

Глава 2

Как компьютеры взаимодействуют в сети

В этой главе вы найдете ответы на следующие вопросы:

- **Что такое эталонная модель OSI?**
- **Каковы функции каждого уровня в модели OSI?**
- **Как определять уровни модели OSI, на которых выполняются конкретные сетевые операции?**
- **Какие возможны расширения модели OSI со стороны IEEE?**

В предыдущей главе вы узнали, что такое компьютерные сети, познакомились с основными типами сетей и поняли, как компьютеры (точнее, работающие на них программы) общаются друг с другом в сети. Теперь рассмотрим принципы взаимодействия компьютеров в сети более подробно.

Чтобы общаться, люди чаще всего используют устную речь. Однако такое непосредственное общение возможно, только если собеседники находятся рядом друг с другом и только в воздушной среде. Но представьте себе, что надо передать данные вашему товарищу, который живет в другом городе, а тем более — в другой стране. Здесь уже не обойтись без целого ряда определенных действий: нужно написать текст на листе бумаги, подписать его, вложить в конверт, указать на нем адреса отправителя и получателя, наклеить марку и отдать почтальону (или бросить в почтовый ящик). Дальнейшая судьба этого письма зависит уже не от вас, а от почтовой службы. Каким-либо способом — на поезде, корабле, самолете или как-то иначе, но письмо доходит до страны и города, где живет ваш друг, затем доставляется в его почтовое отделение и, наконец, попадает к нему в почтовый ящик. Только тогда ваш адресат получает возможность открыть конверт и прочитать ваше сообщение. Заметим, что если какая-либо из стадий доставки не сработает, например, из-за отсутствия почтальона или различий в

правилах записи адресов в разных странах, то информация до вашего друга так и не дойдет.

Точно так же поступают и компьютеры при общении в сети. Способов непосредственного общения у них нет — разговаривать друг с другом компьютеры пока еще не научились. Поэтому, чтобы общаться, им приходится прибегать к целому ряду последовательно выполняемых процедур, называемых *сетевыми протоколами*. Чтобы протоколы работали надежно и согласованно, каждая операция в них строго регламентируется. А чтобы программы и оборудование разных производителей могли взаимодействовать друг с другом, протоколы должны соответствовать определенным *промышленным стандартам*.

Протокол — набор правил и процедур, регулирующих порядок взаимодействия компьютеров в сети.

За долгие годы существования компьютерных сетей было создано великое множество различных протоколов — как открытых (опубликованных для бесплатного применения), так и закрытых (разработанных коммерческими компаниями и требующих лицензирования для их использования). Однако все эти протоколы принято соотносить с так называемой *эталонной моделью взаимодействия открытых систем (Open Systems Interconnection Reference Model)*, или просто *моделью OSI*. Ее описание было опубликовано в 1984 г. Международной организацией по стандартизации (International Standards Organization, ISO), поэтому для нее часто используется другое название — *модель ISO/OSI*. Эта модель представляет собой набор спецификаций, описывающих сети с неоднородными устройствами, требования к ним, а также способы их взаимодействия.

Структура модели OSI

Модель OSI имеет вертикальную структуру, в которой все сетевые функции распределены между семью *уровнями* (рис. 2.1). Каждому такому уровню соответствуют строго определенные операции, оборудование и протоколы.

Реальное взаимодействие уровней, т. е. передача информации внутри одного компьютера, возможно *только по вертикали и только с соседними уровнями* (выше- и нижележащими).

Логическое взаимодействие (в соответствии с правилами того или иного протокола) осуществляется *по горизонтали* — с *аналогичным уровнем другого компьютера* на противоположном конце линии связи. Каждый более высокий уровень пользуется услугами нижележащего уровня, зная, в каком виде и каким способом (т. е. через какой *интерфейс*) нужно передать ему данные.

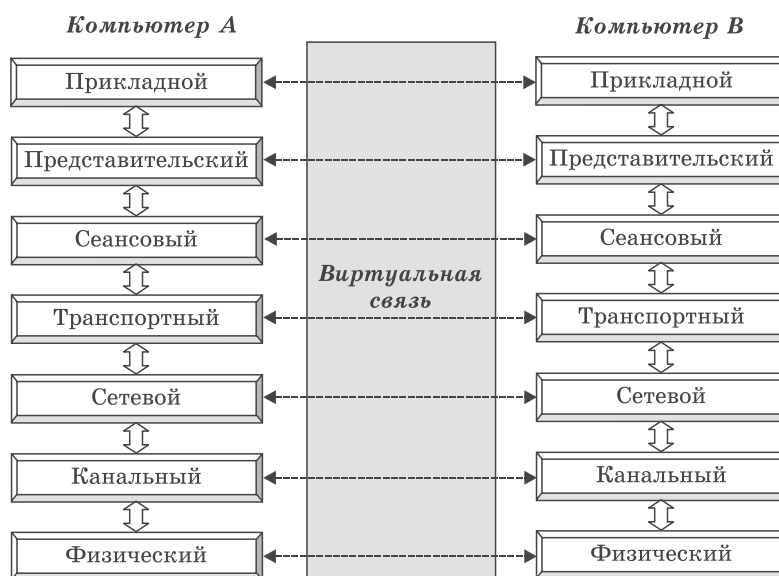


Рис. 2.1. Взаимосвязи между уровнями модели OSI

Задача более низкого уровня — принять данные, добавить свою информацию (например, форматирующую или адресную, которая необходима для правильного взаимодействия с аналогичным уровнем на другом компьютере) и передать данные дальше. Только дойдя до самого нижнего, физического уровня сетевой модели, информация попадает в среду передачи и достигает компьютера-получателя. В нем она проходит сквозь все слои в обратном порядке, пока не достигнет того же уровня, с которого была послана компьютером-отправителем.

Как видим, все это очень похоже на наш пример с работой почты — программы общаются по сети примерно так же, как вы со своим другом по почте. Ваш лист бумаги с текстом передается с верхнего уровня вниз, проходя множество необходимых стадий. При этом он «обрастает» служебной информацией (конверт определенного вида, адрес на конверте, почтовый индекс) и подвергается определенной обработке (почтальон в отделении забирает письмо, на конверт наклеивают марки, ставят штемпели, а после сортировки письмо попадает в контейнер для перевозки почты в другой город). Так ваша информация доходит до самого нижнего уровня — почтового транспорта, которым она перевозится в пункт назначения. Там происходит обратный процесс: открывается контейнер, письмо извлекается, считывается адрес, после чего почтальон доставляет письмо вашему другу. А затем ваш друг получает информацию в первоначальном виде — когда извлекает лист из конверта, проверяет подпись и читает текст.

Таким образом, вы с вашим другом *логически имеете прямую связь*, и детали доставки вас мало заботят. Почтальоны также имеют прямую связь: почтальон в чужом городе получит в точности то, что вы передали своему почтальону — конверт с письмом и адресной информацией. Почтальонов при этом не волнуют проблемы, например, железнодорожников, которые в действительности и осуществляли перевозку почтовой корреспонденции.

Теперь познакомимся поближе с уровнями модели OSI и определим сетевые услуги, которые они предоставляют смежным уровням.

Уровни модели OSI



Можно предположить, что контрольная сумма (CRC) как способ контроля правильности передачи данных появилась одновременно с первыми ЭВМ. Но оказывается, что идея «контрольной суммы» была впервые изобретена... церковниками, озабоченными большим количеством расхождений в текстах переписываемых вручную Библий (еще до изобретения книгопечатания): ведь при каждом таком копировании писцы не только повторяли все ошибки своего оригинала, но и добавляли новые. Из этой проблемы был найден следующий выход. На специальном совещании высших духовных чинов был выбран и утвержден некий канонический вариант Библии. В нем были подсчитаны количества слов и букв в каждой главе. Переписчик же, закончив свою работу, должен был подсчитать эти количества в сделанной копии и сравнить с полагающимися для оригинала.

- **Уровень 0** — не определен в общей схеме (на рис. 2.1), но весьма важен для понимания. Здесь представлены посредники, по которым собственно и происходит передача сигналов: кабели различных типов, радио-, ИК-сигналы и т. д. На этом уровне ничего не описывается, уровень 0 предоставляет физическому уровню 1 только *среду передачи*.
- **Уровень 1 — Физический (Physical)**. Здесь осуществляется передача неструктурированного потока битов, полученных от вышележащего канального уровня 2, по физической среде — например, в виде электрических или световых сигналов. Физический уровень отвечает за *поддержание связи (link)* и детально описывает электрические, оптические, механические и функциональные интерфейсы со средой передачи: напряжения, частоты, длины волн, типы коннекторов, число и функциональность контактов, схемы кодирования сигналов и т. д.
- **Уровень 2 — Канальный (Data Link)**. Обеспечивает *безошибочную передачу данных*, полученных от вышележащего сетевого уровня 3, через физический уровень 1, который сам по себе отсутствия ошибок не гарантирует и может исказить данные. Информация на этом уровне помещается в *кадры (frames)*, где в начале (*заголовке кадра*) содержатся адреса получателя и отправителя, а также управляющая информация, а в конце — *контрольная сумма*, позволяющая выявить возникающие при передаче ошибки (рис. 2.2).

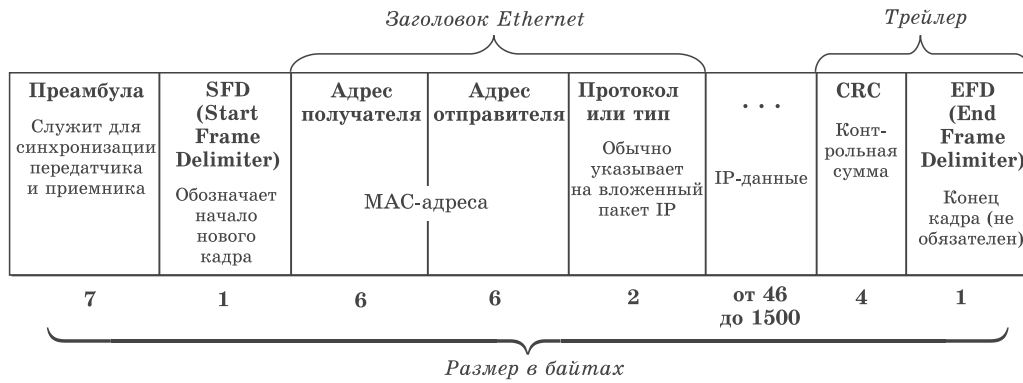


Рис 2.2. Структура кадра

При *получении данных* на канальном уровне определяются начало и конец кадра в потоке битов, сам кадр извлекается из потока и проверяется на наличие ошибок. Поврежденные при передаче кадры, а также кадры, для которых не получено *подтверждение о приеме*, пересылаются заново (*ретранслируются*). Наконец, на канальном уровне обеспечивается управление доступом к среде передачи.

Канальный уровень довольно сложен, поэтому в соответствии со стандартами IEEE (Institute of Electrical and Electronics Engineers), выпущенными в феврале 1980 г. в рамках «Проекта 802» (*Project 802*), его часто разбивают на два подуровня (рис. 2.3): *управления доступом к среде (Media Access Control, MAC)* и *управления логической связью (Logical Link Control, LLC)*.

Уровень MAC обеспечивает совместный доступ сетевых адаптеров к физическому уровню, определение границ кадров, распознавание *адресов назначения кадров* (эти адреса часто называют *физическими*, или *MAC-адресами*).

Уровень LLC, действующий над уровнем MAC, отвечает за установление канала связи и за безошибочную посылку и прием сообщений с данными.

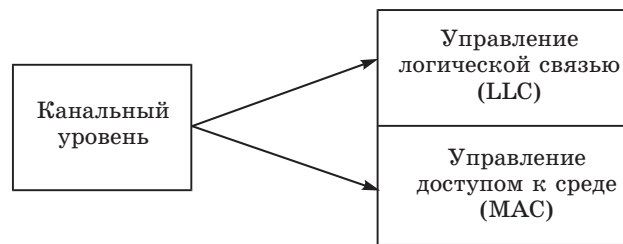


Рис 2.3. Разделение канального уровня на подуровни MAC и LLC

- **Уровень 3 — Сетевой (Network).** Отвечает за *обеспечение связи* между любыми, даже находящимися в разных концах земного шара, точками в сети. Этот уровень осуществляет *проводку сообщений по сети*, которая может состоять из множества отдельных сетей, соединенных множеством линий связи. Такая доставка требует *маршрутизации*, т. е. *определения пути доставки сообщения*, а также решения задач управления потоками данных и обработки ошибок передачи.
- **Уровень 4 — Транспортный (Transport).** Гарантирует *доставку информации* от одного компьютера другому. На этом уровне компьютера-отправителя большие блоки данных разбиваются на более мелкие *пакеты*, которые доставляются компьютеру-получателю *в нужной последовательности, без потерь и дублирования*. На транспортном уровне компьютера-получателя пакеты вновь собираются в исходные блоки данных. Таким образом, транспортный уровень *завершает процесс передачи данных*, скрывая от более высоких уровней все детали и проблемы, связанные с доставкой информации *любого объема* между любыми точками во всей сети.
- **Уровень 5 — Сеансовый (Session).** Позволяет двум *сетевым приложениям* на разных компьютерах *устанавливать, поддерживать и завершать соединение*,

называемое *сетевым сеансом*. Этот уровень также отвечает за восстановление аварийно прерванных сеансов связи. Кроме того, на пятом уровне выполняется преобразование удобных для людей имен компьютеров в сетевые адреса (*распознавание имен*), а также реализуются функции *защиты сеанса*.

- **Уровень 6 — Представительский, или Уровень представления данных (Presentation).** Определяет *форматы* передаваемой между компьютерами информации. Здесь решаются такие задачи, как *перекодировка* (перевод информации в вид, понятный для всех участвующих в обмене компьютеров), сжатие и распаковка данных, шифрование и дешифровка, поддержка сетевых файловых систем и т. д.
- **Уровень 7 — Прикладной (Application), или Уровень Приложений.** Обеспечивает интерфейс взаимодействия программ, работающих на компьютерах в сети. Именно с помощью этих программ пользователь получает доступ к таким сетевым услугам, как обмен файлами, передача электронной почты, удаленный терминальный доступ и т. д.



К моменту появления модели OSI уже существовали и показали высокую эффективность другие *наборы (стеки) протоколов*, например *стек TCP/IP*. Поэтому построенный в полном соответствии с описанной выше моделью *набор протоколов OSI* так и не получил широкого распространения. Большинство современных сетевых архитектур и наборов протоколов соответствуют этой модели лишь до определенной степени. Несмотря на это, сама модель ISO/OSI до сих пор широко используется для описания взаимодействия в сетевых средах.



Вопросы и задания

1. Что понимается под термином «сетевой протокол»?
2. Какие сетевые функции осуществляются в модели OSI?
3. Какой уровень, согласно модели OSI, отвечает за выбор маршрута передачи данных?
4. На каком уровне модели OSI взаимодействуют программы, обеспечивающие передачу сообщений электронной почты?