности сети и условий возникновения отказов.

Функции анализа протоколов

Обычно портативные многофункциональные приборы поддерживают декодирование и анализ только основных протоколов локальных сетей, таких как протоколы стеков TCP/IP, Novell NetWare, NetBIOS и Banyan VINES.

В некоторых многофункциональных приборах отсутствует возможность декодирования захваченных пакетов, как в анализаторах протоколов, а вместо этого собирается статистика о наиболее важных пакетах, свидетельствующих о наличии проблем в сетях. Например, при анализе протоколов стека TCP/IP собирается статистика по пакетам протокола ICMP, с помощью которого маршрутизаторы сообщают конечным узлам о возникновении разного рода ошибок. Для ручной проверки достижимости узлов сети в приборы включается поддержка утилиты IP Ping, а также аналогичных по назначению утилит NetWare Ping и NetBIOS Ping.

7.3.6. Мониторинг локальных сетей на основе коммутаторов

Наблюдение за трафиком

Так как перегрузки процессоров портов и других обрабатывающих элементов коммутатора могут приводить к потерям кадров, то функция наблюдения за распределением трафика в сети, построенной на основе коммутаторов, очень важна.

Однако если сам коммутатор не снабжен встроенным агентом SNMP для каждого своего порта, то задача слежения за трафиком, традиционно решаемая в сетях с разделяемыми средами с помощью установки в сеть внешнего анализатора протоколов, очень усложняется.

Обычно в традиционных сетях анализатор протоколов или многофункциональный прибор подключался к свободному порту концентратора, что позволяло ему наблюдать за всем трафиком, передаваемым между любыми узлами сети.

Если же анализатор протокола подключить к свободному порту коммутатора, то он не зафиксирует почти ничего, так как кадры ему передавать никто не будет, а чужие кадры в его порт также направляться не будут. Единственный вид трафика, который будет фиксировать анализатор, – это трафик широковещательных пакетов, которые будут передаваться всем узлам сети, а также трафик кадров с неизвестными коммутатору адресами назначения. В случае когда сеть разделена на виртуальные сети, анализатор протоколов будет фиксировать только широковещательный трафик своей виртуальной сети.

Чтобы анализаторами протоколов можно было по-прежнему пользоваться и в коммутируемых сетях, производители коммутаторов снабжают свои устройства функцией зеркального отображения трафика любого порта на специальный порт. К специальному порту подключается анализатор протоколов, а затем на коммутатор подается команда через его модуль SNMP-управления для отображения трафика какого-либо порта на специальный порт.

Наличие функции зеркализации портов частично снимает проблему, но оставляет некоторые вопросы. Например, как просматривать одновременно трафик двух портов или трафик порта, работающего в полнодуплексном режиме.

Более надежным способом слежения за трафиком, проходящим через порты коммутатора, является

замена анализатора протокола на агенты RMON MIB для каждого порта коммутатора.

Агент RMON выполняет все функции хорошего анализатора протокола для протоколов Ethernet и Token Ring, собирая детальную информацию об интенсивности трафика, различных типах плохих кадров, о потерянных кадрах, причем самостоятельно строя временные ряды для каждого фиксируемого параметра. Кроме того, агент RMON может самостоятельно строить матрицы перекрестного трафика между узлами сети, которые очень нужны для анализа эффективности применения коммутатора.

Так как агент RMON, реализующий все 9 групп объектов Ethernet, стоит весьма дорого, то производители для снижения стоимости коммутатора часто реализуют только первые несколько групп объектов RMON MIB. Другим приемом снижения стоимости коммутатора является использование одного агента RMON для нескольких портов. Такой агент по очереди подключается к нужному порту, позволяя снять с него требуемые статистические данные.

Управление виртуальными сетями

Виртуальные локальные сети VLAN порождают проблемы для традиционных систем управления на платформе SNMP как при их создании, так и при наблюдении за их работой.

Как правило, для создания виртуальных сетей требуется специальное программное обеспечение компании-производителя, которое работает на платформе системы управления, например HP Open View. Сами платформы систем управления этот процесс поддержать не могут в основном из-за долгого отсутствия стандарта на виртуальные сети. Можно надеяться, что появление стандарта 802.1Q изменит ситуацию в этой области.

Наблюдение за работой виртуальных сетей также создает проблемы для традиционных систем управления. При создании карты сети, включающей виртуальные сети, необходимо отображать как физическую структуру сети, так и ее логическую структуру, соответствующую связям отдельных узлов виртуальной сети. При этом по желанию администратора система управления должна уметь отображать соответствие логических и физических связей в сети, то есть на одном физическом канале должны отображаться все или отдельные пути виртуальных сетей.

К сожалению, многие системы управления либо вообще не отображают виртуальные сети, либо делают это очень неудобным для пользователя способом, что вынуждает обращаться к менеджерам компаний-производителей для решения этой задачи.

Выводы

- Мониторинг и анализ сети представляют собой важные этапы контроля работы сети. Для выполнения этих этапов разработан ряд средств, применяемых автономно в тех случаях, когда применение интегрированной системы управления экономически неоправданно.
- В состав автономных средств мониторинга и анализа сети входят встроенные средства диагностики, анализаторы протоколов, экспертные системы, сетевые анализаторы, кабельные сканеры и тестеры, многофункциональные приборы.
- Анализаторы протоколов чаще всего представляют собой специальное программное обеспечение для персональных компьютеров и ноутбуков, которое переводит сетевой адаптер компьютера в режим «беспорядочного» захвата всех кадров. Анализатор протоколов выполняет декодирование захваченных кадров для вложенных пакетов протоколов всех уровней, включая прикладной.
- Сетевые анализаторы представляют собой прецизионные приборы для сертификации кабельных систем по международным стандартам. Кроме того, эти устройства могут выполнять некоторые функции анализаторов протоколов.
- Кабельные сканеры являются портативными приборами, которые могут измерить электрические параметры кабелей, а также обнаружить место повреждения кабеля. Кабельные тестеры представляют собой наиболее простые портативные приборы, способные обнаружить неисправность кабеля.
- Многофункциональные портативные приборы сочетают в себе функции кабельных сканеров и анализаторов протоколов. Они снабжены многострочными дисплеями, контекстно-чувствительной системой помощи, встроенным микропроцессором с программным обеспечением и позволяют выполнять комплексную проверку сегментов сети на всех уровнях,

от физического (что не умеют делать анализаторы протоколов), до прикладного. Отличаются от анализаторов протоколов поддержкой только базового набора протоколов локальных сетей.

Вопросы и упражнения

1. К какой из пяти стандартных функциональных групп системы управления относится функция концентратора Ethernet по обнулению поля данных в кадрах, поступающих на порты, к которым не подключен узел назначения?
2. К какому уровню модели TMN относится большинство выпускаемых сегодня систем управления?
3. Как объяснить, что наличие в одном сегменте сети NetWare сравнительно небольшого числа (3 %) ошибочных кадров Ethernet резко снижает пропускную способность сети. Рассчитайте коэффициент снижения полезной пропускной способности сети, если при передаче файлов используется метод квитирования с простоями, причем тайм-аут ожидания квитанции составляет 0,5 с, сервер тратит на подготовку очередного кадра данных 20 мкс после получения квитанции от клиентской станции, а клиентская станция отсылает квитанции через 30 мкс после получения очередного кадра данных от сервера. Служебная информация протоколов верхних уровней занимает в кадре Ethernet 58 байт, причем данные передаются в кадрах Ethernet с полем данных максимального размера в 1500 байт, а квитанции помещаются в заголовке протокола прикладного уровня.
4. Какая функция в системах управления системами соответствует функции построения карты сети в системах управления сетями?
5. Какое свойство агента, поддерживающего RMON MIB, послужило поводом назвать данную MIB базой управляющих данных для *удаленного* мониторинга?
6. Какие действия предпринимает агент SNMP, если его сообщение о сбое управляемого устройства, посланное с помощью команды trap, потеряется?
7. Можно ли построить систему управления, работающую без платформы управления?
8. Относится ли средство, называемое community string, к средствам аутентификации?
9. Какую базу данных использует протокол CMIP для воздействия сразу на группу агентов?
10. У вас есть подозрение, что часть коллизий в вашей сети вызвана электромагнитными наводками. Сможет ли анализатор протоколов прояснить ситуацию?

ЗАКЛЮЧЕНИЕ

Сетевые специалисты утверждают, что 50 % знаний в этой динамичной области техники полностью устаревает за 5 лет. Можно, конечно, спорить о точном количестве процентов и лет, но факт остается фактом: набор базовых технологий, представления о перспективности той или иной технологии, подходы и методы решения ключевых задач и даже понятия о том, какие задачи при создании сетей являются ключевыми – все это изменяется очень быстро и часто неожиданно. И примеров, подтверждающих такое положение дел, можно привести достаточно много.

Качество транспортного обслуживания клиентов корпоративной локальной сети в начале 90-х годов мало волновало сетевых администраторов – пропускной способности в 10 или 100 Мбит/с при передаче небольших текстовых файлов хватало на всех, и методы тонкого ее распределения между клиентами мало кого интересовали. А в конце 90-х годов все споры о том, какую технологию применять на магистрали локальной сети, сводятся именно к этой проблеме – хватит ли для победы технологии Gigabit Ethernet простой схемы приоритетного обслуживания в коммутаторах или чашу весов перевесят сложные методы обеспечения гарантированной полосы пропускания технологии ATM.

Непостоянство сетевого мира демонстрирует другой пример. Технически элегантная технология lOOVG-AnyLAN, успешно начавшая свою жизнь в 1995 году, уже через два года была признана всеми настолько бесперспективной, что весьма авторитетный журнал Data Communications International занес ее в список 25 наиболее заметных неудач за все время существования компьютерных сетей. Да и перспективы технологии ATM, которая по праву считается одной из наиболее важных технологий 90-х годов, сейчас подвергаются существенной переоценке. Ожидание скорых перемен, связанных с приходом единой транспортной технологии для всех типов сетей, сменилось гораздо более скептическим и осторожным отношением. Сегодня большинство специалистов считает, что ATM вряд

ли будет когда-либо широко применяться в локальных сетях, а в глобальных сетях ее роль еще долго будет ограничена передачей данных, оставляя на неопределенное время голосовой трафик сетям с коммутацией каналов. Меняются не только технологии, но и эмпирические законы, на основе которых долгое время принимались проектные решения. Например, с правилом 80–20 % о пропорциях локального и внешнего трафика произошло то же, что в свое время с законом Гроша – сегодня, чтобы добиться хорошего результата, оба эти утверждения нужно применять «с точностью до наоборот». Ну, а примеры революционных перемен, которые принес в мир сетей Internet, стали уже классическими.

Но, несмотря на обилие примеров, нельзя абсолютизировать изменчивость сетевых технологий. Ведь остаются «другие» 50 % – это те знания о компьютерных сетях, которые составляют фундамент образования сетевого специалиста. Независимо от того, какие технологии будут применяться в локальных и глобальных сетях через 5 или 10 лет, данные будут передаваться на основе метода коммутации пакетов, которые могут называться и иначе – кадрами, ячейками или как-нибудь еще, но суть метода от этого не изменится. Коммуникационные протоколы будут образовывать иерархический стек, а надежность передачи данных будет обеспечиваться за счет повторной передачи пакетов.

И этот, «другой» перечень примеров стабильности сетевого мира можно продолжать так же долго, как и первый, потому что многие идеи и подходы, составляющие становой хребет сетевых и компьютерных технологий, просто переходят из технологии в технологию, несколько трансформируясь и приспосабливаясь к требованиям времени. Одной из иллюстраций этого тезиса является та же технология lOOVG-AnyLAN. В этой технологии для разрешения конфликтов при доступе к разделяемой среде используется центральный арбитр, встроенный в концентратор. В локальных сетях такой подход ранее не использовался, но он широко применялся и применяется в компьютерах, например, при доступе периферийных устройств к общей шине ввода/вывода. И хотя технология lOOVG-AnyLAN уже была отмечена как неперспективная, в книге ее описание помещено не случайно. Читатель должен быть готов к тому, что скоро может появиться новая сетевая технология, применяющая в той или иной форме универсальную идею централизованного арбитража. Еще один пример. Для понимания недавно появившихся технологий ускоренной маршрутизации IP-трафика в локальных сетях (NetFlow, Fast IP и т. п.) достаточно увидеть в них комбинацию двух базовых идей – классической IP-маршрутизации «пакет за пакетом» и не менее классического подхода глобальных сетей, используемого при образовании виртуального канала – маршрутизации первого пакета и коммутации остальных.

Как знание аксиом в математике позволяет приходить к новым выводам, так и знание основополагающих сетевых концепций позволяет легко разбираться в новых, пусть даже на первый взгляд и очень сложных, технологиях. Авторы надеются, что книга, которую вы прочитали, создала стабильный запас базовых знаний, которые останутся с вами надолго и станут тем инструментом, с помощью которого вы сможете обновлять переменную «половину» знаний о постоянно изменяющемся мире компьютерных сетей.

ПРИЛОЖЕНИЕ

Ответы на вопросы

Далее приведены ответы на вопросы, не требующие развернутого обсуждения.

Глава 1

3. Нет, сетевыми приложениями называют распределенные приложения, то есть приложения, состоящие из нескольких частей, каждая из которых может выполняться на отдельном компьютере сети.
8. Физическая топология – звезда, логическая топология – общая шина.
9. B, D.
12. В каждом из перечисленных случаев кадр появится на всех портах всех устройств сети.
13. Кадр, посланный компьютеру B, появится на портах 5, 6. Кадр, посланный компьютеру C, появится на портах 5, 7,12,13. Кадр, посланный компьютеру D, появится на портах 1, 3, 5, 7, 8, 11, 12, 15, 16, 17.
16. Модель OSI стандартизует количество, функции и названия уровней системных средств

взаимодействия.

17. Стек OSI стандартизует конкретный набор протоколов.
18. Количество уровней могло бы быть и меньше (например, в результате передачи функций представительного уровня сеансовому или прикладному уровням) или больше (например, путем выделения из канального уровня в отдельный уровень подуровня доступа к среде). Семь уровней является одним из нескольких возможных рациональных решений.
20. Нет.
21. IEEE.
22. Стек TCP/IP, Internet или DoD. Стек Microsoft или NetBIOS/SMB. Стек IPX/ SPX или Novell.
24. Время реакции, пропускная способность, задержка передачи.
25. Синхронность.
26. Готовность, отказоустойчивость, безопасность, расширяемость, масштабируемость, прозрачность.

Глава 2

1. Могут.
2. Используйте для расчета формулу Шеннона.
3. Используйте для расчета формулу Найквиста. Так как для широкополосных каналов дуплексный режим обеспечивается с помощью техники TDM, то полученную величину разделите на 2.
4. Ответы приведены в таблице.

	Задержка распространения	Задержка передачи
Для витой пары	0,33 мкс	10,24 мкс
Для коаксиального кабеля	6,6 мкс	102,4 мкс
Для спутникового канала	0,24с	8 мс

5. На линию будет передан кадр 0010 0100 1010 0101 0111 1101 0010 1011 0100 0110 0.
8. Учитывая частоту появления символов, можно выбрать следующую кодировку: O - 1, A - 01, D - 001, B - 0001, C - 00001, F - 00000. В этой кодировке для передачи указанной последовательности потребуется 35 бит. При использовании кодов ASCII требуется 128 бит. При использовании кодов равной длины, учитывая, что в последовательность входит только 6 различных символов, можно обойтись кодами длиной 3 бита, что для всей последовательности составит 48 бит. Следовательно, компрессия достигается в обоих случаях.
10. Уменьшить.
11. Чем сеть надежней, тем окно больше.
12. Нельзя перераспределить пропускную способность между абонентами при молчании некоторых их них.
13. Для трафика компьютерных сетей – способ коммутации пакетов.

Глава 3

2. B.
3. B, C, D – являются. A, E – не являются,.
4. Преамбула и начальный ограничитель нужны для вхождения приемника в битовую и байтовую синхронизацию с передатчиком.
5. Сетевые адапреры и повторители
7. Для устойчивого распознавания коллизий.
9. Названия 1-го типа кадров - 802.3/LLC, 802.3/802.2, Novell 802.2; 2-го типа кадров - Raw 802.3, Novell 802.3; 3-го типа кадров - Ethernet DIX, Ethernet II; 4-го типа кадров – Ethernet SNAP.
10. При ответе на этот вопрос следует учитывать разные факторы: характеристики сетевых адаптеров, используемый протокол сетевого уровня, тип операционной системы. В частности, в сети, работающей по протоколу IPX, даже компьютеры с современными адаптерами,

распознающими тип кадра автоматически, не смогут взаимодействовать друг с другом, если они используют разные форматы кадров.

11. Реакция концентратора зависит от его производителя, чаще всего порт отключается при слишком длительной передаче (jabber) и слишком интенсивных коллизиях. Все концентраторы отключают порт при отсутствии ответных импульсов link test.
13. С увеличением коэффициента использования производительность сети экспоненциально падает.
14. Технология, работающая на меньшей скорости, поддерживает большую максимальную длину сети.
15. Из соображений приемлемого затухания сигнала.
16. Расчет времени двойного оборота должен показать корректность сети.
19. Это время является произведением времени удержания маркера и максимального количества станций в кольце.
22. Сетевые адаптеры и концентраторы, подключенные по схемам DAS и DAC соответственно.
23. Нет, продолжение работы при однократном обрыве кабеля возможно не всегда, а только при двойном подключении всех узлов к кольцу.
24. Кольцо распадется на два несвязных сегмента.
25. Использование таблицы соответствия MAC-адресов узлов сети портам устройства.
26. C, D, E.
28. С обеспечением условий распознавания коллизий.

Глава 4

2.

	Горизонтальная подсистема	Вертикальная подсистема	Подсистема кампуса
Неэкранированная витая пара	+	- +	-
Экранированная витая пара	+	- +	-
Толстый коаксиальный кабель	+	+-	-
Тонкий коаксиальный кабель	+	-	-
Оптоволоконный кабель	+-	+	+
Беспроводная связь	+	-	-

3. Магистральную часть сети, которая объединяет сети большинства подразделений предприятия или сетей доступа поставщика территориальных услуг.
5. Да, сетевой адаптер, соединенный с коммутатором, может работать в дуплексном режиме, а в остальных случаях – нет.
6. Концентратор FDDI – стандартным способом, а концентраторы остальных технологий – нестандартным.
7. Поддержка управления по протоколу SNMP, блокировка порта при подключении узла с несанкционированным MAC-адресом, доставка данных в неискаженном виде только узлу назначения.
9. Для исключения необходимости использования перекрестных кабелей.
10. Путем пассивного слежения за адресами источников проходящих кадров.
11. Мост/коммутатор автоматически учтет их существование при отправке новыми компьютерами первого кадра в сеть.
12. Размер адресной таблицы говорит о назначении моста – чем больше размер, тем для более высокого уровня в иерархии сети (рабочая группа, отдел, магистраль здания) предназначен данный мост. Если таблица переполнится, то мост будет засорять сеть «псевдоширковещательными» кадрами в тех случаях, когда адрес назначения не попал в таблицу из-за ее недостаточного размера.
13. Да.

14. Вручную заблокировать некоторые порты у некоторых мостов, чтобы исключить петли.
15. C.
16. Они могут соединяться связями произвольной топологии.
17. Маршрутизаторы могут передавать данные по резервным связям, а мосты нет.
18. Если стековые концентраторы имеют несколько изолированных внутренних сегментов, то использование двух концентраторов, объединенных в стек, будет лучшим вариантом, так как стек концентраторов более экономичен (за счет общих модулей управления и питания) и позволяет программно менять состав рабочих групп. В противном случае нужно применять два отдельных концентратора.
19. В одноранговой сети, где роль серверов выполняют обычно несколько компьютеров, замена концентратора коммутатором приведет к росту производительности сети во всех трех случаях. В сети NetWare с одним сервером к такому результату приведет только вариант B.
22. В полудуплексном режиме – с помощью методов обратного давления и агрессивного захвата среды, в дуплексном режиме – с помощью механизма управления потоком стандарта 802.3x.
24. Нет.
26. Некоторые дополнительные функции, свойственные дорогим коммутаторам, требуют полной буферизации пакетов.

Глава 5

4. Да.
5. C (компьютеры, подключенные к разным сегментам, могут обмениваться данными, только в том случае, если ОС Windows NT сконфигурирована как программный маршрутизатор).
7. IP, ICMP, RIP, OSPF, ARP и некоторые другие.
8. Протокол IP не гарантирует доставку пакета.
9. Средствами уровня межсетевого взаимодействия ошибки могут быть обнаружены, но не исправлены.
10. Окно определено на множестве байт, а единицей данных, получение которой подтверждается квитанцией, является сегмент.
11. A, B, C, D.
12. E, F.
13. Общее количество IP-адресов определяется разрядностью адреса и равно 2^{32}. Адреса класса A имеют в старшем разряде 0, оставшийся 31 разряд дает 2^{31} комбинаций, что составляет 50 % всего адресного пространства. Адреса класса B имеют фиксированное значение двух старших разрядов 10, и для образования адресов этого класса используется 30 разрядов, что дает 25 % общего адресного пространства. Аналогично рассуждая, получаем, что адреса класса C составляют 12,5 % всего множества IP-адресов.
14. Не могут быть адресами конечных узлов A, C, E, F, I, J, K, L.
15. Номер подсети – 198.65.12.64, максимальное число узлов – 14.
16. Максимальное число абонентов 255. Маска – 255.255.255.0.
17. Максимальное количество подсетей 64, маска – 255.255.255.252.
18. Для правильной маршрутизации пакетов в сети с использованием масок достаточно того, что маски передаются протоколами маршрутизации RIP-2, OSPF или устанавливаются вручную для каждой записи таблицы маршрутизации.
19. Преимущества: экономное расходование адресов и уменьшение количества записей в таблицах маршрутизации. Проблема – перенумерация сетей.
20. Чем короче префикс, тем большее количество IP-адресов может входить в этот пул, и наоборот.
21. Такое сочетание адреса сети и маски дает совпадение с любым IP-адресом.
22. Отличается: маршрутизатор принимает и обрабатывает только кадры с MAC-адресом, совпадающим с адресом его порта, причем в дальнейшей обработке MAC-адрес не используется, а коммутатор принимает кадры с любыми MAC-адресами, и дальнейшая обработка основана на значении MAC-адреса.
24. Самая простая метрика – количество хопов, то есть количество промежуточных маршрутизаторов, которые нужно преодолеть пакету до сети назначения, кроме того, используются метрики, учитывающие пропускную способность, вносимые задержки и

надежность сетей, а также любые комбинации этих метрик.

26. D.
27. C.
30. Достаточно стандартной конфигурации.

Глава 6

1. Модемы используют для передачи данных модулированную синусоиду, а устройства DSU/CSU – импульсы или потенциальные сигналы.
2. Выделенные цифровые каналы T1 или E1.
3. Синхронный, так как он повышает полезную пропускную способность на 20 % при одной и той же битовой скорости.
4. В современном модеме поддерживаются два уровня – физический и канальный.
5. Нет, так как оно двухпроводное, а канал E1 использует четрехпроводное окончание. Но если имеются два обычных окончания, то тогда подключение может оказаться возможным при подходящем качестве проводов окончания.
6. Можно использовать различные услуги: три коммутируемых канала типа B интерфейса PRI сети ISDN, объединенных в один логический канал; три выделенных (полупостоянных) канала интерфейса PRI сетей ISDN, объединенных в один логический канал; выделенный дробный цифровой канал T1 или E1, постоянный виртуальный канал сети frame relay.
7. 28.
8. Может.
9. Сервер удаленного доступа, подключенный своими асинхронными портами к интерфейсам мэйнфрейма и портом Ethernet к локальной сети. Пользователь мэйнфрейма может соединиться с сервером удаленного доступа в режиме терминала, а затем запустить протокол терминального доступа, например telnet, к любым узлам сети, которые этот протокол поддерживают.
10. С помощью ручного набора Hayes-команд.
11. Услугу «Доступ к сети X.25 через канал типа D».
12. Восемь выделенных (полупостоянных) каналов типа B, объединенных в один логический канал.
13. Для коммутатора X.25 – 16 кадров, а для коммутатора frame relay – 8.
14. Процент дошедших кадров будет выше во втором случае, так как в первом некоторые кадры будут сразу отброшены, а во втором они будут только отмечены признаком DE=1, но не отброшены, так как сеть недогружена.
16. Для ABR. Для других категорий услуг предварительное резервирование параметров трафика и контроль соглашения делают управление потоком данных излишним.
17. Коммутация на основе VPI.
18. Не более 33,6 Кбит/с.
20. В превышении тайм-аута ожидания положительной квитанции протокола NetBUEI из-за задержек в очередях сети frame relay.

Глава 7

1. Управление безопасностью.
2. К уровню управления элементами сети.
3. Резкое снижение пропускной способности сети NetWare при появлении ошибочных кадров объясняется большой величиной тайм-аута в единственном протоколе стека, исправляющем ошибки при передаче файлов, – протоколе NCP.
4. Функция учета используемых программных и аппаратных средств.
5. Интеллектуальные функции накопления и обработки данных, удобные при удаленным мониторинге.
6. Никаких.
7. Можно, но достаточно трудоемко.
8. Нет.
9. Дерево включения.
10. Нет, так как анализаторы протоколов не работают на физическом уровне.

РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА

1. Стандарты по локальным вычислительным сетям: Справочник. В. К. Щербо, В. М. Киреичев, С. И. Самойленко; под ред. С. И. Самойленко. – М.: Радио и связь, 1990.
2. Практическая передача данных: Модемы, сети и протоколы. Ф. Дженнингс; перев. с англ. – М.: Мир, 1989.
3. Сети ЭВМ: протоколы стандарты, интерфейсы. Ю. Блэк; перев. с англ. – М.: Мир, 1990.
4. Fast Ethernet. Л. Куинн, Р. Рассел. - BHV-Киев, 1998.
5. Коммутация и маршрутизация IP/IPX трафика. М. В. Кульгин, АйТи. – М.: Компьютер-пресс, 1998.
6. Волоконная оптика в локальных и корпоративных сетях связи. А. Б. Семенов, АйТи. – М.: Компьютер-пресс, 1998.
7. Протоколы Internet. С. Золотов. – СПб.: BHV – Санкт-Петербург, 1998.
8. Персональные компьютеры в сетях TCP/IP. Крейг Хант; перев. с англ. – BHV– Киев, 1997.
9. Вычислительные системы, сети и телекоммуникации. Пятибратов и др. – ФИС, 1998.
10. Высокопроизводительные сети. Энциклопедия пользователя. Марк А. Спортак и др.; перев. с англ. – Киев, ДиаСофт, 1998.
11. Средства связи для «последней мили». Денисьев и Мирошников. - Эко-Трендз, 1998.
12. Синхронные цифровые сети SDH. Н. Н. Слепов. – Эко-Трендз, 1998.
13. Сети предприятий на основе Windows NT для профессионалов. Стерн, Монти; перев. с англ. – СПб.: Питер, 1999.
14. Networking Essentials. Сертификационный экзамен – экстерном (экзамен 70-058). Дж. Стюарт, Эд Титтель, Курт Хадсон; перев с англ. – СПб.: Питер Ком, 1999.
15. Основы построения сетей. Учебное руководство для специалистов MCSE (+CD-ROM). Дж. Челлис, Ч. Перкинс, М. Стриб; перевод с англ. – Лори, 1997.
16. Компьютерные сети. Учебный курс, 2-е изд. (+CD-ROM). - MicrosoftPress, Русская редакция, 1998.
17. Сетевые средства Microsoft Windows NT Server 4.0; перев. с англ. СПб.: – BHV – Санкт-Петербург, 1997.
18. Ресурсы Microsoft Windows NT Server 4.0. Книга 1; перев. с англ. СПб.: - BHV -Санкт-Петербург, 1997.
19. Толковый словарь по вычислительной технике; перев. с англ. – М.: Издательский отдел «Русская редакция» ТОО «Channel Trading Ltd.», 1995.
20. Emerging Communications Technologies, 2/e, Uyless Black, Prentice Hall Professional, 1997.
21. Telecommunications for Managers, 3/e, Stanford H. Rowe, Prentice Hall, 1995.
22. Data and Computer Communications, 5/e, William Stallings, Prentice Hall, 1997.
23. ISDN and Broadband ISDN with Frame Relay and ATM, 3/e, William Stallings, Prentice Hall, 1995.
24. Data Communications, Computer Networks and Open Systems, Fred Halsall, Adisson-Wesley, 1996.
25. Internetworking with TCP/IP: Principles, Protocols, and Architecture, Duglas E. Comer, Prentice Hall, 1995.
26. TCP/IP Network Administration, 2/e, Craig Hunt, O'Reilly & Associates, 1998.
27. Computer Networks, Andrew S. Tanenbaum, Prentice Hall, 1996.

В.Г. Олифер, Н. А. Олифер

Компьютерные сети. Принципы, технологии, протоколы

Главный редактор В. Усманов
Ответственный редактор Е. Строганова
Литературный редактор А. Жданов
Художник Н. Биржаков
Верстка и иллюстрации П. Быстрое
Корректор В. Листова

Лицензия ЛР№066333от23.02.99.

Подписано в печать 02.02.01. Формат 70x 100'/16
Усл. п. л. 54,6. Доп. тираж 10 000 экз. Заказ №2856.

Издательство «Питер».
196105, Санкт-Петербург, ул. Благодатная, 67.

Налоговая льгота – общероссийский классификатор продукции ОК 005-93,
том 2; 953000–книга и брошюры.

Отпечатано с фотоформ в ГПП «Печатный двор»
Министерства РФ по делам печати, телерадиовещания
и средств массовых коммуникаций.
197110, Санкт-Петербург, Чкаловский пр., 15.

СОДЕРЖАНИЕ

От авторов ... 3
Для кого эта книга ... 4
Структура книги ... 4
Благодарности ... 6
Глава 1. Общие принципы построения вычислительных сетей ... 6
1.1. От централизованных систем - к вычислительным сетям ... 6
1.1.1. Эволюция вычислительных систем ... 6
Системы пакетной обработки ... 6
Многотерминальные системы – прообраз сети ... 7
Появление глобальных сетей ... 8
Первые локальные сети ... 8
Создание стандартных технологий локальных сетей ... 9
Современные тенденции ... 10
1.1.2. Вычислительные сети – частный случай распределенных систем ... 11
Мультипроцессорные компьютеры ... 11
Многомашинные системы ... 11
Вычислительные сети ... 12
Распределенные программы ... 13
1.1.3. Основные программные и аппаратные компоненты сети ... 14
1.1.4. Что дает предприятию использование сетей ... 14
Выводы ... 17
1.2. Основные проблемы построения сетей ... 17
1.2.1. Связь компьютера с периферийными устройствами ... 17
1.2.2. Простейший случай взаимодействия двух компьютеров ... 19
1.2.3. Проблемы физической передачи данных по линиям связи ... 21
1.2.4. Проблемы объединения нескольких компьютеров ... 22
Топология физических связей ... 23
Организация совместного использования линий связи ... 24
Адресация компьютеров ... 26
1.2.5. Ethernet–пример стандартного решения сетевых проблем ... 28
1.2.6. Структуризация как средство построения больших сетей ... 29
Физическая структуризация сети ... 30
Логическая структуризация сети ... 31
1.2.7. Сетевые службы ... 35
Выводы ... 36
1.3. Понятие «открытая система» и проблемы стандартизации ... 37
1.3.1. Многоуровневый подход. Протокол. Интерфейс. Стек протоколов ... 37
1.3.2. Модель OSI ... 40

1.3.3. Уровни модели OSI ... 42
Физический уровень ... 42
Канальный уровень ... 43
Сетевой уровень ... 44
Транспортный уровень ... 45
Сеансовый уровень ... 46
Представительный уровень ... 46
Прикладной уровень ... 46
Сетезависимые и сетенезависимые уровни ... 46
1.3.4. Понятие «открытая система» ... 48
1.3.5. Модульность и стандартизация ... 49
1.3,6. Источники стандартов ... 50
1.3.7. Стандартные стеки коммуникационных протоколов ... 52
Стек OSI ... 52
Стек TCP/IP ... 53
Стек IPX/SPX ... 54
Стек NetBIOS/SMB ... 55
Выводы ... 56
1.4. Локальные и глобальные сети ... 56
1.4.1. Особенности локальных, глобальных и городских сетей ... 56
1.4.2. Отличия локальных сетей от глобальных ... 57
1.4.3. Тенденция к сближению локальных и глобальных сетей ... 58
Выводы ... 59
1.5. Сети отделов, кампусов и корпораций ... 60
1.5.1. Сети отделов ... 60
1.5.2. Сети кампусов ... 61
1.5.3. Корпоративные сети ... 62
Выводы ... 63
1.6. Требования, предъявляемые к современным вычислительным сетям ... 64
1.6.1. Производительность ... 64
1.6.2. Надежность и безопасность ... 66
1.6.3. Расширяемость и масштабируемость ... 67
1.6.4. Прозрачность ... 67
1.6.5. Поддержка разных видов трафика ... 68
1.6.6. Управляемость ... 69
1.6.7. Совместимость ... 69
Выводы ... 69
Вопросы и упражнения ... 70
Глава 2. Основы передачи дискретных данных ... 71
2.1. Линии связи ... 72
2.1.1. Типы линий связи ... 72
2.1.2. Аппаратура линий связи ... 73
2.1.3. Характеристики линий связи ... 74
Типы характеристик и способы их определения ... 74
Спектральный анализ сигналов на линиях связи ... 75
Амплитудно-частотная характеристика, полоса пропускания и затухание ... 77
Пропускная способность линии ... 79
Связь между пропускной способностью линии и ее полосой пропускания ... 81
Помехоустойчивость и достоверность ... 82
2.1.4. Стандарты кабелей ... 83
Кабели на основе неэкранированной витой пары ... 84
Кабели на основе экранированной витой пары ... 85
Коаксиальные кабели ... 86
Волоконно-оптические кабели ... 86
Выводы ... 88

2.2. Методы передачи дискретных данных на физическом уровне ... 89
2.2.1. Аналоговая модуляция ... 89
Методы аналоговой модуляции ... 90
Спектр модулированного сигнала ... 90
2.2.2. Цифровое кодирование ... 92
Требования к методам цифрового кодирования ... 92
Потенциальный код без возвращения к нулю ... 93
Метод биполярного кодирования с альтернативной инверсией ... 94
Потенциальный код с инверсией при единице ... 94
Биполярный импульсный код ... 95
Манчестерский код ... 95
Потенциальный код 2B1Q ... 95
2.2.3. Логическое кодирование ... 96
Избыточные коды ... 96
Скрэмблирование ... 96
2.2.4. Дискретная модуляция аналоговых сигналов ... 98
2.2.5. Асинхронная и синхронная передачи ... 101
Выводы ... 101
2.3. Методы передачи данных канального уровня ... 102
2.3.1. Асинхронные протоколы ... 103
2.3.2. Синхронные символьно-ориентированные и бит-ориентированные протоколы ... 103
Символьно-ориентированные протоколы ... 104
Бит-ориентированные протоколы ... 105
Протоколы с гибким форматом кадра ... 106
2.3.3. Передача с установлением соединения и без установления соединения ... 106
2.3.4. Обнаружение и коррекция ошибок ... 108
Методы обнаружения ошибок ... 108
Методы восстановления искаженных и потерянных кадров ... 109
2.3.5. Компрессия данных ... 111
Выводы ... 112
2.4. Методы коммутации ... 113
2.4.1. Коммутация каналов ... 114
Коммутация каналов на основе частотного мультиплексирования ... 115
Коммутация каналов на основе разделения времени ... 116
Общие свойства сетей с коммутацией каналов ... 118
Обеспечение дуплексного режима работы на основе технологий FDM TDM и WDM ... 118
2.4.2. Коммутация пакетов ... 119
Принципы коммутации пакетов ... 119
Виртуальные каналы в сетях с коммутацией пакетов ... 121
Пропускная способность сетей с коммутацией пакетов ... 121
2.4.3. Коммутация сообщений ... 123
Выводы ... 124
Вопросы и упражнения ... 125
Глава 3. Базовые технологии локальных сетей ... 125
3.1, Протоколы и стандарты локальных сетей ... 125
3.1.1. Общая характеристика протоколов локальных сетей ... 125
3.1.2. Структура стандартов IEEE 802.x ... 127
Выводы ... 129
3.2. Протокол LLC уровня управления логическим каналом (802.2) ... 130
3.2.1. Три типа процедур уровня LLC ... 131
3.2.2. Структура кадров LLC. Процедура с восстановлением кадров LLC2 ... 131
Выводы ... 133
3.3. Технология Ethernet (802.3) ... 134
3.3.1. Метод доступа CSMA/CD ... 134
Этапы доступа к среде ... 135

Возникновение коллизии ... 135
Время двойного оборота и распознавание коллизий ... 137
3.3.2. Максимальная производительность сети Ethernet ... 139
3.3.3. Форматы кадров технологии Ethernet ... 140
Кадр 802.3/LLC ... 141
Кадр Raw 802.3/Novell 802.3 ... 142
Кадр Ethernet DIX/Ethernet II ... 143
Кадр Ethernet SNAP ... 143
Использование различных типов кадров Ethernet ... 143
3.3.4. Спецификации физической среды Ethernet ... 144
Стандарт 10Base-5 ... 144
Стандарт 10 Base-2 ... 147
Стандарт 10 Base-T ... 148
Оптоволоконный Ethernet ... 150
Домен коллизий ... 151
Общие характеристики стандартов Ethernet 10 Мбит/с ... 151
3.3.5. Методика расчета конфигурации сети Ethernet ... 152
Расчет PDV ... 153
Расчет PVV ... 154
Выводы ... 155
3.4. Технология Token Ring (802.5) ... 156
3.4.1. Основные характеристики технологии ... 156
3.4.2. Маркерный метод доступа к разделяемой среде ... 156
3.4.3. Форматы кадров Token Ring ... 158
Маркер ... 158
Кадр данных и прерывающая последовательность ... 158
Приоритетный доступ к кольцу ... 160
3.4.4. Физический уровень технологии Token Ring ... 160
Выводы ... 161
3.5. Технология FDDI ... 162
3.5.1. Основные характеристики технологии ... 162
3.5.2. Особенности метода доступа FDDI ... 164
3.5.3. Отказоустойчивость технологии FDDI ... 165
3.5.4 Физический уровень технологии FDDI ... 166
3.5.5. Сравнение FDDI с технологиями Ethernet и Token Ring ... 167
Выводы ... 168
3.6. Fast Ethernet и lOOVG-AnyLAN как развитие технологии Ethernet ... 169
3.6.1. Физический уровень технологии Fast Ethernet ... 170
Физический уровень 100Base-FX – многомодовое оптоволокно, два волокна ... 172
Физический уровень 1 OOBase-TX - витая пара UTP Cat 5 или STP Type 1, две пары ... 172
Физический уровень 100Base-T4 – витая пара UTP Cat 3, четыре пары ... 173
3.6.2. Правила построения сегментов Fast Ethernet при использовании повторителей ... 174
Ограничения длин сегментов DTE-DTE ... 174
Ограничения сетей Fast Ethernet, построенных на повторителях ... 175
3.6.3. Особенности технологии 100VG-AnyLAN ... 177
Выводы ... 179
3.7. Высокоскоростная технология Gigabit Ethernet ... 180
3.7.1. Общая характеристика стандарта ... 180
3.7.2. Средства обеспечения диаметра сети в 200 м на разделяемой среде ... 182
3.7.3. Спецификации физической среды стандарта 802.3z ... 182
Многомодовый кабель ... 182
Одномодовый кабель ... 183
Твинаксиальный кабель ... 183
3.7.4. Gigabit Ethernet на витой паре категории 5 ... 183
Выводы ... 184

Вопросы и упражнения ... 185
Глава 4. Построение локальных сетей по стандартам физического и канального уровней ... 187
4.1. Структурированная кабельная система ... 187
4.1.1. Иерархия в кабельной системе ... 187
4.1.2. Выбор типа кабеля для горизонтальных подсистем ... 189
4,1.3. Выбор типа кабеля для вертикальных подсистем ... 191
4.1.4. Выбор типа кабеля для подсистемы кампуса ... 192
Выводы ... 193
4.2. Концентраторы и сетевые адаптеры ... 193
4.2.1. Сетевые адаптеры ... 193
Функции и характеристики сетевых адаптеров ... 193
Классификация сетевых адаптеров ... 194
4.2.2. Концентраторы ... 195
Основные и дополнительные функции концентраторов ... 195
Отключение портов ... 197
Поддержка резервных связей ... 198
Защита от несанкционированного доступа ... 198
Многосегментные концентраторы ... 200
Управление концентратором по протоколу SNMP ... 201
Конструктивное исполнение концентраторов ... 202
Выводы ... 203
4.3. Логическая структуризация сети с помощью мостов и коммутаторов ... 204
4.3.1. Причины логической структуризации локальных сетей ... 204
Ограничения сети, построенной на общей разделяемой среде ... 204
Преимущества логической структуризации сети ... 206
Структуризация с помощью мостов и коммутаторов ... 208
4.3.2. Принципы работы мостов ... 210
Алгоритм работы прозрачного моста ... 210
Мосты с маршрутизацией от источника ... 212
Ограничения топологии сети, построенной на мостах ... 215
4.3.3. Коммутаторы локальных сетей ... 216
4.3.4. Полнодуплексные протоколы локальных сетей ... 221
Изменения в работе MAC-уровня при полнодуплексной работе ... 221
Проблема управления потоком данных при полнодуплексной работе ... 222
4.3.5. Управления потоком кадров при полудуплексной работе ... 224
Выводы ... 225
4.4. Техническая реализация и дополнительные функции коммутаторов ... 226
4.4.1. Особенности технической реализации коммутаторов ... 226
Коммутаторы на основе коммутационной матрицы ... 227
Коммутаторы с общей шиной ... 228
Коммутаторы с разделяемой памятью ... 228
Комбинированные коммутаторы ... 229
Конструктивное исполнение коммутаторов ... 229
4.4.2. Характеристики, влияющие на производительность коммутаторов ... 231
Скорость фильтрации и скорость продвижения ... 231
Коммутация «на лету» или с буферизацией ... 232
Размер адресной таблицы ... 233
Объем буфера кадров ... 234
4.4.3. Дополнительные функции коммутаторов ... 234
Поддержка алгоритма Spanning Tree ... 234
Трансляция протоколов канального уровня ... 238
Возможности коммутаторов по фильтрации трафика ... 238
Приоритетная обработка кадров ... 239
4.4.4. Виртуальные локальные сети ... 240
4.4.5. Типовые схемы применения коммутаторов в локальных сетях ... 244

Сочетание коммутаторов и концентраторов ... 244
Стянутая в точку магистраль на коммутаторе ... 245
Распределенная магистраль на коммутаторах ... 246
Выводы ... 247
Вопросы и упражнения ... 248
Глава 5. Сетевой уровень как средство построения больших сетей ... 250
5.1. Принципы объединения сетей на основе протоколов сетевого уровня ... 250
5.1.1. Ограничения мостов и коммутаторов ... 250
5.1.2. Понятие internetworking ... 251
5.1.3. Принципы маршрутизации ... 253
5.1.4. Протоколы маршрутизации ... 256
5.1.5. Функции маршрутизатора ... 259
Уровень интерфейсов ... 260
Уровень сетевого протокола ... 261
Уровень протоколов маршрутизации ... 262
5.1.6. Реализация межсетевого взаимодействия средствами TCP/IP ... 262
Многоуровневая структура стека TCP/IP ... 262
Уровень межсетевого взаимодействия ... 262
Основной уровень ... 263
Прикладной уровень ... 263
Уровень сетевых интерфейсов ... 263
Соответствие уровней стека TCP/IP семиуровневой модели ISO/OSI ... 264
Выводы ... 266
5.2. Адресация в IP-сетях ... 267
5.2.1. Типы адресов стека TCP/IP ... 267
5.2.2. Классы IP-адресов ... 267
5.2.3. Особые IP-адреса ... 269
5.2.4. Использование масок в IP-адресации ... 270
5.2.5. Порядок распределения IP-адресов ... 271
5.2.6. Автоматизация процесса назначения IP-адресов ... 272
5.2.7. Отображение IP-адресов на локальные адреса ... 272
5.2.8. Отображение доменных имен на IP-адреса ... 275
Организация доменов и доменных имен ... 275
Система доменных имен DNS ... 277
Выводы ... 279
5.3. Протокол IP ... 279
5.3.1. Основные функции протокола IP ... 279
5.3.2. Структура IP-пакета ... 280
5.3.3. Таблицы маршрутизации в IP-сетях ... 282
Назначение полей таблицы маршрутизации ... 284
Источники и типы записей в таблице маршрутизации ... 285
5.3.4. Маршрутизация без использования масок ... 286
5.3.5. Маршрутизация с использованием масок ... 289
Использование масок для структуризации сети ... 289
Использование масок переменной длины ... 292
Технология бесклассовой междоменной маршрутизации CIDR ... 295
5.3.6. Фрагментация IP-пакетов ... 298
5.3.7. Протокол надежной доставки TCP-сообщений ... 300
Порты ... 301
Сегменты и потоки ... 302
Соединения ... 302
Реализация скользящего окна в протоколе TCP ... 303
Выводы ... 304
5.4. Протоколы маршрутизации в IP-сетях ... 305
5.4.1. Внутренние и внешние протоколы маршрутизации Internet ... 305

5.4.2. Дистанционно-векторный протокол RIP ... 307
Построение таблицы маршрутизации ... 307
Этап 2 – рассылка минимальных таблиц соседям ... 308
Этап 4 – рассылка новой, уже не минимальной, таблицы соседям ... 309
Этап 5 – получение RIP-сообщений от соседей и обработка полученной информации ... 309
Адаптация RIP-маршрутизаторов к изменениям состояния сети ... 310
Методы борьбы с ложными маршрутами в протоколе RIP ... 313
5.4.3. Протокол «состояния связей» OSPF ... 313
Выводы ... 316
5.5. Средства построения составных сетей стека Novell ... 317
5.5,1. Общая характеристика протокола IPX ... 317
5.5.2. Формат пакета протокола IPX ... 318
5.5.3. Маршрутизация протокола IPX ... 320
Выводы ... 321
5.6. Основные характеристики ... 322
маршрутизаторов и концентраторов ... 322
5.6.1. Маршрутизаторы ... 322
Классификация маршрутизаторов по областям применения ... 322
Основные технические характеристики маршрутизатора ... 323
Дополнительные функциональные возможности маршрутизаторов ... 324
5.6.2. Корпоративные модульные концентраторы ... 326
5.6.3. Стирание граней между коммутаторами и маршрутизаторами ... 327
Соотношение коммутации и маршрутизации в корпоративных сетях ... 327
Отказ от маршрутизации ... 328
Коммутаторы 3-го уровня с классической маршрутизацией ... 328
Маршрутизация потоков ... 329
Выводы ... 330
Вопросы и упражнения ... 331
Глава 6. Глобальные сети ... 333
6.1. Основные понятия и определения ... 334
6.1.1. Обобщенная структура и функции глобальной сети ... 334
Транспортные функции глобальной сети ... 334
Высокоуровневые услуги глобальных сетей ... 335
Структура глобальной сети ... 336
Интерфейсы DTE-DCE ... 338
6.1.2. Типы глобальных сетей ... 340
Выделенные каналы ... 341
Глобальные сети с коммутацией каналов ... 341
Глобальные сети с коммутацией пакетов ... 342
Магистральные сети и сети доступа ... 343
Выводы ... 345
6.2. Глобальные связи на основе выделенных линий ... 346
6.2.1. Аналоговые выделенные линии ... 346
Типы аналоговых выделенных линий ... 346
Модемы для работы на выделенных каналах ... 347
6.2.2. Цифровые выделенные линии ... 349
Технология плезиохронной цифровой иерархии PDH ... 349
Технология синхронной цифровой иерархии SONET/SDH ... 352
Применение цифровых первичных сетей ... 358
Устройства DSU/CSU для подключения к выделенному каналу ... 359
6.2.3. Протоколы канального уровня для выделенных линий ... 360
Протокол SLIP ... 360
Протоколы семейства HDLC ... 362
Протокол PPP ... 362
6.2.4. Использование выделенных линий для построения корпоративной сети ... 364

Выводы ... 366
6.3. Глобальные связи на основе сетей с коммутацией каналов ... 368
6.3.1. Аналоговые телефонные сети ... 369
Организация аналоговых телефонных сетей ... 369
Модемы для работы на коммутируемых аналоговых линиях ... 370
6.3.2. Служба коммутируемых цифровых каналов Switched 56 ... 372
6.3.3. ISDN – сети с интегральными услугами ... 373
Цели и история создания технологии ISDN ... 373
Пользовательские интерфейсы ISDN ... 374
Подключение пользовательского оборудования к сети ISDN ... 376
Адресация в сетях ISDN ... 377
Стек протоколов и структура сети ISDN ... 379
Использование служб ISDN в корпоративных сетях ... 380
Выводы ... 381
6.4. Компьютерные глобальные сети с коммутацией пакетов ... 382
6.4.1. Принцип коммутации пакетов с использованием техники виртуальных каналов ... 382
6.4.2. Сети X.25 ... 385
Назначение и структура сетей X.25 ... 385
Адресация в сетях X.25 ... 387
Стек протоколов сети X.25 ... 387
6.4.3. Сети Frame Relay ... 391
Назначение и общая характеристика ... 391
Стек протоколов frame relay ... 392
Поддержка качества обслуживания ... 394
Использование сетей frame relay ... 397
6.4.4. Технология ATM ... 398
Основные принципы технологии ATM ... 399
Стек протоколов ATM ... 404
Уровень адаптации AAL ... 404
Протокол ATM ... 406
Категории услуг протокола ATM и управление трафиком ... 409
Передача трафика IP через сети ATM ... 412
Сосуществование ATM с традиционными технологиями локальных сетей ... 414
Использование технологии ATM ... 416
Выводы ... 417
6.5. Удаленный доступ ... 418
4.5.1. Основные схемы глобальных связей при удаленном доступе ... 418
Типы взаимодействующих систем ... 419
Типы поддерживаемых служб ... 419
Типы используемых глобальных служб ... 420
6.5.2. Доступ компьютер - сеть ... 421
Удаленный узел ... 422
Удаленное управление и терминальный доступ ... 423
Почта ... 424
6.5.3. Удаленный доступ через промежуточную сеть ... 424
Общая схема двухступенчатого доступа ... 424
Технологии ускоренного доступа к Internet через абонентские окончания телефонных и кабельных сетей ... 425
Выводы ... 427
Вопросы и упражнения ... 428
Глава 7. Средства анализа и управления сетями ... 429
7.1. Функции и архитектура систем управления сетями ... 429
7.1.1. Функциональные группы задач управления ... 429
7.1.2. Многоуровневое представление задач управления ... 432
7.1.3. Архитектуры систем управления сетями ... 434

Схема менеджер – агент ... 434
Структуры распределенных систем управления ... 436
Платформенный подход ... 438
Выводы ... 438
7.2. Стандарты систем управления ... 439
7.2.1. Стандартизуемые элементы системы управления ... 439
7.2.2. Стандарты систем управления на основе протокола SNMP ... 439
Концепции SNMP-управления ... 439
Примитивы протокола SNMP ... 440
Структура SNMP MIB ... 441
Форматы и имена объектов SNMP MIB ... 442
Формат сообщений SNMP ... 444
Спецификация RMON MIB ... 446
Недостатки протокола SNMP ... 448
7.2.3. Стандарты управления OSI ... 448
Агенты и менеджеры ... 449
Управление системами, управление уровнем и операции уровня ... 450
Информационная модель управления ... 450
Управляющие знания и деревья знаний ... 451
Использование древовидных баз данных для хранения управляющих знаний ... 451
Правила определения управляемых объектов ... 453
Протокол CMIP и услуги CMIS ... 454
Сравнение протоколов SNMP и CMIP ... 455
Выводы ... 455
7.3. Мониторинг и анализ локальных сетей ... 456
7.3.1. Классификация средств мониторинга и анализа ... 456
7.3.2. Анализаторы протоколов ... 457
7.3.3. Сетевые анализаторы ... 459
7.3.4. Кабельные сканеры и тестеры ... 459
7.3.5. Многофункциональные портативные приборы мониторинга ... 459
Интерфейс пользователя ... 459
Функции проверки аппаратуры и кабелей ... 460
Функции сбора статистики ... 461
Функции анализа протоколов ... 463
7.3.6. Мониторинг локальных сетей на основе коммутаторов ... 463
Наблюдение за трафиком ... 463
Управление виртуальными сетями ... 464
Выводы ... 464
Вопросы и упражнения ... 465
Заключение ... 465
Приложение ... 466
Ответы на вопросы ... 466
Глава 1 ... 466
Глава 2 ... 467
Глава 3 ... 467
Глава 4 ... 468
Глава 5 ... 469
Глава 6 ... 470
Глава 7 ... 470
Рекомендуемая литература ... 471
Содержание ... 472