

2. Основные проблемы построения сетей

При создании компьютерных сетей их разработчикам пришлось решить много проблем. В этой лекции 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.

2.1. Связь компьютера с периферийными устройствами

Для обмена данными между компьютером и периферийным устройством (ПУ) в компьютере предусмотрен внешний интерфейс (рисунок 2.1), то есть набор проводов, соединяющих компьютер и периферийное устройство, а также набор правил обмена информацией по этим проводам (протокол работы интерфейса). Примерами интерфейсов, используемых в компьютерах, являются параллельный интерфейс Centronics, предназначенный для подключения первых принтеров, и последовательный интерфейс RS-232C, через который раньше подключались мышь, модем и много других устройств. Сегодня им на смену пришла шина USB.

Как вам уже известно из курса ОС, 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 по-разному — одни лучше, а другие хуже — в зависимости от опыта и способностей программистов, их разработавших.

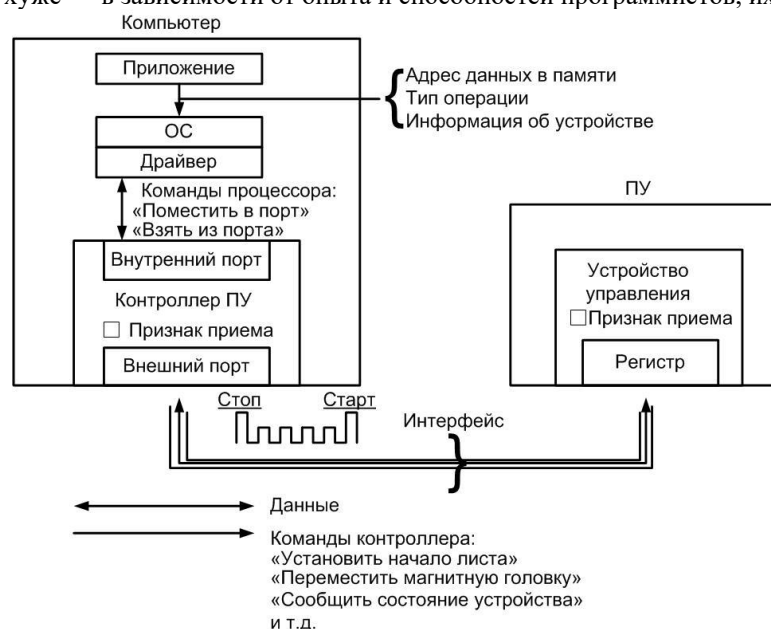


Рисунок 2.1. Связь компьютера с периферийным устройством

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».

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 (рисунок 2.2). Предположим, что на дворе 1991 год, и мы связали эти компьютеры по кабелю связи через COM-порты, которые, как известно, реализовали интерфейс RS-232C (такое соединение часто называли нуль-модемным). Пусть для определенности компьютеры работали 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, 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.

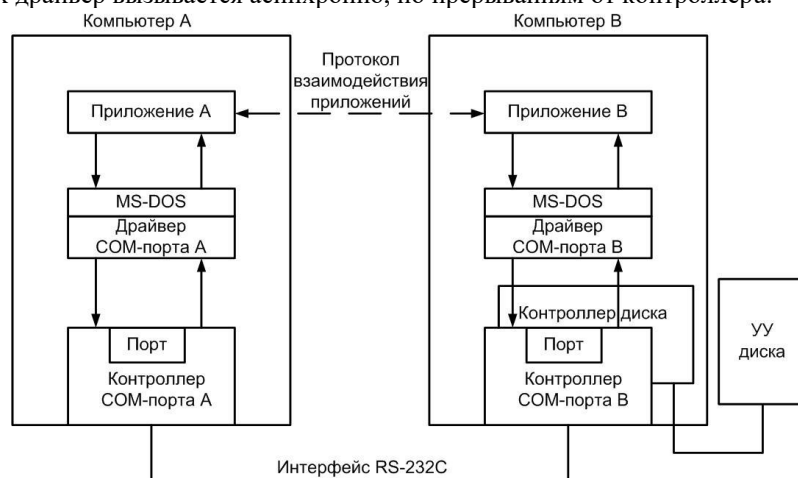


Рисунок 2.2. Взаимодействие двух компьютеров

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 СОМ-портов. Для определенности назовем такие программы 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 СОМ-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 СОМ-порта передает считанные данные 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 существовало большое количество программ, которые действительно самостоятельно решали все задачи по межмашинному обмену данными, например файловый менеджер 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 сказано в лекции 1, 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 рисунке 2.3. 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, работающих в сети с более сложными аппаратными связями компьютеров.

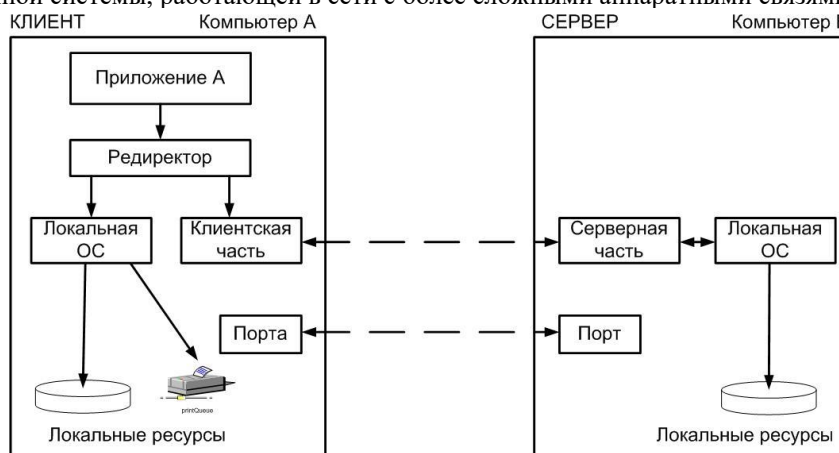


Рисунок 2.3. Взаимодействие программных компонентов при связи двух компьютеров

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.

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 значительным 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*. 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. Модуляция использовалась в глобальных сетях при передаче данных через аналоговые телефонные каналы связи, которые были разработаны для передачи голоса в аналоговой форме и поэтому плохо подходили для непосредственной передачи импульсов. Сегодня модуляция широко используется в беспроводных технологиях передачи данных.

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 декодирует 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, оптоволоком и т. п. Каждый тип передающей среды обладает определенными электрическими характеристиками, влияющими на способ использования данной среды, и определяя скорость передачи сигналов, способ их кодирования и некоторые другие параметры.

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 (рисунок 2.4, 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 получается из полностью связанной путем удаления некоторых возможных связей (рисунок 2.4, 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 (рисунок 2.4, в) распространена в беспроводных сетях (а в первое время была и самой распространенной топологией для локальных сетей). В этом случае компьютеры подключаются к одному общему каналу передачи данных. 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 или же использование радиоволн. 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* (рисунок 2.4, г). В этом случае каждый компьютер подключается отдельным кабелем к общему устройству, называемому *концентратором*, который находится в центре сети (*в наши дни концентраторы заменили на коммутаторы с целью повышения производительности 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 (рисунок 2.4, д). В настоящее время иерархическая звезда является самым распространенным типом топологии связей как в локальных, так и глобальных сетях.

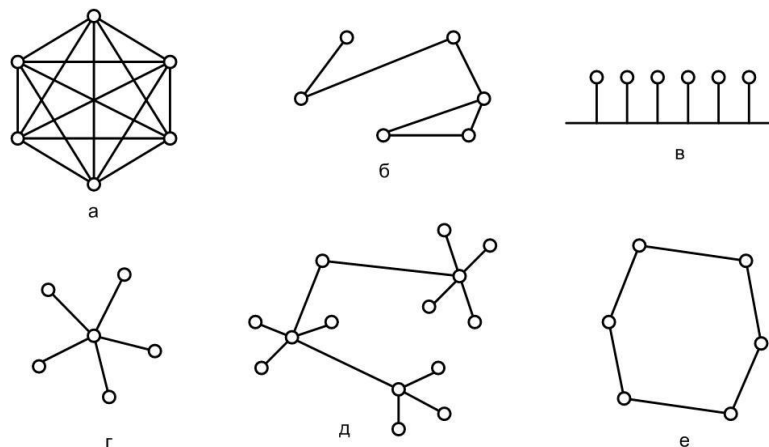


Рисунок 2.4. Типовые топологии сетей

В сетях с *кольцевой* конфигурацией (рисунок 2.4, 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.

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* (рисунок 2.5).

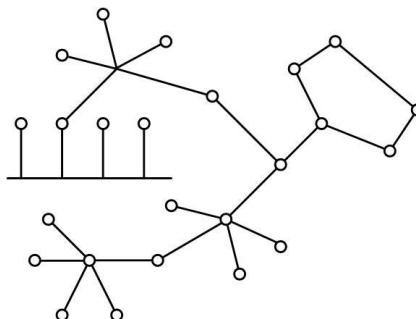


Рисунок 2.5. Смешанная топология

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 использовались очень часто. Этот подход, в частности, был реализован в широко распространенных ранее классических технологиях Ethernet и Token Ring. Однако в настоящее время от разделяемых сред передачи данных в **проводных** локальных сетях практически повсеместно отказались. Это связано с тем, что за достигаемое таким образом удешевление сети приходилось 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 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 переход к их использованию начался с конца 90-х годов XX века.

В глобальных сетях отказ от разделяемых линий связи объясняется техническими причинами. Здесь большие временные задержки распространения сигналов принципиально ограничивают применимость техники разделения линий 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.

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 символическое представление например, 224main 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-archl.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 составляющая полного имени, то есть имя ftp-archl.

- *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. Типичными представителями адресов этого типа являются IP-адреса. В 4 версии IP-протокола поддерживается двухуровневая иерархия, адрес делится на номер сети и 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 составляющие. 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.

2.5. Ethernet — пример стандартного решения сетевых проблем

Рассмотрим, каким образом описанные выше общие подходы к решению наиболее важных проблем построения сетей были воплощены в знакомой нам из курса ТЛВС 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. 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, как вы помните, был принят в 1980 году. 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 (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» (рисунок 2.6). 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 MAC-адрес присваиваемый, как правило, производителем. Передача данных происходит со скоростью 10 Мбит/с. Эта величина являлась пропускной способностью сети Ethernet, хотя мы показали что в зависимости от размера кадров и загрузки сети реальная скорость передачи данных могла существенно отличаться от этого значения.

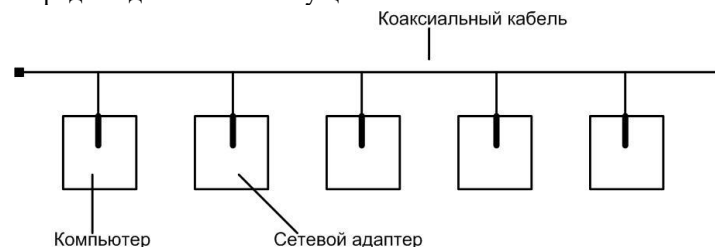


Рисунок 2.6. Сеть Ethernet

Суть случайного метода доступа как вы помните, состояла в следующем: компьютер в сети Ethernet мог 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 являлась процедура определения доступности среды.

После того как компьютер убедился, что сеть свободна, он начинал передачу, при этом «захватывая» среду. Время монопольного использования разделяемой среды одним узлом ограничивалось временем передачи одного кадра. *Кадр* — это единица данных, которыми обменивались компьютеры в сети Ethernet. Кадр имел уже известный нам из курса ТЛВС фиксированный формат и наряду с полем данных содержал различную служебную информацию, например адрес получателя и адрес отправителя.

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

Иногда могла возникать ситуация, когда одновременно два или более компьютера решали, что сеть свободна, и начинали передавать информацию. Такая ситуация, называемая *коллизией*, препятствовала правильной передаче данных по сети. В стандарте Ethernet поэтому был предусмотрен алгоритм обнаружения и корректной обработки коллизий. Вероятность возникновения коллизии зависела от интенсивности сетевого трафика. После обнаружения коллизии сетевые адаптеры, которые пытались

передать свои кадры, прекращали передачу и после паузы случайной длительности пытались снова получить доступ к среде и передать тот кадр, который вызвал коллизиию.

Главным достоинством сетей Ethernet, благодаря которому они стали такими популярными, являлась их экономичность. Для построения сети достаточно было 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 требовали наличия дополнительного устройства — концентратора.

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

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

Другие базовые сетевые технологии — Token Ring, FDDI, 10VGAny-LAN, хотя и обладали многими индивидуальными чертами, в то же время имели 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 были связаны с особенностями используемого метода доступа к разделяемой среде. Так, отличия технологии Ethernet от технологии Token Ring во многом определялись спецификой заложенных в них методов разделения среды — случайного алгоритма доступа в Ethernet и метода доступа путем передачи маркера в Token Ring.

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 узла 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, как нам уже известно, технология Ethernet на тонком коаксиальном кабеле позволяла использовать кабель длиной не более 185 метров, к которому можно было подключить не более 30 компьютеров. Однако, если компьютеры интенсивно обменивались информацией между собой, приходилось 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 как вам известно из курса ТЛВС 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 (рисунок 2.7). Повторитель позволяет преодолеть ограничения на длину линий связи за счет улучшения качества передаваемого сигнала — восстановления его мощности и амплитуды, улучшения фронтов и т. п.

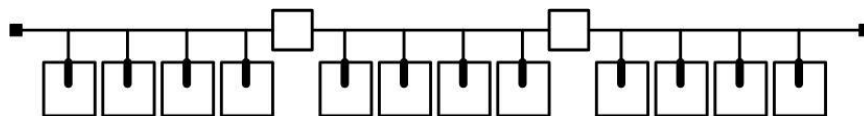


Рисунок 2.7. Повторитель позволяет увеличить длину сети Ethernet

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

Концентраторы характерны практически для всех базовых технологий локальных сетей — Ethernet,

Token Ring, FDDI, 100VG-AnyLAN, Fast Ethernet, Gigabit Ethernet.

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 (рисунок 2.8, а). А концентратор Token Ring (рисунок 2.8, б) повторяет входные сигналы, поступающие с некоторого порта, только на одном порту — на том, к которому подключен следующий в кольце компьютер.

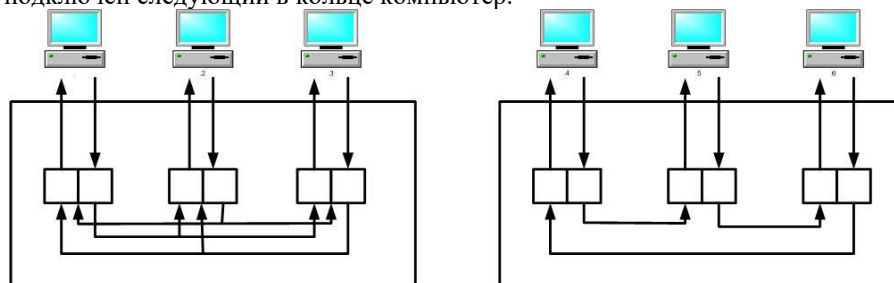


Рисунок 2.8. Концентраторы различных технологий

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 рисунке 2.9, а, имеет физическую топологию кольца. 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.

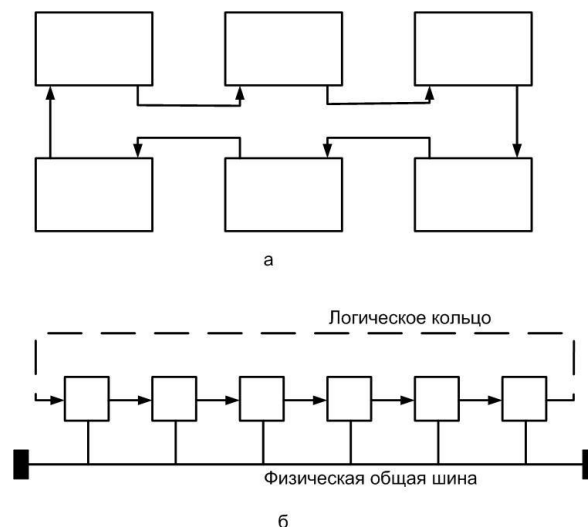


Рисунок 2.9. Логическая и физическая топологии сети

Сеть, показанная на рисунке 2.9, 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.

Другим примером несовпадения физической и логической топологий сети является уже рассмотренная сеть на рисунке 2.8, 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 занятости среды, захват среды, распознавание и обработка 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 по 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. На начальном этапе внедрения локальных сетей наиболее интенсивный обмен данными наблюдался между компьютерами, принадлежащими к одной подсети, и только небольшая часть обращений происходила к ресурсам компьютеров, находящихся вне локальных рабочих групп. 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 рисунок 2.10, 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 сети, 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 (рисунок 2.10, б).

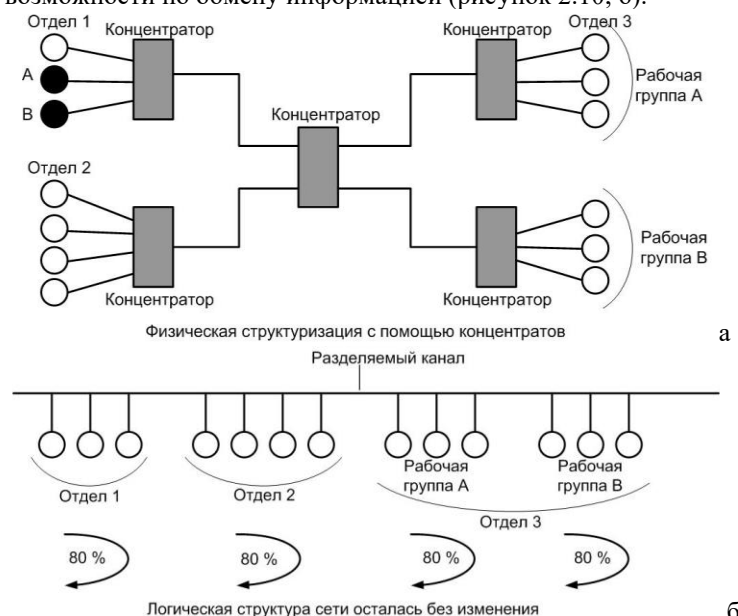


Рисунок 2.10. Противоречие между логической структурой сети и структурой информационных потоков

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, то и пропускная способность линий связи и коммуникационного оборудования, соединяющего отделы, может быть значительно ниже внутреннего трафика сети отдела. Распространение трафика, предназначенного для компьютеров некоторого сегмента сети, только в пределах этого сегмента, называется *локализацией* трафика. Таким образом, *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 рисунке 2.11 показана сеть, которая была получена из сети с центральным концентратором (см. рисунок 2.10) 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.

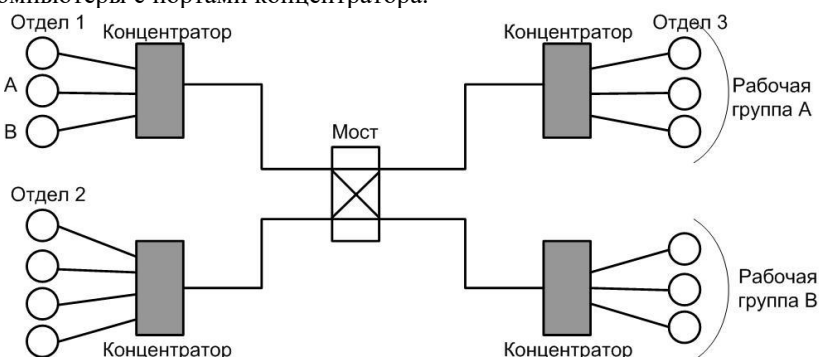


Рисунок 2.11. Логическая структуризация сети с помощью моста

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.

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 (например локальную и глобальную сеть). 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, а для их объединения всегда используется оборудование, обеспечивающее локализацию трафика, — мосты, коммутаторы, маршрутизаторы и шлюзы.

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, прочесть 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 доступа к базе данных 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 реализована 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.

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 физические 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.