**ПРОТОКОЛЫ СЕТЕВОГО ВЗАИМОДЕЙСТВИЯ**

СОДЕРЖАНИЕ

1. ВВЕДЕНИЕ
2. ПРОТОКОЛЫ СЕТЕВОГО ВЗАИМОДЕЙСТВИЯ
   1. Что такое уровни взаимодействия
   2. Модель сетевого взаимодействия
   3. Практическая реализация уровневого взаимодействия
   4. Модель ISO/OSI
   5. Функции уровней модели ISO/OS
   6. Протоколы взаимодействия приложений и протоколы транспортной подсистемы
   7. Особенности протоколов, используемых в локальных и глобальных сетях
   8. Характеристика популярных стеков коммуникационных протоколов
3. ЗАКЛЮЧЕНИЕ
4. СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ
5. ПРИЛОЖЕНИЯ

**ВВЕДЕНИЕ**

Протокол сетевого взаимодействия - это система правил, позволяющих двум или более субъектам коммуникационной системы передавать информацию посредством любого вида изменения физической величины. Протокол определяет правила, синтаксис, семантику и синхронизацию связи, а также возможные методы восстановления ошибок. Протоколы могут быть реализованы аппаратными средствами, программным обеспечением или их комбинацией.

Протоколы сетевого взаимодействия используют четко определенные форматы для обмена различными сообщениями. Каждое сообщение имеет точное значение, предназначенное для того, чтобы вызвать ответ из диапазона возможных ответов, заранее определенных для данной конкретной ситуации. Указанное поведение обычно не зависит от того, как оно должно быть реализовано. Протоколы связи должны быть согласованы соответствующими сторонами. Для достижения соглашения протокол может быть превращен в технический стандарт. Язык программирования описывает то же самое для вычислений, поэтому существует тесная аналогия между протоколами и языками программирования: протоколы являются для коммуникации тем же, чем языки программирования являются для вычислений. Альтернативная формулировка гласит, что протоколы относятся к коммуникации, а алгоритмы - к вычислению.

Несколько протоколов часто описывают различные аспекты одного сообщения. Группа протоколов, предназначенных для совместной работы, называется набором протоколов; при реализации в программном обеспечении они представляют собой стек протоколов.

Протоколы сетевого взаимодействия публикуются целевой группой по инженерному обеспечению Интернета (IETF). Институт инженеров электротехники и электроники (англ. IEEE - Institute of Electrical and Electronics Engineers) обрабатывает проводные и беспроводные сети, а международная организация по стандартизации (англ. ISO - International Organization for Standardization) - другие типы сетей. ITU-T (Сектор стандартизации электросвязи Международного союза электросвязи) обрабатывает телекоммуникационные протоколы и форматы для коммутируемой телефонной сети общего пользования (PSTN). По мере сближения PSTN и интернета происходит также сближение стандартов.

Одно из первых применений термина протокол в контексте коммутации данных встречается в меморандуме под названием протокол для использования в сети передачи данных NPL, написанном Роджером Скантлбери и Китом Бартлеттом в апреле 1967 года.

ARPANET (от англ. Advanced Research Projects Agency Network) — компьютерная сеть, созданная в 1969 году в США Агентством Министерства обороны США по перспективным исследованиям (DARPA) и явившаяся прототипом сети Интернет. В ARPANET отправной точкой для обмена данными между хостами в 1969 году был протокол 1822 года, который определял передачу сообщений в IMP. Программа управления сетью (далее от англ. – RFC) для ARPANET была впервые реализована в 1970 году, интерфейс которой позволял прикладному программному обеспечению подключаться через ARPANET, реализуя коммуникационные протоколы более высокого уровня, что является ранним примером концепции многослойности протоколов.

Исследование сетей в начале 1970-х годов Робертом Э. Каном и Винтом Серфом привело к разработке программы управления передачей данных (TCP). Спецификации RFC 675 была написана Серфом с Далалом в декабре 1974 года.

Программное обеспечение TCP было переработано в виде модульного стека протоколов. Первоначально названный IP / TCP, он был установлен в ARPANET для производственного использования в январе 1983 года. Разработка полного набора протоколов к 1989 году, описанного в RFC 1122 и RFC 1123, заложила основу для роста TCP/IP как комплексного набора протоколов в качестве основного компонента формирующегося Интернета.

Более поздние международные работы привели к созданию модели OSI, опубликованной в 1984 году. В конце 1980-х и начале 1990-х годов инженеры, организации и государства стали поляризованными в вопросе о том, какой стандарт, модель OSI или набор интернет-протоколов, приведет к созданию наилучших и наиболее надежных компьютерных сетей.

Обмен информацией между устройствами через сеть или другие носители информации регулируется правилами и соглашениями, которые могут быть изложены в спецификациях протоколов связи. Природа коммуникации, фактический обмен данными и любое поведение, зависящее от состояния, определяется этими спецификациями. В цифровых вычислительных системах правила могут быть выражены алгоритмами и структурами данных. Протоколы относятся к общению так же, как алгоритмы или языки программирования - к вычислениям.

Операционные системы обычно содержат набор взаимодействующих процессов, которые манипулируют общими данными для взаимодействия друг с другом. Эта связь регулируется хорошо понятными протоколами, которые могут быть встроены в сам код процесса. В отличие от этого, поскольку нет общей памяти, системы взаимодействия должны взаимодействовать друг с другом с помощью общей среды передачи. Передача данных не обязательно является надежной, и отдельные системы могут использовать различное оборудование или операционные системы.

Для реализации сетевого протокола программные модули протокола взаимодействуют с фреймворком, реализованным в операционной системе компьютера. Этот фреймворк реализует сетевую функциональность операционной системы. Когда алгоритмы протокола выражены в портативном языке программирования, программное обеспечение протокола может быть сделано независимым от операционной системы. Наиболее известными фреймворками являются модель TCP/IP и модель OSI.

В то время, когда был создан интернет, абстрактное многослойное проектирование оказалось успешным подходом к проектированию как компиляторов, так и операционных систем, и, учитывая сходство между языками программирования и коммуникационными протоколами, первоначально монолитные сетевые программы были разложены на сотрудничающие протоколы. Это привело к появлению концепции многоуровневых протоколов, которая в настоящее время составляет основу проектирования протоколов.

Системы обычно не используют один протокол для обработки передачи данных. Вместо этого они используют набор взаимодействующих протоколов, иногда называемый набором протоколов. Некоторые из наиболее известных пакетов протоколов - TCP/IP, IPX/SPX, X.25, AX.25.

Протоколы могут быть организованы на основе функциональных возможностей в группы, например, существует группа транспортных протоколов. Функциональные возможности отображаются на слои, каждый из которых решает отдельный класс задач, связанных, например, с функциями приложения, транспорта, интернета и сетевого интерфейса. Для передачи сообщения необходимо выбрать протокол на каждом уровне. Выбор следующего протокола осуществляется путем расширения сообщения с помощью селектора протокола для каждого уровня.

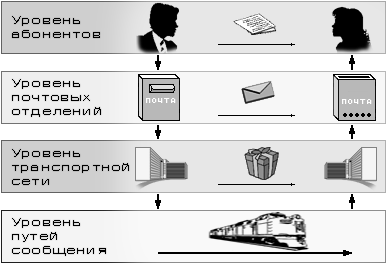
**2. ПРОТОКОЛЫ СЕТЕВОГО ВЗАИМОДЕЙСТВИЯ**

**2.1 Что такое уровни взаимодействия**

При описании практически любого взаимодействия можно выделять различные уровни. Например, представьте себе, что двум людям, проживающим в разных населенных пунктах, необходимо обмениваться какой-либо информацией, и они используют для этого традиционный способ посылки писем. Уже во взаимодействии такого рода можно выделить несколько уровней:

* уровень пользователей, обменивающихся письмами, и использующих для этой цели почтовую службу;
* уровень почтовой службы, осуществляющей пересылку корреспонденции между почтовыми отделениями населенных пунктов и использующей для работы услуги транспортной сети;
* уровень транспортной сети, обеспечивающий доставку грузов по путям сообщения между населенными пунктами;
* уровень путей сообщения, обеспечивающий возможность физической доставки грузов между населенными пунктами.

В случае, если не существует прямых путей сообщения между населенными пунктами, к этой схеме между уровнями почтовой службы и транспортной сети добавляется еще один уровень – уровень отделений по перевозке почты, обеспечивающих правильную перегрузку почтовых отправлений на транспортных узлах, а также выбор альтернативных путей пересылки в случае выхода из строя транспортных линий.

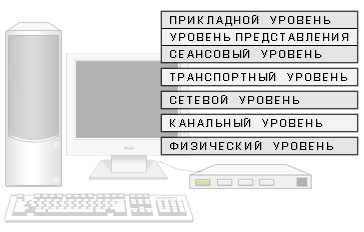


*Пример уровневого взаимодействия*

Разделение процесса взаимодействия на уровни позволяет функционально изолировать различные средства, участвующие в этом процессе по принципу - "каждый занимается своим делом". Это позволяет обеспечить достаточную гибкость при расширении функциональности этих средств. Так, например, выделение уровня транспортной сети, позволяет при необходимости обеспечить транспортировку между населенными пунктами не только почтовых грузов, но и пассажиров, не требуя для этого перестройки путей сообщения. Выделение почтовой службы обеспечивает возможность пересылки не только писем, но и посылок, переводов и т.п., используя стандартные средства транспортной сети и опосредованно – существующие пути сообщения протоколов сетевого взаимодействия.  
  
Взаимодействие в компьютерных сетях также можно описывать с помощью уровней. В настоящее время для этих целей широко используется так называемая модель взаимодействия открытых систем (Open Systems Interconnection, OSI).

**2.2 Модель сетевого взаимодействия**

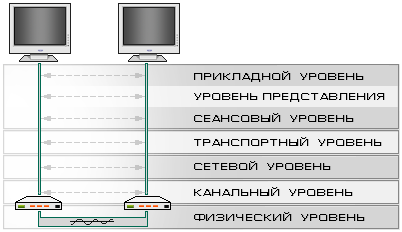
В 1984 году Международной Организацией по Стандартизации (International Standard Organization, ISO) была разработана *модель взаимодействия открытых систем* (Open Systems Interconnection, OSI). Модель представляет собой международный стандарт для проектирования сетевых коммуникаций и предполагает уровневый подход к построению сетей. Каждый уровень модели обслуживает различные этапы процесса взаимодействия. Посредством деления на уровни сетевая модель OSI упрощает совместную работу оборудования и программного обеспечения. Модель OSI разделяет сетевые функции на семь уровней: прикладной, уровень представления, сессионный, транспортный, сетевой, канальный и физический.



*Уровни модели OSI*

При уровневой организации процесса взаимодействия должны соблюдаться следующие требования:

* компоненты одного уровня одной системы могут взаимодействовать с компонентами только того же уровня другой системы;
* в рамках одной системы компоненты какого-либо уровня могут взаимодействовать только с компонентами смежных (вышележащего и нижележащего) уровней.



*Порядок уровневого взаимодействия*

Набор правил, определяющих порядок взаимодействия средств, относящихся к одному и тому же уровню и функционирующих в разных системах, называется протоколом (protocol). Правила взаимодействия между собой средств, относящихся к смежным уровням и функционирующих в одной системе, называются интерфейсом (interface).

**2.3 Практическая реализация уровневого взаимодействия**

На практике протоколы и интерфейсы регламентируют технические требования, предъявляемые к программным и аппаратным средствам. Программные (аппаратные) модули, предназначенные для обеспечения практического взаимодействия, определяемого тем или иным протоколом (или интерфейсом), обычно называют реализацией протокола (интерфейса).  
Хотя различные компоненты, относящиеся к различным уровням сетевой модели формально должны быть функционально независимыми друг от друга, при практической разработке протоколов такая независимость не всегда выдерживается. Это объясняется тем, что попытка добиться точного соответствия эталонной модели может привести к неэффективности работы программно-аппаратного обеспечения, реализующего протокол. В настоящее время наблюдается два типа отклонений, возникающих при реализации уровневого взаимодействия:

* функции некоторых уровней могут объединяться одним протоколом и наоборот, – функции одного уровня могут делиться между различными протоколами;
* функционирование протокола какого-либо уровня подразумевают использование только определенных протоколов нижележащего уровня.

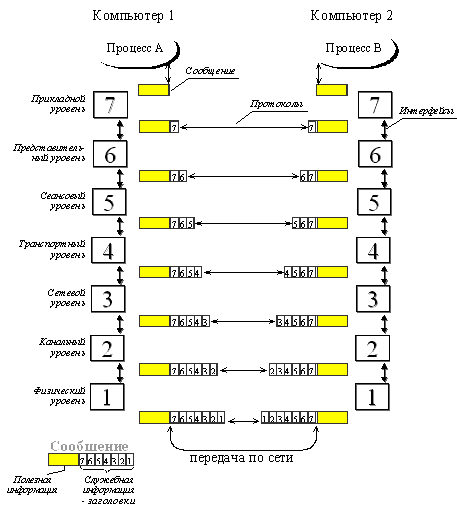
Поэтому разработка практических методов сетевого взаимодействия, как правило, подразумевает разработку не отдельных протоколов, а целых наборов протоколов. Такие наборы обычно включают в себя протоколы, относящиеся к нескольким смежным уровням эталонной модели OSI, и называются стеками (или семействами, наборами) протоколов (protocol stack, protocol suite). Наиболее известным стеком протоколов, обеспечивающим взаимодействие в сети Интернет, является стек протоколов TCP/IP  
Поскольку при реализации протоколов допускаются отклонения от эталонной модели, стеки протоколов могут предполагать собственную схему деления на уровни. В частности, стек протоколов TCP/IP разделяет весь процесс сетевого взаимодействия на четыре уровня. На предложенном ниже рисунке показано соответствие уровней модели OSI и уровней стека TCP/IP.

|  |  |
| --- | --- |
| **Уровни модели OSI** | **Уровни стека TCP/IP** |
| прикладной представления сессионный | уровень приложения |
| транспортный | транспортный уровень |
| сетевой | межсетевой уровень |
| канальный физический | уровень сетевого интерфейса |

*Соответствие уровней модели OSI и уровней стека TCP/IP*

**2.4 Модель 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 межсетевого обмена 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 представительному уровню. 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).

**2.5 Функции уровней модели ISO/OS**

*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 стандартизуются 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, 100VG-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 мультиплексирования 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.

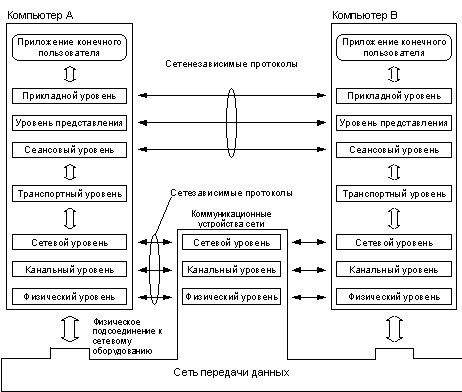
**2.6 Протоколы взаимодействия приложений и протоколы транспортной подсистемы**

Функции всех уровней модели OSI могут быть отнесены к одной из двух групп: либо к функциям, зависящим от конкретной технической реализации сети, либо к функциям, ориентированным на работу с приложениями.

Три нижних уровня - физический, канальный и сетевой - являются сетезависимыми, то есть протоколы этих уровней тесно связаны с технической реализацией сети, с используемым коммуникационным оборудованием. Например, переход на оборудование FDDI означает полную смену протоколов физического и канального уровня во всех узлах сети.

Три верхних уровня - сеансовый, уровень представления и прикладной - ориентированы на приложения и мало зависят от технических особенностей построения сети. На протоколы этих уровней не влияют никакие изменения в топологии сети, замена оборудования или переход на другую сетевую технологию. Так, переход от Ethernet на высокоскоростную технологию АТМ не потребует никаких изменений в программных средствах, реализующих функции прикладного, представительного и сеансового уровней.

Транспортный уровень является промежуточным, он скрывает все детали функционирования нижних уровней от верхних уровней. Это позволяет разрабатывать приложения, независящие от технических средств, непосредственно занимающихся транспортировкой сообщений.



*Рис. 1.2. Сетезависимые и сетенезависимые уровни модели OSI*

Рисунок 1.2 показывает уровни модели OSI, на которых работают различные элементы сети. Компьютер, с установленной на нем сетевой ОС, взаимодействует с другим компьютером с помощью протоколов всех семи уровней. Это взаимодействие компьютеры осуществляют через различные коммуникационные устройства: концентраторы, модемы, мосты, коммутаторы, маршрутизаторы, мультиплексоры. В зависимости от типа, коммуникационное устройство может работать либо только на физическом уровне (повторитель), либо на физическом и канальном (мост и коммутатор), либо на физическом, канальном и сетевом, иногда захватывая и транспортный уровень (маршрутизатор).

**2.7 Особенности протоколов, используемых в локальных и глобальных сетях**

В настоящее время наблюдается тенденция к сближению протоколов локальных и глобальных сетей. Ярким примером являются протоколы технологии АТМ, работающие без изменений как в тех, так и в других сетях. Тем не менее, большинство протоколов, используемых сегодня, относятся либо к локальным, либо к глобальным сетям и не могут применяться не по прямому назначению.

Различия между протоколами локальных и глобальных сетей происходят в основном из-за различий между свойствами каналов, использующихся в этих сетях.

Каналы локальных сетей имеют небольшую длину и высокое качество, а каналы глобальных сетей - наоборот, большую длину и низкое качество.

Небольшая длина каналов локальных сетей создала возможность совместного использования их узлами сети в режиме разделения времени. Практически все протоколы локальных сетей имеют версию работы на разделяемых средах передачи данных, хотя более поздние протоколы (Fast Ethernet, Gigabit Ethernet) имеют также и версию работы на индивидуальных каналах в полнодуплексном режиме. Большая протяженность каналов глобальных сетей делает нерациональными любые процедуры разделения канала во времени, так как длительность этих процедур становится слишком большой. Поэтому каналы глобальных сетей используются всегда на индивидуальной основе как связи типа "точка - точка".

Высокое качество кабелей локальных сетей послужило причиной отказа от использования в протоколах локальных сетей процедур восстановления искаженных и потерянных кадров. Этих процедур нет ни в протоколах семейства Ethernet, ни у протокола Token Ring, ни у протокола FDDI. В то же время в протоколах глобальных сетей, ориентирующихся на каналы плохого качества, процедурам восстановления кадров всегда уделялось большое внимание. Например, в сетях Х.25 восстановлением кадров занимаются сразу два смежных протокола - LAP-B на канальном уровне и протокол Х.25/3 - на сетевом.

Начало массового использования цифровых оптоволоконных каналов в глобальных сетях, обеспечивающих высокое качество передачи данных, послужило причиной разработки протоколов глобальных сетей нового поколения, в которых отсутствуют процедуры восстановления кадров. Такой особенностью обладают, например, сети frame relay и ATM.

Таким образом, одно из отличий протоколов локальных и глобальных сетей преодолено за счет продвижения глобальных сетей навстречу локальным. Второе отличие сегодня снимается за счет быстрого внедрения в локальные сети техники микросегментации, отказывающейся от использования разделяемых сред и предоставляющей каждому узлу сети индивидуальный коммутируемый канал. В результате, протоколы локальных и глобальных сетей все больше сближаются, а существование технологии АТМ доказывает, что принципиальных причин для существования между этими классами протоколов четкой границы сегодня не существует и ее окончательное исчезновение - не за горами.

**2.8 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 рекомендуемому моделью 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.

**ЗАКЛЮЧЕНИЕ**

Очевидно, что рано или поздно компьютеры, расположенные в разных точках земного шара, по мере увеличения своего количества должны были обрести некие средства общения. Такими средствами стали компьютерные сети. Сети бывают локальными и глобальными. Локальная сеть - это сеть, объединяющая компьютеры, географически расположенные на небольшом расстоянии друг от друга - например, в одном здании. Глобальные сети служат для соединения сетей и компьютеров, которых разделяют большие расстояния - в сотни и тысячи километров. Интернет относится к классу глобальных сетей.

Простое подключение одного компьютера к другому - шаг, необходимый для создания сети, но не достаточный. Чтобы начать передавать информацию, нужно убедиться, что компьютеры "понимают" друг друга. Как же компьютеры "общаются" по сети? Чтобы обеспечить эту возможность, были разработаны специальные средства, получившие название "протоколы". Протокол - это совокупность правил, в соответствии с которыми происходит передача информации через сеть. Понятие протокола применимо не только к компьютерной индустрии. Даже те, кто никогда не имел дела с Интернетом, скорее всего работали в повседневной жизни с какими-либо устройствами, функционирование которых основано на использовании протоколов. Так, обычная телефонная сеть общего пользования тоже имеет свой протокол, который позволяет аппаратам, например, устанавливать факт снятия трубки на другом конце линии или распознавать сигнал о разъединении и даже номер звонящего.

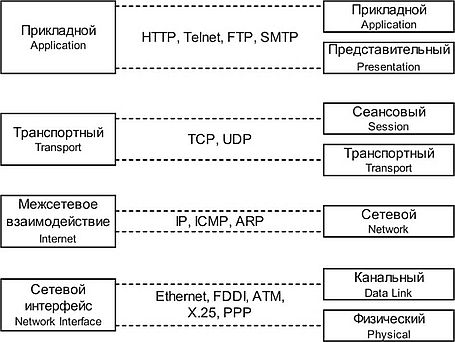
Исходя из этой естественной необходимости, миру компьютеров потребовался единый язык (то есть протокол), который был бы понятен каждому из них.

**СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ**

1. Распределенные системы. Принципы и парадигмы / Э. Таненбаум, М. ван Стеен. — СПб.: Питер, 2003. — с. 83-93 — (Серия «Классика computer science»).
2. Р. Д. Теннент (1981): Принципы программирования языков 10-я печать. Прентис-Холл.
3. Hunt, Craig. TCP/IP Network Administration. — 3rd Edition. — O'Reilly Media, Inc..
4. Radia Perlman: Interconnections: Bridges, Routers, Switches, and Internetworking Protocols. 2nd Edition. Addison-Wesley 1999.
5. Douglas E. Comer (2000). Internetworking with TCP/IP - Principles, Protocols and Architecture (4th ed.). Prentice Hall.

**ПРИЛОЖЕНИЯ**

1. Модель ISO/OSI и стек протоколов TCP/IP



1. Видео «Основы сетей передачи данных. Модель OSI и стек протоколов TCP IP. Основы Ethernet» - <https://www.youtube.com/watch?v=Z-a7MNStFQs> .