Компьютерная сеть - это группа компьютеров, соединённая каналами связи.

Сети находятся в частном ведении пользователя и занимают некоторую территорию и по территориальному признаку делятся на:

1. Локальные вычислительные сети (LAN) Local area networks - расположены в одном здании, часто называются корпоративными

2. Распределённые компьютерные сети (WAN) Wide Area Networks - соединены каналами связи, расположены на расстоянии

Различают два вида сети:

1. Коммуникационная сеть

2. Информационная сеть

Коммуникационная сеть предназначена для передачи данных, связана также с преобразованием данных.

Они различаются по типу используемых физических средств соединения

Информационная сеть состоит из информационных систем, предназначена для хранения информации

На базе коммуникационных сетей может быть постоена база информационных сетей.

Под информационной системой следует понимать поставщика и потребителя информации.

Преимущества работы в сети:

1. Совместное использование информации

2. Совместное использование аппаратных средств

3. Совместное использование программных средств

4. Обмен сообщениями

Недостатки работы в сети:

1. Быстрое распространение вирусов

2. Возможность взлома информации

Архитектура сети определяет основные элементы сети, определяет общую логическую организацию, техническое обеспечение и программное обеспечение, а так же описывает методы кодирования

Архитектура также определяет принципы функционирования и интерфейс пользователя

Первый вид архитекруты это одноранговая архитектура

Второй вид это клиент-серверная архитектура

Третий - гибридная архитектура

Одноранговая сеть - это компьютерная сеть, основанная на равноправии участников. В этой сети нет как такового выделенного сервера. Как правило каждый компьютер функционирует и как сервер и как клиент в одноранговой сети. Все пользователи самостоятельно решают что и кому можно сделать на своем компьютере. Одноранговая сеть используется в небольшой организации, компьютеров в сети в основном меньше 10.

Архитектура клиент-сервер, клиент-серверная архитектура - это концепция информационной сети, в которой основная часть ее ресурсов сосредоточена в серверах, обслуживающих своих клиентов. Данная архитектура состоит из двух основных объектов - сервера и клиента.

Сервер - это объект, предоставляющий сервис другим объектам сети по их запросам.

Сервис - это процесс обслуживания клиентов.

Клиенты - это рабочие станции, которые использую ресурсы сервера и предоставляют удобные интерфейсы пользователям.

Сервер работает по заданиям клиентов и управляет исполнением их заданий. После выполнения каждого задания сервер посылает результат выполнения клиенту, пославшему это задание.

Сервисная функция в архитектуре клиент-сервера описывается компексом прикладных программ, в соответсвии с которыми выполняются разнообразные прикладные процессы.

Гибридные сети (смешанные) включают в себя признаки как одноранговых сетей так и сетей, построенных по клиент-серверному принципу. В смешанной сети узел является сервером для одной части компьютеров и может быть клиентов для другого сервера.

ФАЙЛОВЫЕ СЕРВЕРЫ

Сервер - это компьюьер, предоставляющий свои ресурсы (принтеры, диски, каталоги и т.п.) другим пользователям сети.

На файловом сервере должна стоять сетевая ОС, а также сетевое ПО.

К сетевому ПО сервера относятся сетевые службы и протоколы, а также средства администрирования сервера.

Файловые серверы могут контролировать доступ пользователей к различным частям файловой системы. Это обычно осуществляется разрешением пользователю присоединить файловую систему или каталог к рабочей станции пользователя, к локальному диску.

По мере возлагаемых на сервер функций и увеличения обслуживаемых ими клиентов происходит всё большая специализация серверов.

Понятие пакета

При обмене данными как между компьютеров в лок. сети так и в глоб. сети любое инф.- ое сообщение разбивается программами передачи данных на небольшие блоки данных, которые называются пакетами, связанно это с тем, что данные обычно содержаться в больших по размерам файлах, если передающий компьютер пошлёт его целиком, то он надолго заполнит канал связи, и свяжет работу всей сети, то есть будет препятствовать взаимодействию других участников сети, кроме того возникновение ошибок при передаче крупных блоков вызовет большие затраты времени, чем на его повторную передачу.

Пакет- это основная единица информации в компьютерных сетях при разбиении данных на пакеты скорость их передачи возрастает на столько, что каждый компьютер сети получает возможность принимать и передавать данные практически одновременно с остальными компьютерами. При разбиении данных на пакеты сетевая ОС к передаваемым данным добавляет специальную добавочную информацию:

1. Заголовок. В заголовке содержится адрес отправителя и получателя, а также время отправки пакета, номер пакета в последовательности отправления сообщений и тд.
2. Данные. Размер данных зависит от типа сети
3. Трейлер. В трейлере содержится информация для проверки безошибочности передачи пакета, при обнаружении ошибки, передача пакета должна повториться

Методы передачи данных.

При соединении по сетевым каналам используют 3 различных метода:

1. Симплексный
2. Полудуплексное
3. Дуплексное

Симплексный канал позволяет перемещать данные только в одном направлении.

Полудуплексное соединение позволяет данным перемещаться в обоих направлениях, но не одновременно, а по очереди.

Дуплексное соединение позволяет данным перемещаться в обоих направления одновременно.

Модели OSI.

Для единого предоставления данных в сетях с неоднородными устройствами и ПО, ISO разработало базовую модель связи ооткрытых систем, которая называется OSI. Эта модель описывает правила и процедуры передачи данных в различных сетевых средах при организации сеанса связи. Основными элементами модели являются уровни, прикладные процессы и физические средства соединения.

1. Физический
2. Канальный
3. Сетевой
4. Транспортный
5. Сеансовый
6. Представительский
7. Прикладной

Каждый уровень модели OSI выполняет определлённую задачу в процессе передачи данных по сети. Базовая модель является основой для разработки сетевых протоколов, данная модель разделяет коммуникационные функции в сети на 7 уровней, каждая из которых обслуживает различные части процесса области взаимодействия открытых систем.

**РАЗДЕЛ 2. СЕТЕВЫЕ АРХИТЕКТУРЫ**

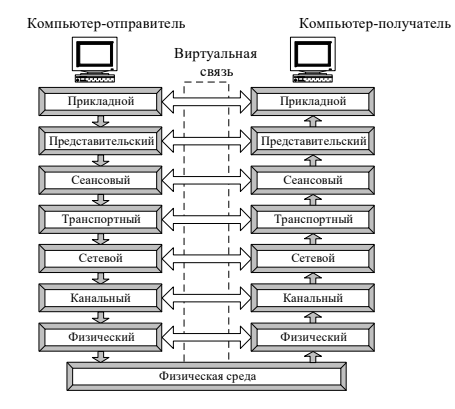
**Модель OSI**

Для единого представления данных в сетях с неоднородными устройствами и программным обеспечением международная организация по стандартам ISO (International Standardization Organization) разработала базовую модель связи открытых систем OSI (Open System Interconnection). Эта модель описывает правила и процедуры передачи данных в

различных сетевых средах при организации сеанса связи. Основными элементами модели  являются уровни, прикладные процессы и физические средства соединения. На рисунке представлена структура базовой модели. Каждый уровень модели OSI выполняет  определенную задачу в процессе передачи данных по сети. Базовая модель является основой  для разработки сетевых протоколов. OSI разделяет коммуникационные функции в сети на  семь уровней, каждый из которых обслуживает различные части процесса области  взаимодействия открытых систем.

****

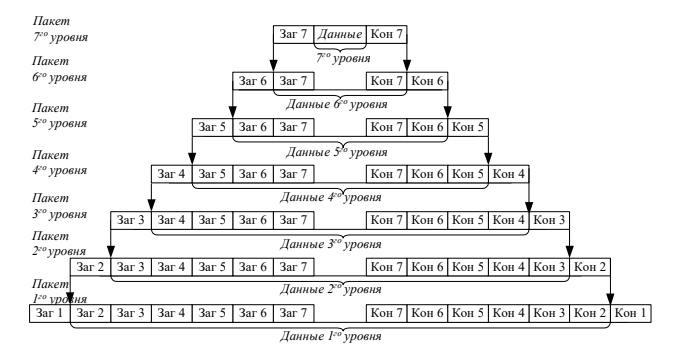
Модель OSI описывает только системные средства взаимодействия, не касаясь  приложений конечных пользователей. Приложения реализуют свои собственные протоколы  взаимодействия, обращаясь к системным средствам. Если приложение может взять на себя  функции некоторых верхних уровней модели OSI, то для обмена данными оно обращается  напрямую к системным средствам, выполняющим функции оставшихся нижних уровней  модели OSI.



Каждый уровень компьютера–отправителя взаимодействует с таким же уровнем  компьютера-получателя, как будто он связан напрямую. Такая связь называется логической  или виртуальной связью. В действительности взаимодействие осуществляется между  смежными уровнями одного компьютера.

Информация на компьютере-отправителе должна пройти через все уровни. Затем она  передается по физической среде до компьютера–получателя и опять проходит сквозь все слои,  пока не доходит до того же уровня, с которого она была послана на компьютере-отправителе.

Перед подачей в сеть данные разбиваются на пакеты. На принимающей стороне пакет  проходит через все уровни в обратном порядке. На каждом уровне протокол этого уровня  читает информацию пакета, затем удаляет информацию, добавленную к пакету на этом же  уровне отправляющей стороной, и передает пакет следующему уровню. Когда пакет дойдет до  *Прикладного* уровня, вся управляющая информация будет удалена из пакета, и данные примут  свой первоначальный вид.



Каждый уровень модели выполняет свою функцию. Чем выше уровень, тем более  сложную задачу он решает.

Отдельные уровни модели OSI удобно рассматривать как группы программ,  предназначенных для выполнения конкретных функций. Один уровень, к примеру, отвечает за  обеспечение преобразования данных из ASCII в EBCDIC и содержит программы необходимые  для выполнения этой задачи.

Каждый уровень обеспечивает сервис для вышестоящего уровня, запрашивая в свою  очередь, сервис у нижестоящего уровня. Верхние уровни запрашивают сервис почти  одинаково: как правило, это требование маршрутизации каких-то данных из одной сети в  другую. Практическая реализация принципов адресации данных возложена на нижние уровни.

Рассматриваемая модель определяет взаимодействие открытых систем разных  производителей в одной сети. Поэтому она выполняет для них координирующие действия по:

− взаимодействию прикладных процессов;

− формам представления данных;

− единообразному хранению данных;

− управлению сетевыми ресурсами;

− безопасности данных и защите информации;

− диагностике программ и технических средств.

**7. Прикладной**

представляет набор интерфейсов, позволяющий получить доступ к

 сетевым службам

**6. Представления**

преобразует данные в общий формат для передачи по сети

**5. Сеансовый**

поддержка взаимодействия (сеанса) между удаленными процессами

**4. Транспортный**

управляет передачей данных по сети, обеспечивает подтверждение

 передачи

**3. Сетевой**

маршрутизация, управление потоками данных, адресация сообщений  для доставки, преобразование логические сетевые адреса и имена в

 соответствующие им физические

**2. Канальный**

**2.1. Контроль логической связи (LLC):**

формирование кадров

**2.2. Контроль доступа к среде (MAC):**

управление доступом к среде

**1. Физический:**

битовые протоколы передачи информации

**Физический уровень (Physical Layer)**

Физический уровень предназначен для сопряжения с *физическими средствами  соединения*. *Физические средства соединения* – это совокупность *физической среды*,  аппаратных и программных средств, обеспечивающая передачу сигналов между системами.  *Физическая среда* – это материальная субстанция, через которую осуществляется передача  сигналов. Физическая среда является основой, на которой строятся физические средства  соединения. В качестве физической среды широко используются эфир, металлы, оптическое  стекло и кварц.

Физический уровень состоит из *Подуровня стыковки со средой* и *Подуровня  преобразования передачи.*

Первый из них обеспечивает сопряжение потока данных с используемым физическим  каналом связи. Второй осуществляет преобразования, связанные с применяемыми  протоколами. Физический уровень обеспечивает физический интерфейс с каналом передачи  данных, а также описывает процедуры передачи сигналов в канал и получения их из канала.  На этом уровне определяются электрические, механические, функциональные и процедурные  параметры для физической связи в системах. Физический уровень получает пакеты данных от вышележащего канального уровня и преобразует их в оптические или электрические сигналы,  соответствующие 0 и 1 бинарного потока. Эти сигналы посылаются через среду передачи на  приемный узел. Механические и электрические / оптические свойства среды передачи  определяются на физическом уровне и включают:

− тип кабелей и разъемов;

− разводку контактов в разъемах;

− схему кодирования сигналов для значений 0 и 1.

Физический уровень выполняет следующие функции:

1. Установление и разъединение физических соединений.

2. Передача сигналов в последовательном коде и прием.

3. Прослушивание, в нужных случаях, каналов.

4. Идентификация каналов.

5. Оповещение о появлении неисправностей и отказов.

Оповещение о появлении неисправностей и отказов связано с тем, что на физическом  уровне происходит обнаружение определенного класса событий, мешающих нормальной  работе сети (столкновение кадров, посланных сразу несколькими системами, обрыв канала,  отключение питания, потеря механического контакта и т. д.). Виды сервиса, предоставляемого  канальному уровню, определяются протоколами физического уровня. Прослушивание канала  необходимо в тех случаях, когда к одному каналу подключается группа систем, но  одновременно передавать сигналы разрешается только одной из них. Поэтому прослушивание  канала позволяет определить, свободен ли он для передачи.

Можно считать этот уровень, отвечающим за аппаратное обеспечение.

Физический уровень может обеспечивать как асинхронную (последовательную) так и  синхронную (параллельную) передачу, которая применяется для некоторых мэйнфреймов и  мини - компьютеров. На Физическом уровне должна быть определена схема кодирования для  представления двоичных значений с целью их передачи по каналу связи. Во многих  локальных сетях используется манчестерское кодирование.

**Канальный уровень (Data Link)**

Единицей информации канального уровня являются *кадры (frame).* Кадры – это  логически организованная структура, в которую можно помещать данные. Задача канального  уровня передавать кадры от сетевого уровня к физическому уровню.

На физическом уровне просто пересылаются биты. При этом не учитывается, что в  некоторых сетях, в которых линии связи используются 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.

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.

Задача канального уровня - брать пакеты, поступающие с сетевого уровня и готовить их  к передаче, укладывая в кадр соответствующего размера. Этот уровень обязан определить, где  начинается и где заканчивается блок, а также обнаруживать ошибки передачи.

0На этом же уровне определяются правила использования физического уровня узлами  сети. Электрическое представление данных в ЛВС (биты данных, методы кодирования данных  и маркеры) распознаются на этом и только на этом уровне. Здесь обнаруживаются и  исправляются (путем требований повторной передачи данных) ошибки.

Канальный уровень обеспечивает создание, передачу и прием кадров данных. Этот  уровень обслуживает запросы сетевого уровня и использует сервис физического уровня для  приема и передачи пакетов. Спецификации IEEE 802.Х делят канальный уровень на два  подуровня:

− *LLC* (*Logical Link Control*) управление логическим каналом осуществляет  логический контроль связи. Подуровень LLC обеспечивает обслуживание сетевого  уровня и связан с передачей и приемом пользовательских сообщений.

− *MAC* (*Media Assess Control)* контроль доступа к среде. Подуровень MAC  регулирует доступ к разделяемой физической среде (передача маркера или  обнаружение коллизий или столкновений) и управляет доступом к каналу связи.  Подуровень *LLC* находится выше подуровня *МАC.*

Канальный уровень определяет доступ к среде и управление передачей посредством  процедуры передачи данных по каналу. При больших размерах передаваемых блоков данных  канальный уровень делит их на кадры и передает кадры в виде последовательностей. При  получении кадров уровень формирует из них переданные блоки данных. Размер блока данных  зависит от способа передачи, качества канала, по которому он передается.

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. Канальный уровень может выполнять следующие виды функций:

1. Организация (установление, управление, расторжение) канальных соединений и  идентификация их портов.

2. Организация и передача кадров.

3. Обнаружение и исправление ошибок.

4. Управление потоками данных.

5. Обеспечение прозрачности логических каналов (передачи по ним данных,  закодированных любым способом).

**Сетевой уровень (Network Layer)**

Сетевой уровень обеспечивает прокладку каналов, соединяющих абонентские и  административные системы через коммуникационную сеть, выбор маршрута наиболее  быстрого и надежного пути.

Сетевой уровень устанавливает связь в вычислительной сети между двумя системами и  обеспечивает прокладку виртуальных каналов между ними. *Виртуальный или логический  канал* - это такое функционирование компонентов сети, которое создает взаимодействующим  компонентам иллюзию прокладки между ними нужного канала. Кроме этого, сетевой уровень  сообщает транспортному уровню о появляющихся ошибках. Сообщения сетевого уровня  принято называть *пакетами* (packet). В них помещаются фрагменты данных. Сетевой уровень  отвечает за их адресацию и доставку.

Прокладка наилучшего пути для передачи данных 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.

Протокол канального уровня 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.

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

Сетевой уровень отвечает за деление пользователей на группы и маршрутизацию  пакетов на основе преобразования MAC-адресов в сетевые адреса. Сетевой уровень  обеспечивает также прозрачную передачу пакетов на транспортный уровень.  Сетевой уровень выполняет функции:

1. Создание сетевых соединений и идентификация их портов.

2. Обнаружение и исправление ошибок, возникающих при передаче через  коммуникационную сеть.

3. Управление потоками пакетов.

4. Организация (упорядочение) последовательностей пакетов.

5. Маршрутизация и коммутация.

6. Сегментирование и объединение пакетов.

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. Однако часто 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. **Транспортный уровень (Transport Layer)**

Транспортный уровень предназначен для передачи пакетов через коммуникационную  сеть. На транспортном уровне пакеты разбиваются на блоки.

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 модели  (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.

Транспортный уровень определяет адресацию физических устройств (систем, их частей)  в сети. Этот уровень гарантирует доставку блоков информации адресатам и управляет этой доставкой. Его главной задачей является обеспечение эффективных, удобных и надежных  форм передачи информации между системами. Когда в процессе обработки находится более  одного пакета, транспортный уровень контролирует очередность прохождения пакетов. Если

проходит дубликат принятого ранее сообщения, то данный уровень опознает это и игнорирует  сообщение.

В функции транспортного уровня входят:

1. Управление передачей по сети и обеспечение целостности блоков данных. 2. Обнаружение ошибок, частичная их ликвидация и сообщение о неисправленных  ошибках.

3. Восстановление передачи после отказов и неисправностей.

4. Укрупнение или разделение блоков данных.

5. Предоставление приоритетов при передаче блоков (нормальная или срочная). 6. Подтверждение передачи.

7. Ликвидация блоков при тупиковых ситуациях в сети.

Начиная с транспортного уровня, все вышележащие протоколы реализуются  программными средствами, обычно включаемыми в состав сетевой операционной системы. Наиболее распространенные протоколы транспортного уровня включают в себя: **Сеансовый уровень (Session layer)**

Сеансовый уровень – это уровень, определяющий процедуру проведения сеансов между  пользователями или прикладными процессами.

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 сначала. На практике немногие приложения используют сеансовый  уровень, и он редко реализуется.

Сеансовый уровень управляет передачей информации между прикладными процессами,  координирует прием, передачу и выдачу одного сеанса связи. Кроме того, сеансовый уровень  содержит дополнительно функции управления паролями, управления диалогом,  синхронизации и отмены связи в сеансе передачи после сбоя вследствие ошибок в  нижерасположенных уровнях. Функции этого уровня состоят в координации связи между  двумя прикладными программами, работающими на разных рабочих станциях. Это  происходит в виде хорошо структурированного диалога. В число этих функций входит  создание сеанса, управление передачей и приемом пакетов сообщений во время сеанса и  завершение сеанса.

На сеансовом уровне определяется, какой будет передача между двумя прикладными  процессами:

− полудуплексной

− дуплексной

В полудуплексном режиме сеансовый уровень выдает тому процессу, который начинает  передачу, *маркер данных*. Когда второму процессу приходит время отвечать, маркер данных  передается ему. Сеансовый уровень разрешает передачу только той стороне, которая обладает  маркером данных.

Сеансовый уровень обеспечивает выполнение следующих функций:

1. Установление и завершение на сеансовом уровне соединения между  взаимодействующими системами.

2. Выполнение нормального и срочного обмена данными между прикладными  процессами.

3. Управление взаимодействием прикладных процессов.

4. Синхронизация сеансовых соединений.

5. Извещение прикладных процессов об исключительных ситуациях. 6. Установление в прикладном процессе меток, позволяющих после отказа либо  ошибки восстановить его выполнение от ближайшей метки.

7. Прерывание в нужных случаях прикладного процесса и его корректное  возобновление.

8. Прекращение сеанса без потери данных.

9. Передача особых сообщений о ходе проведения сеанса.

Сеансовый уровень отвечает за организацию сеансов обмена данными между  оконечными машинами. Протоколы сеансового уровня обычно являются составной частью  протоколов трех верхних уровней модели.

**Уровень представления данных (Presentation layer)**

Уровень представления данных или представительский уровень представляющие  данные, передаваемые между прикладными процессами, в нужной форме данные. Этот уровень обеспечивает то, что информация, передаваемая прикладным уровнем,  будет понятна прикладному уровню в другой системе. В случаях необходимости уровень  представления в момент передачи информации выполняет преобразование форматов данных в  некоторый общий формат представления, а в момент приема, соответственно, выполняет  обратное преобразование. Таким образом, прикладные уровни могут преодолеть, например,  синтаксические различия в представлении данных. Такая ситуация может возникнуть в ЛВС с  неоднотипными компьютерами (*IBM PC и Macintosh*), которым необходимо обмениваться  данными. Так, в полях баз данных информация должна быть представлена в виде букв и цифр,  а зачастую и в виде графического изображения. Обрабатывать же эти данные нужно,  например, как числа с плавающей запятой.

Эта система служит для описания структуры файлов, а также позволяет решить  проблему шифрования данных. На этом уровне может выполняться шифрование и  дешифрование данных, благодаря которым секретность обмена данными обеспечивается сразу  для всех прикладных сервисов. Примером такого протокола является протокол *Secure Socket Layer (SSL),* который обеспечивает секретный обмен сообщениями для протоколов  прикладного уровня стека TCP/IP. Этот уровень обеспечивает преобразование данных  (кодирование, компрессия и т.п.) прикладного уровня в поток информации для транспортного уровня.

Представительный уровень выполняет следующие основные функции:

1. Генерация запросов на установление сеансов взаимодействия прикладных  процессов.

2. Согласование представления данных между прикладными процессами. 3. Реализация форм представления данных.

4. Засекречивание данных.

5. Передача запросов на прекращение сеансов.

Протоколы уровня представления данных обычно являются составной частью  протоколов трех верхних уровней модели.

**Прикладной уровень (Application layer)**

Прикладной уровень обеспечивает прикладным процессам средства доступа к области  взаимодействия, является верхним (седьмым) уровнем и непосредственно примыкает к  прикладным процессам. В действительности прикладной уровень – это 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. Специальные  элементы прикладного сервиса обеспечивают сервис для конкретных прикладных программ,  таких как программы пересылки файлов и эмуляции терминалов. Если, например программе  необходимо переслать файлы, то обязательно будет использован *протокол передачи, доступа  и управления файлами* FTAM (File Transfer, Access, and Management). В модели OSI *прикладная программа*, которой нужно выполнить конкретную задачу (например, обновить  базу данных на компьютере), посылает конкретные данные в виде *Дейтаграммы* на  *прикладной уровень*. Одна из основных задач этого уровня - определить, как следует  обрабатывать запрос прикладной программы, другими словами, какой вид должен принять  данный запрос.

Единица данных, которой оперирует прикладной уровень, обычно называется  сообщением (message).

Прикладной уровень выполняет следующие функции:

1. Описание форм и методов взаимодействия прикладных процессов.

2. Выполнение различных видов работ:

− передача файлов;

− управление заданиями;

− управление системой и т.д.

3. Идентификация пользователей по их паролям, адресам, электронным подписям; − Определение функционирующих абонентов и возможности доступа к новым  прикладным процессам;

− Определение достаточности имеющихся ресурсов;

− Организация запросов на соединение с другими прикладными процессами; − Передача заявок представительскому уровню на необходимые методы описания  информации;

− Выбор процедур планируемого диалога процессов;

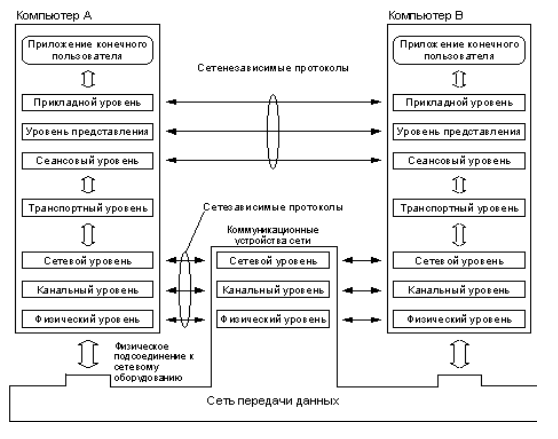
− Управление данными, которыми обмениваются прикладные процессы и  синхронизация взаимодействия прикладных процессов;

− Определение качества обслуживания (время доставки блоков данных,  допустимой частоты ошибок);

− Соглашение об исправлении ошибок и определении достоверности данных; − Согласование ограничений, накладываемых на синтаксис (наборы символов,  структура данных).

Указанные функции определяют виды сервиса, которые прикладной уровень  предоставляет прикладным процессам. Кроме этого, прикладной уровень передает  прикладным процессам сервис, предоставляемый физическим, канальным, сетевым,  транспортным, сеансовым и представительским уровнями.

На *прикладном уровне* необходимо предоставить в распоряжение пользователей уже  переработанную информацию. С этим может справиться системное и пользовательское  программное обеспечение.



**Вывод:**

Три нижних уровня (физический, канальный и сетевой) - являются сетизависимыми, то  есть протоколы этих уровней тесно связаны с технической реализацией сети, с используемым  коммуникационным оборудованием.

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.

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

Топологии сетей

Топология компьютерной сети – это из расположения компьютера в сети относительно друг друга и способ соединения их линиями связи.

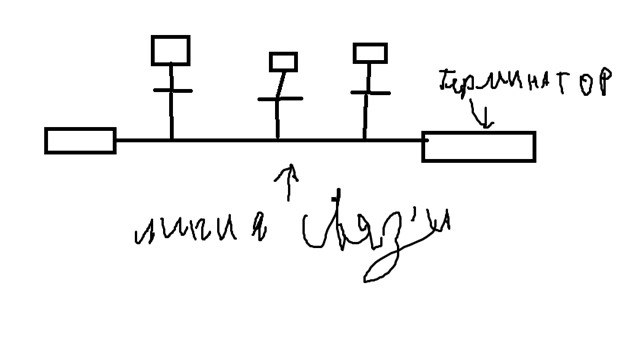
Существует три базовые топологии:

- Шина

- Кольцо

- Звезда

При топологии шина все компьютеры параллельно подключаются к одной линии связи. Информация от каждого компьютера передаётся всем остальным. На концах кабеля находятся терминаторы для предотвращения отражения сигнала.



Топология шина своей структурой предполагает идентичностью сетевого оборудования, а также равноправие всех компьютеров по доступу в сети.

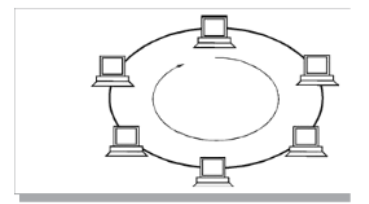
Компьютеры в шине могут передавать данные только по очереди, т.к. линия связи в этом случае единственная (полудуплексное, по очереди в два направления)

Если несколько компьютеров будут передавать информацию одновременно, то она исказится в результате наложения.

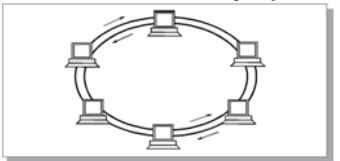
В топологии шина отсутствует явно выраженные центральный компьютер, через который передается вся информации (сервер). Это увеличивает ее надежность.

В большинстве случаев при использовании шины требуется минимальное кол-во соединительного кабеля по сравнению с другими топологиями.

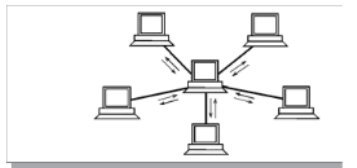
Поскольку центральный компьютер отсутствует, разрешение возможных конфликтов ложится на сетевое оборудование компьютеров. В связи с этим сетевая аппаратура при топологии шина сложнее чем при других топологиях. Важно преимущество шины состоит в том, что при отказе любого из компьютеров остальные могут нормально работать.



Кольцо — это топология, в которой каждый компьютер соединен линиями связи двумя другими: от одного он получает информацию, а другому передает. Передача информации в кольце всегда производится только в одном направлении. Важная особенность кольца состоит в том, что каждый компьютер ретранслирует (восстанавливает, усиливает) приходящий к нему сигнал, то есть выступает в роли репитер. Обрыв или короткое замыкание в любом из кабелей кольца делает работу всей сети невозможной.

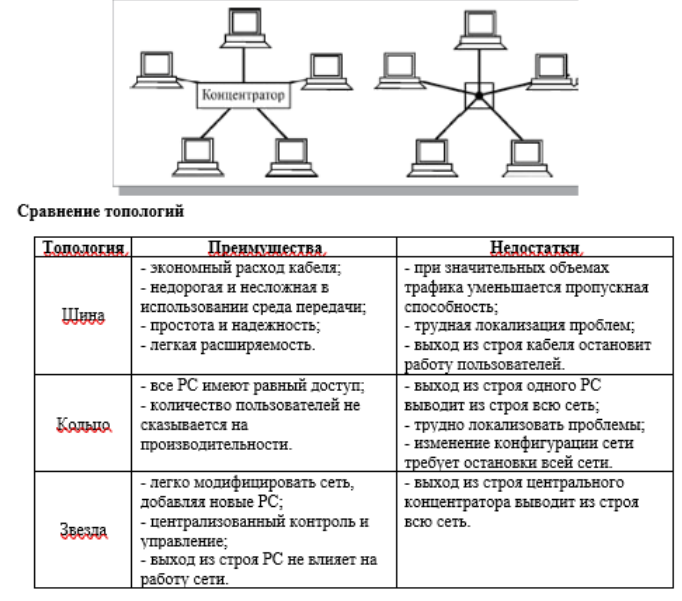


Иногда сеть с толологией кольцо выполняется на основе двух параллельных кольцевых линий связи, передающих информацию в противоположных направлениях Цель подобного решения — увеличение (в идеале — вдвое) скорости передачи информации по сети. К тому же при повреждении одного из кабелей сеть может работать с другим кабелем (правда, предельная скорость уменьшится).



Звезда — это топология сети с выделенным центром, к которому подключаются все остальные абоненты. Обмен информацией идет исключительно через центральный компьютер, на который ложится большая нагрузка, поэтому ничем другим, кроме сети, он, как правило, заниматься не может. Выход из строя периферийного компьютера или его сетевого оборудования никак не отражается на функционировании оставшейся части сети, зато любой отказ центрального компьютера делает сеть полностью неработоспособной.

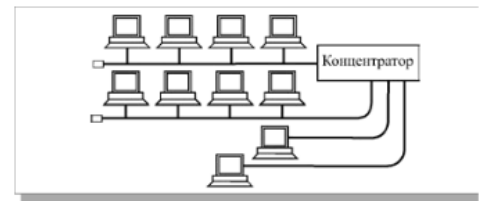
Звезда, где есть центральный компьютер называют активной звездой. Существует также топология, называемая пассивной звездой. В настоящее время она распространена гораздо более широко, чем активная звезда. В центре сети с данной топологией помещается не компьютер, а специальное устройство — концентратор.



Комбинированные топологии

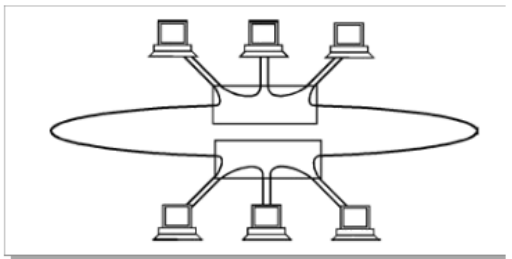
Довольно часто применяются комбинированные топологии, среди которых наиболее распространены звёздно-шинная и звёздно-кольцевая.

Звезда-шина



В звездно-шинной топологии используется комбинация шины и пассивной звезды. К концентратору подключаются как отдельные компьютеры, так и целые шинные сегменты. На самом деле реализуется физическая топология шина, включающая все компьютеры сети. В данной топологии может использоваться и несколько концентраторов, соединенных между собой и образующих так называемую магистральную, опорную шину.

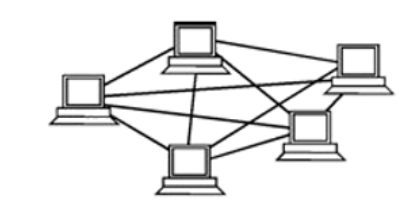
Звезда-кольцо



В случае звездно-кольцевой топологии в кольцо объединяются не сами

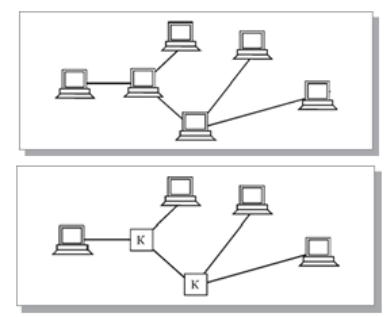
компьютеры, а специальные концентраторы, к которым, в свою очередь, подключаются компьютеры с помощью звездообразных двойных линий связи. В действительности все компьютеры сети включаются в замкнутое кольцо, так как внутри концентраторов линии связи образуют замкнутый контур.

Сотовая



Ячеистая (сотовая) топология характеризуется наличием избыточных связей между устройствами. Например, в истинной сети с сеточной структурой существует прямая связь между всеми устройствами сети. Основным достоинством сети с сотовой структурой является ее отказоустойчивость. Другие преимущества включают в себя гарантированную пропускную способность канала связи и то, что такие сети достаточно легко диагностировать. К недостаткам сотовой топологии относятся сложность инсталляции и реконфигурации, а также стоимость поддержки избыточных каналов.

Дерево



Нередко применяется также сетевая топология дерево, которую можно

рассматривать как комбинацию нескольких звезд. Причем, как и в случае звезды, дерево может быть активным пассивным. При активном дереве в центрах объединения нескольких линий связи находятся центральные компьютеры, а при пассивном — концентраторы.

Уровни сетевой архитектуры:

- Физическая топология - географическая схема расположения компьютеров и прокладки кабелей

- Логическая топология - структура связей, характер распространения сигналов по сети.

- Топология управления обменом - принцип и последовательность передачи права на захват сети между отдельными компьютерами.

- Информационная топология - направление потоков информации, передаваемой по сети.