

Международная Организация по Стандартам (International Standards Organization, ISO) разработала модель, которая четко определяет различные уровни взаимодействия систем, дает им стандартные имена и указывает, какую работу должен делать каждый уровень. Эта модель называется моделью взаимодействия открытых систем (Open System Interconnection, OSI) или моделью ISO/OSI.

В модели OSI взаимодействие делится на семь уровней или слоев (рис. 1.1). Каждый уровень имеет дело с одним определенным аспектом взаимодействия. Таким образом, проблема взаимодействия декомпозирована на 7 частных проблем, каждая из которых может быть решена независимо от других. Каждый уровень поддерживает интерфейсы с выше- и нижележащими уровнями.

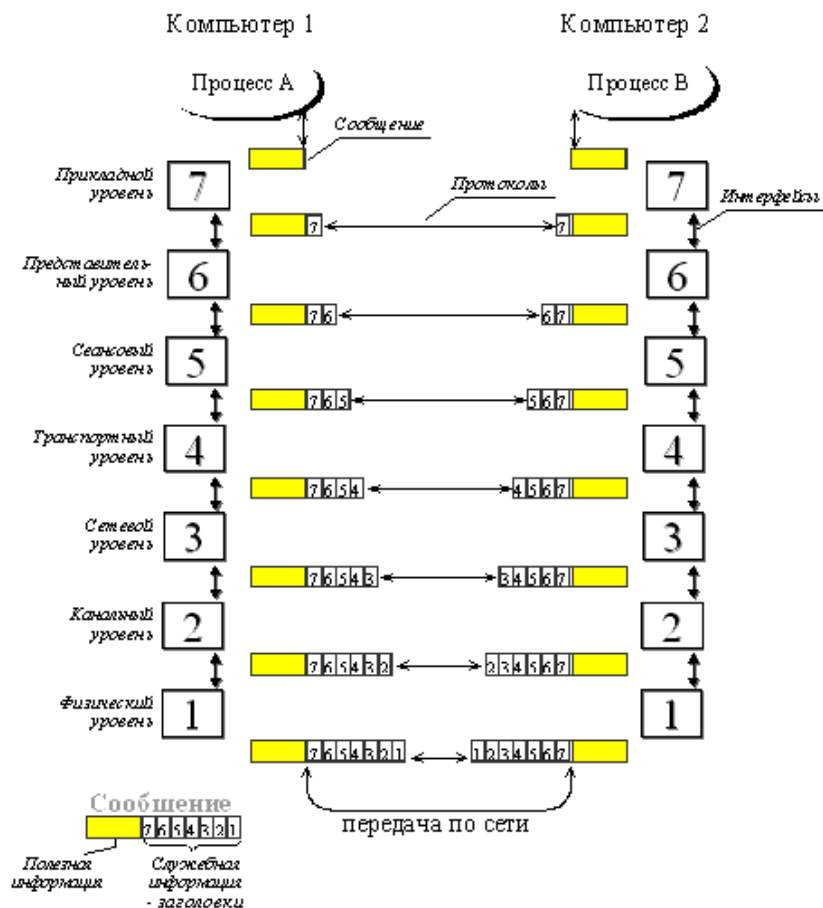


Рис. 1.1. Модель взаимодействия открытых систем ISO/OSI

Модель OSI описывает только системные средства взаимодействия, не касаясь приложений конечных пользователей. Приложения реализуют свои собственные протоколы взаимодействия, обращаясь к системным средствам. Следует иметь в виду, что приложение может взять на себя функции некоторых верхних уровней модели OSI, в таком случае, при необходимости межсетевых обменов оно обращается напрямую к системным средствам, выполняющим функции оставшихся нижних уровней модели OSI.

Приложение конечного пользователя может использовать системные средства взаимодействия не только для организации диалога с другим приложением, выполняющимся на другой машине, но и просто для получения услуг того или иного сетевого сервиса, например, доступа к удаленным файлам, получение почты или печати на разделяемом принтере.

Итак, пусть приложение обращается с запросом к прикладному уровню, например к файловому сервису. На основании этого запроса программное обеспечение прикладного уровня формирует сообщение стандартного формата, в которое помещает служебную информацию (заголовок) и, возможно, передаваемые данные. Затем это сообщение направляется представителю уровня. Представительный уровень добавляет к сообщению свой заголовок и передает результат вниз сеансовому уровню, который в свою очередь добавляет свой заголовок и т.д. Некоторые реализации протоколов предусматривают наличие в сообщении не только заголовка, но и концевика. Наконец, сообщение достигает самого низкого, физического уровня, который действительно передает его по линиям связи.

Когда сообщение по сети поступает на другую машину, оно последовательно перемещается вверх с уровня на уровень. Каждый уровень анализирует, обрабатывает и удаляет заголовок своего уровня, выполняет соответствующие данному уровню функции и передает сообщение вышележащему уровню.

Кроме термина "сообщение" (message) существуют и другие названия, используемые сетевыми специалистами для обозначения единицы обмена данными. В стандартах ISO для протоколов любого уровня используется такой термин как "протокольный блок данных" - Protocol Data Unit (PDU). Кроме этого, часто используются названия кадр (frame), пакет (packet), дейтаграмма (datagram).

Функции уровней модели ISO/OSI

Физический уровень. Этот уровень имеет дело с передачей битов по физическим каналам, таким, например, как коаксиальный кабель, витая пара или оптоволоконный кабель. 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, такие как требования к фронтам импульсов, уровням напряжения или тока передаваемого сигнала, тип кодирования, скорость передачи сигналов. Кроме этого, здесь стандартизируются типы разъемов и назначение каждого контакта.

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

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

Канальный уровень. На физическом уровне просто пересылаются биты. При этом не учитывается, что в некоторых сетях, в которых линии связи используются (разделяются) попеременно несколькими парами взаимодействующих компьютеров, физическая среда передачи может быть занята. Поэтому одной из задач канального уровня является проверка доступности среды передачи. Другой задачей канального уровня является реализация механизмов обнаружения и коррекции ошибок. Для этого на канальном уровне биты группируются в наборы, называемые кадрами (frames). Канальный уровень обеспечивает корректность передачи каждого кадра, помещая специальную последовательность бит в начало и конец каждого кадра, чтобы отметить его, а также вычисляет контрольную сумму, суммируя все байты кадра определенным способом и добавляя контрольную сумму к кадру. Когда кадр приходит, получатель снова вычисляет контрольную сумму полученных данных и сравнивает результат с контрольной суммой из кадра. Если они совпадают, кадр считается правильным и принимается. Если же контрольные суммы не совпадают, то фиксируется ошибка.

В протоколах канального уровня, используемых в локальных сетях, заложена определенная структура связей между компьютерами и способы их адресации. Хотя канальный уровень и обеспечивает доставку кадра между любыми двумя узлами локальной сети, он это делает только в сети с совершенно определенной топологией связей, именно той топологией, для которой он был разработан. К таким типовым топологиям, поддерживаемым протоколами канального уровня локальных сетей, относятся общая шина, кольцо и звезда. Примерами протоколов канального уровня являются протоколы Ethernet, Token Ring, FDDI, 10VG-AnyLAN.

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

В глобальных сетях, которые редко обладают регулярной топологией, канальный уровень обеспечивает обмен сообщениями между двумя соседними компьютерами, соединенными индивидуальной линией связи. Примерами протоколов "точка - точка" (как часто называют такие протоколы) могут служить широко распространенные протоколы PPP и LAP-B.

Сетевой уровень. Этот уровень служит для образования единой транспортной системы, объединяющей несколько сетей с различными принципами передачи информации между конечными узлами. Рассмотрим функции сетевого уровня на примере локальных сетей. Протокол канального уровня локальных сетей обеспечивает доставку данных между любыми узлами только в сети с соответствующей *типовой топологией*. Это очень жесткое ограничение, которое не позволяет строить сети с развитой структурой, например, сети, объединяющие несколько сетей предприятия в единую сеть, или высоконадежные сети, в которых существуют избыточные связи между узлами. Для того, чтобы, с одной стороны, сохранить простоту процедур передачи данных для типовых топологий, а с другой стороны, допустить использование произвольных топологий, используется дополнительный сетевой уровень. На этом уровне вводится понятие "сеть". В данном случае под сетью понимается совокупность компьютеров, соединенных между собой в соответствии с одной из стандартных типовых топологий и использующих для передачи данных один из протоколов канального уровня, определенный для этой топологии.

Таким образом, внутри сети доставка данных регулируется канальным уровнем, а вот доставкой данных между сетями занимается сетевой уровень.

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 номера сети и номера компьютера в этой сети.

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. Для того, чтобы передать сообщение от отправителя, находящегося в одной сети, получателю, находящемуся в другой сети, нужно совершить некоторое количество транзитных передач (*hops*) между сетями, каждый раз выбирая подходящий маршрут. Таким образом, маршрут представляет собой последовательность маршрутизаторов, через которые проходит пакет.

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

На сетевом уровне определяется два вида протоколов. Первый вид относится к определению правил передачи пакетов с данными конечных узлов от узла к маршрутизатору и между маршрутизаторами. Именно эти протоколы обычно имеют в виду, когда говорят о протоколах сетевого уровня. К сетевому уровню относят и другой вид протоколов, называемых *протоколами обмена маршрутной информацией*. С помощью этих протоколов маршрутизаторы собирают информацию о топологии межсетевых соединений. Протоколы сетевого уровня реализуются программными модулями операционной системы, а также программными и аппаратными средствами маршрутизаторов.

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

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

Выбор класса сервиса транспортного уровня определяется, с одной стороны, тем, в какой степени задача обеспечения надежности решается самими приложениями и протоколами более высоких, чем транспортный, уровней, а с другой стороны, этот выбор зависит от того, насколько надежной является вся система транспортировки данных в сети. Так, например, если качество каналов передачи связи очень высокое, и вероятность возникновения ошибок, не обнаруженных протоколами более низких уровней, невелика, то разумно воспользоваться одним из облегченных сервисов транспортного уровня, не обремененных многочисленными проверками, квитированием и другими приемами повышения надежности. Если же транспортные средства изначально очень ненадежны, то целесообразно обратиться к наиболее развитому сервису транспортного уровня, который работает, используя максимум средств для обнаружения и устранения ошибок - с помощью предварительного установления логического соединения, контроля доставки сообщений с помощью контрольных сумм и циклической нумерации пакетов, установления тайм-аутов доставки и т.п.

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

Сеансовый уровень. Сеансовый уровень обеспечивает управление диалогом для того, чтобы фиксировать, какая из сторон является активной в настоящий момент, а также предоставляет средства синхронизации. Последние позволяют вставлять контрольные точки в длинные передачи, чтобы в случае отказа можно было вернуться назад к последней контрольной точке, вместо того, чтобы начинать все с начала. На практике немногие приложения используют сеансовый уровень, и он редко реализуется.

Уровень представления. Этот уровень обеспечивает гарантию того, что информация, передаваемая прикладным уровнем, будет понятна прикладному уровню в другой системе. При необходимости уровень представления выполняет преобразование форматов данных в некоторый общий формат представления, а на приеме, соответственно, выполняет обратное преобразование. Таким образом, прикладные уровни могут преодолеть, например, синтаксические различия в представлении данных. На этом уровне может выполняться шифрование и дешифрование данных, благодаря которому секретность обмена данными обеспечивается сразу для всех прикладных сервисов. Примером протокола, работающего на уровне представления, является протокол Secure Socket Layer (SSL), который обеспечивает секретный обмен сообщениями для протоколов прикладного уровня стека TCP/IP.

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

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

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

Характеристика популярных стеков коммуникационных протоколов

Итак, взаимодействие компьютеров в сетях происходит в соответствии с определенными правилами обмена сообщениями и их форматами, то есть в соответствии с определенными протоколами. Иерархически организованная совокупность протоколов, решающих задачу взаимодействия узлов сети, называется стеком коммуникационных протоколов.

Существует достаточно много стеков протоколов, широко применяемых в сетях. Это и стеки, являющиеся международными и национальными стандартами, и фирменные стеки, получившие распространение благодаря распространенности оборудования той или иной фирмы. Примерами популярных стеков протоколов могут служить стек IPX/SPX фирмы Novell, стек TCP/IP, используемый в сети Internet и во многих сетях на основе операционной системы UNIX, стек OSI международной организации по стандартизации, стек DECnet корпорации Digital Equipment и некоторые другие.

Использование в сети того или иного стека коммуникационных протоколов во многом определяет лицо сети и ее характеристики. В небольших сетях может использоваться исключительно один стек. В крупных корпоративных сетях, объединяющих различные сети, параллельно используются, как правило, несколько стеков.

В коммуникационном оборудовании реализуются протоколы нижних уровней, которые в большей степени стандартизованы, чем протоколы верхних уровней, и это является предпосылкой для успешной совместной работы оборудования различных производителей. Перечень протоколов, поддерживаемых тем или иным коммуникационным устройством, является одной из наиболее важных характеристик этого устройства.

Компьютеры реализуют коммуникационные протоколы в виде соответствующих программных элементов сетевой операционной системы, например, протоколы канального уровня, как правило, выполнены в виде драйверов сетевых адаптеров, а протоколы верхних уровней в виде серверных и клиентских компонент сетевых сервисов.

Умение хорошо работать в среде той или иной операционной системы является важной характеристикой коммуникационного оборудования. Часто можно прочитать в рекламе сетевого адаптера или концентратора, что он разрабатывался специально для работы в сети NetWare или UNIX. Это означает, что разработчики аппаратуры оптимизировали ее характеристики применительно к тем протоколам, которые используются в этой сетевой операционной системе, или к данной версии их реализации, если эти протоколы используются в различных ОС. Из-за особенностей реализации протоколов в различных ОС, в качестве одной из характеристик коммуникационного оборудования используется его сертифицированность на возможность работы в среде данной ОС.

На нижних уровнях - физическом и канальном - практически во всех стеках используются одни и те же протоколы. Это хорошо стандартизованные протоколы Ethernet, Token Ring, FDDI и некоторые другие, которые позволяют использовать во всех сетях одну и ту же аппаратуру.

Протоколы сетевого и более высоких уровней существующих стандартных стеков отличаются большим разнообразием и, как правило, не соответствуют рекомендуемой модели ISO разбиению на уровни. В частности, в этих стеках функции сеансового и представительного уровня чаще всего объединены с прикладным уровнем. Такое несоответствие связано с тем, что модель ISO появилась как результат обобщения уже существующих и реально используемых стеков, а не наоборот.

Стек OSI

Следует различать стек протоколов OSI и модель OSI. В то время, как модель OSI концептуально определяет процедуру взаимодействия открытых систем, декомпозируя задачу на 7 уровней, стандартизирует назначение каждого уровня и вводит стандартные названия уровней, стек OSI - это набор вполне конкретных спецификаций протоколов, образующих согласованный стек протоколов. Этот стек протоколов поддерживает правительство США в своей программе GOSIP. Все компьютерные сети, устанавливаемые в правительственных учреждениях после 1990 года, должны либо непосредственно поддерживать стек OSI, либо обеспечивать средства для перехода на этот стек в будущем. Тем не менее, стек OSI более популярен в Европе, а не в США, так как в Европе меньше установлено старых сетей, использующих свои собственные протоколы. В Европе также ощущается большая потребность в общем стеке, так как здесь имеется большое количество разных стран.

Это международный, независимый от производителей стандарт. Он может обеспечить взаимодействие между корпорациями, партнерами и поставщиками. Это взаимодействие осложняется из-за проблем с адресацией, именованием и безопасностью данных. Все эти проблемы в стеке OSI частично решены. Протоколы OSI

требуют больших затрат вычислительной мощности центрального процессора, что делает их более подходящими для мощных машин, а не для сетей персональных компьютеров. Большинство организаций пока только планируют переход к стеку OSI. Из тех, кто работает в этом направлении, можно назвать Военно-морское ведомство США и сеть NFSNET. Одним из крупнейших производителей, поддерживающих OSI, является компания AT&T. Ее сеть Stargroup полностью базируется на стеке OSI.

По вполне очевидным причинам стек OSI в отличие от других стандартных стеков полностью соответствует модели взаимодействия OSI, он включает спецификации для всех семи уровней модели взаимодействия открытых систем (рис. 1.3).

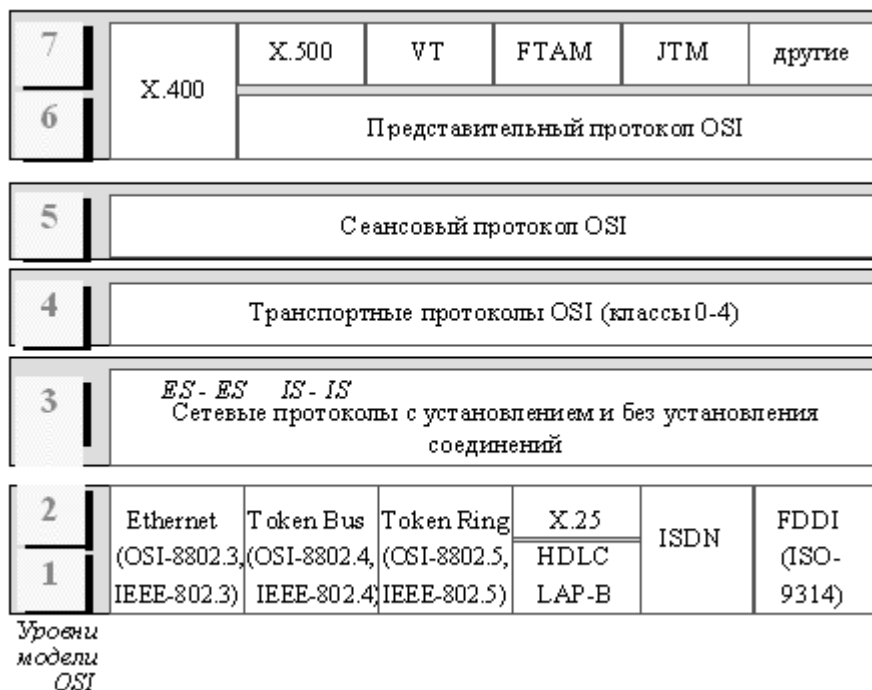


Рис. 1.3. Стек OSI

На **физическом и канальном уровнях** стек OSI поддерживает протоколы Ethernet, Token Ring, FDDI, а также протоколы LLC, X.25 и ISDN. Эти протоколы будут подробно обсуждены в других разделах пособия.

Сервисы **сетевого, транспортного и сеансового уровней** также имеются в стеке OSI, однако они мало распространены. На сетевом уровне реализованы протоколы как без установления соединений, так и с установлением соединений. Транспортный протокол стека OSI в соответствии с функциями, определенными для него в модели OSI, скрывает различия между сетевыми сервисами с установлением соединения и без установления соединения, так что пользователи получают нужное качество обслуживания независимо от нижележащего сетевого уровня. Чтобы обеспечить это, транспортный уровень требует, чтобы пользователь задал нужное качество обслуживания. Определены 5 классов транспортного сервиса, от низшего класса 0 до высшего класса 4, которые отличаются степенью устойчивости к ошибкам и требованиями к восстановлению данных после ошибок.

Сервисы **прикладного уровня** включают передачу файлов, эмуляцию терминала, службу каталогов и почту. Из них наиболее перспективными являются служба каталогов (стандарт X.500), электронная почта (X.400), протокол виртуального терминала (VT), протокол передачи, доступа и управления файлами (FTAM), протокол пересылки и управления работами (JTM). В последнее время ISO сконцентрировала свои усилия именно на сервисах верхнего уровня.

X.400 - это семейство рекомендаций Международного консультативного комитета по телеграфии и телефонии (CCITT), в которых описываются системы пересылки электронных сообщений. На сегодняшний день рекомендации X.400 являются наиболее популярным протоколом обмена сообщениями. Рекомендации X.400 описывают модель системы обмена сообщениями, протоколы взаимодействия между всеми компонентами этой системы, а также множество видов сообщений и возможности, которыми обладает отправитель по каждому виду отправляемых сообщений.

Рекомендации X.400 определяют следующий минимально необходимый набор услуг, предоставляемых пользователям: управление доступом, ведение уникальных системных идентификаторов сообщений, извещение о доставке или недоставке сообщения с указанием причины, индикация типа содержания

сообщения, индикация преобразования содержания сообщения, временные отметки при передаче и доставке, выбор категории доставки (срочная, несрочная, нормальная), многоадресная доставка, задержанная доставка (до определенного момента времени), преобразование содержимого для взаимодействия с несовместимыми почтовыми системами, например, со службами телексной и факсимильной связей, запрос о том, доставлено ли конкретное сообщение, списки рассылки, которые могут иметь вложенную структуру, средства защиты сообщений от несанкционированного доступа, базирующиеся на асимметричной криптосистеме публичных ключей.

Целью рекомендаций **X.500** является выработка стандартов глобальной справочной службы. Процесс доставки сообщения требует знания адреса получателя, что при больших размерах сетей представляет собой проблему, поэтому необходимо иметь справочную службу, помогающую получать адреса отправителей и получателей. В общем виде служба X.500 представляет собой распределенную базу данных имен и адресов. Все пользователи потенциально имеют право войти в эту базу данных, используя определенный набор атрибутов.

Над базой данных имен и адресов определены следующие операции:

- чтение - получение адреса по известному имени,
- запрос - получение имени по известным атрибутам адреса,
- модификация, включающая удаление и добавление записей в базе данных.

Основные проблемы реализации рекомендаций X.500 проистекают из масштабности этого проекта, претендующего на роль всемирной справочной службы. Поэтому программное обеспечение, реализующее рекомендации X.500, получается весьма громоздким и предъявляет высокие требования к производительности аппаратуры.

Учет рекомендаций X.400 и X.500 при проектировании систем электронной почты делает принципиально возможной и концептуально простой стыковку почтовых систем разных производителей.

Протокол **VT** решает проблему несовместимости различных протоколов эмуляции терминалов. Сейчас пользователю персонального компьютера, совместимого с IBM PC, для одновременной работы с компьютерами VAX, IBM 3090 и HP9000 нужно приобрести три различные программы для эмуляции терминалов различных типов и использующих разные протоколы. Если бы каждый хост-компьютер имел бы в своем составе программное обеспечение протокола эмуляции терминала ISO, то и пользователю бы понадобилась только одна программа, поддерживающая протокол VT. В своем стандарте ISO аккумулировала широко распространенные функции эмуляции терминалов.

Передача файлов - это наиболее распространенный компьютерный сервис. Доступ к файлам, как к локальным, так и к удаленным, нужен всем приложениям - текстовым редакторам, электронной почте, базам данных или программам удаленного запуска. ISO предусматривает такой сервис в протоколе **FTAM**. Наряду со стандартом X.400, это наиболее популярный стандарт стека OSI. FTAM предусматривает средства для локализации и доступа к содержимому файла и включает набор директив для вставки, замены, расширения и очистки содержимого файла. FTAM также предусматривает средства для манипулирования файлом как единым целым, включая создание, удаление, чтение, открытие, закрытие файла и выбор его атрибутов.

Протокол пересылки и управления работами **JTM** позволяет пользователям пересылать работы, которые должны быть выполнены на хост-компьютере. Язык управления заданиями, который обеспечивает передачу работ, указывает хост-компьютеру, какие действия и с какими программами и файлами должны быть выполнены. Протокол JTM поддерживает традиционную пакетную обработку, обработку транзакций, ввод удаленных заданий и доступ к распределенным базам данных.

Стек TCP/IP

Стек TCP/IP, называемый также стеком DoD и стеком Internet, является одним из наиболее популярных и перспективных стеков коммуникационных протоколов. Если в настоящее время он распространен в основном в сетях с ОС UNIX, то реализация его в последних версиях сетевых операционных систем для персональных компьютеров (Windows NT, NetWare) является хорошей предпосылкой для быстрого роста числа установок стека TCP/IP.

Стек был разработан по инициативе Министерства обороны США (Department of Defence, DoD) более 20 лет назад для связи экспериментальной сети ARPAnet с другими спутниковыми сетями как набор общих протоколов для разнородной вычислительной среды. Сеть ARPA поддерживала разработчиков и исследователей в военных областях. В сети ARPA связь между двумя компьютерами осуществлялась с

использованием протокола Internet Protocol (IP), который и по сей день является одним из основных в стеке TCP/IP и фигурирует в названии стека.

Большой вклад в развитие стека TCP/IP внес университет Беркли, реализовав протоколы стека в своей версии ОС UNIX. Широкое распространение ОС UNIX привело и к широкому распространению протокола IP и других протоколов стека. На этом же стеке работает всемирная информационная сеть Internet, чье подразделение Internet Engineering Task Force (IETF) вносит основной вклад в совершенствование стандартов стека, публикуемых в форме спецификаций RFC.

Так как стек TCP/IP был разработан до появления модели взаимодействия открытых систем ISO/OSI, то, хотя он также имеет многоуровневую структуру, соответствие уровней стека TCP/IP уровням модели OSI достаточно условно.

Структура протоколов TCP/IP приведена на рисунке 1.4. Протоколы TCP/IP делятся на 4 уровня.

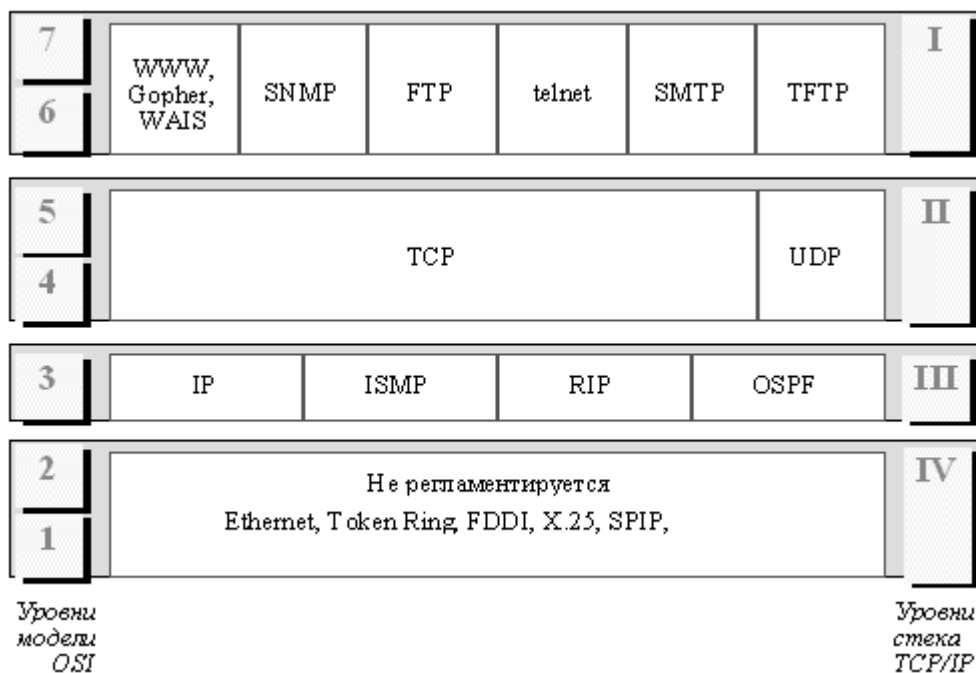


Рис. 1.4. Стек TCP / IP

Самый нижний (**уровень IV**) - уровень межсетевых интерфейсов - соответствует физическому и каналному уровням модели OSI. Этот уровень в протоколах TCP/IP не регламентируется, но поддерживает все популярные стандарты физического и канального уровня: для локальных каналов это Ethernet, Token Ring, FDDI, для глобальных каналов - собственные протоколы работы на аналоговых коммутируемых и выделенных линиях SLIP/PPP, которые устанавливают соединения типа "точка - точка" через последовательные каналы глобальных сетей, и протоколы территориальных сетей X.25 и ISDN. Разработана также специальная спецификация, определяющая использование технологии ATM в качестве транспорта канального уровня.

Следующий уровень (**уровень III**) - это уровень межсетевого взаимодействия, который занимается передачей дейтаграмм с использованием различных локальных сетей, территориальных сетей X.25, линий специальной связи и т. п. В качестве основного протокола сетевого уровня (в терминах модели OSI) в стеке используется протокол **IP**, который изначально проектировался как протокол передачи пакетов в составных сетях, состоящих из большого количества локальных сетей, объединенных как локальными, так и глобальными связями. Поэтому протокол IP хорошо работает в сетях со сложной топологией, рационально используя наличие в них подсистем и экономно расходуя пропускную способность низкоскоростных линий связи. Протокол IP является дейтаграммным протоколом.

К уровню межсетевого взаимодействия относятся и все протоколы, связанные с составлением и модификацией таблиц маршрутизации, такие как протоколы сбора маршрутной информации **RIP** (Routing Internet Protocol) и **OSPF** (Open Shortest Path First), а также протокол межсетевых управляющих сообщений **ICMP** (Internet Control Message Protocol). Последний протокол предназначен для обмена информацией об ошибках между маршрутизатором и шлюзом, системой-источником и системой-приемником, то есть для организации обратной связи. С помощью специальных пакетов ICMP сообщается о невозможности доставки

пакета, о превышении времени жизни или продолжительности сборки пакета из фрагментов, об аномальных величинах параметров, об изменении маршрута пересылки и типа обслуживания, о состоянии системы и т.п.

Следующий уровень (**уровень II**) называется основным. На этом уровне функционируют протокол управления передачей **TCP** (Transmission Control Protocol) и протокол дейтаграмм пользователя **UDP** (User Datagram Protocol). Протокол TCP обеспечивает устойчивое виртуальное соединение между удаленными прикладными процессами. Протокол UDP обеспечивает передачу прикладных пакетов дейтаграммным методом, то есть без установления виртуального соединения, и поэтому требует меньших накладных расходов, чем TCP.

Верхний уровень (**уровень I**) называется прикладным. За долгие годы использования в сетях различных стран и организаций стек TCP/IP накопил большое количество протоколов и сервисов прикладного уровня. К ним относятся такие широко используемые протоколы, как протокол копирования файлов FTP, протокол эмуляции терминала telnet, почтовый протокол SMTP, используемый в электронной почте сети Internet и ее российской ветви РЕЛКОМ, гипертекстовые сервисы доступа к удаленной информации, такие как WWW и многие другие. Остановимся несколько подробнее на некоторых из них, наиболее тесно связанных с тематикой данного курса.

Протокол **SNMP** (Simple Network Management Protocol) используется для организации сетевого управления. Проблема управления разделяется здесь на две задачи. Первая задача связана с передачей информации. Протоколы передачи управляющей информации определяют процедуру взаимодействия сервера с программой-клиентом, работающей на хосте администратора. Они определяют форматы сообщений, которыми обмениваются клиенты и серверы, а также форматы имен и адресов. Вторая задача связана с контролируемыми данными. Стандарты регламентируют, какие данные должны сохраняться и накапливаться в шлюзах, имена этих данных и синтаксис этих имен. В стандарте SNMP определена спецификация информационной базы данных управления сетью. Эта спецификация, известная как база данных MIB (Management Information Base), определяет те элементы данных, которые хост или шлюз должен сохранять, и допустимые операции над ними.

Протокол пересылки файлов **FTP** (File Transfer Protocol) реализует удаленный доступ к файлу. Для того, чтобы обеспечить надежную передачу, FTP использует в качестве транспорта протокол с установлением соединений - TCP. Кроме пересылки файлов протокол, FTP предлагает и другие услуги. Так пользователю предоставляется возможность интерактивной работы с удаленной машиной, например, он может распечатать содержимое ее каталогов, FTP позволяет пользователю указывать тип и формат запоминаемых данных. Наконец, FTP выполняет аутентификацию пользователей. Прежде, чем получить доступ к файлу, в соответствии с протоколом пользователи должны сообщить свое имя и пароль.

В стеке TCP/IP протокол FTP предлагает наиболее широкий набор услуг для работы с файлами, однако он является и самым сложным для программирования. Приложения, которым не требуются все возможности FTP, могут использовать другой, более экономичный протокол - простейший протокол пересылки файлов **TFTP** (Trivial File Transfer Protocol). Этот протокол реализует только передачу файлов, причем в качестве транспорта используется более простой, чем TCP, протокол без установления соединения - UDP.

Протокол **telnet** обеспечивает передачу потока байтов между процессами, а также между процессом и терминалом. Наиболее часто этот протокол используется для эмуляции терминала удаленной ЭВМ.

Стек IPX/SPX

Этот стек является оригинальным стеком протоколов фирмы Novell, который она разработала для своей сетевой операционной системы NetWare еще в начале 80-х годов. Протоколы Internetwork Packet Exchange (IPX) и Sequenced Packet Exchange (SPX), которые дали имя стеку, являются прямой адаптацией протоколов XNS фирмы Xerox, распространенных в гораздо меньшей степени, чем IPX/SPX. По количеству установок протоколы IPX/SPX лидируют, и это обусловлено тем, что сама ОС NetWare занимает лидирующее положение с долей установок в мировом масштабе примерно в 65%.

Семейство протоколов фирмы Novell и их соответствие модели ISO/OSI представлено на рисунке 1.5.

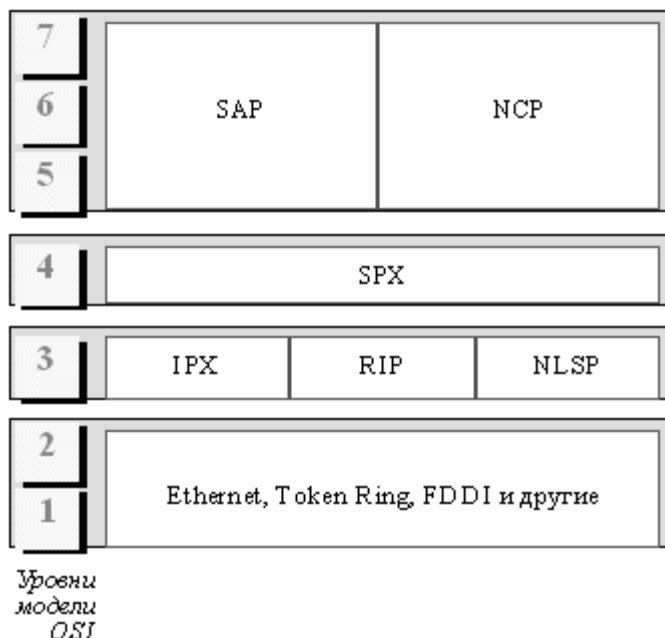


Рис. 1.5. Стек IPX / SPX

На **физическом и канальном уровнях** в сетях Novell используются все популярные протоколы этих уровней (Ethernet, Token Ring, FDDI и другие).

На **сетевом уровне** в стеке Novell работает протокол **IPX**, а также протоколы обмена маршрутной информацией **RIP** и **NLSP** (аналог протокола OSPF стека TCP/IP). IPX является протоколом, который занимается вопросами адресации и маршрутизации пакетов в сетях Novell. Маршрутные решения IPX основаны на адресных полях в заголовке его пакета, а также на информации, поступающей от протоколов обмена маршрутной информацией. Например, IPX использует информацию, поставляемую либо протоколом RIP, либо протоколом NLSP (NetWare Link State Protocol) для передачи пакетов компьютеру назначения или следующему маршрутизатору. Протокол IPX поддерживает только дейтаграммный способ обмена сообщениями, за счет чего экономно потребляет вычислительные ресурсы. Итак, протокол IPX обеспечивает выполнение трех функций: задание адреса, установление маршрута и рассылку дейтаграмм.

Транспортному уровню модели OSI в стеке Novell соответствует протокол SPX, который осуществляет передачу сообщений с установлением соединений.

На верхних **прикладном, представительном и сеансовом уровнях** работают протоколы NCP и SAP. Протокол **NCP** (NetWare Core Protocol) является протоколом взаимодействия сервера NetWare и оболочки рабочей станции. Этот протокол прикладного уровня реализует архитектуру клиент-сервер на верхних уровнях модели OSI. С помощью функций этого протокола рабочая станция производит подключение к серверу, отображает каталоги сервера на локальные буквы дисководов, просматривает файловую систему сервера, копирует удаленные файлы, изменяет их атрибуты и т.п., а также осуществляет разделение сетевого принтера между рабочими станциями.

SAP (Service Advertising Protocol) - протокол объявления о сервисе - концептуально подобен протоколу RIP. Подобно тому, как протокол RIP позволяет маршрутизаторам обмениваться маршрутной информацией, протокол SAP дает возможность сетевым устройствам обмениваться информацией об имеющихся сетевых сервисах.

Серверы и маршрутизаторы используют SAP для объявления о своих сервисных услугах и сетевых адресах. Протокол SAP позволяет сетевым устройствам постоянно корректировать данные о том, какие сервисные услуги имеются сейчас в сети. При старте серверы используют SAP для оповещения оставшейся части сети о своих услугах. Когда сервер завершает работу, то он использует SAP для того, чтобы известить сеть о прекращении действия своих услуг.

В сетях Novell серверы NetWare 3.x каждую минуту рассылают широковещательные пакеты SAP. Пакеты SAP в значительной степени засоряют сеть, поэтому одной из основных задач маршрутизаторов, выходящих на глобальные связи, является фильтрация трафика SAP-пакетов и RIP-пакетов.

Особенности стека IPX/SPX обусловлены особенностями ОС NetWare, а именно ориентацией ее ранних версий (до 4.0) на работу в локальных сетях небольших размеров, состоящих из персональных компьютеров со скромными ресурсами. Поэтому Novell нужны были протоколы, на реализацию которых требовалось минимальное количество оперативной памяти (ограниченной в IBM-совместимых компьютерах под управлением MS-DOS 640 Кбайтами) и которые бы быстро работали на процессорах небольшой вычислительной мощности. В результате, протоколы стека IPX/SPX до недавнего времени хорошо работали в локальных сетях и не очень - в больших корпоративных сетях, так как слишком перегружали медленные глобальные связи широкоэмитательными пакетами, которые интенсивно используются несколькими протоколами этого стека (например, для установления связи между клиентами и серверами).

Это обстоятельство, а также тот факт, что стек IPX/SPX является собственностью фирмы Novell и на его реализацию нужно получать у нее лицензию, долгое время ограничивали распространенность его только сетями NetWare. Однако к моменту выпуска версии NetWare 4.0, Novell внесла и продолжает вносить в свои протоколы серьезные изменения, направленные на приспособление их для работы в корпоративных сетях. Сейчас стек IPX/SPX реализован не только в NetWare, но и в нескольких других популярных сетевых ОС - SCO UNIX, Sun Solaris, Microsoft Windows NT.

Стек NetBIOS/SMB

Фирмы Microsoft и IBM совместно работали над сетевыми средствами для персональных компьютеров, поэтому стек протоколов NetBIOS/SMB является их совместным детищем. Средства NetBIOS появились в 1984 году как сетевое расширение стандартных функций базовой системы ввода/вывода (BIOS) IBM PC для сетевой программы PC Network фирмы IBM, которая на прикладном уровне (рис. 1.6) использовала для реализации сетевых сервисов протокол SMB (Server Message Block).

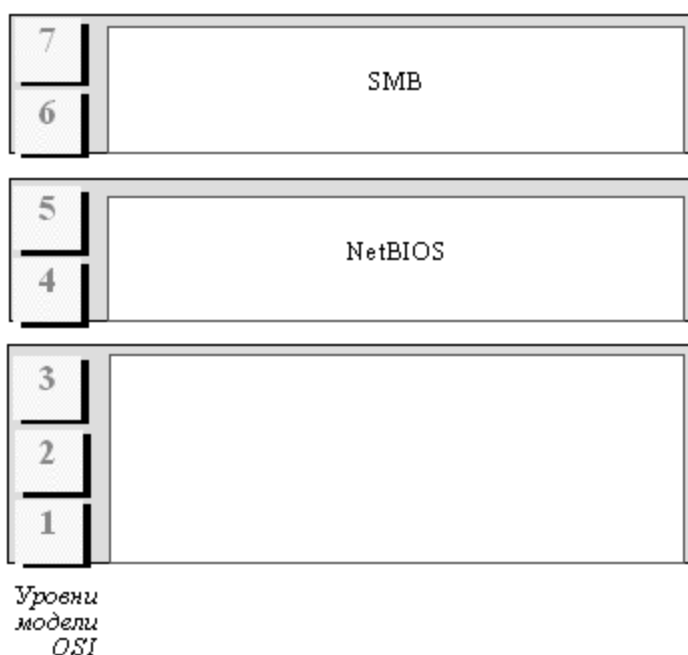


Рис. 1.6. Стек NetBIOS / SMB

Протокол **NetBIOS** работает на трех уровнях модели взаимодействия открытых систем: **сетевом, транспортном и сеансовом**. NetBIOS может обеспечить сервис более высокого уровня, чем протоколы IPX и SPX, однако не обладает способностью к маршрутизации. Таким образом, NetBIOS не является сетевым протоколом в строгом смысле этого слова. NetBIOS содержит много полезных сетевых функций, которые можно отнести к сетевому, транспортному и сеансовому уровням, однако с его помощью невозможна маршрутизация пакетов, так как в протоколе обмена кадрами NetBIOS не вводится такое понятие как сеть. Это ограничивает применение протокола NetBIOS локальными сетями, не разделенными на подсети. NetBIOS поддерживает как дейтаграммный обмен, так и обмен с установлением соединений.

Протокол **SMB**, соответствующий прикладному и представительному уровням модели OSI, регламентирует взаимодействие рабочей станции с сервером. В функции SMB входят следующие операции:

- Управление сессиями. Создание и разрыв логического канала между рабочей станцией и сетевыми ресурсами файлового сервера.

- **Файловый доступ.** Рабочая станция может обратиться к файл-серверу с запросами на создание и удаление каталогов, создание, открытие и закрытие файлов, чтение и запись в файлы, переименование и удаление файлов, поиск файлов, получение и установку файловых атрибутов, блокирование записей.
- **Сервис печати.** Рабочая станция может ставить файлы в очередь для печати на сервере и получать информацию об очереди печати.
- **Сервис сообщений.** SMB поддерживает простую передачу сообщений со следующими функциями: послать простое сообщение; послать широковещательное сообщение; послать начало блока сообщений; послать текст блока сообщений; послать конец блока сообщений; переслать имя пользователя; отменить пересылку; получить имя машины.

Из-за большого количества приложений, которые используют функции API, предоставляемые NetBIOS, во многих сетевых ОС эти функции реализованы в виде интерфейса к своим транспортным протоколам. В NetWare имеется программа, которая эмулирует функции NetBIOS на основе протокола IPX, существуют программные эмуляторы NetBIOS для Windows NT и стека TCP/IP.