**Протоколы поочередного доступа**

Как уже упоминалось, двумя желательными свойствами протокола множественного доступа являются, во-первых, возможность единствен­ного активного узла передавать свои данные с максимальной пропуск­ной способностью канала *R* бит/с, во-вторых, возможность для каждого из *М* активных узлов передавать свои данные со скоростью *R/M* бит/с. Протоколы ALOHA и CSMA удовлетворяют первому требованию, но не удовлетворяют второму. Это подвигло исследователей на создание нового класса протоколов — **протоколов поочередного доступа.** Как и в случае с протоколами произвольного доступа, их существуют де­сятки, и у каждого есть множество вариантов. Здесь мы рассмотрим два наиболее важных протокола поочередного доступа. Первый из них — **протокол опроса.** При использовании протокола опроса один из узлов должен быть назначен главным (управляющим). Главный узел пооче­редно **опрашивает** все узлы. Например, сначала главный узел посылает сообщение узлу 1, сообщая ему, что он может передать некоторое мак­симальное количество кадров. После того как узел 1 передает несколько кадров, главный узел разрешает передать некоторое количество кадров узлу 2. (Главный узел может определить момент завершения передачи очередным узлом по отсутствию сигнала в канале.) Данная процедура продолжается бесконечно, при этом главный узел в цикле опрашивает все узлы.

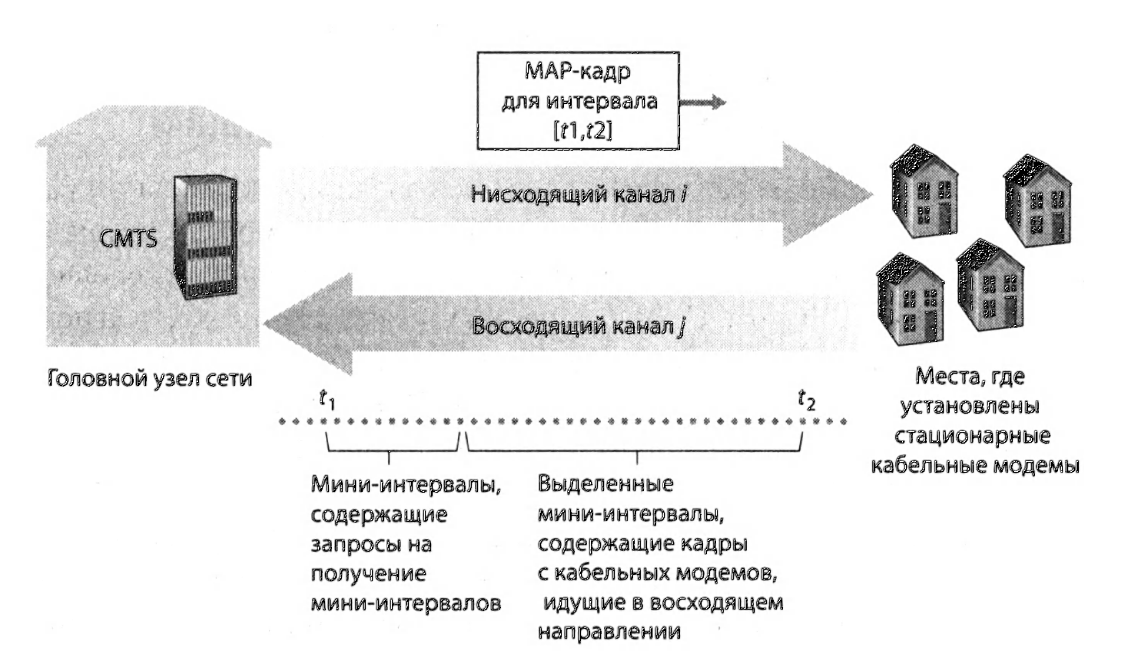
Протокол опроса устраняет коллизии и пустые кванты, от которых страдают протоколы произвольного доступа. Таким образом, эффектив­ность протокола опроса значительно выше. Однако у протокола опро­са есть несколько недостатков. Первый заключается в том, что опреде­ленное время тратится протоколом опроса на саму процедуру опроса, то есть на выдачу узлу разрешения на передачу. Например, если только один узел является активным, тогда он не сможет передавать со сред­ней скоростью, равной полной пропускной способности канала, так как после отправки активным узлом разрешенного количества кадров глав­ный узел будет опрашивать остальные узлы в каждом цикле. Второй недостаток является еще более серьезным. Он заключается в том, что при выходе из строя главного узла вся деятельность канала прекраща­ется. В качестве примеров протоколов опроса можно привести 802.15 и Bluetooth.

Второй протокол поочередного доступа — **протокол с передачей маркера.** В этом протоколе главного узла не существует. Все узлы, при­соединенные к широковещательному каналу, обмениваются небольшим специальным кадром, называемым **маркером** (токеном). Порядок об­мена маркера фиксирован. Например, узел 1 всегда посылает маркер узлу 2, а узел 2 всегда посылает маркер узлу 3 и т. д.; а узел *N* всегда посылает маркер узлу 1. Получив маркер, узел удерживает его, только если у него есть данные для передачи; в противном случае он немедлен­но передает маркер следующему узлу. Если к моменту получения мар­кера у узла есть кадры для передачи, он отправляет некое максимальное количество кадров, после чего пересылает маркер следующему узлу. Пе­редача маркера осуществляется децентрализованно и обладает высокой эффективностью. Но проблемы могут возникнуть и в данной схеме. На­пример, выход из строя одного узла может вывести из строя весь канал, а если какой-либо узел забудет передать маркер, потребуется специаль­ная процедура вывода канала из тупиковой ситуации. За многие годы было разработано множество протоколов с передачей маркера, в част­ности, FDDI (протокол волоконно-оптического интерфейса передачи данных) и протокол IEEE 802.5 для передачи маркера по сети с коль­цевой конфигурацией, в каждом из которых приходилось решать эти и другие неприятные вопросы.

**DOCSIS: протокол канального уровня для кабельного доступа в Интернет**

В предыдущих трех разделах мы познакомились с тремя большими классами протоколов множественного доступа: протоколы с разделени­ем канала, протоколы произвольного доступа и протоколы поочередно­го доступа. Очень удобно показать приемы работы с ними на примере кабельной сети доступа, поскольку здесь мы найдем различные аспекты использования *каждого* из этих трех классов протоколов множествен­ного доступа!

Как вы помните, кабельная сеть доступа обычно со­единяет несколько тысяч клиентских кабельных модемов с терминаль­ной станцией кабельных модемов (CMTS), расположенной в головном узле компьютерной сети. Стандарт передачи данных по коаксиальному кабелю (DOCSIS) описывает архитектуру кабельной сети доступа и применяемые в ней протоколы. Стандарт DOCSIS задействует муль­типлексирование с частотным разделением (FDM) для разделения нисходящих (от CMTS к модему) и восходящих (от модема к CMTS) сетевых сегментов на множество частотных каналов. Ширина каждого нисходящего канала составляет 6 МГц, максимальная пропускная спо­собность — около 40 Мбит/с на канал (хотя на практике такая частота передачи данных по кабельному модему достигается редко). Максималь­ная ширина восходящего канала составляет 6,4 МГц, а его максималь­ная пропускная способность — примерно 30 Мбит/с. Все восходящие и нисходящие каналы являются широковещательными. Кадры переда­ются по нисходящему каналу головной станцией и принимаются всеми кабельными модемами, подключенными к этому каналу. Поскольку го­ловная станция всего одна, в этом направлении не возникает проблем с множественным доступом. Управление восходящими потоками — бо­лее интересная и технически сложная задача, поскольку один и тот же канал одновременно используется множеством кабельных модемов, от­правляющих свои данные на головную станцию. Естественно, в таких случаях могут возникать коллизии.

Как показано на рис. 5.14, все восходящие каналы работают с разде­лением времени на интервалы (подобно механизму TDM). Каждый интервал содержит последовательность мини-интервалов, в ходе которых кабельные модемы могут передавать информацию CMTS. CMTS явно выделяет конкретным кабельным модемам право на передачу информа­ции в ходе тех или иных мини-интервалов. Для этого CMTS отправляет по нисходящему каналу специальное управляющее сообщение, называе­мое MAP, в котором указывает, какой кабельный модем (располагающий данными для отправки) может их передать в течение мини-интервала, приходящегося на период, обозначенный в МАР-сообщении. Посколь­ку мини-интервалы явно выделяются кабельным модемам, головная станция может гарантировать, что в ходе мини-интервала никаких кол­лизий не возникнет.

**Рис. 5.14. Восходящие и нисходящие каналы между головной узловой станцией**

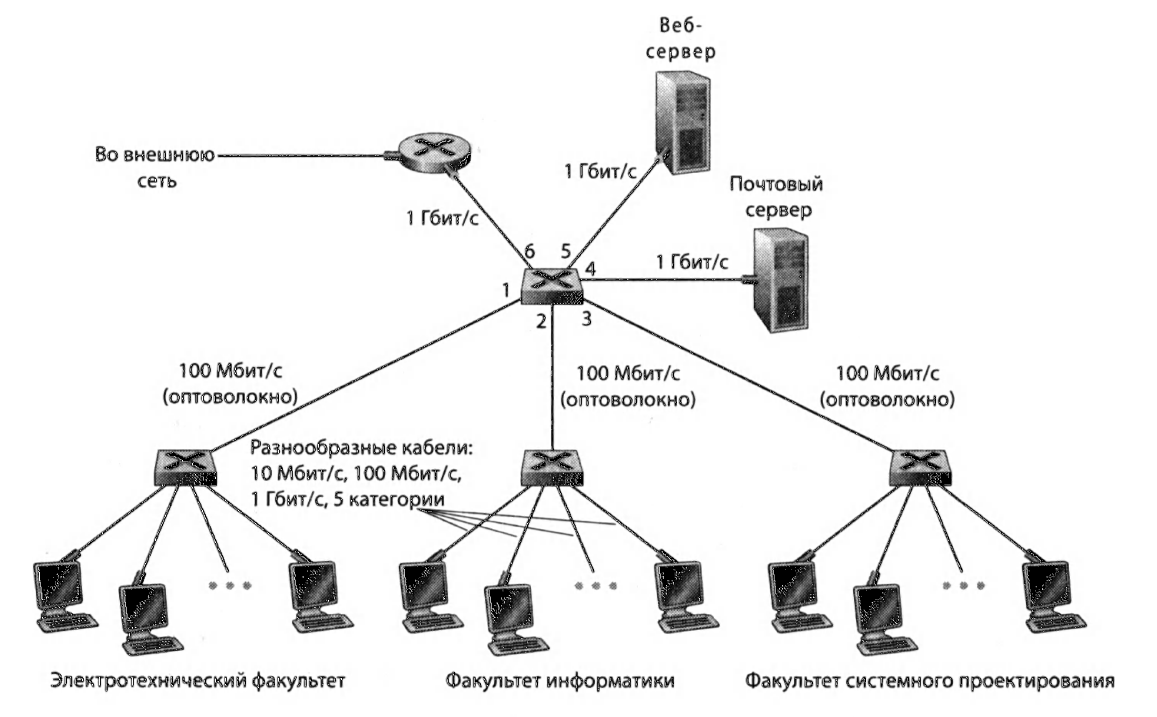
**и кабельными модемами**

Но как же CMTS узнает, какие из кабельных модемов располагают данными для отправки? Для этого модемы отправляют CMTS кадры с запросом мини-интервалов. Для передачи таких запросов выделяются специальные мини-интервалы, предназначенные именно для этой цели, как показано на рис. 5.14. Такие кадры с запросами на получение мини-интервалов отправляются с расчетом на произвольный доступ, поэтому между ними могут возникать коллизии. Кабельный модем не может пу­тем опроса узнать, занят ли восходящий канал, равно как не может об­наружить коллизии. Вместо этого модем логически заключает, что его кадр с запросом мини-интервала претерпел коллизию, если не получает отклика на запрошенное выделение ресурса в следующем нисходящем управляющем сообщении. Когда таким образом логически установлен факт коллизии, модем применяет экспоненциальный двоичный алгоритм задержки для повторного запроса о мини-интервале на будущий квант времени. Если в восходящем канале не так много трафика, то кабельный модем может передавать кадры с данными в течение тех квантов, которые номинально выделяются для запросов о мини-интервалах (следователь­но, модему не приходится ждать получения такого мини-интервала).

Итак, кабельная сеть доступа — превосходный образец практическо­го применения протоколов множественного доступа. Здесь мы наблю­даем и технологии FDM и TDM, и произвольный доступ, и централизо­ванно выделяемые кванты времени — все в одной сети!

**Локальная сеть с коммутируемым доступом**

сетям. На рис. 5.15 представлена такая сеть, в которой при помощи че­тырех коммутаторов соединяются три факультета, два сервера и марш­рутизатор.

**Рис. 5.14. Университетская сетьосходящие и нисходящие каналы между головной узловой станциейи кабельными модемами**

Поскольку эти коммутаторы работают на канальном уровне, они пе­реключают кадры канального уровня (а не дейтаграммы сетевого уров­ня) и не различают сетевых адресов, а также не используют алгоритмов маршрутизации — в частности, RIP или OSPF, которые обеспечивают прокладку путей по сети в коммутаторах уровня 2. Как мы вскоре уви­дим, для перенаправления пакетов в сети коммутаторов используются не сетевые адреса, а адреса канального уровня. Мы начнем изучение коммутации в локальных сетях с адресации канального уровня. Затем мы поговорим о широко известном протоколе Ethernet. После этого мы обратимся к тому, как функционируют коммутаторы канального уровня, а потом изучим, как такие коммутаторы применяются для создания крупномас­штабных локальных сетей.

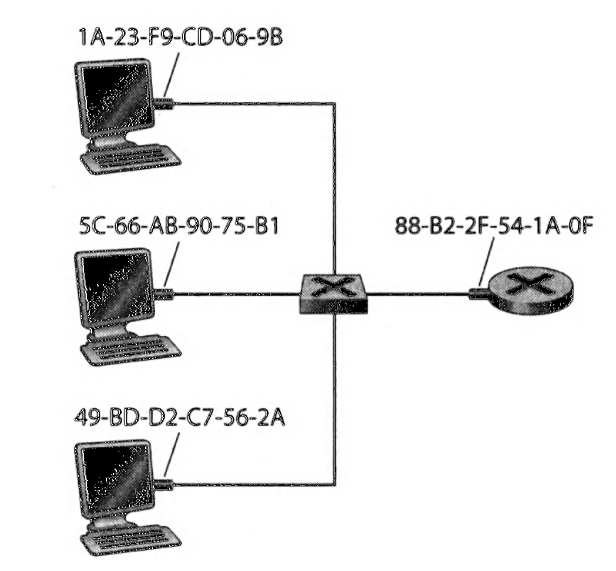
**Адресация канального уровня и протокол ARP**

У хостов и маршрутизаторов есть адреса канального уровня. Воз­можно, сейчас вам это кажется удивительным — ведь ранее было ска­зано, что у хостов и маршрутизаторов есть и адреса сетевого уровня. Мо­жет возникнуть вопрос: зачем же нужны адреса как на канальном, так и на сетевом уровне? Сейчас мы не только опишем синтаксис и функционирование адресации канального уровня, но и постараемся объяснить, почему два уровня адресации не только полезны, но и, в сущ­ности, необходимы. Кроме того, мы рассмотрим протокол разрешения адресов (ARP), который обеспечивает трансляцию (преобразование) IP-адресов в адреса канального уровня.

**МАС-адреса**

В действительности адрес канального уровня есть не у узла, а у адап­тера (то есть у сетевого интерфейса). Следовательно, хост или маршру­тизатор с несколькими сетевыми интерфейсами будет иметь несколько адресов канального уровня. При этом важно отметить, что те интер­фейсы коммутаторов канального уровня, которые соединяются с хоста­ми и маршрутизаторами, не снабжаются такими адресами. Дело в том, что задача коммутатора канального уровня — передавать дейтаграммы между хостами и маршрутизаторами. Коммутатор прозрачно выполня­ет эту работу — то есть хосту или маршрутизатору не приходится явно сообщать адрес кадра тому коммутатору, через который этот кадр про­ходит. Это иллюстрирует рис. 5.16. Адрес в локальной сети, или **LAN-адрес,** также называют **физическим адресом, Ethernet-адресом** или

**МАС-адресом** (Media Access Control — управление доступом к среде передачи). В большинстве локальных сетей (включая Ethernet-сети и беспроводные локальные сети стандарта 802.11) МАС-адрес представ­ляет собой 6-байтовое число, что позволяет использовать 248 возможных адресов. Как показано на рис. 5.16, эти 6-байтовые адреса, как правило, изображаются в шестнадцатеричном виде, при этом каждый байт адреса записывается как пара шестнадцатеричных цифр. Хотя адрес адаптера в локальной сети изначально задумывался как постоянный, в настоя­щее время существует возможность программно изменять МАС-адрес адаптера. Мы будем считать, что МАС-адрес адаптера является постоянным.

**Рис. 5.16. Каждый из интерфейсов, подключенных к локальной сети, имеет**

**уникальный МАС-адрес**

Одно интересное свойство МАС-адресов заключается в том, что не может существовать двух адаптеров с одинаковыми адресами. Это может показаться удивительным, так как сетевые адаптеры произво­дятся в разных странах, разными производителями. Как может компа­ния, производящая адаптеры в Тайване, гарантировать, что адреса ее устройств будут отличаться от адресов адаптеров, производимых другой компанией в Бельгии? Дело в том, что физическим адресным простран­ством управляет институт IEEE (Institute of Electrical and Electronics Engineers — Институт инженеров по электротехнике и электронике). В частности, когда компания хочет выпускать адаптеры, она приобре­тает блок адресного пространства, состоящий из 224 адресов. IEEE выделяет блок из 224 адресов, фиксируя старшие 24 бита физического адреса и позволяя компании создавать уникальные комбинации из младших 24 разрядов для каждого адаптера.

Таким образом, МАС-адреса адаптеров образуют плоскую (а не ие­рархическую) структуру и не изменяются при перемещении адаптеров. У ноутбука с Ethernet-картой всегда один и тот же М АС-адрес независи­мо от того, где находится этот компьютер. Вспомним, что IP-адреса, на­против, образуют иерархическую структуру (то есть делятся на сетевую и хостовую части), и при перемещении хоста IP-адрес узла должен быть изменен. Таким образом, МАС-адрес адаптера аналогичен номеру кар­точки социального страхования (эти номера также образуют плоскую адресную структуру и не изменяются при перемещении их владельца). IP-адрес можно сравнить с почтовым адресом человека — иерархиче­ским и изменяющимся при смене места жительства владельца. Многие люди активно пользуются как номерами социального страхования, так и почтовыми адресами. Аналогично, на хосте и интерфейсе полезно иметь как адрес сетевого уровня, так и МАС-адрес.

Когда адаптер хочет переслать кадр другому адаптеру в той же ло­кальной сети, передающий адаптер помещает в кадр адрес получателя в локальной сети. Как мы вскоре увидим, коммутатор может передавать входящий кадр на все свои интерфейсы. Далее мы увидим, что в се­тях 802.11 также выполняется широковещательная передача кадров. Это значит, что адаптер может получить кадр, который ему не предназначен. Следовательно, получив кадр, адаптер проверяет, совпадает ли целевой МАС-адрес в этом кадре с собственным МАС-адресом этого адаптера. Если зафиксировано совпадение, принимающий адаптер извлекает из кадра дейтаграмму и передает ее вверх по стеку протоколов. Если совпа­дения нет, то адаптер отбрасывает этот кадр, не передавая дейтаграмму сетевого уровня вверх по стеку протоколов. Таким образом, узел прини­мает и обрабатывает только те кадры, которые адресованы ему.

Однако иногда передающий узел *бывает* заинтересован в том, чтобы все адаптеры в локальной сети приняли и *обработали* кадр, который он посылает. В этом случае передающий адаптер помещает в поле адреса получателя специальный **широковещательный адрес** MAC. В локаль­ных сетях, использующих 6-байтовые адреса (как, например, Ethernet или 802.11), широковещательный адрес представляет собой строку из 48 двоичных единиц (то есть FF-FF-FF-FF-FF-FF в шестнадцатерич-ной нотации).

**ПРИНЦИПЫ В ДЕЙСТВИИ**

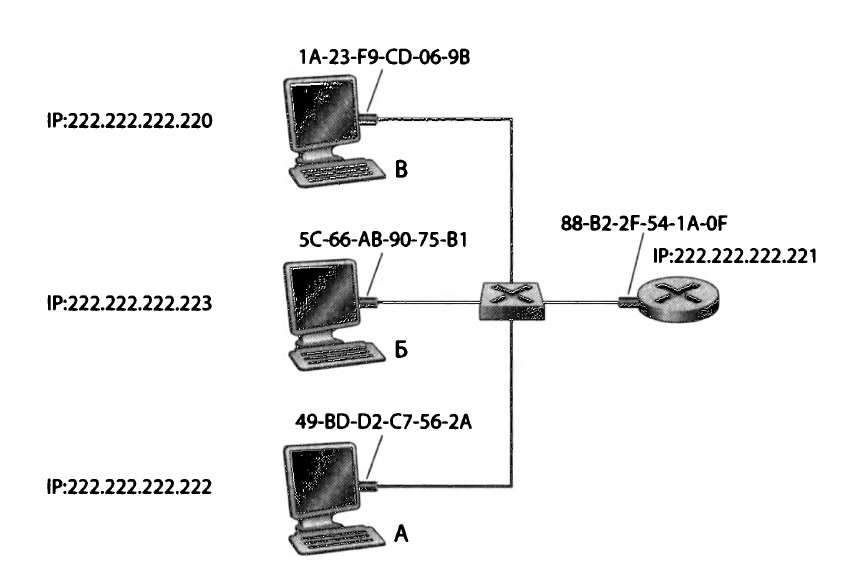
**Обеспечение независимости уровней**

Есть несколько причин, по которым узлам помимо адресов сете­вого уровня выделяются МАС-адреса. Во-первых, локальные сети предназначены для работы с произвольными протоколами сетево­го уровня, а не только с протоколом IP и Интернетом. Если бы вме­сто «нейтральных» МАС-адресов адаптерам назначались IP-адреса, адаптерам было бы трудно поддерживать другие протоколы сете­вого уровня (например, IPX или DECnet). Во-вторых, если бы адап­теры должны были использовать IP-адреса вместо МАС-адресов, тогда адрес сетевого уровня пришлось бы хранить в оперативной памяти адаптера и перенастраивать его при каждом перемещении устройства или при включении питания. Правда, можно было бы во­обще не использовать адреса в адаптерах и передавать данные (как правило, IP-дейтаграммы) каждого полученного адаптером кадра вверх по стеку протоколов. После этого родительский узел мог бы проверять соответствие адреса сетевого уровня. Недостаток такого варианта состоит в том, что в этом случае родительский узел пре­рывался бы каждым кадром, посланным по локальной сети, вклю­чая кадры, предназначающиеся другим узлам той же широковеща­тельной сети. Таким образом, для большей независимости уровней в стеке протоколов уровни должны обладать собственными схема­ми адресации. Мы уже ознакомились с тремя типами адресов: име­нами хостов на прикладном уровне, IP-адресами на сетевом уровне и МАС-адресами на канальном уровне.

**Протокол разрешения адресов**

При передаче дейтаграмм используются и адреса сетевого уровня (например, IP-адреса Интернета), и адреса канального уровня (то есть МАС-адреса), поэтому возникает потребность в преобразовании одних адресов в другие. В Интернете эту работу выполняет **протокол разре­шения адресов** (Address Resolution Protocol, ARP).

Чтобы понять, зачем нужен протокол ARP, рассмотрим сеть, изобра­женную на рис. 5.17. В этом простом примере у каждого узла есть IP-адрес, а у адаптера каждого узла есть МАС-адрес. IP-адреса, как обычно, представляются в виде четырех десятичных чисел, а МАС-адреса пока­заны в шестнадцатеричном виде. В этом разделе мы будем считать, что коммутатор передает все кадры широковещательно. То есть как только коммутатор получает кадр на любом интерфейсе, он перенаправляет этот кадр на все другие свои интерфейсы. В следующем разделе будет более подробно описано, как именно работают коммутаторы.



**Рис. 5.17. Каждый интерфейс в локальной сети имеет IP-адрес и МАС-адрес**

Теперь предположим, что хост с IP-адресом 222.222.222.220 хочет послать IP-дейтаграмму хосту 222.222.222.222. В данном примере и ис­ходный, и целевой хост относятся к одной и той же подсети, их адреса­ция организована так, как описано в разделе 4.2.2. Чтобы выполнить эту задачу, передающий хост должен отправить адаптеру не только IP-дей­таграмму, но также МАС-адрес узла 222.222.222.222. Получив IP-дейта­грамму и МАС-адрес узла, адаптер передающего хоста формирует кадр канального уровня, содержащий МАС-адрес принимающего узла, и пе­редает кадр в локальную сеть.

В этом разделе нам требуется ответить на важный вопрос: каким образом передающий хост определяет МАС-адрес хоста с IP-адресом 222.222.222.222? Как вы уже догадались, он делает это с помощью мо­дуля ARP. В данном случае ARP-модуль исходного хоста принима­ет в качестве ввода любой IP-адрес из своей локальной сети, а в ответ возвращает соответствующий МАС-адрес. Здесь хост-отправитель с IP-адресом 222.222.222.220 указывает своему ARP-модулю IP-адрес 222.222.222.222. На это ARP-модуль отвечает соответствующим LAN-адресом узла, то есть адресом 49-BD-D2-C7-56-2A.

Итак, ARP-модуль преобразует IP-адрес в МАС-адрес узла. Во мно­гом это аналогично системе DNS (см. раздел 2.5), преобразующей имена хостов в IP-адреса. Однако важное различие между этими двумя схемами преобразования адресов заключается в том, что DNS преобразует имена хостов в IP-адреса во всем Интернете, тогда как протокол ARP занимается только IP-адресами в пределах одной локальной сети. Если бы узел в Калифорнии попытался узнать LAN-адрес для IP-адреса узла в Миссисипи, протокол ARP вернул бы ошибку.

Теперь, выяснив, что делает протокол ARP, посмотрим, как он это делает. У ARP-модуля каждого узла есть оперативная память, в которой хранится **ARP-таблица.** В этой таблице прописаны IP-адреса хостов ло­кальной сети и соответствующие им МАС-адреса. На рис. 5.18 показа­но, как могла бы выглядеть ARP-таблица для узла 222.222.222.220. Для каждой пары адресов в таблице также содержится поле **предписанного времени жизни** (Time To Live, **TTL),** в котором указывается, когда дан­ная запись будет удалена из таблицы. Обратите внимание, что в таблице не обязательно содержатся записи для всех узлов локальной сети. На­пример, записи для одних узлов могут быть удалены, так как время их жизни истекло, тогда как записи для других узлов вообще могут никогда не попасть в эту таблицу. Типичное значение времени жизни — 20 мин с момента помещения записи в ARP-таблицу.

|  |  |  |
| --- | --- | --- |
| **IP-адрес** | **МАС-адрес** | **TTL** |
| **222.222.222.221** | **88-B2-2F-54-1A-0F** | **13:45:00** |
| **222.222.222.223** | **5С-66-АВ-90-75-В1** | **13:52:00** |

**Рис. 5.18. Так могла бы выглядеть ARP-таблица для хоста 222.222.222.220**

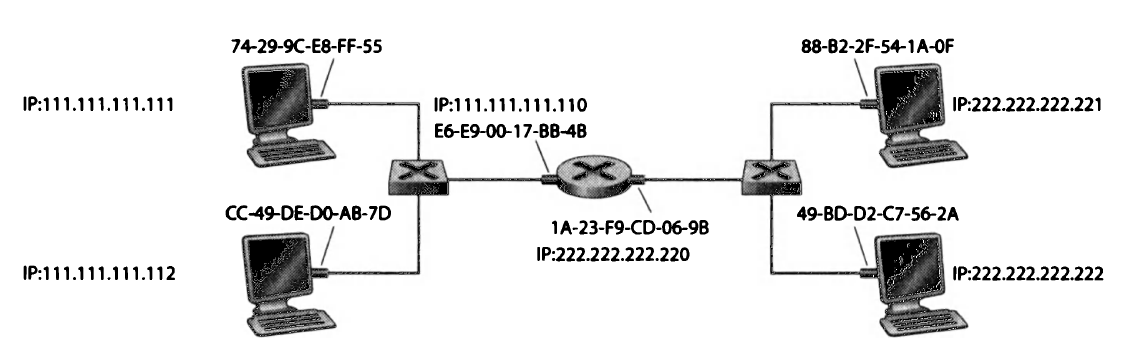
Теперь предположим, что хост 222.222.222.220 хочет отправить дейтаграмму с указанием IP-адреса другому хосту или маршрутизато­ру той же локальной сети. Для этого передающему узлу нужно по IP-адресу получающего узла узнать его МАС-адрес. Эта задача неслож­ная, если в ARP-таблице передающего узла содержится запись для узла-получателя. Но что делать, если такой записи в ARP-таблице нет? Например, пусть узел 222.222.222.220 желает переслать дейтаграм­му узлу 222.222.222.222. В этом случае передающий узел определяет нужный ему адрес при помощи протокола ARP. Сначала передающий узел формирует специальный **ARP-пакет.** В ARP-пакете содержится несколько полей, среди которых есть IP-адреса и МАС-адреса пере­дающего и принимающего узлов. Для обоих ARP-пакетов (запроса и ответа) используется один и тот же формат. Цель ARP-пакета с за­просом состоит в том, чтобы опросить все остальные узлы локальной сети и определить LAN-адрес, соответствующий интересующему нас IP-адресу.

Итак, в нашем примере узел 222.222.222.220 передает ARP-пакет с запросом своему адаптеру вместе с указанием переслать этот пакет по широковещательному МАС-адресу FF-FF-FF-FF-FF-FF. Адаптер ин­капсулирует ARP-пакет в кадр канального уровня, указывает широко­вещательный адрес в поле адреса получателя и передает кадр в подсеть. Если вспомнить нашу аналогию с номерами карточек социального стра­хования и почтовыми адресами, то можно заметить, что ARP-запрос аналогичен человеку, выкрикивающему в переполненном зале с офис­ными кабинками некоторой компании (например, «СоцСтрах»): «Ка­кой номер карточки социального страхования у сотрудника из кабины 112 офиса 13 компании «СоцСтрах», Подольск, Московская область?» Кадр с ARP-запросом принимается всеми остальными адаптерами под­сети, и (поскольку в запросе использовался широковещательный адрес) каждый адаптер передает содержащийся в кадре ARP-пакет своему узлу. Каждый узел проверяет, совпадает ли его IP-адрес с указанным IP-адре­сом получателя в ARP-пакете. Узел, обнаруживший совпадение, посы­лает запрашивающему узлу ответный ARP-пакет с указанным в нем соответствующим МАС-адресом. После этого запрашивающий узел 222.222.222.220 может обновить свою ARP-таблицу и отправить IP-дей­таграмму, заключенную в кадр канального уровня, где МАС-адрес на­значения соответствует адресу хоста или маршрутизатора, ответившего на предыдущий ARP-запрос.

Следует сделать два интересных замечания о протоколе ARP. Во-пер­вых, ARP-запрос посылается в широковещательном кадре, а ответ пере­дается в стандартном кадре. Прежде чем продолжить чтение, подумайте, почему. Во-вторых, в протоколе ARP реализован принцип самонастрой­ки (plug-and-play), так как ARP-таблица узла формируется автоматиче­ски — ее не нужно настраивать администратору вычислительной сети. И если узел отсоединяется от подсети, соответствующая ему запись по истечении времени жизни удаляется из таблицы.

Зачастую студенты интересуются, к протоколам какого уровня от­носится ARP — сетевого или канального? Как мы убедились, пакет ARP инкапсулируется в кадр канального уровня и, следовательно, архитек­турно располагается выше канального уровня. Однако в ARP-пакете есть поля, содержащие адреса канального уровня, что позволяет считать его протоколом канального уровня. Правда, у него есть и адреса сетевого уровня, что позволяет с тем же успехом относить его к протоколам сетевого уровня. Таким образом, правильнее всего трактовать его как граничный протокол, не вписывающийся в тот простой стек протоко­лов, который мы изучили ранее. Да, на практике встречаются и такие сложности!

**Передача дейтаграммы узлу за пределы подсети**

Теперь должно быть ясно, как работает протокол ARP, когда узел хо­чет послать дейтаграмму другому узлу *той же самой подсети.* Но теперь мы рассмотрим более сложную ситуацию, в которой хост подсети хочет послать дейтаграмму сетевого уровня хосту, находящемуся *за предела­ми подсети* (то есть в другую подсеть). Обсудим этот вопрос на при­мере сети, состоящей из двух подсетей, соединенных маршрутизатором (рис. 5.19).

**Рис. 5.19. Две подсети, соединенные маршрутизатором**

На рис. 5.19 следует отметить ряд интересных вещей. У каждого хоста есть ровно один IP-адрес и один адаптер. Но, как было показа­но в разделе 4.4 главы 4, у маршрутизатора есть по одному IP-адресу для *каждого* интерфейса. У каждого интерфейса маршрутизатора есть собственный ARP-модуль и собственный адаптер. У изображенного на рис. 5.19 маршрутизатора два интерфейса, поэтому у него два IP-адреса, два ARP-модуля и два адаптера. Разумеется, у каждого адаптера есть свой МАС-адрес.

Обратите также внимание, что адреса всех интерфейсов подсети 1, имеют вид 111.111.111/24, а адреса всех интерфейсов, соединен­ных с подсетью 2, имеют вид 222.222.222/24. Следовательно, все ин­терфейсы, подключенные к подсети 1, имеют адреса вида 111.111.111. ххх, а все интерфейсы, подключенные к подсети 2, имеют адреса вида [222.222.222.xxx](http://222.222.222.xxx).

Теперь предположим, что хост 111.111.111.111 из подсети 1 хочет переслать IP-дейтаграмму хосту 222.222.222.222 из подсети 2. Переда­ющий хост, как обычно, отправляет дейтаграмму своему адаптеру. Но при этом передающий хост должен указать адаптеру МАС-адрес. Какой МАС-адрес следует использовать? Соблазнительно предположить, что это должен быть локальный адрес адаптера хоста 222.222.222.222, то есть 49-BD-D2-C7-56-2A. Однако это неверно. Если передающий адап­тер будет использовать этот МАС-адрес, тогда ни один из адаптеров подсети 1 не станет передавать такую IP-дейтаграмму своему сетевому уровню, так как адрес получателя в кадре не совпадет с М АС-адресом ни одного адаптера подсети 1. Переданная дейтаграмма просто умрет и по­падет в свой дейтаграммный рай.

Если внимательно посмотреть на рис. 5.19, то можно заметить, что для передачи дейтаграммы в подсеть 2 ее нужно направлять интерфейсу маршрутизатора по адресу 111.111.111.110, а этот маршрутизатор рас­полагается на расстоянии одного перехода на пути к конечному узлу. Таким образом, в поле МАС-адреса получателя кадра следует указы­вать адрес адаптера интерфейса маршрутизатора 111.111.111.110, то есть Е6-Е9-00-17-ВВ-4В. Как передающему хосту узнать МАС-адрес узла 111.111.111.110? С помощью протокола ARP, разумеется! Как только пе­редающий адаптер получает этот МАС-адрес, он создает кадр и посылает его в подсеть 1. Адаптер маршрутизатора в подсети 1 видит, что данный кадр канального уровня адресован ему, и поэтому передает кадр сетево­му уровню маршрутизатора. Ура! IP-дейтаграмма была успешно переме­щена от хоста-отправителя на маршрутизатор! Но мы еще не закончили. Нам нужно еще переслать дейтаграмму от маршрутизатора получателю. Теперь маршрутизатор должен определить, по какому интерфейсу сле­дует переслать дейтаграмму. Как было показано ранее, выбор делается при помощи таблицы перенаправления, хранящейся на маршрутизаторе. В этой таблице маршрутизатор находит запись, по которой определяет, что дейтаграмму следует отправить через интерфейс 222.222.222.220. Этот интерфейс передает дейтаграмму своему адаптеру, который поме­щает ее в новый кадр и посылает этот кадр в подсеть 2. Теперь уже МАС-адрес кадра указывает на его конечного получателя. И как же маршрути­затор узнает его МАС-адрес? При помощи протокола ARP, конечно!

**Стандарт Ethernet**

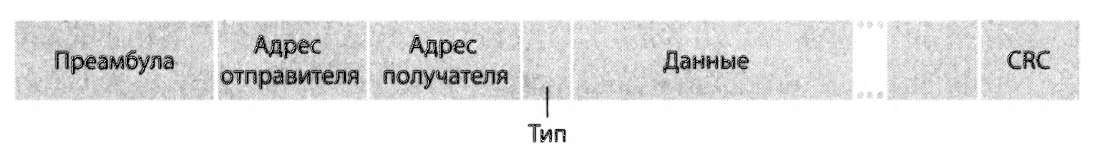
Технология Ethernet очень популярна на рынке локальных сетей. В 80-е годы и в начале 90-х технологии Ethernet приходилось конкуриро­вать с множеством альтернативных технологий локальных сетей, включая маркерный доступ по сети с кольцевой конфигурацией (token ring), FDDI и ATM. Некоторым из этих технологий удалось на несколько лет захватить часть рынка. Но технология Ethernet, разработанная в середине 70-х, про­должает расти и развиваться до сих пор, а в последние годы она прочно за­няла доминирующее положение на рынке. Сегодня Ethernet представляет собой превалирующую технологию локальных сетей, и, похоже, такая си­туация сохранится в обозримом будущем. Значение Ethernet в локальных сетях так же велико, как и значение Интернета в глобальных сетях.

Успеху технологии Ethernet способствовало множество причин. Во-первых, Ethernet была первой локальной сетью, получившей широкое распространение. Благодаря этому администраторы вычислительной сети очень близко познакомились с технологией Ethernet и уже не хотели переходить на другие технологии локальных сетей, когда те появились на рынке. Во-вторых, такие технологии, как сети маркерного доступа, FDDI и ATM, были более сложными и дорогими, чем Ethernet, что еще больше препятствовало переходу на них. В-третьих, наиболее сильным стимулом перехода на другие технологии локальных сетей (например, FDDI или ATM), как правило, являлась более высокая скорость пере­дачи данных. Однако Ethernet всегда отвечала ударом на удар, и каждая новая версия Ethernet не уступала конкурентам, а то и превосходила их по данному параметру. В начале 90-х была представлена коммутируемая Ethernet-сеть, что позволило снова повысить эффективную скорость передачи данных. Наконец, благодаря популярности Ethernet аппарату­ра Ethernet (в частности, адаптеры и коммутаторы) стала массово про­изводиться и дешеветь.

Первая локальная Ethernet-сеть была придумана Бобом Меткалфом и Дэвидом Боггсом в середине 70-х. В этой сети для соединения узлов использовался коаксиальный кабель. Сами топологии сетей Ethernet в основном не изменились ни в 80-е годы, ни до середины 90-х. Ethernet с шинной топологией представляет собой широковещательную локаль­ную сеть, все проходящие по которой кадры обрабатываются *всеми* адаптерами, подключенными к шине. Здесь можно вспомнить протокол множественного доступа Ethernet CSMA/CD и алгоритм двоичной экс­поненциальной выдержки.

К концу 1990-х большинство компаний и университетов заменили свои локальные сети Ethernet-конфигурациями, применяя звездообраз­ную топологию сети на основе концентратора. В такой системе хосты (и маршрутизаторы) напрямую подключаются к концентратору при помо­щи медного кабеля «витая пара». **Концентратор** (хаб) — это устройство физического уровня, которое оперирует отдельными битами, а не кадра­ми. Когда с интерфейса поступает бит, представляющий собой нуль или единицу, концентратор просто воспроизводит его, а потом передает с бо­лее высокой энергией на все другие интерфейсы. Соответственно, сеть Ethernet со звездообразной топологией, использующая концентратор, также является широковещательной локальной сетью. Когда концен­тратор получает бит с любого из своих интерфейсов, он посылает копии этого бита на все другие интерфейсы. В частности, если концентратор одновременно получает кадры с двух разных интерфейсов, происходит коллизия, и узлы, породившие эти кадры, должны повторить передачу.

В начале 2000-х сеть Ethernet претерпела еще одно коренное эво­люционное изменение. В конфигурациях сетей Ethernet по-прежнему использовалась звездообразная топология, но концентратор в ее центре был заменен **коммутатором** (свич). Ниже в этой главе мы подробнее по­говорим о коммутируемых сетях Ethernet. Пока достаточно оговориться, что коммутатор не только значительно снижает количество коллизий, но и отлично справляется с промежуточным хранением пакетов. Одна­ко в отличие от маршрутизаторов, работающих на всех уровнях вплоть до третьего, коммутатор работает только до второго уровня.

**Рис. 5.20. Структура Ethernet-кадра**

**Структура Ethernet-кадра**

Разобравшись в структуре кадра, которая схематически показана на рис. 5.20, мы довольно много узнаем о технологии Ethernet. Чтобы сде­лать наше обсуждение более предметным, рассмотрим передачу IP-дей­таграммы от одного хоста другому в ситуации, когда оба хоста находятся в одной и той же локальной Ethernet-сети (например, как на рис. 5.17). (Отметим, однако, что, хотя в нашем примере полезная нагрузка пред­ставляет собой IP-дейтаграмму, кадр Ethernet может переносить и другие пакеты сетевого уровня.) Пусть физический адрес передающего адаптера А равен АА-АА-АА-АА-АА-АА, а физический адрес прини­мающего адаптера, адаптера В — ВВ-ВВ-ВВ-ВВ-ВВ-ВВ. Передающий адаптер инкапсулирует IP-дейтаграмму в Ethernet-кадр и отправляет кадр физическому уровню. Принимающий адаптер получает кадр от физического уровня, извлекает IP-дейтаграмму и передает ее сетевому уровню. Рассмотрим в этом контексте шесть полей Ethernet-кадра, по­казанных на рис. 5.20.

* *Поле данных (от 46 до 1500 байт).* Это поле содержит I Р-дейтаграмму. **Максимальный размер передаваемого блока** (Maximal Transfer Unit, **MTU)** в Ethernet-сети составляет 1500 байт. Это означает, что если размер IP-дейтаграммы превышает 1500 байт, то хост должен разбить ее на отдельные фрагменты (см. подраздел 4.4.1). Мини­мальный размер поля данных равен 46 байт. Другими словами, если размер IP-дейтаграммы меньше 46 байт, то данные, помещаемые в это поле, дополняются байтами-заполнителями. Сетевой уровень получает дейтаграмму от канального уровня с этими дополнитель­ными байтами и отсекает все лишнее сам, ориентируясь на поле дли­ны в заголовке IP-дейтаграммы.
* *Адрес получателя (6 байт).* Это поле содержит МАС-адрес при­нимающего адаптера, а именно ВВ-ВВ-ВВ-ВВ-ВВ-ВВ. Получив Ethernet-кадр с адресом получателя ВВ-ВВ-ВВ-ВВ-ВВ-ВВ или ши­роковещательным адресом MAC, адаптер В передает содержимое поля данных сетевому уровню. В противном случае он отбрасывает кадр.
* *Адрес отправителя (6 байт).* Это поле содержит МАС-адрес адап­тера, передающего кадр в локальную сеть, а именно АА-АА-АА-АА-АА-АА.
* *Поле типа (2 байта).* Поле типа позволяет локальной Ethernet-сети мультиплексировать протоколы сетевого уровня. Чтобы понять, что это означает, вспомним, что хосты могут помимо протокола IP ис­пользовать и другие протоколы. В самом деле, любой хост может поддерживать несколько протоколов сетевого уровня — разные про­токолы для разных приложений. По этой причине, получив Ethernet-кадр, адаптер В должен знать, какому протоколу сетевого уровня он должен передать (то есть демультиплексировать) содержимое поля данных. Каждому сетевому протоколу (например, IP, Novell IPX или AppleTalk) присвоен зафиксированный в стандарте номер типа. Об­ратите внимание, что поле типа аналогично полю протокола в дейтаграмме сетевого уровня и полю номера порта сегмента транспортно­го уровня. Все эти поля служат для связи протокола одного уровня с протоколом уровнем выше.
* *CRC (4 байта).* Как было показано ранее, назначение поля CRC заключается в том, чтобы получающий адаптер (адаптер Б) мог определить, не исказился ли кадр при передаче, то есть обна­ружить ошибки в кадре.
* *Преамбула (8 байт).* Ethernet-кадр начинается с 8-байтового поля преамбулы. В каждый из первых 7 байт преамбулы записывается значение 10101010, ав последний байт — значение 10101011. Первые 7 байт должны «разбудить» принимающие адаптеры и помочь им синхронизировать свои таймеры с часами отправителя. Как уже от­мечалось, адаптер А должен передать кадр со скоростью 10 Мбит/с, 100 Мбит/с или 1 Гбит/с в зависимости от типа локальной Ethernet-сети. Однако поскольку в реальном мире ничто не совершенно, ско­рость передачи всегда будет несколько *отличаться* от номинала. Величина этого отклонения скорости другим адаптерам локальной сети *заранее* не известна. Таким образом, первые 62 бита преамбулы, представляющие собой чередующиеся нули и единицы, позволяют приемнику с достаточной точностью настроиться на скорость пере­датчика, а последние два разряда (две единицы подряд) сообщают адаптеру Б, что преамбула закончилась и следом идет уже первый информационный байт поля кадра. Адаптер Б понимает, что следую­щие 6 байт содержат адрес получателя. Конец кадра адаптер может распознать просто по отсутствию сигнала в линии.

**ИСТОРИЯ**

**Боб Меткалф и Ethernet**

Готовясь к получению степени доктора философии в Гарварде в на­чале 70-х годов, Боб Меткалф работал над проектом ARPAnet в Мас­сачусетсом технологическом институте (MIT). Во время обучения он познакомился с работами Абрамсона над сетью ALOHA и прото­колами произвольного доступа. Получив степень незадолго до на­чала работы в исследовательском центре Xerox PARC корпорации Xerox в Пало-Альто, он посетил Абрамсона и его коллег в Гавайском университете, где в течение трех месяцев изучал сеть ALOHAnet. В центре Xerox PARC Меткалф занялся компьютерами Alto, во мно­гом послужившими прототипами первых персональных компьюте­ров 80-х годов. Меткалф осознал необходимость разработки недорогого способа соединения подобных компьютеров в сеть. Таким образом, вооружившись знаниями о сетях ARPAnet и ALOHAnet, а также о протоколах произвольного доступа, Меткалф вместе со своим коллегой Дэвидом Боггсом спроектировал Ethernet-сеть.

Оригинальная Ethernet-сеть, разработанная Меткалфом и Боггсом, работала на скорости 2,94 Мбит/с и соединяла до 256 хостов на рас­стоянии до одной мили. Меткалфу и Боггсу удалось уговорить боль­шую часть исследователей Xerox PARC принять участие в тестирова­нии этой сети. Затем Меткалф сумел объединить корпорации Xerox, Digital и Intel в целях разработки стандарта Ethernet со скоростью 10 Мбит/с, ратифицированного институтом IEEE. Однако компания Xerox не проявила большого интереса к коммерциализации техно­логии Ethernet, поэтому в 1979 году Меткалф основал собственную компанию, 3Com, которая занялась разработкой и коммерциализа­цией сетевых технологий, включая технологию Ethernet. В частно­сти, в начале 80-х годов компания 3Com подготовила и выпустила на рынок Ethernet-карты для очень популярных в то время персональ­ных компьютеров IBM PC. В 1990 году Меткалф покинул компанию 3Com. В это время в ней работали 2000 сотрудников, а доходы со­ставляли 400 млн долларов.

Все технологии Ethernet предоставляют сетевому уровню службу без установления соединения. Таким образом, когда адаптер А хочет послать дейтаграмму адаптеру Б, он инкапсулирует ее в Ethernet-кадр и передает этот кадр в локальную сеть, даже не выполнив процеду­ры рукопожатия с адаптером Б. Подобная не требующая соединения служба на канальном уровне (уровень 2) аналогична дейтаграммной службе протокола IP (уровень 3) и службе протокола UDP (уро­вень 4).

Все разновидности технологий Ethernet предоставляют сетевому уровню ненадежную службу. В частности, когда адаптер Б получает кадр от адаптера А, он выполняет CRC-контроль кадра, но не передает в ответ подтверждения, что кадр успешно прошел проверку (также не передается и отрицательная квитанция, если контрольная сумма не со­впала со значением поля CRC). Когда кадр не проходит такой проверки, принимающий адаптер просто отбрасывает его. То есть у адаптера А нет ни малейшего представления о том, прошел переданный им кадр CRC-контроль или нет. Благодаря отсутствию на канальном уровне службы надежной доставки технология Ethernet остается простой и дешевой. Но это также означает, что в потоке дейтаграмм, переданных сетевому уровню, могут быть пропуски.

Если в потоке данных возникают пропуски, потому что протокол Ethernet теряет кадры, видит ли эти пропуски приложение на хосте Б? Как отмечалось ранее, это полностью зависит от используемого при­ложением протокола (UDP или TCP). Если это протокол UDP, тогда приложению самому придется заниматься проблемой потерянных ка­дров. Если же это протокол TCP, то проблему пропущенных кадров возьмет на себя протокол TCP, с помощью механизма подтверждений и интервалов ожидания заставляя хост А передавать пропущенные дан­ные повторно. Обратите внимание, когда протокол TCP передает дан­ные повторно, они снова проходят через тот же Ethernet-адаптер. Так что в этом смысле Ethernet может повторять передачу данных. Однако следует помнить, что Ethernet-сеть не догадывается о том, что переда­ет данные повторно, оставаясь в неведении, получает ли она новенькую дейтаграмму с новыми данными или дейтаграмму с данными, которые уже были переданы как минимум один раз.

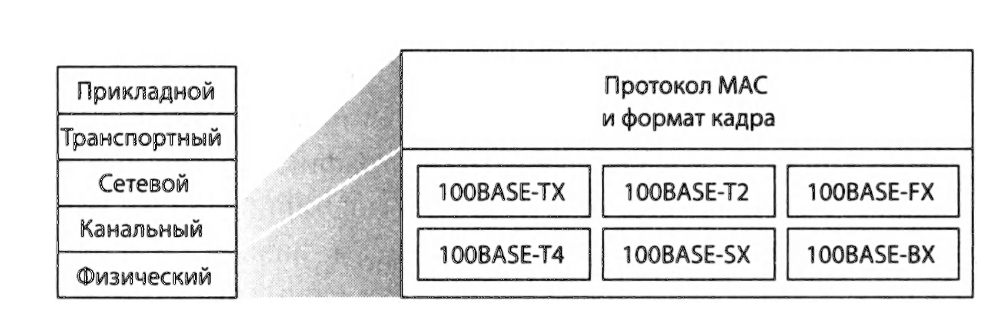
**Технологии Ethernet**

Выше мы говорили об Ethernet как о едином стандарте протоколов. На самом же деле существует *множество* разновидностей Ethernet, которые обозначаются довольно сложными аббревиатурами, напри­мер, 10BASE-T, 10BASE-2, 100BASE-T, 1000BASE-LX и 10GBASE-T. Эти и многие другие технологии Ethernet были стандартизированы за прошедшие годы рабочей группой IEEE 802.3 CSMA/CD (Ethernet). Сокращения на первый взгляд могут показаться устрашающими, но на самом деле в них прослеживается определенный порядок. Первая часть аббревиатуры указывает на скорость, применяемую в стандарте: 10,100,1000 или 10G означает, соответственно, 10 Мбит/с, 100 Мбит/с, 1 Гбит/с и 10 Гбит/с. BASE означает смодулированный Ethernet — то есть физический носитель передает только трафик Ethernet; практиче­ски все стандарты 802.3 касаются именно немодулированного Ethernet. Последняя часть аббревиатуры описывает сам физический носитель. Для Ethernet существуют спецификации как канального, так *и* физи­ческого уровня, а передача данных осуществляется по разнообразным физическим носителям, включая коаксиальный кабель, медный кабель и оптоволокно. Вообще буква Т означает медный кабель типа «витая пара».

Передача на более длинные дистанции осуществляет­ся при помощи повторителя. Это устройство физического уровня, кото­рое принимает на входе сигнал и воспроизводит его на выходе. Коакси­альный кабель, показанный на рис. 5.20, красноречиво свидетельствует, что Ethernet является средством широковещательной передачи данных. Все кадры, передаваемые интерфейсом, принимаются на всех осталь­ных интерфейсах, а протокол Ethernet CDMA/CD красиво решает про­блему множественного доступа. Узлы просто подключаются к кабелю и — вуаля\ — у нас есть локальная сеть.

За прошедшие годы технология Ethernet претерпела ряд эволюци­онных изменений, и нынешний Ethernet весьма отличается от ориги­нальной шинной топологии, работавшей с применением коаксиально­го кабеля. Сегодня в большинстве конфигураций узлы подключаются к коммутатору при помощи двухточечных сегментов, изготовленных из медного кабеля «витая пара» или оптоволоконного кабеля, как показано на рис. 5.15-5.17.

В середине 90-х появился стандарт Ethernet со скоростью передачи данных 100 Мбит/с — то есть в 10 раз быстрее, чем 10 Мбит/с. В нем сохранились оригинальный протокол Ethernet MAC и формат кадра, но были описаны более высокоскоростные физические уровни для медно­го кабеля (100BASE-T) и для оптоволоконного кабеля (100BASE-FX, 100BASE-SX, 100BASE-BX). На рис. 5.21 представлены эти различные стандарты, а также обычный протокол Ethernet MAC и формат кадра.

****

**Рис. 5.21**. Стандарты Ethernet 100 Мбит/с: канальный уровень общий,

физические уровни отличаются

Максимальная длина кабеля для Ethernet 100 Мбит/с составляет 100 метров при передаче по витой паре и до нескольких километров, если используется оптоволокно. Таким образом удается соединять ком­мутаторы Ethernet в разных зданиях.

Гигабитный Ethernet является усовершенствованием исключитель­но успешных стандартов 10 Мбит/с и 100 Мбит/с. Гигабитный Ethernet обеспечивает максимальную физическую передачу данных со скоростью 1000 Мбит/с, а также полную совместимость со всей огромной базой уже установленного Ethernet-оборудования. Согласно документу IEEE 802.3z, стандарт Gigabit Ethernet реализует следующий функционал:

* Использует стандартный формат кадра Ethernet (рис. 5.20) и обес­печивает обратную совместимость с технологиями 10BASE-T и 100BASE-T. Таким образом, обеспечивается удобная интеграция Gigabit Ethernet в уже развернутую техническую базу Ethernet-оборудования.
* Допускает использование двухточечных каналов, а также разделяе­мых широковещательных каналов. При работе с двухточечными каналами используются коммутаторы, а при работе с широковеща­тельными — концентраторы, как было описано выше. В терминоло­гии Gigabit Ethernet концентратор называется *буферизованным рас­пределителем.*
* Использует протокол CSMA/CD при работе с широковещательны­ми каналами. Чтобы обеспечивать при этом приемлемую эффектив­ность работы, максимальное расстояние между узлами строго огра­ничивается.
* Обеспечивает полнодуплексную эксплуатацию двухточечных кана­лов на скорости 1000 Мбит/с в обоих направлениях.

Изначально технология Gigabit Ethernet работала по оптоволокон­ному кабелю, но теперь также допускает эксплуатацию кабеля кате­гории 5 UTP. Технология Ethernet 10 Гбит/с была стандартизирована в 2007 году, после чего локальные Ethernet-сети стали еще мощнее.

Завершая обсуждение технологии Ethernet, постараемся ответить на вопрос, который вы уже наверняка перед собой ставили. Во времена шинных топологий и звездообразных топологий с применением кон­центратора в Ethernet, разумеется, использовалась широковещательная связь (о которой мы говорили в разделе 5.3). Когда применялся такой механизм связи, между кадрами могли возникать коллизии, если узлы передавали данные одновременно. Для того чтобы справляться с кол­лизиями, стандарт Ethernet предусматривал использование протокола CSMA/CD, который действительно очень эффективен при прокладке кабельных локальных сетей на небольших территориях. Но в настоящее время в сетях Ethernet преобладают звездообразные топологии с при­менением коммутаторов, где реализуется коммутация пакетов с проме­жуточным хранением и передачей. В таком случае, сохраняется ли се­годня необходимость в использовании МАС-протокола? Как мы вскоре увидим, коммутатор координирует свои операции по передаче данных и никогда не передает более одного кадра на конкретный интерфейс в определенный момент времени. Более того, современные коммутато­ры являются полнодуплексными — то есть обеспечивают передачу ка­дров в обоих направлениях без интерференции. Иными словами, в ком­мутируемых локальных Ethernet-сетях коллизии отсутствуют, а значит, не нужен и протокол MAC!

Как мы могли убедиться, современный Ethernet *очень сильно* отли­чается от исходной технологии, изобретенной Меткалфом и Боггсом около 30 лет назад. Скорости передачи информации возросли на три порядка, кадры Ethernet передаются по самым разным носителям, пре­обладает коммутируемая организация сетей Ethernet, и даже протокол MAC зачастую оказывается не нужен! Можно ли все это *по-прежнему* называть словом Ethernet? Разумеется, «да, по определению». Инте­ресно отметить, что, несмотря на все перечисленное, в этой технологии сохранилась одна постоянная, которая ничуть не изменилась за 30 лет. Это формат кадра Ethernet. Пожалуй, именно формат кадра является истинным и нестареющим ядром стандарта Ethernet.