

Ижевский государственный технический университет

А.В. Абилов

СЕТИ ПЕРЕДАЧИ ДАННЫХ

Часть 1. Обзор сетей передачи данных

Глава 1. Введение в компьютерные сети
Глава 2. Многоуровневая сетевая иерархия

Ижевск 2007

ЧАСТЬ 1

ОБЗОР СИСТЕМ И СЕТЕЙ ПЕРЕДАЧИ ДАННЫХ

Передача данных (ПД) в сетях связи становится все более актуальной сферой человеческой деятельности. Например, такие продукты, как MP3 проигрыватели и сотовые телефоны уже стали не просто обыденными и повседневными, но и доступными для всех. Информационные технологии в последнее время развиваются стремительно. Пользователям таких технологий необходимо лишь базовые знания о том, как их использовать. Однако тем, кто разрабатывает и внедряет такие технологии необходимо владеть принципами их работы и взаимодействия в рамках структуры связи, представленной на рис. 1.



Рис. 1. Структура связи

Передача данных

Основная концепция передачи информации заключается в способности сетей передавать данные из одного места в другое на расстоянии. Для освоения основных принципов передачи информации необходимо знать систему связи, способы представления различных типов данных и организацию связи (режимы связи).

Сети

Передача информации между удаленными объектами может осуществляться посредством сети, соединяющей компьютеры, мультимедиа- и сетевые

устройства. Любая сеть характеризуется тремя основными понятиями: распределенная обработка, критерии и структура.

Локальные и глобальные сети

Сети связи можно поделить на две основные категории: локальные (Local Area Networks – LAN) и глобальные (Wide Area Networks – WAN). Они различаются по своим характеристикам и функциям. Обычно LAN включает в себя компьютеры и периферийные устройства в пределах одного здания или кампуса (нескольких зданий). Обычно LAN обслуживает одну организацию или какое либо подразделение. WAN как правило состоит из локальных сетей удаленных друг от друга на большие расстояния и связанных между собой.

Интернет

Интернет состоит из множества локальных и глобальных сетей связанных между собой устройствами взаимодействия. Функционирование Интернет обеспечивается не только физическим соединением LAN и WAN, но и стандартами протоколами взаимодействия.

Протоколы и стандарты

Для передачи информации в сетях связи большое значение имеют протоколы и стандарты. Протоколы описывают определенные правила функционирования, стандарты обеспечивают использование совместимых протоколов различными производителями оборудования и операторами услуг связи. Протоколы и стандарты охватывают как область передачи информации (алгоритмы взаимодействия), так и сети (принципы построения, структуры).

Сетевые модели

Сетевые модели служат для организации, унификации и управления аппаратными и программными средствами передачи информации в сетях связи. Хотя термин «Сетевая модель» относится к сетям связи, он также охватывает область передачи информации.

ГЛАВА 1

ВВЕДЕНИЕ

Передача информации и информационные сети оказывают все большее влияние на все сферы деятельности. Компьютерные сети позволяют мгновенно обмениваться информацией на любом расстоянии. Рост количества персональных компьютеров меняют способы ведения бизнеса и способствует ускоренному развитию промышленности, науки и образования. Такое же влияние оказывает развитие технологий передачи данных и информационных сетей. Основной целью таких сетей является обмен информацией различного вида: текст, аудио, видео и т.д. В первой главе рассматриваются основные принципы передачи информации, построения сетей, основы Интернет, а также общие сведения о протоколах и стандартах.

1.1. Передача данных

Для передачи данных от источника к получателю необходима организация связи между ними. Связь может осуществляться локально или удаленно. Локальная связь происходит непосредственно между людьми, например, посредством голоса. Удаленная связь организуется с помощью средств передачи информации. Широко используемый термин «Телекоммуникации» (Telecommunication), к которому традиционно относятся телефония, телеграфия и телерадиовещание, означает связь на расстоянии. Передача данных (Data communication) означает обмен данными, представленными в какой либо заранее оговоренной форме, между устройствами по определенной среде.

Эффективность передачи информации зависит от трех основных характеристик: доставка, достоверность и своевременность. *Доставка* – система связи должна обеспечить передачу информации тому пользователю, для которого она предназначена. *Достоверность* – способность системы связи передавать информацию без искажений либо с искажениями в пределах допустимого уровня. *Своевременность* – информация должна быть доставлена вовремя. Например, для передачи аудио- или видеоинформации необходимо обеспечить доставку в правильной последовательности (звуковых фрагментов или видеокадров) и с минимальными задержками.

Система передачи данных

В упрощенном представлении систему связи можно представить в виде совокупности пяти компонентов, которые представлены на рис. 1.1.

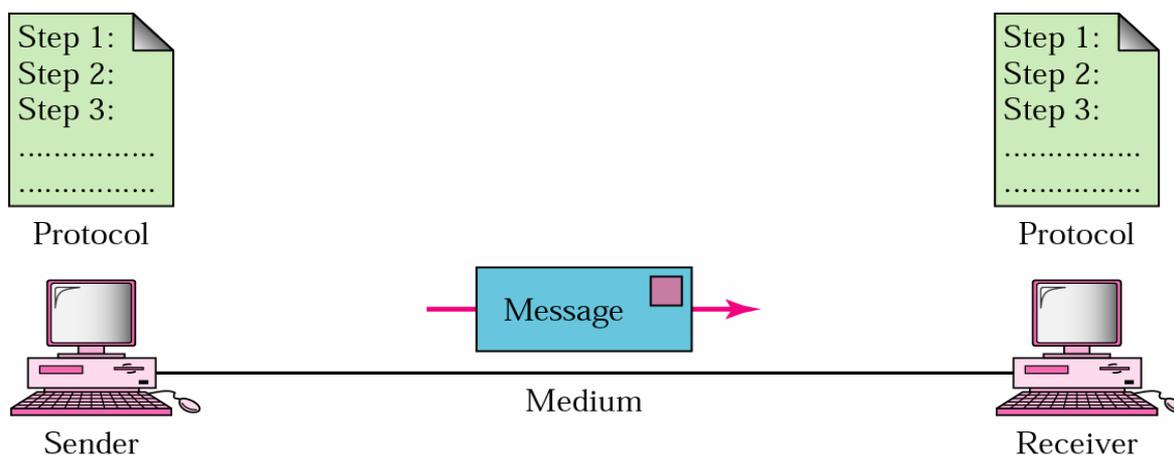


Рис. 1.1. Пять компонентов системы связи

Сообщение (Message) – информация (данные) для передачи. Может быть представлены как текст, числа, изображения, аудио, видео или их комбинации. *Отправитель* (Sender) – устройство, посылающее данные, например, компьютер, рабочая станция, телефонный аппарат, видеокамера и т.д. *Приемник* (Receiver) – устройство, принимающее данные, например, компьютер, рабочая станция, телефонный аппарат, телевизор, радиоприемник и т.д. *Среда передачи* (Transmission medium) – физическая среда, по которой информация передается от отправителя приемнику. *Протокол* (Protocol) – совокупность правил, определяющих передачу информации. Эти правила едины для отправителя и приемника. Без протокола устройства могут быть подключены друг к другу, но без возможности обмена информацией, например, аналогично как люди, пытающиеся общаться на разных языках, слышат друг друга, но не понимают.

Представление данных

Информация в зависимости от вида может быть представлена в различных формах: текст, числа, изображения, аудио и видео.

Текст

Текстовая информация представляется в виде последовательности двоичных битов (0 или 1). Количество бит в кодовом слове зависит от количества символов в языке. Например, английский язык состоит из 26 символов (A, B, C, ..., Z) для представления заглавных букв, 26 символов для представления маленьких букв, 10 символов (0, 1, 2, ..., 9) для представления чисел и символы (., ?, :, ...!) для представления знаков пунктуации. Также используются такие символы как пробел, табуляция для выравнивания и удобства чтения текста. Набор кодовых слов представляет совокупность текстовых символов.

ASCII

Код ASCII (American Standard Code for Information Interchange) разработан Американским национальным институтом стандартизации (ANSI). Этот код использует 7 бит для каждого символа и определяет 128 различных символов.

Расширенный ASCII

Добавляется один дополнительный бит 0 в левой части кода с целью доведения до размера одного байта (8 бит). Например, начальный код – 00000000, последний код 01111111.

Unicode

Для поддержки символов многих других языков необходима большая длина кода. Для этой цели разработан Unicode, использующий 16 бит, который может представлять до 16536 символов. Различные группы комбинаций используются для разных языков, определенные комбинации также используются для представления графических и специальных символов.

ISO

Международная организация по стандартизации ISO разработала код длиной 32 бит, который может представлять до 4 294 967 296 символов.

Числа

Коды, подобные ASCII, не используются для представления чисел. Арифметические числа непосредственно конвертируются двоичных комбинаций для удобства математических операций над ними.

Изображения

Изображения также представляются в виде последовательности двоичных комбинаций. Однако механизм представления изображений отличается от представления чисел. Изображение делится на элементы (пиксели) в виде матрицы точек. Размер пикселей зависит от разрешения, с увеличением количества пикселей увеличивается разрешение изображения, однако это требует большего объема памяти. Каждому пикселю ставится в соответствие двоичная комбинация, размер которой зависит от типа изображения. Для представления пикселя черно-белого изображения достаточно одного бита. Для полутонных изображений необходимо увеличение бит в кодовой комбинации.

Для того, чтобы представить цветное изображение, необходимо каждый пиксель разложить на три цвета: красный, зеленый и синий. Оцененная интенсивность каждого из цветовых составляющих представляется в виде кодовой комбинации (обычно 8 бит). Таким образом каждый пиксель цветного изображения кодируется тремя цветовыми кодами.

Аудио

Звук по своей природе является непрерывным и отличается от формы представления текста, чисел или изображений. Например, микрофон преобразует акустический аудио сигнал в непрерывный электрический.

Видео

Видеоинформация представляется в виде последовательности изображений, которые могут изменяться. Отображает действие во времени и сопровождается аудио информацией.

Режимы передачи данных

Передача данных между двумя устройствами может осуществляться в одном из трех режимов: симплексный, полудуплексный или дуплексный.

Симплексный режим

В симплексном режиме связь однонаправленная. Только одно из двух устройств может использовать канал для передачи, другое устройство только принимает информацию.

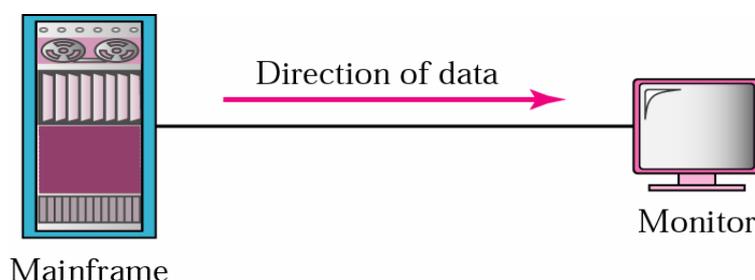


Рис. 1.2. Симплекс

Примерами симплексного режима являются клавиатура (передает) или монитор (принимает).

Полудуплексный режим

В полудуплексном режиме каждое из устройств может как передавать, так и принимать данные, но с разделением во времени. Когда одно устройство передает, другое может только принимать и наоборот.

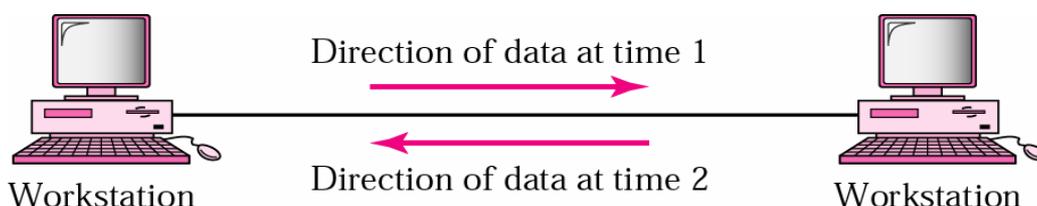


Рис. 1.3. Полудуплекс

В один момент времени общая емкость канала связи занимает только одним передающим устройством. Примерами полудуплексного режима связи могут быть устройства связи типа walkie-talkie, радики, радилюбительская КВ связь и др.

Дуплексный режим (полный дуплекс)

В полнодуплексном режиме два взаимодействующих устройства передают и принимают одновременно.

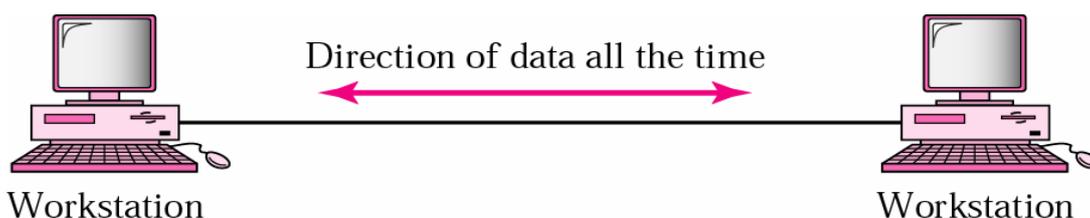


Рис. 1.4. Полный дуплекс

Сигналы, проходящие в обоих направлениях одновременно, разделяют общую емкость канала связи. Такое разделение может осуществляться двумя основными способами: канал должен содержать две отдельные физические линии, одна для передачи и другая для приема либо общая емкость канала используется для сигналов в обоих направлениях, например, посредством частотного разделения. Самым распространенным примером дуплексной связи является телефонная сеть, где абонент может одновременно говорить и слушать.

1.2. Сети

Сеть представляет собой совокупность устройств (часто называемых *узлами*) соединенных между собой с помощью линий связи. Узлами могут быть компьютеры, принтеры, или любые другие устройства, способные передавать и/или принимать данные.

Распределенная обработка

К большинству сетей применимо понятие *распределенная обработка*, где какая либо задача делится среди множества компьютеров. Вместо одного мощного компьютера, выполнение всех задач, возлагается на отдельные компьютеры меньшей производительности.

Требования к сети

Любая сеть должна удовлетворять определенным требованиям в соответствии с рядом критериев. Наиболее важными из них, как правило, являются: производительность, надежность и информационная безопасность.

Производительность

Производительность может оцениваться многими способами, например, время доставки сообщений, время отклика на запрос (реакции). Производительность сети также зависит от ряда факторов, включая количество пользователей, среды передачи, пропускной способности устройств и каналов связи и эффективности программного обеспечения.

Надежность

Надежность сети можно оценить интенсивностью отказов, временем, которое затрачивается на восстановление сети или его элементов после отказов и отказоустойчивостью в экстремальных условиях, например, удары молнии.

Информационная безопасность

Информационная безопасность подразумевает защиту данных от неавторизованного доступа.

Физические структуры

В простейшем случае любой канал можно представить на рисунке в виде линии, соединяющей два узла. Существует два основных типа соединений: соединение “точка-точка” (point-to-point) и многоточечное соединение (multipoint).

Типы соединений

Соединение “точка-точка” обеспечивает выделенный канал между двумя узлами связи. Общая пропускная способность канала используется для передачи данных только между этими двумя узлами и никакими другими более. Возможно применение как проводных линий, непосредственно соединяющих узлы, так и, например, радиорелейные или спутниковые каналы (рис. 1.5). Примером может служить система инфракрасного удаленного управления “пульт-телевизор”.

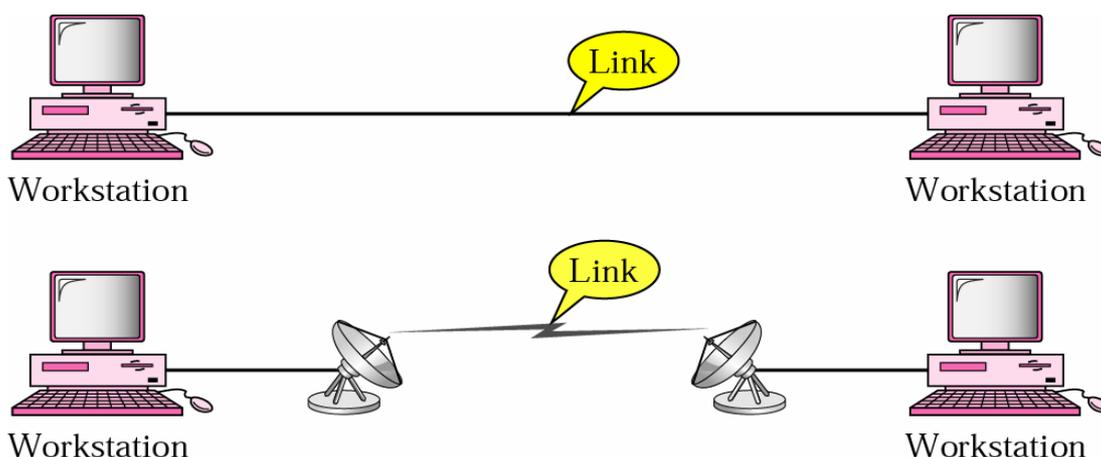


Рис. 1.5. Соединение “точка-точка”

Многоточечное соединение предоставляет каналы, используемые более чем двумя узлами (рис. 1.6).

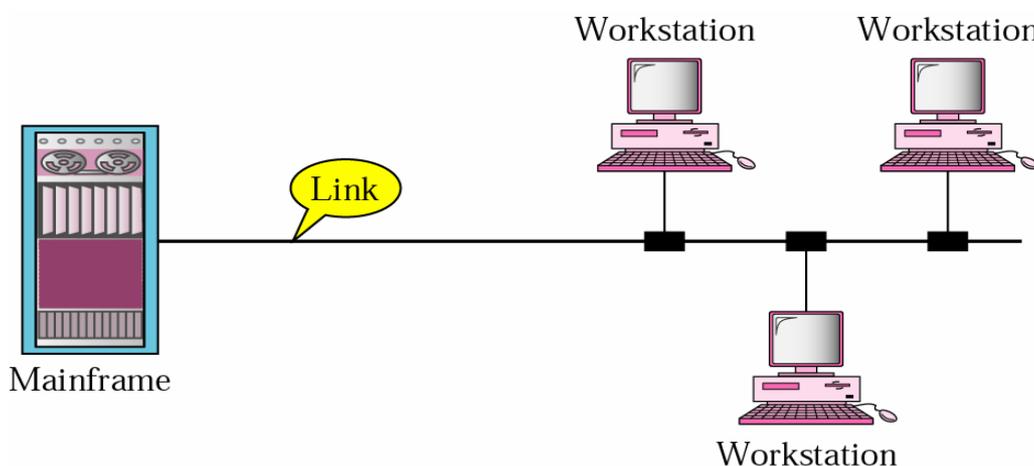


Рис. 1.6. Многоточечное соединение

При таком соединении емкость одного канала совместно используется более чем двумя устройствами путем пространственного и/или временного разделения.

Физическая топология сети

Термин “*физическая топология*” подразумевает представление сетевой структуры, отражающей физические взаимосвязи узлов. Два или более взаимосвязанных узла образуют одну или более линии, две и более линии уже форми-

руют сетевую топологию. Сетевая топология – это геометрическое представление взаимосвязи всех линий и взаимодействующих друг с другом узлов. Существует 4 основные сетевые топологии (рис. 1.7): *полносвязная (mesh)*, *звездообразная (star)*, *шинная (bus)*, *кольцевая (ring)*.

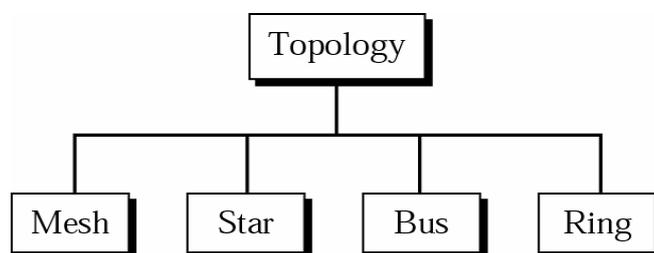


Рис. 1.7. Сетевые топологии

В *полносвязной топологии (mesh)* каждый узел имеет выделенную линию «точка-точка» ко всем другим узлам сети. Термин «*выделенная линия*» означает, что такая линия пропускает трафик только между двумя конкретными устройствами соединенными друг с другом. В полносвязной сети имеется $n(n-1)/2$ физических линий для n узлов, при этом каждый узел должен иметь $n-1$ портов ввода-вывода (I/O).

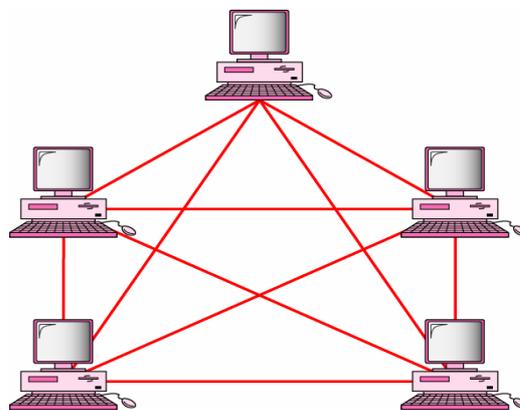


Рис. 1.8. Полносвязная топология

Полносвязная топология имеет следующие достоинства. Применение выделенных линий гарантирует отсутствие перегрузок вследствие совместного использования каналов многими узлами при ряде других топологий. Полносвязная топология обладает повышенной живучестью сети, т.е. неисправности на одних линиях не влияют на работу других вследствие их независимости. Такая независимость обеспечивает информационную безопасность, так как физические границы предотвращают для сторонних пользователей доступ к передаваемой информации. Это также облегчает процедуры определения мест повреждений на участках связи. Средства сетевого управления позволяют обнаруживать точные участки повреждений и их причины.

Основными недостатками полносвязной топологии являются большое количество кабелей и портов ввода-вывода каждого узла. Это приводит к сложности инсталляции сети и ее физической реконфигурации и, как следствие, к относительной дороговизне. По этим причинам такая топология используется ограниченно, преимущественно на магистральном уровне.

При *звездообразной топологии (star)* каждый узел имеет выделенную линию «точка-точка» только к центральному контроллеру, обычно называемому «hub». Узлы не соединяются друг с другом непосредственно и hub действует как посредник, транслирующий передаваемые данные от одного узла к другому.

В звездообразной топологии каждый узел имеет лишь одну линию и один порт ввода-вывода, что значительно сокращает затраты на оборудование и построение сети, а также облегчает ее инсталляцию и реконфигурацию. Например, удаление из сети одного узла повлечет за собой удаление лишь одной линии. Повреждения на одном из участков сети также не влияют на работу остальных участков, что свидетельствует о высокой живучести сети со звездообразной топологией. Устройство «Hub» может использоваться для мониторинга неисправностей сетевого оборудования. Однако количество кабелей в этом случае больше чем в шинной и кольцевой топологиях.

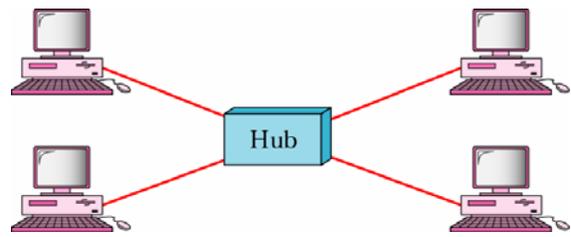


Рис. 1.9. Звездообразная топология

Шинная топология (bus) использует многоточечный тип соединения, где для связи всех узлов в сети применяется один кабель в качестве магистрали передачи данных. Узлы подключаются к общей шине с помощью распределительных линий (drop lines) и соединителей (tap). Места соединения в кабеле вызывают затухания сигнала. Это накладывает определенные ограничения на количество соединителей и длину кабеля.

К преимуществам шинной топологии относится легкость инсталляции оборудования и разводки кабеля, где узлы могут подключаться к общей шине линиями различной длины. Общая протяженность кабеля значительно меньше, чем при полностью связанной и звездообразной топологиях.

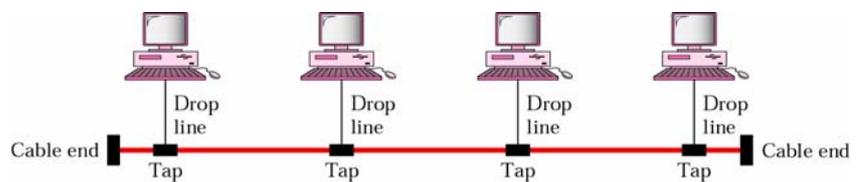


Рис. 1.10. Шинная топология

Недостатки включают сложность при переключениях, перестановок и добавления новых узлов.

Отражения сигнала от переходников может быть причиной ухудшения качества связи. Добавление новых узлов может потребовать коренной модификации сети или замены шинного кабеля. Обрывы и потеря физических контактов на шинной линии приводит к утрате работоспособности всей сети.

В сетях с *кольцевой топологией* каждый узел подключается по принципу «точка-точка» лишь с двумя соседними узлами. Сигнал передается (циркулирует) в одном направлении от узла к узлу до тех пор, пока не достигнет приемника. Каждый узел в кольце выполняет функции повторителя, т.е. если узел принимает сообщение, предназначенное для другого узла, он выполняет функции регенерации сигнала.

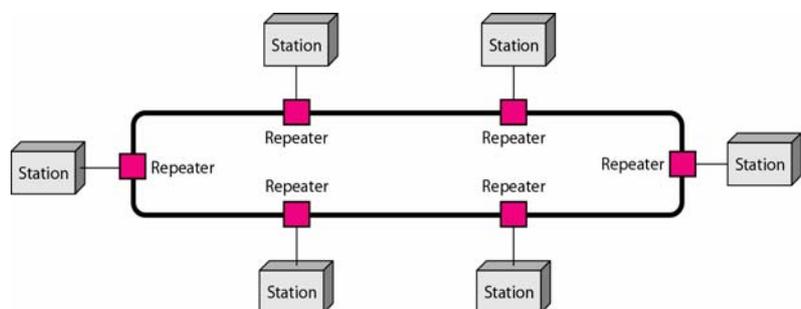


Рис. 1.11. Кольцевая топология

К преимуществам такой сети относится простота инсталляции и реконфигурации. Каждый узел соединяется только с его соседними узлами (физически или логически). Добавление новых узлов или удаление существующих требует изменений на двух участках сети. К ограничениям относятся длина кольцевой линии и количество узлов. Сигнал постоянно циркулирует по кольцу и если узел не принимает предназначенный ему сигнал в течение определенного времени, то это свидетельствует о неполадках в сети и может индицироваться программными средствами.

Передача сигнала в одном направлении может оказаться недостатком, т.к. в случае неисправности на каком либо участке вся сеть может потерять работоспособность. Эту проблему возможно решить применением двойного кольца либо способности сети менять направление передачи.

Типы сетей

Одним из признаков классификации сетей является их размер. Чаще всего рассматриваются три основных типа: локальные (Local-area network – LAN), муниципальные (Metropolitan-area network – MAN) и глобальные (Wide-area network – WAN) сети (рис. 1.12).

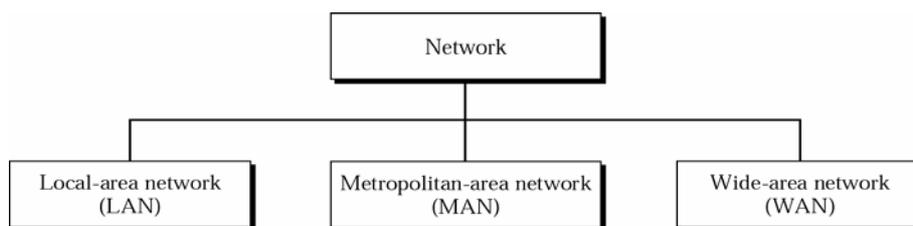


Рис. 1.12. Типы сетей

Локальные сети (LAN)

Являются частными сетями, которые, как правило, соединяют узлы в пределах офиса, здания или кампуса. В зависимости от нужд организации и используемых технологий локальная сеть может в простейшем случае, например, состоять из двух компьютеров и одного принтера с общим доступом, либо включать множество компьютеров и периферийных устройств. Чаще всего локальные сети охватывают территорию от нескольких метров до 1 км. Их используют для объединения компьютеров и рабочих станций с общим доступом к ресурсам. Ресурсы могут включать как аппаратные, так и программные средства либо данные.

Примером может быть использование одного компьютера с диском памяти большой емкости в качестве сервера. На этом сервере могут храниться необходимые программные средства и использоваться всей рабочей группой. Размер локальной сети может определяться лицензионными ограничениями по количеству пользователей на одну копию установленного на сервере программного обеспечения.

Чаще всего локальные сети отличаются от других типов сетей использованием одного типа среды передачи. Если LAN занимает одно здание или комнату и состоит из небольшого количества компьютеров, то используется одна топология (рис. 1.13а). При большом количестве компьютеров и протяженности LAN может быть разделена на сегменты с различной топологией, которые соединены магистральной линией (рис. 1.13б).

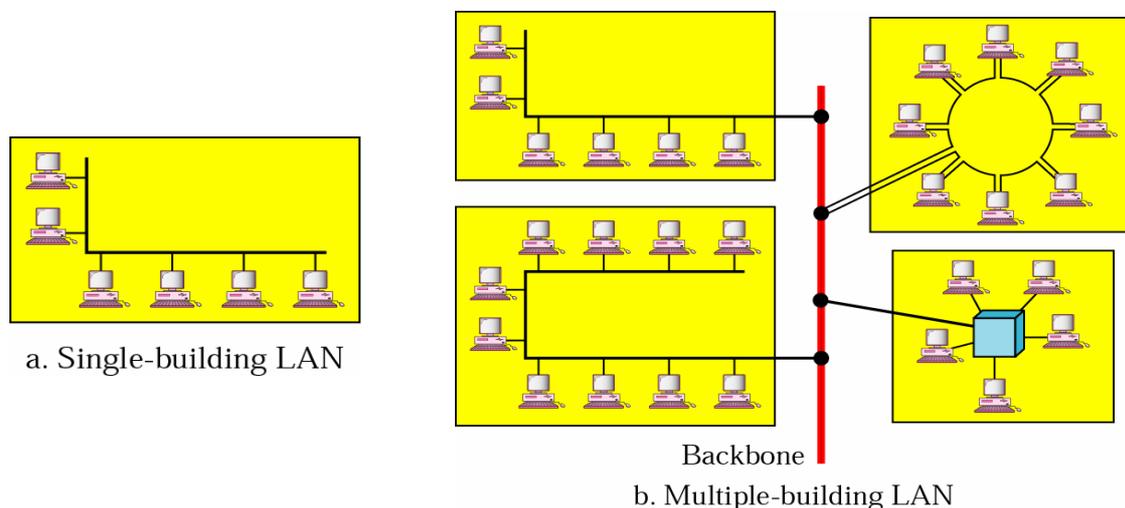


Рис. 1.13. Локальная сеть: а) одна топология; б) несколько топологий

Возможно использование шинной, кольцевой и звездообразной топологии. Скорость передачи определяется используемым стандартом локальной сети.

Муниципальные сети (MAN)

Объединяют узлы в пределах города. Самым распространенным примером MAN является сеть кабельного телевидения. Другим примером может быть сеть, связывающая несколько локальных сетей. Например, одна компания, имеющая несколько офисов (подразделений) в городе, может использовать MAN для связи локальных сетей (рис. 1.14).

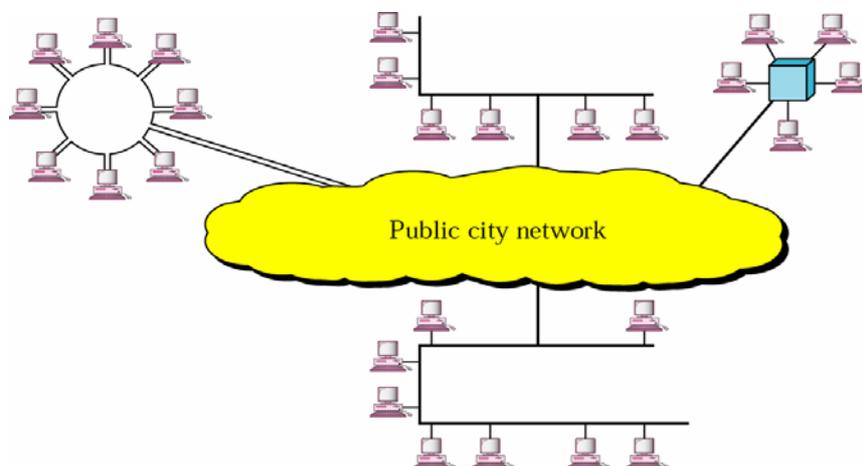


Рис. 1.14. Муниципальная сеть

Примерно до конца 90-х годов сети кабельного телевидения использовались исключительно для приема. Когда Интернет стал привлекать к себе массовую аудиторию, множество кабельных сетей стали постепенно превращаться в муниципальную компьютерную сеть, где по тем же линиям в неиспользуемой части спектра передаются цифровые данные (в обоих направлениях).

MAN может либо полностью принадлежать частной компании, которая владеет всем оборудованием и линиями. В другом случае линии и узлы, соединяющие локальные сети, могут принадлежать другой компании. Недавние разработки, связанные с высокоскоростным беспроводным доступом в Интернет, привели к созданию нового типа муниципальных сетей по стандарту IEEE 802.16, широко известному как стандарт WiMAX.

Глобальные сети (WAN)

Обеспечивают обмен информацией на территориях большой протяженности и могут охватывать территорию страны, континента или всего мира (рис. 1.15).

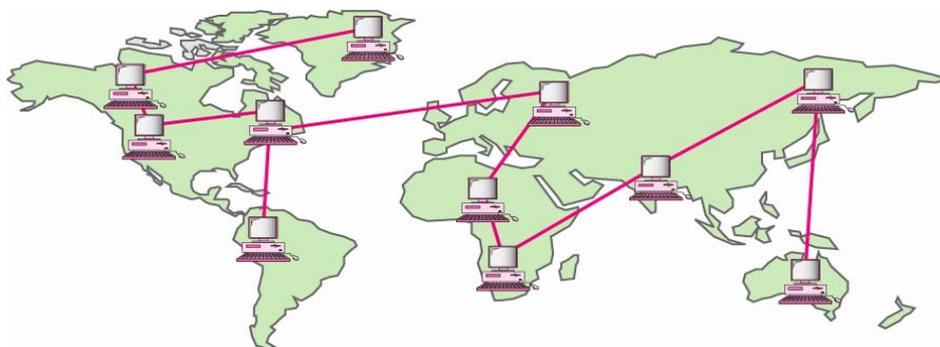


Рис. 1.15. Глобальная сеть

В отличие от локальных сетей глобальные чаще всего могут использовать как оборудование и каналы других компаний и организаций (аренда), так и собственное. Глобальная сеть, принадлежащая одной компании, является корпоративной сетью и используется исключительно для обеспечения собственных нужд. Самым известным примером глобальной сети является сеть Интернет.

1.3. Интернет

Интернет стал одним из основных средств получения информации в наши дни. Он повлиял на способы ведения бизнеса, а также отдыха и развлечений. Обмен электронными письмами, оплата счетов, чтение газет, заказ билетов, бронирование мест в отелях, покупки и многое другое – все это теперь возможно благодаря Интернет.

История Интернет

Сеть представляет собой группу взаимосвязанных и взаимодействующих друг с другом узлов (компьютеры, принтеры и др.). Две и более взаимосвязанных сети образуют *интернет* (с маленькой буквой *и*). Совокупность множества

взаимодействующих сетей, разбросанных по всему миру, образуют глобальную сеть *Интернет* (с большой буквы *I*). В наши дни Интернетом пользуются как частные лица, так и различные организации в большинстве стран мира. Количество пользователей составляет сотни миллионов и непрерывно растет.

В середине 60-х годов многие университеты и промышленные компании США уже обзавелись достаточно мощными на то время компьютерами, однако все они были обособлены. Возникла необходимость их соединения для оперативного обмена результатами исследований. Для этого была создана единая научная организация под покровительством Министерства обороны США, ARPA (Advanced Research Projects Agency – Управление перспективного планирования научно-исследовательских работ). Эта организация не имела ни ученых, ни лабораторий в своем штате и занималось тем, что выбирало из множества предлагаемых университетами и компаниями проектов наиболее перспективные и организовывало выделение грантов под эти проекты. Соединение компьютеров в сеть потребовалось для того, что бы ученые могли более эффективно обмениваться результатами исследований, тем самым, уменьшая стоимость исследовательских работ и исключая их дублирование.

В 1967 году ARPA представило свой проект (ARPANET) небольшой сети для связи нескольких компьютеров. Идея заключалась в том, что каждый компьютер (хост-компьютер), принадлежащий какой-либо организации, подключается к специализированному компьютеру, называемому IMP (Interface Message Processor). Несколько таких IMP компьютеров соединялись друг с другом линиями связи, обеспечивающими передачу информации со скоростью 56 кбит/с., и образовывали сеть. Для повышения надежности каждый IMP должен был соединяться как минимум с двумя IMP. Каждый узел связи фактически должен был состоять из IMP и хост-компьютера.

К 1969 году сеть ARPANET стала реальностью. В декабре этого года удалось запустить экспериментальную сеть, состоящую из четырех узлов, которые располагались в Калифорнийском университете в Лос-Анджелесе (UCLA), Калифорнийском университете в Санта-Барбаре (UCSB), Исследовательском институте Стэнфорда (SRI. Stanford Research Institute) и Университете штата Юта. В этих университетах имелись в то время различные и совершенно несовместимые хост-компьютеры. Сеть быстро росла по мере создания новых IMP и подключения к ним новых организаций и вскоре охватила все штаты. Связь между хост-компьютерами обеспечивалась посредством специально разработанного программного обеспечения, называемого NCP – Network Control Protocol (протокол управления сетью).

По мере роста сети стало ясно, что имеющиеся протоколы сети ARPANET малопригодны для работы с объединенными сетями. В результате дополнительных исследований в 1973 году был предложен новый протокол доставки пакетов от узла к узлу, названный *Transmission Control Protocol (TCP)*. Он был специально разработан для управления обменом данными по интернетам, что становилось все более актуальным по мере подключения новых сетей к ARPANET. В 1974 году было решено разделить TCP на два отдельных протокола: *TCP – Transmission Control Protocol* (протокол управления передачей) и

IP – Internetworking Protocol (протокол межсетевого взаимодействия). IP отвечает за маршрутизацию пакетов, в то время как TCP выполняет функции более высокого уровня – сегментация пакетов, их транспортировка и обнаружение ошибок. Общий протокол передачи и взаимодействия известен в настоящее время как *TCP/IP*.

К концу 70-х годов стало ясно, что сеть ARPANET оказывает большое влияние на исследовательские работы университетов, позволяя ученым обмениваться информацией и совместно работать над проектами, но она принадлежала Министерству обороны и имела ограничения к доступу. Национальный научный фонд США (NSF. National Science Foundation) создал правительственную сеть, состоящую из магистрали и региональных сетей. Эта сеть получила имя NSFNET, имела связь с ARPANET и с самого начала использовала протокол TCP/IP. Сеть развивалась высокими темпами и в 1990 году была передана во владение некоммерческой корпорации ANS (Advanced Networks and Services, Inc.).

После того как TCP/IP стал единственным официальным протоколом и сети NSFNET и ARPANET объединились, рост стал экспоненциальным. К сети стали присоединяться множество региональных сетей, сети разных стран и континентов стали объединяться. Примерно с середины 80-х годов множество сетей стали называть интернетью (internet), а впоследствии Интернетом (Internet).

В конце 1991 года Конгресс США утвердил закон о создании сети NREN (National Research and Education Network – государственная научно-исследовательская и образовательная сеть). Для облегчения перехода с одних сетей на другие и обеспечения взаимоувязанности различных сетей заключил контракт с четырьмя различными сетевыми операторами об организации пунктов доступа к сети (NAP, Network Access Point). В результате концепция единой магистрали была заменена коммерчески управляемой конкурентной инфраструктурой. В других странах и регионах в 90-х годах также были построены сети, сравнимые с NSFNET.

В период примерно с 1970 по 1990 год у Интернета были четыре основные сферы применения: электронная почта, новости (конференции, форумы), удаленный доступ, перенос файлов. Вплоть до 1990-х годов Интернет был популярен среди академических, государственных и промышленных организаций. Новое приложение, WWW (World Wide Web – всемирная паутина) позволило размещать в сети в виде страниц для общего доступа информацию в виде текста, изображения, звука и видео, со встроенными ссылками на другие страницы. Это привело к массовому распространению Интернета по всему миру. Таким образом, Интернет из закрытой сети для военных и ученых превратился в услугу общего доступа. Точное количество пользователей сети Интернет определить уже невозможно, но оно исчисляется сотнями миллионов.

Архитектура Интернет

Сегодня Интернет представляет собой совокупность множества глобальных, муниципальных и локальных сетей, соединенных друг с другом линиями связи и узлами коммутации. Состав Интернет постоянно меняется, добавляются новые сети и технологии, удаляются устаревшие сегменты и сети организаций,

прекративших свое существование. Интернет не управляется одной компанией и его работа обеспечивается множеством частных компаний. Пользователи Интернет получают доступ к сети через Интернет-провайдеров (ISP, Internet Service Provider). Существуют провайдеры международного, национального, регионального и локального уровней. Общая концепция построения сети Интернет представлена на рис 1.16.

Международные Интернет-провайдеры (International ISP)

Находятся на верхнем уровне иерархии сети и связывают друг с другом различные национальные сети.

Национальные Интернет-провайдеры (National ISP)

Обеспечивают магистральные сети, создаваемые и поддерживаемые крупными коммерческими компаниями. Для обеспечения взаимодействия пользователей разных сетей, магистральные сети разных компаний соединяются друг с другом с помощью мощных маршрутизаторов, называемых пунктами доступа к сети (*NAP, Network Access Point*). Каждую магистраль должен представлять по крайней мере один маршрутизатор. Такие маршрутизаторы объединяются в локальную сеть, благодаря чему пакеты могут передаваться с маршрутизатора на маршрутизатор, то есть, фактически, с магистрали на магистраль. Кроме того, наиболее крупные магистрали могут быть связаны друг с другом на прямую, образуя равноранговую сеть.

Региональные Интернет-провайдеры (Regional ISP)

Сети региональных провайдеров подключаются к одному или более NSP и находятся на третьем уровне иерархии. Обеспечивают меньшие по сравнению с NSP скорости передачи.

Локальные Интернет-провайдеры (Local ISP)

Обеспечивают непосредственный доступ пользователей к услугам сети Интернет. Локальные ISP могут быть соединены к региональным ISP либо непосредственно к национальным. Примером локального ISP может выступать провайдер предоставляющий услуги Интернет общего пользования; компания, обеспечивающая услуги Интернет для своих сотрудников, а также непрофильные организации, такие как университеты, школы и др.

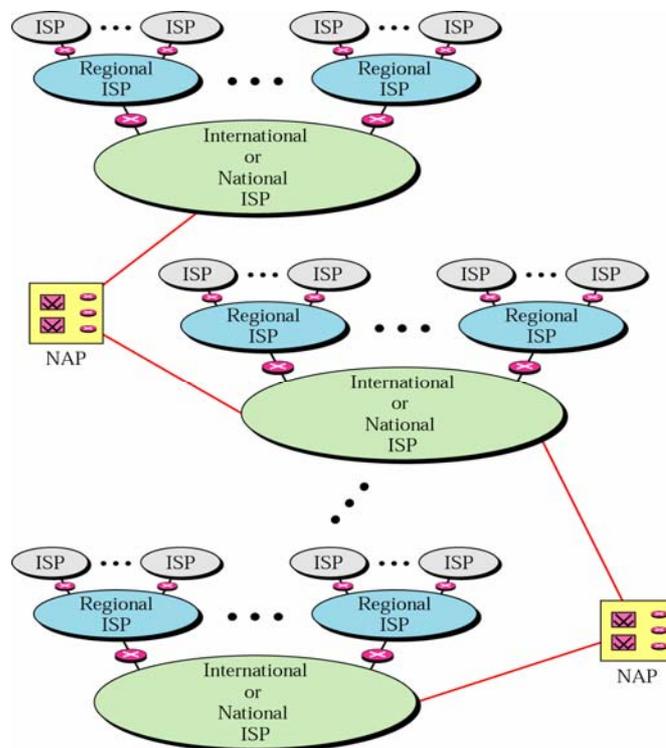


Рис. 1.16. Архитектура сети Интернет

1.4. Протоколы и стандарты

В этом разделе рассматриваются два широко используемых термина: протоколы и стандарты.

Протоколы

В компьютерных сетях передача данных осуществляется между объектами разных систем. Здесь под понятием *объект* подразумевается устройство, способное принимать и передавать информацию. Однако два объекта разных систем не в состоянии понимать друг друга. Для решения этой проблемы два объекта должны взаимодействовать по заранее определенному протоколу.

Процедура взаимодействия двух узлов может быть описана в виде набора правил. *Протоколом* это набор формализованных правил, определяющих последовательность и формат сообщений, которыми обмениваются сетевые компоненты. Ключевыми элементами протокола являются синтаксис, семантика и синхронизация.

Синтаксис

Описывает структуру или формат данных, определяющих порядок, в котором они представляются. Например, простой протокол мог бы подразумевать первые 8 бит данных как адрес источника, следующие 8 бит как адрес приемника и остальные биты блока данных – само сообщение.

Семантика

Характеризует значение каждого блока данных. Как интерпретируется конкретная комбинация, и какие действия должны быть предприняты на основе этой интерпретации? Например, означает ли адрес в блоке данных маршрут передачи или конечный пункт доставки сообщения.

Синхронизация

Дает ответы на вопросы: когда и на какой скорости должны посылаться данные. Например, если источник посылает данные на скорости 100 Мбит/с, а приемник способен обрабатывать данные лишь на скорости 1 Мбит/с, то это приведет к перегрузке приемника и большая часть данных будет потеряна.

Стандарты

Так как все компоненты сети должны работать согласованно, важным является принятие многочисленных стандартов, которые, если не во всех, то хотя бы в большинстве случаев, гарантировали бы совместимость оборудования и программ различных фирм-изготовителей. Большинство стандартов, принимаемых сегодня, носят открытый характер, так как возможность легкого взаимодействия с продуктами конкурентов не снижает, а наоборот, повышает ценность изделия, так как его можно применить в большем количестве работающих сетей, построенных на продуктах разных производителей.

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

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

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

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

Организации по стандартизации в сфере передачи данных

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

Органы по созданию стандартов

Международная организация по стандартизации (International Organization for Standardization, ISO, часто называемая также International Standards Organization) представляет собой ассоциацию ведущих национальных организаций по стандартизации разных стран. Главным достижением ISO явилась модель взаимодействия открытых систем OSI, которая в настоящее время является концептуальной основой стандартизации в области вычислительных

сетей. В соответствии с моделью OSI этой организацией был разработан стандартный стек коммуникационных протоколов OSI.

Международный союз электросвязи (International Telecommunication Union, ITU) – организация, являющаяся в настоящее время специализированным органом Организации Объединенных Наций. Ранее наиболее значительную роль в стандартизации телекоммуникационных сетей и сетей передачи данных играл постоянно действующий *Международный консультативный комитет по телефонии и телеграфии (МККТТ) (Consultative Committee on International Telegraphy and Telephony, CCITT)*. В результате проведенной в 1993 году реорганизации CCITT несколько изменил направление своей деятельности и сменил название – теперь он называется *Сектором стандартизации в области телекоммуникаций (Telecommunication Standardization Sector, ITU-T)*, Основу деятельности ITU-T составляет разработка международных стандартов в области телефонии, телематических служб (электронной почты, факсимильной связи, телетекста, телекса и т. д.), передачи данных, аудио- и видеосигналов. За годы своей деятельности ITU-T выпустил огромное число рекомендаций-стандартов. Свою работу ITU-T строит на изучении опыта сторонних организаций, а также на результатах собственных исследований. Раз в четыре года издаются труды ITU-T в виде так называемой «Книги», которая на самом деле представляет собой целый набор обычных книг, сгруппированных в выпуски, которые, в свою очередь, объединяются в тома. Каждый том и выпуск содержат логически взаимосвязанные рекомендации.

Институт инженеров по электротехнике и радиоэлектронике (Institute of Electrical and Electronics Engineers, IEEE) – крупнейшее в мире инженерно-техническое общество, способствующее развитию теоретическим и производственным аспектам в области электротехники, электроники, радиотехники и построению сетей. В 1981 году рабочая группа 802 этого института сформулировала основные требования, которым должны удовлетворять локальные вычислительные сети. Группа 802 определила множество стандартов, из них самыми известными являются стандарты 802.1, 802.2, 802.3, 802.5 и 802.11, которые описывают общие понятия, используемые в области локальных сетей.

Ассоциация электронной промышленности (Electronic Industries Association, EIA) – промышленно-торговая группа производителей электронного и сетевого оборудования; проявляет значительную активность в разработке стандартов для проводов, коннекторов и других сетевых компонентов. Ее наиболее известный стандарт – RS-232C.

Американский национальный институт стандартов (American National Standards Institute, ANSI) – эта организация представляет США в Международной организации по стандартизации ISO. Комитеты ANSI ведут работу по разработке стандартов в различных областях вычислительной техники и передачи данных.

Регулирующие органы

Все аспекты функционирования сетей передачи данных подлежат регулированию со стороны органов власти государства, например, Федеральная ко-

миссия по связи в США (Federal Communications Commission, FCC) или Министерство информационных технологий и связи Российской Федерации.

Стандартизация Интернет

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

Основной организацией по стандартизации Интернет является Internet Society (ISOC) – профессиональное сообщество, которое занимается общими вопросами эволюции и роста Internet как глобальной коммуникационной инфраструктуры. Под управлением ISOC работает Internet Architecture Board (IAB) - организация, в ведении которой находится технический контроль и координация работ для Internet. IAB координирует направление исследований и новых разработок для стека TCP/IP и является конечной инстанцией при определении новых стандартов Internet.

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

ГЛАВА 2

МНОГОУРОВНЕВАЯ СЕТЕВАЯ ИЕРАРХИЯ

Универсальный тезис о пользе стандартизации, справедливый для всех отраслей, в компьютерных сетях приобретает особое значение. Суть сети – это соединение разного оборудования, а значит, проблема совместимости является одной из наиболее острых. В сетях передачи данных идеологической основой стандартизации является многоуровневый подход к разработке средств сетевого взаимодействия.

Для передачи данных сеть использует комбинацию аппаратных и программных средств. Аппаратные средства состоят из физического оборудования, которое преимущественно обеспечивает перенос сигналов от одного узла сети в другой. Так как сами по себе услуги, предоставляемые сетью, являются более сложными, по сравнению с простой передачей сигналов, то для реализации логики работы сети необходимы программные средства.

Организация взаимодействия между устройствами в сети является сложной задачей. Как известно, для решения сложных задач используется универсальный прием – декомпозиция, то есть разбиение одной сложной задачи на несколько более простых задач-модулей. Например, задача отправки E-mail с одной точки в другую может быть поделена на несколько задач, каждая из которых выполняется отдельным программным модулем или пакетом. Каждый такой программный модуль одного уровня использует результаты выполнения программы предыдущего уровня. Таким образом, при декомпозиции часто используется многоуровневый подход.

2.1. Иерархия задач

Многоуровневый подход заключается в том, что все множество задач разбивают на уровни, которые образуют иерархию, то есть имеются вышележащие и нижележащие уровни. Для выполнения своих задач каждый уровень обращается с запросами только непосредственно примыкающему нижележащему уровню. С другой стороны, результаты работы некоторого уровня, могут быть переданы только соседнему вышележащему уровню. Такая иерархическая декомпозиция задачи предполагает четкое определение функции каждого уровня и интерфейсов между ними. Интерфейс определяет набор функций, которые нижележащий уровень предоставляет вышележащему. В результате достигается относительная независимость уровней, а значит, и возможность их легкой замены. Таким образом, средства сетевого взаимодействия также могут быть представлены в виде иерархически организованного множества уровней. При этом модули нижнего уровня могут, например, решать все вопросы, связанные с надежной передачей электрических сигналов между двумя соседними узлами.

Модули более высокого уровня организуют транспортировку сообщений в пределах всей сети, пользуясь для этого средствами упомянутого нижележащего уровня. А на верхнем уровне работают модули, предоставляющие пользователям доступ к различным службам – файловой, печати и т.п. Иерархически организованный набор протоколов, достаточный для организации взаимодействия узлов в сети, называется *стеком коммуникационных протоколов*.

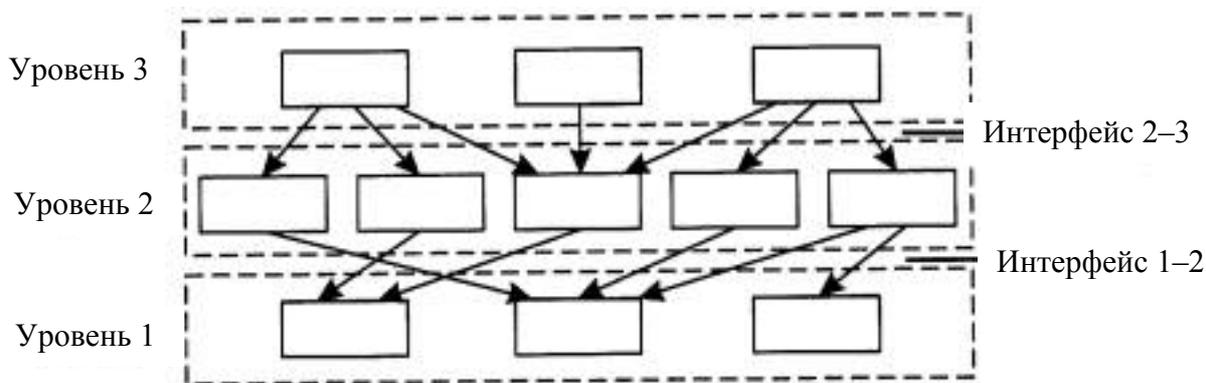


Рис. 2.1. Многоуровневый подход – создание иерархии задач

В процессе обмена сообщениями участвуют две стороны и необходимо организовать согласованную работу двух «иерархий» многоуровневого представления. При передаче сообщений оба участника сетевого обмена (узла) должны принять соглашения для всех уровней. Уровень N одного узла поддерживает связь с уровнем N другого узла. Формализованные правила и соглашения, определяющие последовательность и формат сообщений, которыми обмениваются компоненты этого уровня, называются *протоколом уровня N* .

Соседние уровни, находящиеся в одном узле, также взаимодействуют друг с другом в соответствии с четко определенными правилами. Эти правила принято называть *интерфейсом* который определяет набор сервисов, предоставляемый данным уровнем соседнему уровню (рис. 2.2).

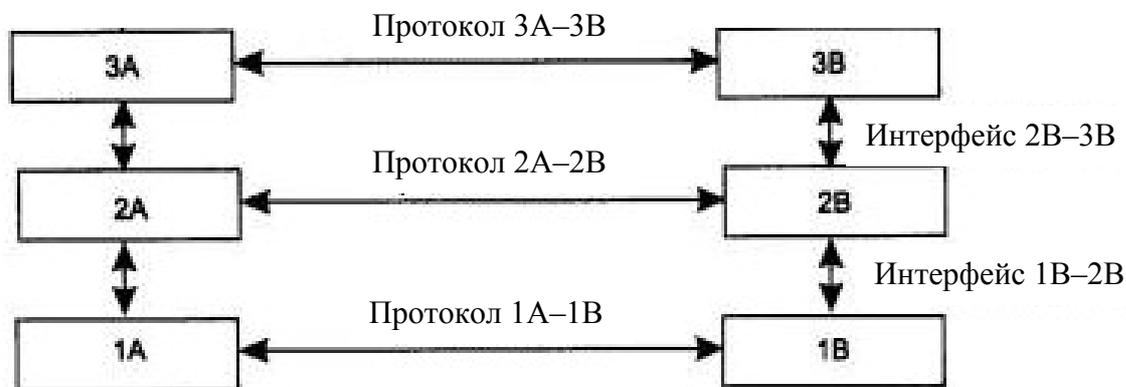


Рис. 2.2. Модель взаимодействия двух узлов

Следует различать понятия протокол и сервис (служба). *Служба* – это набор операций, которые более низкий уровень предоставляет более высокому, т.е. нижний уровень является поставщиком сервиса, а верхний – его потребителем. Протокол, в свою очередь, задается набором правил в рамках равнорангового (одноуровневого) процесса. Таким образом, службы связаны с межуровневыми интерфейсами, а протоколы – равноранговыми процессами.

Рассмотрим многоуровневый подход на простом примере отправки почтового письма. Решение такой задачи иллюстрируется на рис. 2.3.

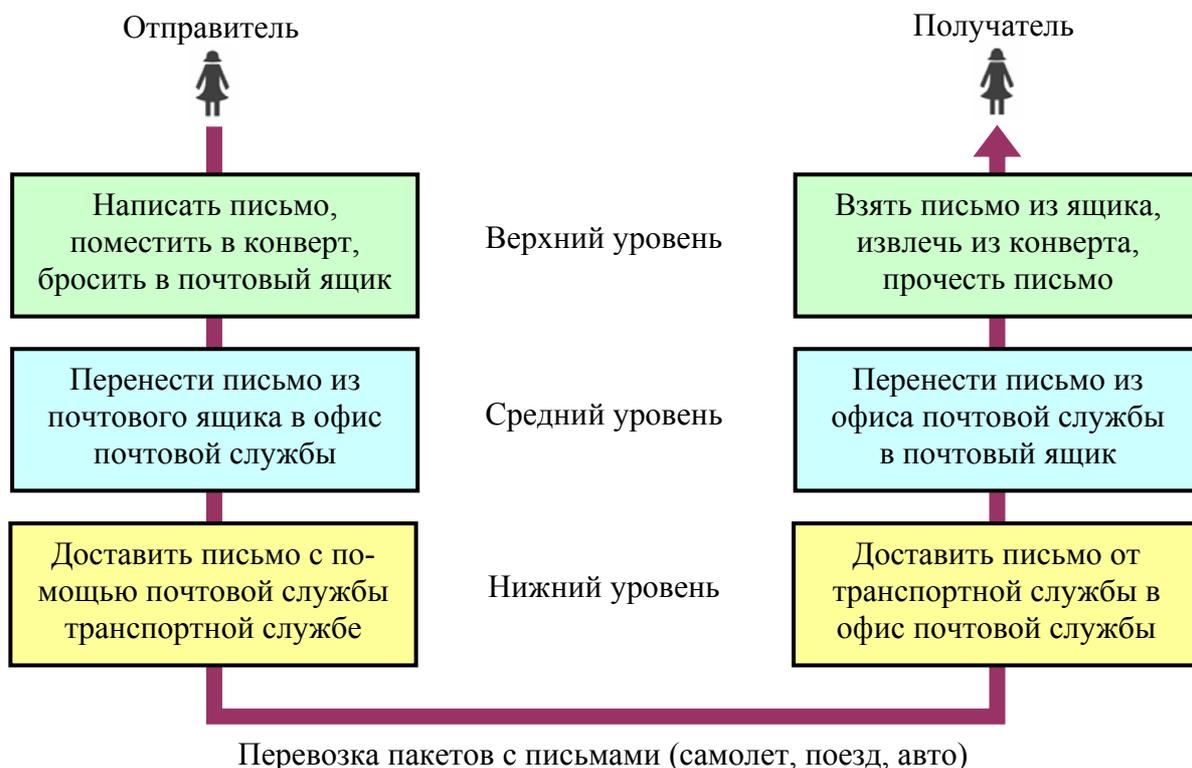


Рис. 2.3. Многоуровневая модель отправки почтового письма

На стороне отправителя

На *верхнем уровне* иерархии задач отправитель пишет письмо, вкладывает его в конверт, пишет на нем адреса отправителя и получателя, бросает письмо в почтовый ящик. *Средний уровень* – почтальон забирает письмо из ящика и доставляет его в ближайшее почтовое отделение. На *нижнем уровне* в почтовом отделении все письма сортируются и заданное письмо передается соответствующей транспортной службе для его перевозки.

На пути к получателю

Письмо доставляется в другой почтовый офис, наиболее близко расположенный к получателю. В качестве средств доставки могут использоваться различный транспорт (авиа, ж/д, судоходный, автодорожный или их комбинации). На пути к почтовому офису получателя письмо может проходить через другие почтовые пункты (в модели на рис. 2.1 не предусмотрено).

На стороне получателя

Нижний уровень – транспортная служба доставляет письмо в почтовый офис получателя. *Средний уровень* – письмо обрабатывается в почтовом отде-

лении и на основе адреса почтальон доставляет его в почтовый ящик получателя. На *верхнем уровне* получатель забирает конверт из ящика, открывает его и читает письмо.

Весь процесс отправки и получения письма подчиняется строгой иерархии. Решение последующей задачи возможно лишь после решения предыдущей. На передающей стороне чтобы письмо было отправлено (забрано почтальоном), необходимо его сначала написать, поместить в конверт и бросить в почтовый ящик. На приемной стороне для того, чтобы получить и прочесть письмо, необходимо его доставить до почтового ящика получателя.

2.2. Эталонная модель Интернет

В настоящее время принято считать, что сети Интернет соответствует пятиуровневая модель, состоящая из следующих уровней (рис. 2.4): физический (Уровень 1), канальный (Уровень 2), сетевой (Уровень 3), транспортный (Уровень 4) и прикладной (Уровень 5). В модели Интернет используются наиболее значимые элементы известной семиуровневой модели OSI. Каждый уровень определяет семейство функций, отличных от функций других уровней. На рис. 2.5. показаны уровни процессы передачи сообщений от узла (Device) А к узлу В.

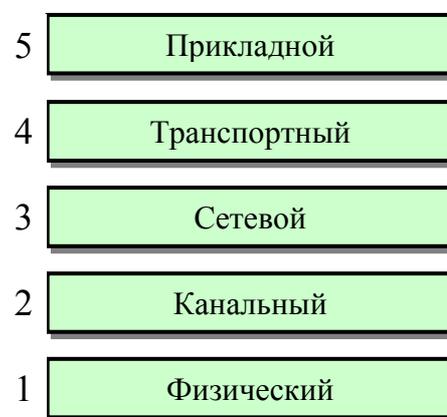


Рис. 2.4. Уровни в модели Интернет

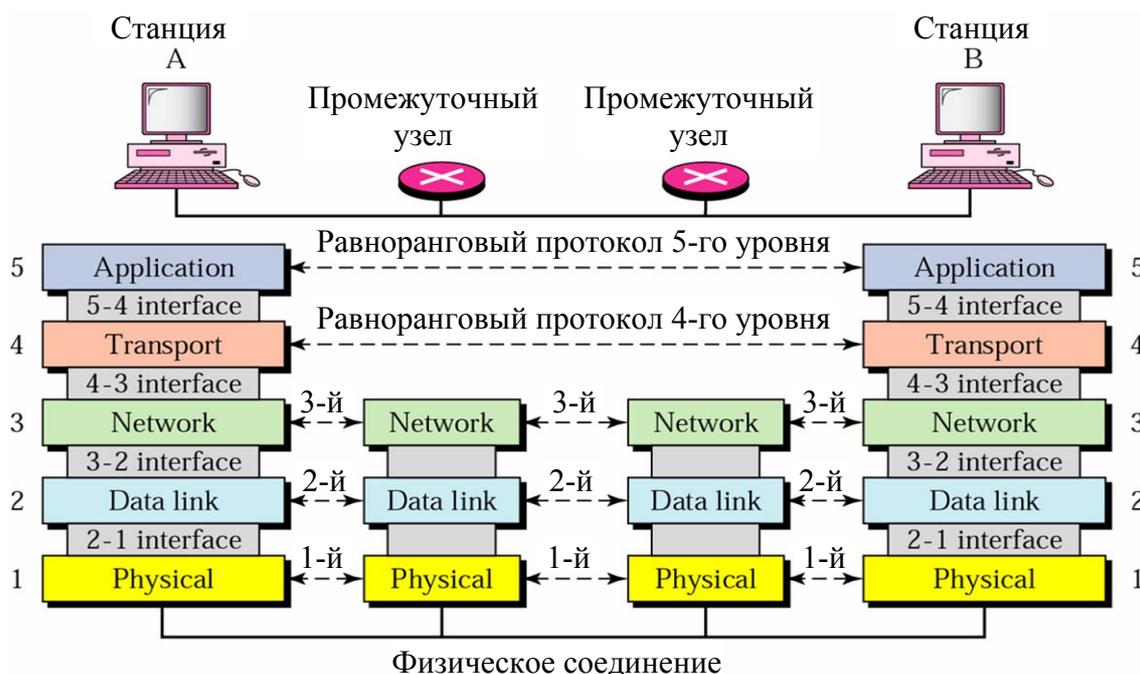


Рис. 2.5. Уровни и равноранговые (peer-to-peer) процессы

В пределах одного узла каждый уровень обращается к сервису нижележащего уровня и предоставляет сервис более высокому уровню. Между различными узлами сети уровни одного ранга взаимодействуют друг с другом для обеспечения процесса передачи информации. Эта взаимосвязь определяется набором правил и соглашений, называемым *протоколом*. Процесс взаимодействия узлов в пределах заданного уровня называется *равноранговым (peer-to-peer) процессом*. Следовательно, передача сообщений между узлами является равноранговым процессом, использующим протоколы соответствующего уровня.

Равноранговые процессы

На физическом уровне взаимосвязь узлов непосредственная: узел А пересылает поток битов непосредственно узлу В. Однако на более высоких уровнях их взаимодействие осуществляется перемещением сообщения вниз по уровням на узле А, транспортировкой по среде передачи и обратным перемещением вверх по уровням на узле В. Каждый уровень на передающем узле добавляет собственную (служебную) информацию к сообщению, которое он принимает от верхнего соседнего уровня и передает данные нижестоящему уровню.

На уровне 1 весь комплект данных, вместе со служебной информацией, преобразуется в форму, необходимую для транспортировки по среде передачи. На приемном узле данные распаковываются от уровня к уровню, принимая необходимые для соответствующего уровня данные и удаляя ненужные. Такая передача данных между уровнями осуществляется посредством интерфейсов между каждой парой смежных уровней. Каждый интерфейс определяет – какую информацию и сервис необходимо предоставить более высокому уровню. Тем самым обеспечивается независимость различных уровней, т.е. возможность модификации одних уровней, не влияя на другие.

Организация уровней

Пять уровней модели Интернет можно разбить на три основные подгруппы. Первые три уровня (физический, канальный и сетевой) относятся к классу *межсетевого уровня* и обеспечивают перенос данных от одного сетевого узла к другому (электрические спецификации передачи, физические соединения, адресация, синхронизация доставки и надежность). Пятый уровень (прикладной) можно трактовать как *уровень поддержки пользователя*, он обеспечивает взаимодействие конкретных прикладных программ (процессов) в окружении других. Например, если передается непрерывный поток видеoinформации, то на приемной стороне он принимается пользователем с помощью определенных программных средств, среди множества других имеющихся. Четвертый уровень (*транспортный*) связывает две вышеупомянутых класса уровней и преобразует данные от нижних уровней в ту форму, которая необходима для пятого уровня, например, формирует исходное сообщение из отдельных сегментов.

На рис. 2.6 представлен развернутый процесс передачи сообщений на основе многоуровневой модели Интернет, где *L5* означает блок данных пятого уровня, *L4* – четвертого и т.д.

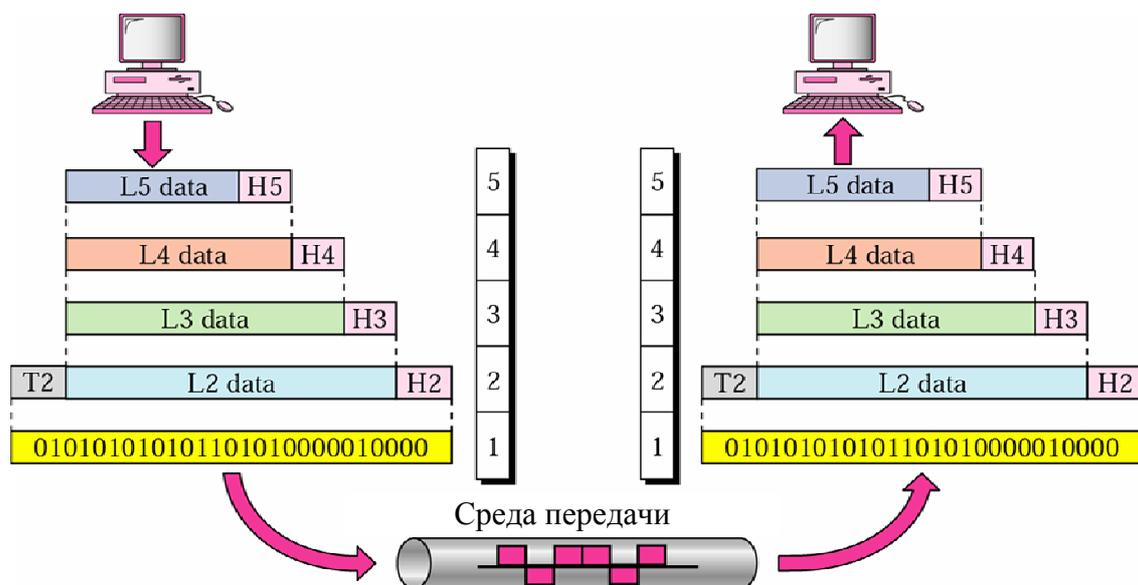


Рис. 2.6. Процесс передачи сообщений на основе пятиуровневой модели Интернет

Процесс передачи начинается на прикладном уровне 5 и снижается вниз от уровня к уровню в последовательном порядке. На каждом уровне, как правило, к блоку данных добавляется заголовок H со служебной информацией соответствующего уровня. На уровне 2 (передачи данных) также добавляется специальная последовательность бит T , обеспечивающая синхронизацию кадров и коррекцию ошибок. Физический уровень 1 преобразует сформированный поток данных (битов) в электромагнитный сигнал и обеспечивает его транспортировку.

На приемном конце сигнал поступает на физический уровень 1, где трансформируется обратно в цифровую форму (поток битов). Затем блоки данных поднимаются вверх в обратном порядке. Когда блок данных достигает следующего вышестоящего уровня, заголовки и дополнительные биты, добавленные на соответствующем уровне передающего узла, удаляются и предпринимаются действия по обработке данных этого уровня. По достижению прикладного уровня 5, сообщение соответствует форме требуемого прикладного процесса (программного обеспечения) и становится доступным получателю.

Функции уровней

Физический уровень (*Physical Layer*)

Этот уровень обеспечивает передачу потока битов по физической среде. К этому уровню имеют отношение характеристики физических сред передачи данных, такие как полоса пропускания, помехозащищенность, волновое сопротивление и другие. На этом же уровне определяются характеристики электрических сигналов, передающих дискретную информацию, например, крутизна фронтов импульсов, уровни напряжения или тока передаваемого сигнала, тип кодирования, скорость передачи сигналов. Кроме этого, здесь стандартизируются типы разъемов и назначение каждого контакта. Физический уровень определяет процедуры и функции, которые должны выполнять физические устройства и интерфейсы для

транспортировки битов. *Преамбула* (специальная последовательность бит) подготавливает приемный узел к приему кадра. Функции физического уровня реализуются во всех устройствах, подключенных к сети. Например, со стороны компьютера функции физического уровня выполняются сетевым адаптером или последовательным портом. Рис. 2.7. иллюстрирует положение физического уровня по отношению к среде передачи и каналному уровню.

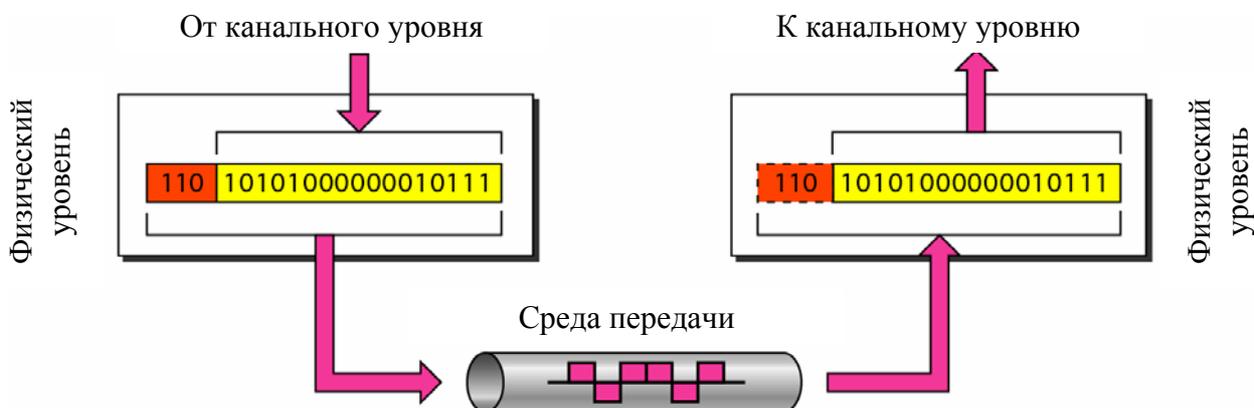


Рис. 2.7. Физический уровень

Физический уровень отвечает за передачу индивидуальных битов от одного узла к другому

Ниже перечислены основные функции физического уровня:

- **Физические характеристики интерфейсов и среды передачи.** Физический уровень определяет характеристики интерфейса между устройствами (узлами) и средой передачи. Он также определяет типы среды передачи.
- **Представление битов.** Данные физического уровня состоят из потока битов (двоичная последовательность нулей и единиц) без какой либо интерпретации. Для передачи по физической среде эти биты должны быть преобразованы в сигналы – электрические или оптические. Здесь определяется тип представления (каким образом последовательность битов преобразуется в сигналы).
- **Скорость передачи данных.** Скорость передачи (бит/с) также задается на физическом уровне и определяет время, затрачиваемое для передачи каждого бита.
- **Битовая синхронизация.** Передатчик и приемник должны иметь не только одинаковую скорость обработки битов, но и быть синхронизированы на уровне бит. Т.е. тактовые генераторы передатчика и приемника должны быть синхронны.
- **Линейная конфигурация.** Определяется тип соединения: «точка-точка» или многоточечное соединение.
- **Физическая топология.** Определяется схема соединения узлов
- **Режим передачи.** Симплексный, полудуплексный или дуплексный

Канальный уровень (Data Link Layer)

Этот уровень преобразует ненадежную среду передачи физического уровня в более надежный канал для доставки данных к следующему сетевому уровню. Иными словами, задача этого уровня – организовать надежный канал, свободный от необнаруженных ошибок, для передачи «сырых» данных физического уровня. Для этого на уровне передачи данных поток битов, поступающих с физического уровня разбивается на кадры. Здесь обеспечивается корректность передачи каждого кадра, помещением специальной последовательности бит в кадр, для его выделения, а также вычисляется контрольная сумма, обрабатывая все байты кадра определенным способом и добавляя контрольную сумму к кадру. Функция исправления ошибок не является обязательной для канального уровня, поэтому в некоторых протоколах этого уровня она отсутствует, например, в Ethernet и Frame Relay. В некоторых сетях одной из задач канального уровня является проверка доступности среды передачи с общим доступом. Рис. 2.8. иллюстрирует отношение канального уровня к сетевому и физическому уровням.

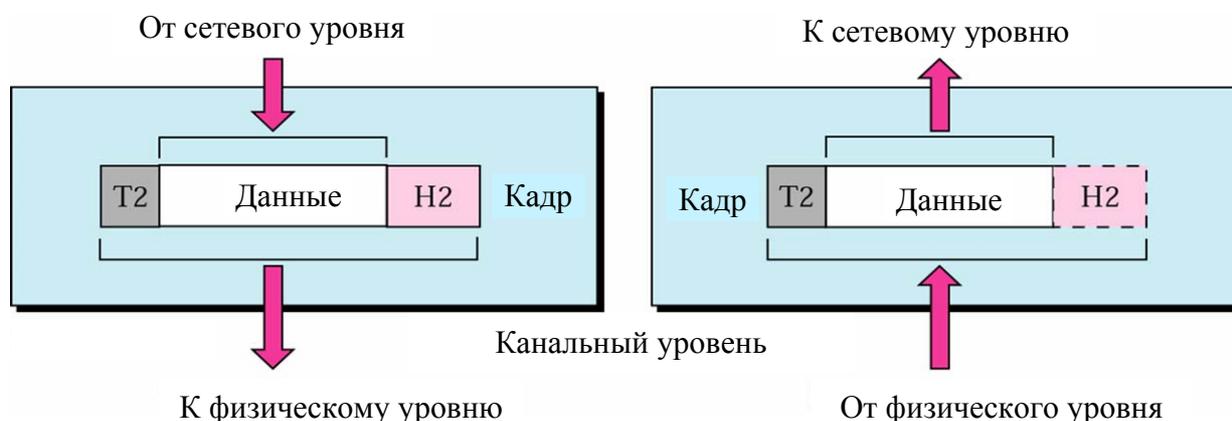


Рис. 2.8. Уровень передачи данных

Канальный уровень отвечает за передачу кадров от одного узла к другому

Ниже перечислены основные функции канального уровня:

- **Кадровая синхронизация.** Уровень передачи данных делит поток битов, принимаемых от физического уровня в блоки данных, называемые кадрами.
- **Физическая адресация.** Если различные кадры необходимо доставлять разным узлам, но в пределах одной сети, то уровень передачи данных добавляет заголовок к кадру для определения его отправителя и/или приемника. Если кадр необходимо доставить к узлу, принадлежащему другой сети, то адрес приемника является адресом промежуточного узла, соединяющего разные сети.
- **Управление потоком.** Если скорость, на которой данные принимаются приемником меньше чем скорость передатчика, то уровень передачи данных обеспечивает механизм управления потоком для пре-

дотвращения переполнения приемника (случай, когда передающая сторона «заваливает» данными медленную принимающую сторону).

- **Коррекция ошибок.** Уровень передачи данных повышает надежность с помощью механизма обнаружения и коррекции поврежденных или потерянных кадров. Обычно этот механизм обеспечивается путем добавления избыточных битов в конце кадра.
- **Управление доступом.** Когда два или более узла пытаются занять одну линию, то для определения очередности доступа во времени необходим соответствующий протокол на уровне передачи данных (алгоритм доступа к среде передачи).

Рис. 2.9. иллюстрирует доставку на канальном уровне методом «от узла к узлу» (*hop-to-hop delivery*).

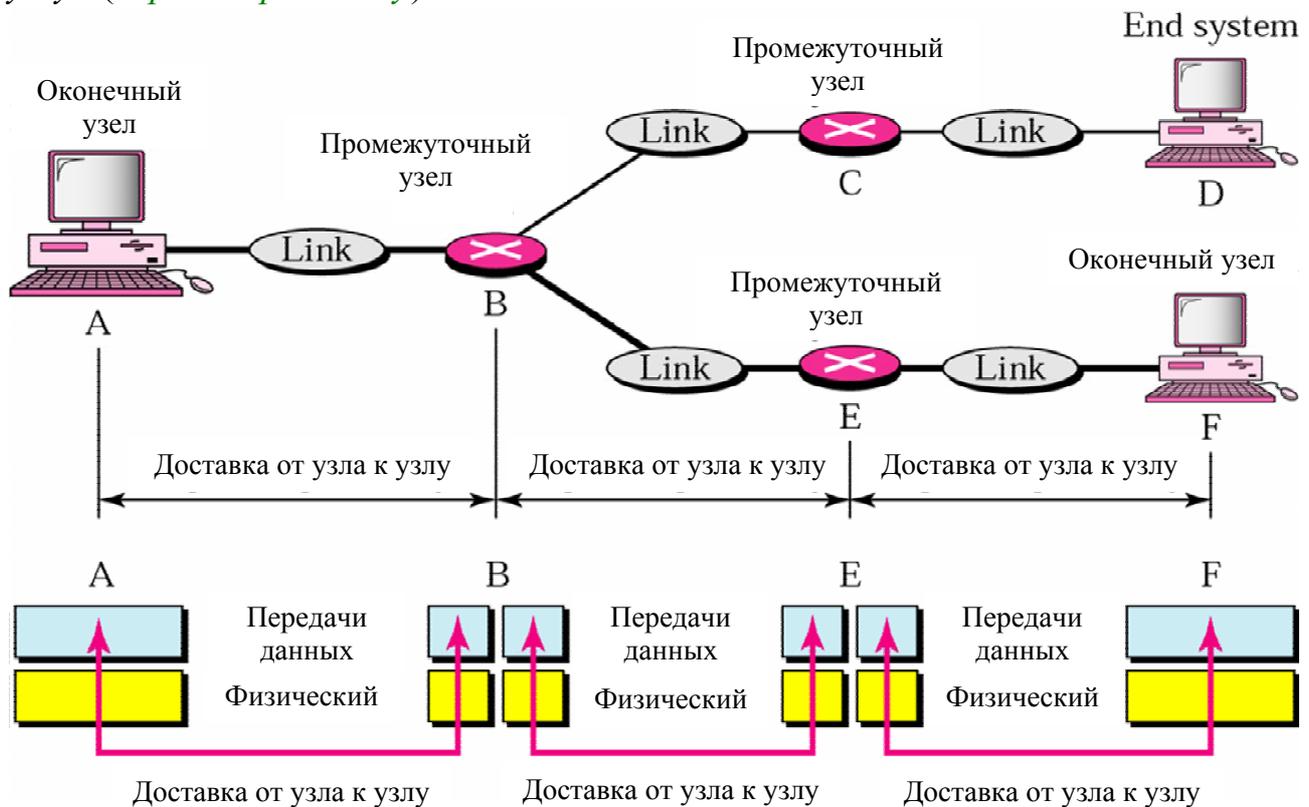


Рис. 2.9. Доставка данных от узла к узлу (*hop-to-hop delivery*)

Примерами протоколов канального уровня являются протоколы Ethernet, Token Ring, FDDI, 100VG-AnyLAN.

Сетевой уровень (Network Layer)

Этот уровень отвечает за доставку пакета от источника к пункту назначения (получателю) между различными сетями, в то время как канальный уровень отвечает за доставку данных между любыми узлами одной сети с соответствующей типовой топологией. Т.е. если два узла подключены к одному общему каналу, то необходимость в сетевом уровне отсутствует. Сетевой уровень поддерживает возможность правильного выбора маршрута передачи сообщения даже в том случае,

если узлы принадлежат разным сетям с разной топологией. Рис. 2.10. иллюстрирует отношение сетевого уровня к канальному и транспортному уровням.

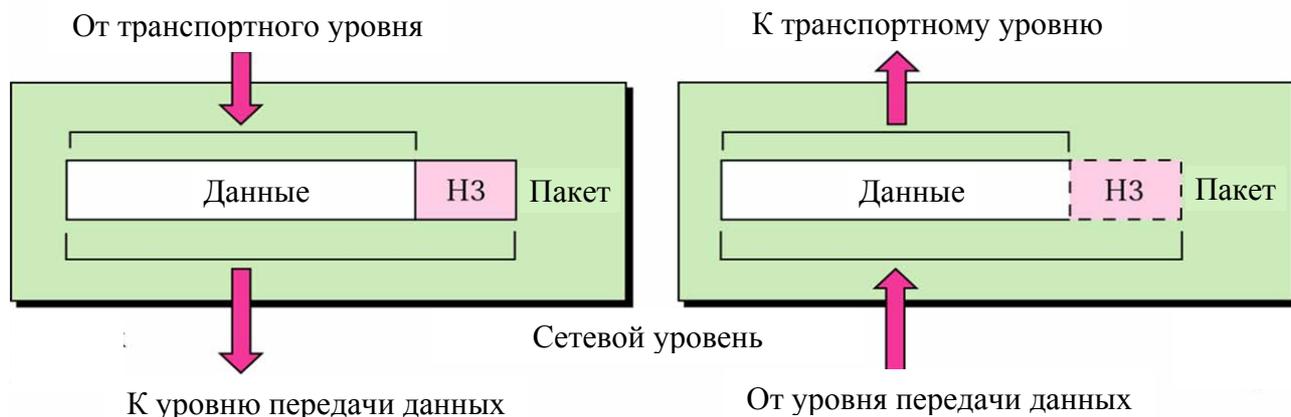


Рис. 2.10. Сетевой уровень

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

Сетевой уровень отвечает за доставку пакетов от источника к получателю

Ниже перечислены основные функции сетевого уровня:

- **Логическая адресация.** Физическая адресация, осуществляемая на уровне передачи данных, решает задачу адресации на локальном уровне (в пределах однотипной сети). Если пакеты пересекают границу одной сети, то необходима другая система адресации для различения узлов источника и получателя разных сетей. Сетевым уровнем добавляется заголовок к пакету, приходящему от верхнего уровня, который кроме другой служебной информации, содержит логические адреса узлов источника и получателя.
- **Маршрутизация.** Когда независимые сети или каналы соединяются для создания взаимоувязанной сети (сети сетей) или крупномасштабной сети, то для маршрутизации или коммутации пакетов к пункту назначения используются маршрутизаторы (routers) или коммутаторы (switches).

Рис. 2.11. иллюстрирует доставку пакетов *от источника к получателю* на сетевом уровне (*source-to-destination delivery*).

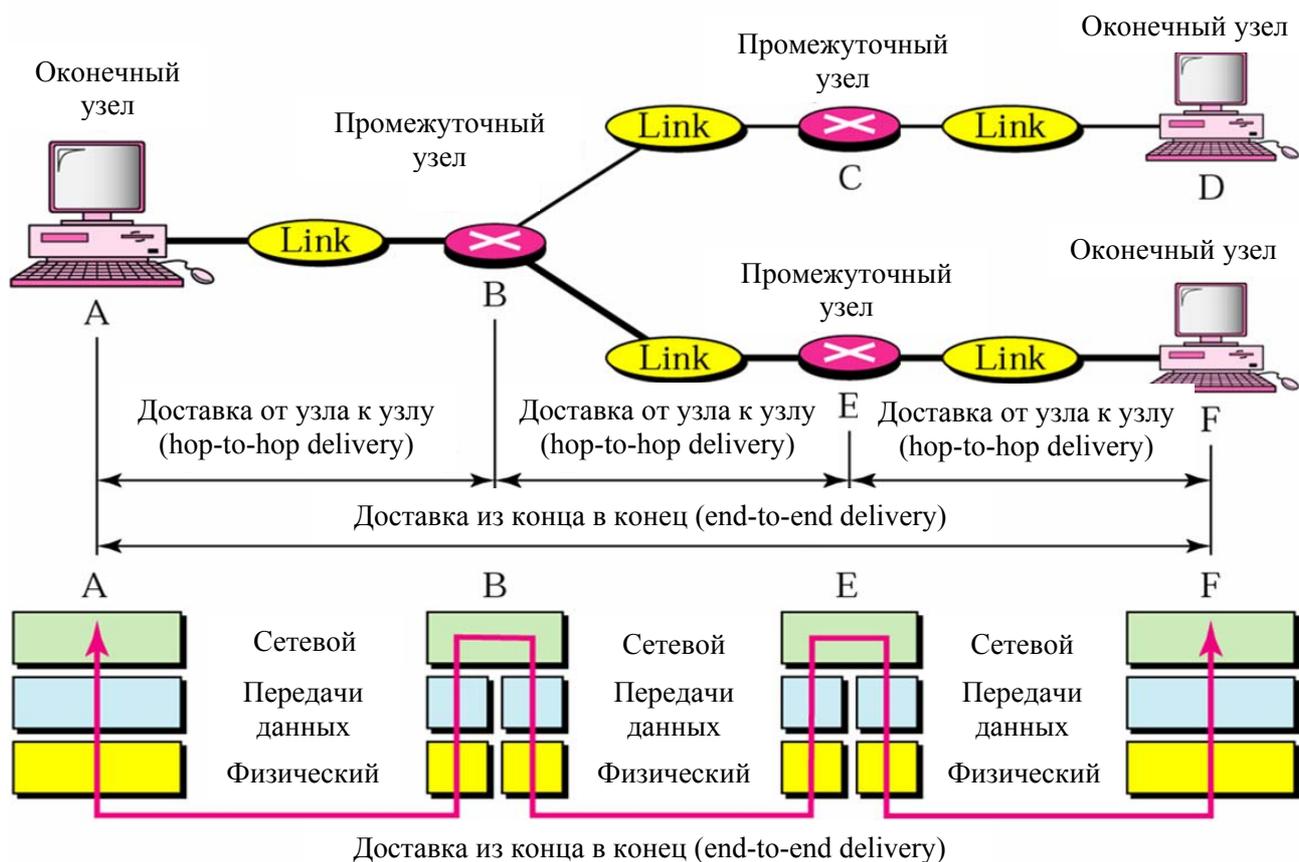


Рис. 2.11. Доставка пакетов от источника к получателю (*source-to-destination delivery*)

Сетевой уровень решает также задачи согласования разных технологий, упрощения адресации в крупных сетях и создания надежных и гибких барьеров на пути нежелательного трафика между сетями. Сообщения сетевого уровня принято называть *пакетами (packets)*. При организации доставки пакетов на сетевом уровне используется понятие «номер сети». В этом случае адрес получателя состоит из старшей части – номера сети и младшей – номера узла в этой сети. Все узлы одной сети должны иметь одну и ту же старшую часть адреса.

Транспортный уровень (Transport Layer)

Этот уровень отвечает за доставку всего сообщения от *процесса к процессу*. Несмотря на то, что сетевой уровень обеспечивает доставку индивидуальных пакетов от источника получателю, однако между этими пакетами никакой связи на этом уровне не фиксируется (например, принадлежат ли пакеты одному сообщению или разным) и все пакеты обрабатываются и доставляются независимо. С другой стороны, транспортный уровень гарантирует, что полное сообщение поступает без каких либо искажений (потерь отдельных пакетов) и в верном порядке (согласно упорядоченным номерам пакетов) обеспечивая как коррекцию ошибок, такт и управление потоком на уровне «от процесса к процессу». На пути от отправителя к получателю пакеты могут быть искажены или утеряны. Транспортный уровень обеспечивает приложениям или верхнему уровню стека – прикладному – передачу данных с той степенью надежности, которая ему требуется. Так как в сети работает много компьютеров и на них

одновременно могут выполняться сразу несколько процессов, каждому из таких процессов необходимо средство, позволяющее указать, с кем он хочет взаимодействовать. Следовательно, необходимо обеспечить идентификацию прикладных процессов отправителя и получателя.

Рис. 2.12. иллюстрирует отношение транспортного уровня к прикладному и сетевому уровням.

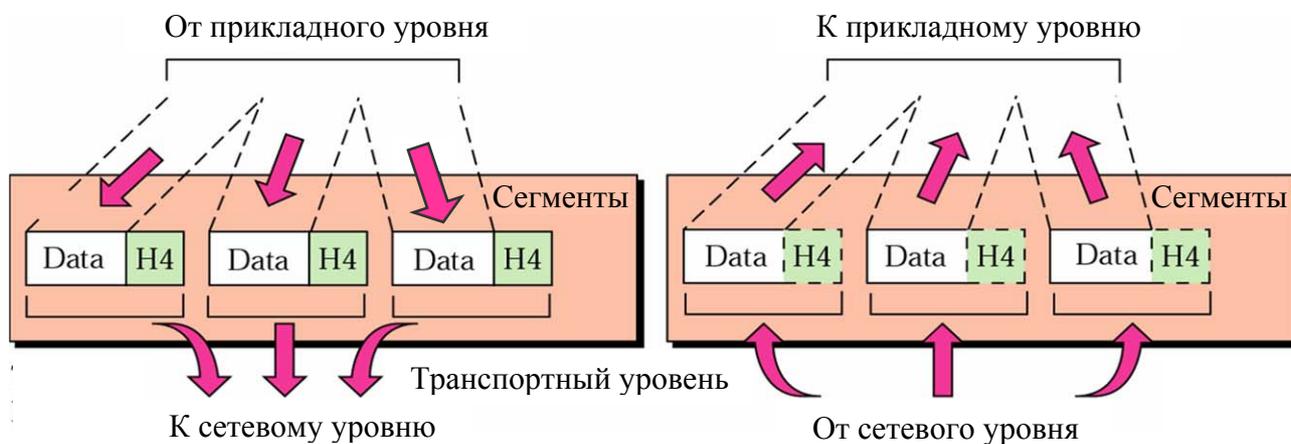


Рис. 2.12. Транспортный уровень

Транспортный уровень отвечает за доставку сообщения от одного процесса к другому

Ниже перечислены основные функции транспортного уровня:

- **Адресация процессов.** Компьютеры часто выполняют одновременно несколько процессов (активных программ). Следовательно, доставка от процесса к процессу означает не только доставку от одного компьютера к другому, но и от определенного процесса одного компьютера к аналогичному процессу другого компьютера. Таким образом, заголовок данных транспортного уровня должен включать тип адреса, называемый *адресом процесса* или *адресом порта*. Сетевой уровень обеспечивает доставку каждого пакета к требуемому компьютеру, а транспортный уровень обеспечивает доставку всего сообщения требуемому процессу этого компьютера.
- **Сегментация и сборка.** Сообщение разделяется на передаваемые сегменты, каждый из которых содержит порядковый номер. По этим номерам на приемном конце транспортный уровень производит сборку исходного сообщения, а также идентификацию и замену пакетов, потерянных при передаче.
- **Управление соединением.** Транспортный уровень может быть ориентированным на установление соединения либо без установления предварительных соединений. В случае без установления соединения каждый сегмент трактуется как независимый пакет. Транспортный уровень, ориентированный на установление соединения перед

доставкой пакетов предварительно осуществляет соединение с транспортным уровнем узла назначения. После передачи всех данных соединение разрывается.

- **Управление потоком.** Транспортный уровень тоже отвечает за управление потоком, как и канальный уровень. Однако на транспортном уровне управление потоком выполняется на уровне от источника к получателю (end-to-end), а не через один канал.
- **Коррекция ошибок.** На транспортном уровне коррекция ошибок также выполняется на уровне от источника к получателю (end-to-end), а не через один канал, как на канальном уровне. Передающий узел (компьютер) удостоверяется в том, что все сообщение поступает на приемный транспортный уровень без ошибок. Коррекция ошибок на транспортном уровне обычно осуществляется путем переприема пакетов с повторной передачи в случае ошибок.

Рис. 2.13. иллюстрирует доставку от процесса к процессу на транспортном уровне (*process-to-process delivery*).

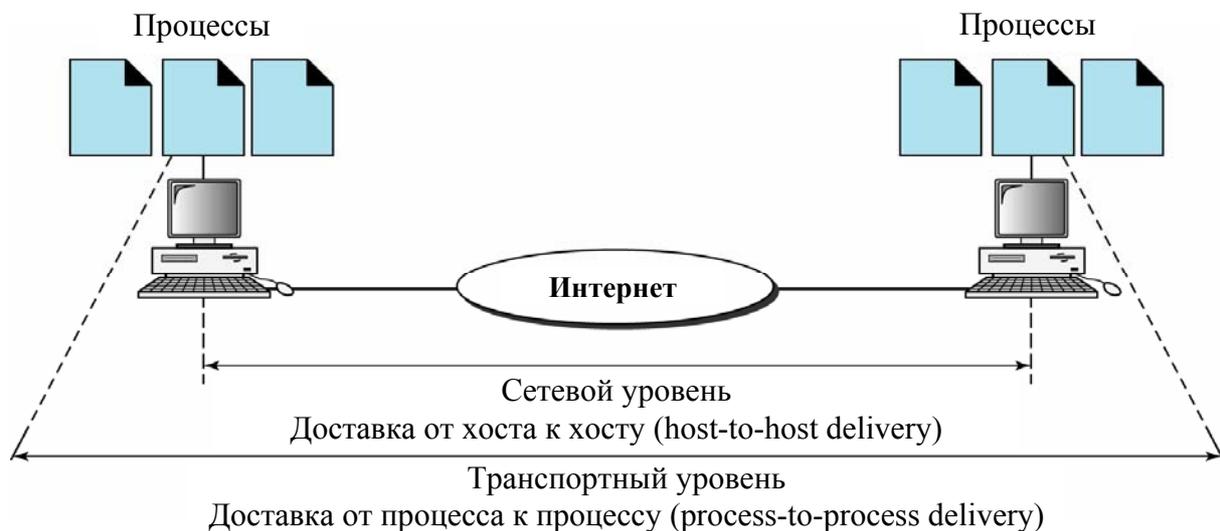


Рис. 2.13. Надежная доставка сообщения от «процесса к процессу»

Протоколы нижних четырех уровней обобщенно называют сетевым транспортом или транспортной подсистемой, так как они полностью решают задачу транспортировки сообщений с заданным уровнем качества в составных сетях с произвольной топологией и различными технологиями. Последний верхний уровень (прикладной) решает задачи предоставления прикладных сервисов на основании имеющейся транспортной подсистемы. Как правило, все протоколы, начиная с транспортного уровня и выше, реализуются программными средствами конечных узлов сети – компонентами их сетевых операционных систем.

В качестве примера транспортных протоколов можно привести протоколы *TCP* и *UDP* стека TCP/IP и протокол *SPX* стека Novell. Наибольшее распространение в сети Интернет получили два сквозных протокола – *TCP* и *UDP*. Первый, *TCP* (Transmission Control Protocol – протокол управления передачей), является надежным протоколом с установлением соединения и обеспечивает надежную

доставку сообщения с одного компьютера (точнее, его процесса) на другой. Кроме того, *TCP* осуществляет управление потоком, чтобы быстрый отправитель не завалил информацией медленного получателя. Второй протокол транспортного уровня, *UDP* (User Data Protocol – пользовательский протокол данных), является ненадежным протоколом без установления соединения, т.е. доставка отдельных пакетов данных (обычно потоковых) не гарантируется. Кроме того, протокол *UDP* предоставляет собственное управление потоком данных. Этот протокол широко используется в приложениях, в которых оперативность важнее точности, например, для передачи речи и видео.

Прикладной уровень (*Application Layer*)

Прикладной уровень – это набор разнообразных протоколов, с помощью которых пользователи (человек или программа) получают доступ к сети и ее разделяемым ресурсам. Прикладной уровень обеспечивает пользователя интерфейсами и поддержку таких служб как электронная почта (протоколы *SMTP* и *POP3*), удаленный доступ к файлам и их передача (протокол *FTP*), доступ к WWW по протоколу гипертекстовых ссылок *HTTP* и др. Сами приложения конечного пользователя (различные программы доступа к службам сети) используют протоколы прикладного уровня. Единица данных, которой оперирует прикладной уровень, обычно называется *сообщением* (*message*).

Рис. 2.15. иллюстрирует отношение прикладного уровня к пользователю и транспортному уровню.

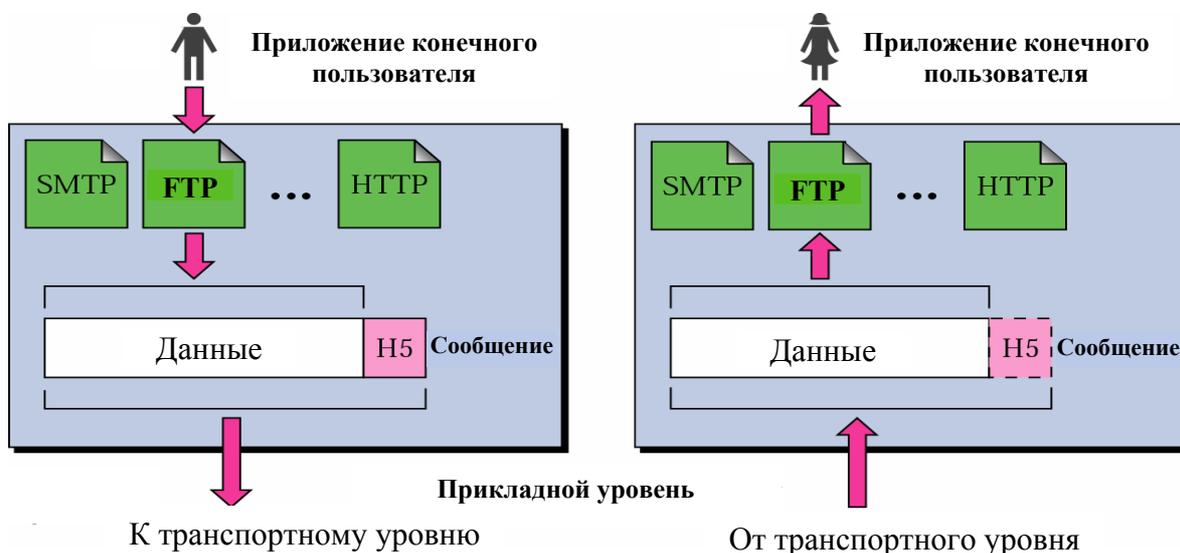


Рис. 2.15. Прикладной уровень

Прикладной уровень отвечает за предоставление различных услуг конечному пользователю

Ниже перечислены основные функции прикладного уровня:

- *Услуги электронной почты.* Реализация протоколов отправки и приема почтовых сообщений.

- *Передача файлов и доступ.* Предоставление пользователям (прикладным программам) доступа к файлам на удаленном компьютере (для их чтения и изменения), загрузка файлов с удаленного компьютера на локальный, управление файлами на удаленном компьютере.
- *Удаленная регистрация (логин).* Предоставление возможности пользователю зарегистрироваться на удаленном компьютере для получения доступа к его ресурсам.
- *Доступ к WWW.* Предоставление пользователям доступ к гипертекстовым страницам по протоколу HTTP.

На рис. 2.16 еще раз дается краткое перечисление обязанностей каждого из уровней модели Интернет.

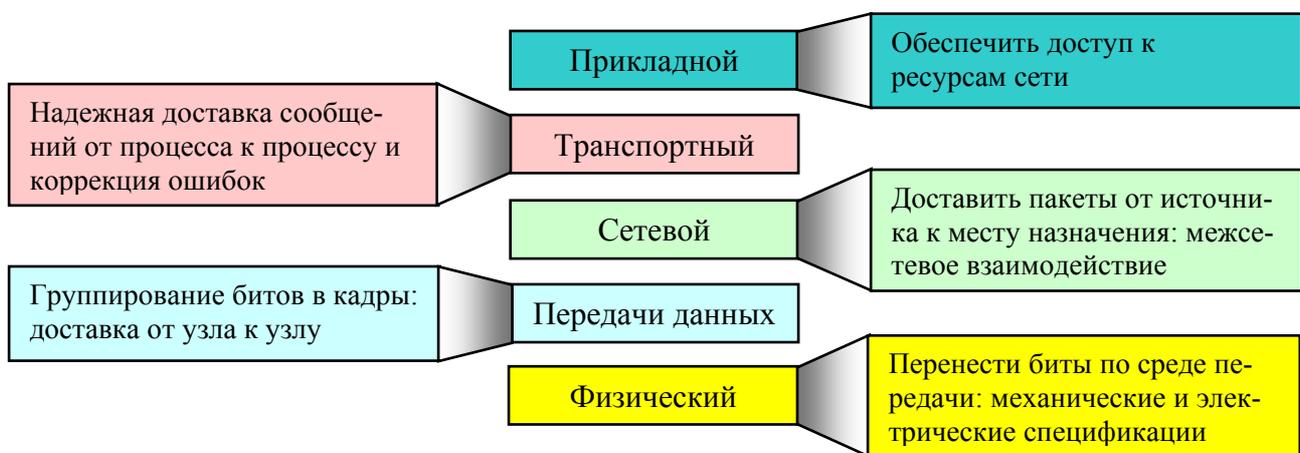


Рис. 2.16. Обязанности уровней Интернет

2.3. Эталонная модель OSI

Эталонная модель *OSI* называется *моделью взаимодействия открытых систем (Open System Interconnection, OSI)*, которая была разработана Международной организацией по стандартизации (*International Organization for Standardization, ISO*) и представлена в конце 70-х годов. *Открытая система* характеризуется набором протоколов, который позволяет взаимодействовать двум различным системам вне зависимости от их архитектуры. Модель *OSI* определяет различные уровни взаимодействия систем, дает им стандартные имена и указывает, какие функции должен выполнять каждый уровень. Основная цель модели *OSI* – показать, как организовать связь между различными системами без изменений в их аппаратном и программном обеспечении.



Рис. 2.17. Уровни в модели OSI

Полное описание этой модели занимает более 1000 страниц текста. До 1990-х годов считалось, что модель *OSI* станет окончательным стандартом передачи данных, однако этого не случилось, и она никогда не была реализована полностью. Не смотря на то, что протоколы, связанные с моделью *OSI*, используются очень редко, сама модель является актуальной и свойства ее уровней очень важны с теоретической точки зрения. Эталонная модель Интернет, основанная на стеке протоколов *TCP/IP*, наоборот, почти не используется, но ее протоколы получили очень широкое распространение. На рис 2.17 показаны уровни модели *OSI*.

Первые четыре уровня в модели *OSI* практически полностью соответствуют уровням в модели *Интернет*. Модель *OSI* имеет два дополнительных уровня: сеансовый уровень и уровень представлений.

Сеансовый уровень (Session Layer)

Сеансовый уровень устанавливает, поддерживает и синхронизирует сеансы связи (взаимодействие) между конечными компьютерами. Здесь обеспечивается управление диалогом, т.е. фиксируется, какая из сторон является активной (отслеживание очередности передачи данных) и предоставляются средства синхронизации, где внутри длинных сообщений вставляются служебные метки. Они позволяют в случае отказа вернуться назад к последней метке и продолжить передачу не сначала, а с того места, на котором она оборвалась.

Рис. 2.18. иллюстрирует отношение сеансового уровня к уровню представлений и транспортному уровню в модели *OSI*.

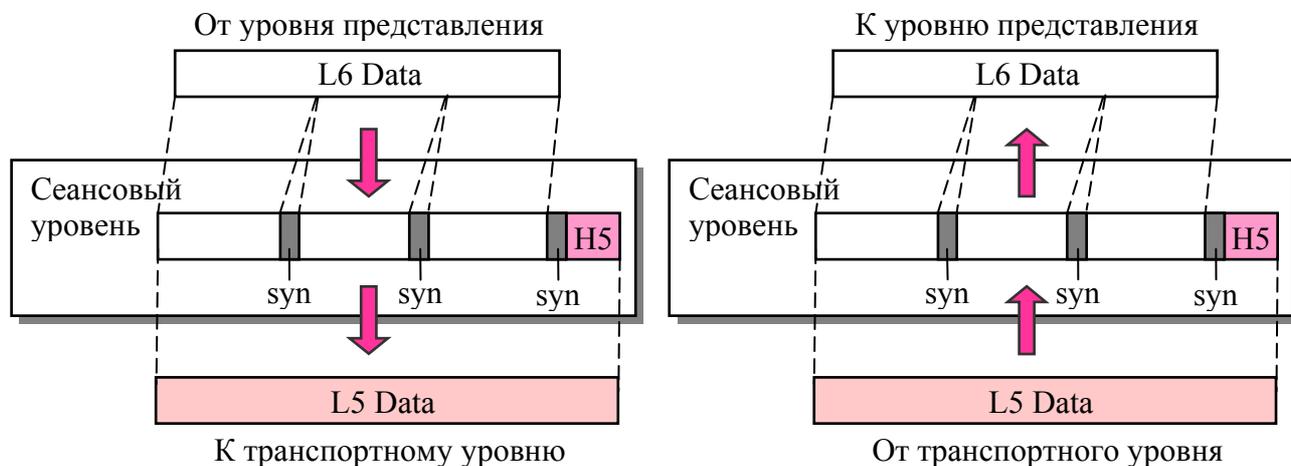


Рис. 2.18. Сеансовый уровень

Сеансовый уровень отвечает за обеспечение сеанса связи между конечными компьютерами

Ниже перечислены основные функции сеансового уровня:

- **Управление диалогом.** Сеансовый уровень позволяет двум системам войти в диалог. Он обеспечивает связь между двумя процессами в полудуплексном либо дуплексном режиме. Например, обмен информацией терминала с сервером в полудуплексном режиме.

- **Синхронизация.** Сеансовый уровень позволяет процессу вставлять в поток данных контрольные точки синхронизации (служебные метки). Например, если необходимо отправить большой по объему файл, содержащий 2000 кбайт, то разумно периодически вставлять метки (допустим через каждые 100 кбайт), обеспечивая независимость приема каждого блока объемом 100 кбайт. В случае нарушения связи, например, в точке 523 кбайт, то повторно необходимо передать лишь с 501 по 523 кбайт, т.к. предыдущие 500 блоков уже считаются приняты.

Уровень представления (*Presentation Layer*)

Представительный уровень имеет дело с формой представления передаваемой по сети информации, не меняя при этом ее содержания. Информация, передаваемая прикладным уровнем одной системы, всегда понятна прикладному уровню другой системы. В отличие от более низких уровней, задача которых – достоверная передача битов и байтов, уровень представления занимается синтаксисом и семантикой передаваемой информации. С помощью средств данного уровня протоколы прикладных уровней могут преодолеть синтаксические различия в представлении данных или же различия в кодах символов. Различные форматы данных преобразуются в некий стандартизованный вид для передачи по сети. На этом уровне может выполняться шифрование и дешифрование данных, благодаря которому секретность обмена данными обеспечивается сразу для всех прикладных служб.

Рис. 2.19. иллюстрирует отношение уровня представления к прикладному и сеансовому уровням в модели *OSI*.

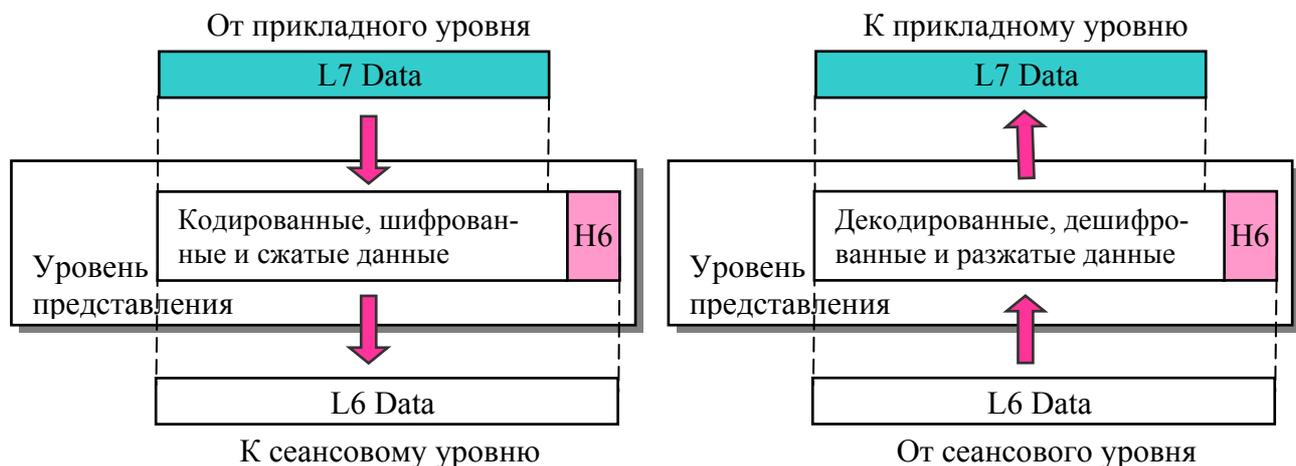


Рис. 2.19. Сеансовый уровень

Уровень представления отвечает за синтаксис и семантику передаваемой информации

Ниже перечислены основные функции уровня представления:

- **Трансляция (кодирование).** Процессы (активные программы) в двух системах обычно обмениваются информацией в форме символьных

строк, чисел и т.д. Если компьютеры используют разные системы кодирования (представления) информации, то уровень представления в этом случае отвечает за взаимопонимание между этими системами кодирования. Уровень представления на стороне отправителя изменяет информацию с собственного формата в некий общий формат. На приемной стороне происходит обратное преобразование общего формата представления информации в формат принимаемого процесса.

- **Шифрование.** Для передачи важной информации, система должна обеспечить ее конфиденциальность. Шифрование означает, что отправитель, перед тем как передать исходную информацию по сети, трансформирует в другую форму, при которой не возможно выявить смысловое содержание без знания специальных ключей, известных лишь отправителю и получателю. Дешифрование производит обратное преобразование принятого сообщения в исходную форму.
- **Компрессия.** Сжатие данных позволяет уменьшить количество бит, содержащихся в сообщении при допустимом уровне искажений. Сжатие информации имеет особенно важное значение при передаче мультимедиа информации (текст, изображения, аудио и видео)

На практике немногие приложения используют сеансовый уровень и уровень представлений, и они редко реализуются в виде отдельных протоколов, хотя функции этих уровней часто объединяют с функциями прикладного уровня и реализуют в одном протоколе. Примером использования шифрования сообщений является протокол Secure Socket Layer (**SSL**), который обеспечивает секретный обмен сообщениями для протоколов прикладного уровня стека TCP/IP.

2.4. Сравнение моделей Интернет и OSI

Стек протоколов TCP/IP был разработан до появления модели OSI. Следовательно уровни стека TCP/IP не точно соответствуют уровням модели OSI. Первоначально TCP/IP имел 4 уровня: **хост-сеть** (host-to-network), **интернет** (internet), **транспорт** (transport) и **приложение** (application). Однако при сравнении моделей Интернет и OSI становится ясно, что уровень «**хост-сеть**» эквивалентен комбинации физического и канального уровней, уровень «**интернет**» соответствует сетевому уровню, «**транспорт**» – транспортному, а уровень «**приложение**» преимущественно выполняет функции сеансового, представительного и прикладного уровней модели OSI. Функции сеансового уровня модели OSI частично выполняет транспортный уровень стека TCP/IP.

Следовательно, в последние годы принято считать, что модель Интернет, основанная на стеке TCP/IP, состоит из 5-ти уровней: **физический**, **канальный**, **сетевой**, **транспортный** и **прикладной**. Первые четыре уровня модели Интернет соответствуют первым четырем уровням модели OSI и три верхних уровня модели

OSI представлены одним прикладным уровнем модели Интернет. На рис. 2.20 показано соответствие модели *OSI* протоколам модели Интернет.

Уровни модели Интернет выстроены иерархически и содержат относительно независимые протоколы. Каждый из протоколов поддерживается одним или более протоколами соседнего нижнего уровня.

На транспортном уровне стек TCP/IP определяет три протокола: TCP (Transmission Control Protocol), UDP (User Datagram Protocol) и SCTP (Stream Control Transmission Protocol). На сетевом уровне основным протоколом является IP (Internetworking Protocol), а также ряд вспомогательных протоколов.

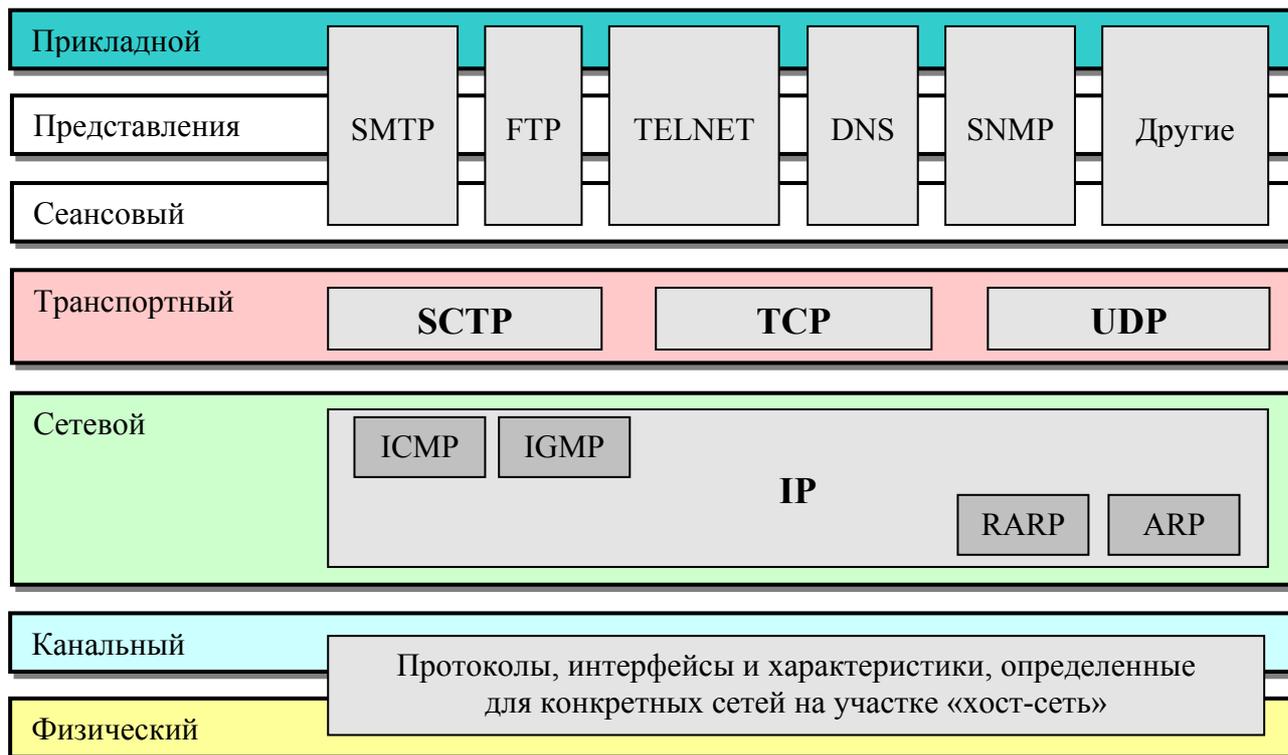


Рис. 2.20. Соответствие модели *OSI* протоколам модели Интернет

Протоколы физического и канального уровня

Стек протоколов TCP/IP модели Интернет не определяет конкретные протоколы. Он поддерживает все стандартизованные протоколы на участке «хост-сеть».

Протоколы сетевого уровня

Сетевой уровень поддерживает протокол межсетевого взаимодействия IP, который, в свою очередь, использует четыре вспомогательных протокола: ARP, RARP, ICMP и IGMP.

Протокол межсетевого взаимодействия IP

IP – это ненадежный протокол передачи без установления соединения с негарантированной доставкой (*best-effort delivery service*). Режим негарантиро-

ванной доставки означает, что IP не осуществляет обнаружение и контроль ошибок. Предполагается ненадежность передачи на нижележащих уровнях и прилагаются «максимальные усилия» для доставки данных вплоть до получателя, но без каких либо гарантий. IP обеспечивает передачу данных в пакетах, называемых дейтаграммами, каждый из которых доставляется независимо. Пакеты могут доставляться различными маршрутами и достигать получателя в другой очередности или с дублированием. Протокол IP не отслеживает маршруты доставки пакетов и не имеет средств для их сортировки при достижении получателя. На каждом участке следования пакета сетевые ресурсы выделяются по возможности, т.е. только те, которые в данный момент свободны.

Протокол преобразования адресов (ARP)

Протокол ARP (Address Resolution Protocol) используется для сопоставления логического адреса с физическим. Например, в локальной сети каждый узел идентифицируется уникальным физическим адресом, который прошивается производителем в сетевой карте этого узла. ARP используется для определения физического адреса узла по известному логическому (сетевому) IP адресу.

Протокол обратного преобразования адресов (RARP)

Протокол RARP (Reverse Address Resolution Protocol) позволяет хосту определить его сетевой IP адрес по известному физическому адресу. Он используется при первом соединении компьютера с сетью.

Протокол управляющих сообщений Интернет (ICMP)

Протокол ICMP (Internet Control Message Protocol) используется хостами и шлюзами для передачи обратно отправителю управляющих пакетов с извещениями о различных проблемах (сообщения запросов и об ошибках).

Протокол групповых сообщений Интернет (IGMP)

Протокол IGMP (Internet Group Message Protocol) используется для реализации алгоритмов одновременной передачи сообщения группе получателей.

Протоколы транспортного уровня

Традиционно и до последнего времени транспортный уровень стека TCP/IP представлялся двумя протоколами: TCP и UDP. IP является протоколом, доставляющий пакеты «от источника к получателю» (host-to-host). UDP и TCP являются протоколами транспортного уровня и отвечают за доставку всего сообщения «процесса к процессу». Новый протокол транспортного уровня, SCTP, разработан для более эффективной поддержки новых приложений.

Протокол дейтаграмм пользователя UDP

Протокол UDP (User Datagram Protocol) является простейшим протоколом транспортного уровня. Это протокол доставки сообщений «от процесса к процессу», который лишь добавляет в заголовке адрес порта, избыточные биты проверки ошибок и значение длины поля информационных данных.

Протокол управления передачей TCP

Протокол TCP (Transmission Control Protocol) предоставляет транспортные услуги для прикладного уровня и является надежным протоколом транспортировки потока сообщений. Этот протокол использует режим с установле-

нием соединения, т.е. перед передачей данных между сторонами предварительно должно быть установлено соединение.

На стороне отправителя для каждой передачи протокол TCP делит поток данных (сообщение) на более мелкие части – сегменты. Каждый сегмент содержит в заголовке порядковый номер для сортировки на приемной стороне вместе с номером подтверждения для принятых сегментов. Каждый сегмент инкапсулируется в отдельный пакет протокола IP. На приемной стороне протокол TCP производит сортировку принятых сегментов на основе порядковых номеров и их сборку в единое сообщение.

Протокол управления потоковой передачей SCTP

Протокол SCTP (Stream Control Transmission Protocol) обеспечивает эффективную поддержку новых приложений, например, «Voice over IP». Протокол SCTP объединяет в себе лучшие свойства протоколов UDP и TCP.

Протоколы прикладного уровня

Прикладной уровень в стеке TCP/IP эквивалентен сессионному, представительскому и прикладному уровням модели OSI. На этом уровне определены множество различных протоколов.

2.5. Адресация

Пример 1

На рис. 2.21 узел с физическим адресом 10 посылает кадр к узлу с физическим адресом 87. На уровне передачи данных заголовок этого кадра содержит физический адрес и другую необходимую информацию. Прицепляемые добавочные биты уровня передачи данных T2 (trailer) используются для обнаружения и коррекции ошибок.



Рис. 2.21. Пример доставки на уровне передачи данных

Пример 2

Узел с сетевым адресом A и физическим адресом 10, размещенный в одной локальной сети LAN, посылает данные к узлу с сетевым адресом P и фи-

зическим адресом 95, размещенному в другой локальной сети (рис. 2.22). Так как эти два узла принадлежат разным сетям, то одного физического адреса недостаточно, так как он имеет лишь локальную юрисдикцию. В этом случае необходимо иметь еще и сетевой (логический) адрес, который обеспечивает адресацию сетей. Пакет на сетевом уровне содержит логические адреса, которые остаются одинаковыми от источника до получателя (A и P , соответственно). При переходе пакета от одной сети к другой, логические адреса остаются постоянными, однако физические адреса меняются. Сети взаимодействуют друг с другом через маршрутизаторы R , на каждом из которых в заголовках пакетов меняются физические адреса соответствующих маршрутизаторов (межсетевых промежуточных узлов).

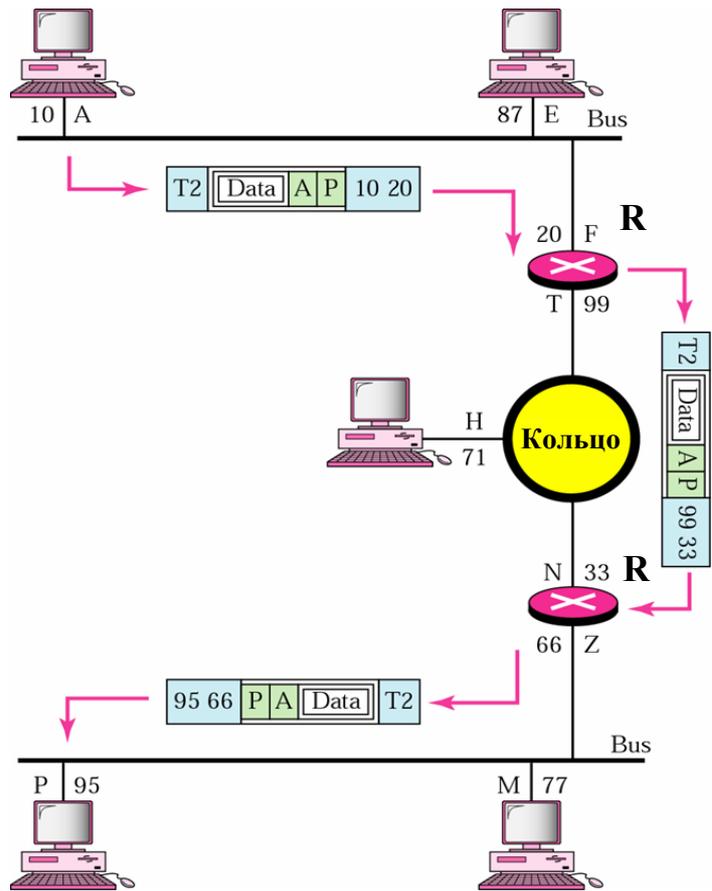


Рис. 2.22. Пример доставки пакета на сетевом уровне

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

Пример 3

Ни рис. 2.23 показан пример взаимодействия на транспортном уровне. Данные, поступающие с верхнего (прикладного) уровня, имеют адреса процессов (портов) j и k (j – адрес процесса отправителя и k – адрес процесса получателя). Так как объем одного блока данных (одного сообщения) обычно превосходит тот объем, который может обработать сетевой уровень, то на транспортном уровне производится деление (сегментация) всего сообщения на отдельные пакеты (два пакета на рис. 2.23). При этом каждый из пакетов снабжается адресами процессов отправителя и получателя (j и k , соответственно). Затем на сетевом уровне каждый пакет дополнительно снабжается сетевыми адресами (A и P). Пакеты могут проходить через различные пути и достигать место назначения в нарушенном порядке. На приемном конце два пакета доставляются к транспортному уровню, который ответственен за удаление заголовков транспортного уровня и сборку двух частей блока данных (сообщения) для доставки к прикладному уровню.

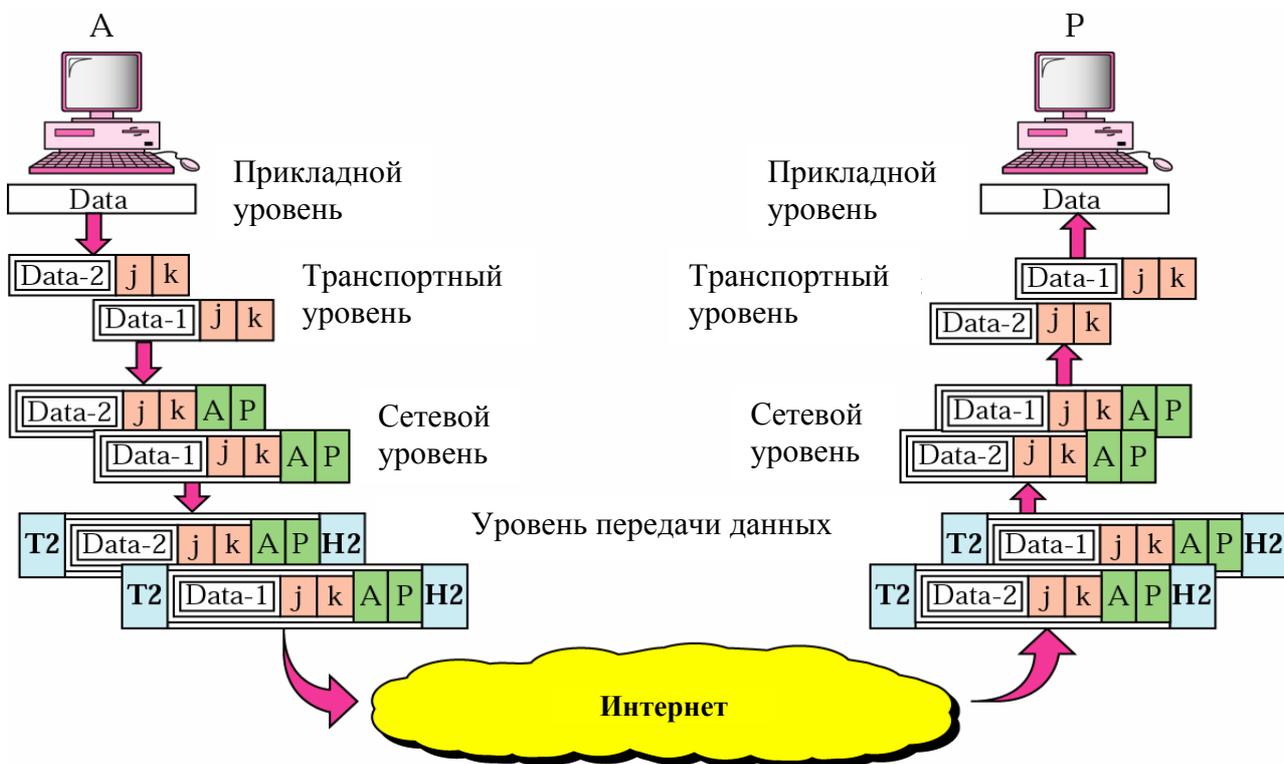


Рис. 2.16. Пример доставки сообщения на транспортном уровне

Литература

1. Behrouz A. Forouzan, Data Communications and Networking, 3rd Ed., 2004, 973 pp.
2. Behrouz A. Forouzan, Data Communications and Networking, 4rd Ed., 2007, 1134 pp.
3. William Stallings, Data and Computer Communications, 7th Ed., 2003, 864 pp.
4. Олифер В.Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, протоколы: Учебник для вузов. 3-е изд., СПб.: Питер, 2007. - 958 с.
5. Компьютерные сети. 4-е изд. / Таненбаум. - СПб.: Питер, 2005. - 992 с.
6. Столлингс В. Передача данных. 4-е изд. - СПб.: Питер, 2004. - 750 с.
7. Столлингс В. Компьютерные системы передачи данных. 6-е изд.:Пер. с англ. - М.: Издательский дом "Вильямс", 2002. - 928 с.
8. Столлингс В. Компьютерные сети, протоколы и технологии Интернета. - СПб.: БХВ-Петербург, 2005. - 832 с.
9. Современные компьютерные сети. 2-е изд. / В. Столлингс. - СПб.: Питер, 2003. - 783 с.
10. Куроуз Дж., Росс К. Компьютерные сети. 2-е изд. - СПб.: Питер, 2004. - 765 с.