Романенко Дмитрий Михайлович 219-1

**Основные понятия и определения**

Сеть — совокупность объектов, образуемых устройствами передачи и обработки данных. Сеть — последовательная, бит-ориентированная передача информации между связанными друг с другом независимыми устройствами.

В классическом варианте сети всегда делили по территориальному признаку на локальные (в рамках одного или нескольких близко расположенных зданий), городские, региональные, глобальные. Однако на современном этапе развития сетей и вычислительной техники наиболее логичной является следующая классификация: локальные и глобальные. При этом ключевым фактором является скорость доступа к информации. В состав сети в общем случае включаются следующие элементы: сетевые компьютеры, каналы связи, различного рода преобразователи сигналов, сетевое или коммуникационное оборудование.

Различают 2 понятия сети: коммуникационная и информационная сеть. Коммуникационная сеть предназначена для передачи данных, а так же выполняет задачи, связанные с преобразованием данных (физические преобразования). Коммуникационные сети различаются по типу используемых физических средств. Информационная сеть предназначена для хранения информации. Она так же может выполнять задачи, связанные с передачей и преобразованием информации. На базе коммуникационной сети может быть построена группа информационных сетей. Под информационной системой следует понимать объект, способный осуществлять хранение, обработку и передачу информации. В состав информационной системы входят компьютеры и ПО, предназначенное для обработки и передачи информации. В случае если информационная система предназначена для решения задача пользователя, то ее называют клиентской рабочей станцией.

Канал связи — путь либо средство, по которому передаются сигналы. Каналы связи создаются по линиям связи при помощи сетевого оборудования и физических средств соединения. Между взаимодействующими информационными системами через физические каналы коммуникационной сети и узлы коммутации накладываются так называемые логические каналы, т.е. это путь передачи данных от одной системы к другой.

Информация в сети передается блоками данных в соответствие с процедурами передачи данных. Эти процедуры называются протоколами передачи данных. Протокол — это совокупность правил, устанавливающих формат и процедуры обмена информации между 2 и более устройствами.

Загрузка сети или трафик — это поток сообщений, проходящих через сеть. В количественном измерении это число проходящих блоков данных и их длинны в единицу времени. На характеристики сети существенное влияние окажут метод доступа и топология сети. ФПод методом доступа понимают способ определения того, какой компьютер следующим сможет использовать сеть для передачи информации. Топология — это описание физических соединений в сети, указывающее, какие рабочие станции связываются друг с другом. Тип топологии оказывает существенное влияние на производительность и надежность работы как отдельных рабочих станций, так и всей сети в целом.

Архитектура — это концепция, определяющая взаимосвязь, структуру и функции взаимодействующих рабочих станций в сети. 3 основные архитектуры: терминал-главный компьютер, клиент-серверная архитектура, одноранговая.

Другой отрицательной особенностью является то что при включении нового узла в сеть оно должно быть разомкнуто (кольцо) а следовательно временно прекращена работа сети. Положительным моментом является то что кольцо характеризуется предсказуемым временем отклика. В чистом виде кольцевая топология используется редко часто выполняет транспортную функцию что будет рассмотрено в звездно-кольцевой технологии. Выделяют так же в виде отдельных топологий топологию цепочка, которая представляет собой общую шину но с токен-ринговским оборудованием и принципами работы кольца.

Звезда – топология при которой все узлы присоединены к концентратору (центральный узел) который устанавливает, поддерживает и разрывает связи между рабочими станциями.

Коммутатор – последовательный доступ, концентратор – широковещательный доступ.

Особенности:

1. Простота настройки
2. Простота поиска неисправностей
3. Простота расширения сети за счет расширения количества узлов
4. Выход из строя центрального узла выводит их строя всю сеть
5. Высокий расход кабеля
6. Безопасность сети зависит от безопасности центрального устройства

Существует разветвленная звезда – это топология в которой несколько отдельных звезд соединены вместе через концентратор.

Ограничение по масштабируемости в силу правила: информация передаваемая между самыми удаленными узлами должна проходить не более 4 точек коммутации.

Древовидные топологии

Под древовидными топологиями понимают комбинацию нескольких звезд но при этом должен существовать корень дерева. Дерево может быть активным (истинным). В активном дереве в связующих точках расположены компьютеры. Пассивное дерево – в центре объединения нескольких линий расположены концентраторы либо хабы. Древовидные топологии характеризуются теми же особенностями что и звезда. Выделяют так же утолщенное дерево, которое является упрощенной схемой построения суперкомпьютеров.

Особенности:

1. Расширение пропускной способности канала (минимум удвоение) при переходе от уровня к уровню по мере движения от листьев к корню.

Вилы комбинированных топологий: наиболее распространенные: звездно-шинная и звездно-кольцевая.

Звездно-шинная – к центральному узлу присоединяются не только отдельные компьютеры но и целые шинные сегменты. Данная топология применяется при модернизации существующей сети с шинными сегментами в целях экономии финансовых средств. В данной топологии может быть несколько концентраторов, соединенных между собой и образующих так называемую магистральную или опорную шину. К каждому из концентраторов подключаются шинные сегменты. Это принято называть звездно-шинным деревом.

Звездно-кольцевая топология – это топология в которой в кольцо объединяются не сами компьютеры а концентраторы а компьютеры подключаются к ним через двойные звездообразные линии связи. За счет линии связи компьютеры объединяются в одно единое кольцо но при этом за счет использования концентратора легко изменять число узлов в сети, легко искать неисправности.

Ячеистая (сеточная) топологии. В них компьютеры связываются друг с другом не одной а несколькими линиями связи, причем выделяют полную и частичную сеточные топологии, которые имеют следующий вид. Компьютер должен быть соединен напрямую со всеми остальными компьютерами напрямую. В частичной напрямую соединяются только самые активные и наиболее важные компьютеры.

**Методы доступа в компьютерных сетях**

4 метода доступа:

1. CSDM/CD – Carrier Sense Multiple Access with Collision Detection

Прослушивание несущей и разрешение коллизий. При необходимости информация (пакет) после того как он готов сетевой адаптер прослушивает сеть и если она свободна начинает передачу. В процессе передачи адаптер продолжает прослушивать сеть на предмет возникновения коллизий (столкновение 2х и более пакетов). Коллизию может обнаружить и другая рабочая станция. В данном случае в сеть отправляется специальный управляющий сигнал, который говорит что все должны прекратить передачу и сбросить принятые но не обработанные пакеты. Далее все рабочие станции замолкают, при этом станции запрограммированы на различные псевдослучайные промежутки времени, через которые вернется в сеть. Далее после возвращения операция повторяется. Предполагается что возвращение произойдет через различные промежутки времени но в случае возникновения сразу же повторной коллизии время молчания увеличивается, тем самым снижая вероятность фактически одновременного возвращения в сеть нескольких рабочих станций. Применяется этот метод с применением арбитражного элемента. В классической реализации в связи с случайностью передачи невозможно гарантировать 2м узлам непрерывную передачу пакетов.

1. TPMA

Множественный доступ с передачей полномочий. Метод доступа предполагает что от рабочей станции другой рабочей станции последовательно передается маркер или токен которые передает право на передач сообщений (в реальности токен – комбинация бит). При получении маркера рабочая станция может присоединить к нему необходимую информацию для передачи. Далее он переносит это сообщение по сети. Каждая рабочая станция последовательно принимает его, проверяет адресную информацию и либо отправляет дальше в сеть либо приступает к обработке. Здесь возможны 2 варианта: если сеть работает без подтверждения передачи, то рабочая станция получившая пакет создает новый пустой маркер и отправляет его в сеть. Если же требуется подтверждение то пакет проверяется, после проверки к маркеру присоединяется новый маркер что все корректно либо сообщение о необходимости повторной передачи и отправляется отправителю. В данной ситуации другие рабочие станции не смогут передавать данные пока предыдущие не закончат работу.

1. TDMA

Множественный доступ с разделением во времени. Данный метод доступа основан на распределении времени работы канала между системами и рабочими станциями. Предполагает использование специального устройства – элементарного генератора, который делит время работы канала на повторяющиеся циклы. Каждый цикл начинается с разграничителя. Цикл включает в себя n-пронумерованных интервалов называемых ячейками. Возможны 2 варианта работы: 1) число ячеек равно числу рабочих станций, за каждой рабочей станцией закрепляется определенная ячейка.

1. FDMA

Множественный доступ с разделением частоты

**Принципы передачи информации по сети**

Вся информация по сети всегда передается в виде отдельных фрагментов либо блоков, которые называют пакетами (кадрами, фреймами). Причем предельная длинна этих пакетов ограничена как со стороны минимального, так и максимального значения. Нижняя граница обычно находится в пределах нескольких десятках байт. Верхняя – несколько килобайт. Важнейшим параметром любой сети будет являться время доступа, которое определяется как временной интервал между моментом готовности абонента к передачи и моментом начала передачи. Что бы данный параметр находился в разумных пределах необходимо установить оптимальный размер пакета. Любая сеть характеризуется какой-либо вероятностью появления ошибок (10 в -5 степени для витой пары) использование пакетов слишком большой длинны может привести к возникновению в передаваемых пакетов неисправляемых ошибок, а соответственно потребуется повтор передачи. К тому же если допустить передачу в одном пакете ГБ информации то возможет «монопольный» захват сети. С минимальной стороны размер пакета так же надо ограничивать, т.к. для обеспечения эффективности использования пропускной способности канала в пакете доля служебной информации должна быть минимальной (служебная информация – адреса, маршрут передачи, номер пакета и т.д.) В итоге существует оптимальная длинна пакета, но она не может быть постоянной. Она зависит от уровня помех, метода управления обменом информации, характер информации, количество узлов в сети. В целом процесс обмена информации представляет собой чередование пакетов между несколькими клиентами.

Структура пакетов для каждой сети определены стандартом на данную сеть и связана прежде всего с аппаратными особенностями сети, выбранной топологией, типом среды передачи, протоколами обмена информации. При этом можно выделить общие принципы формирования структуры пакета, которые подходят для любых сетей. Пакет чаще всего содержит следующие 7 полей: преамбула, идентификатор приемника, идентификатор передатчика, управляющая информация, данные, контрольная сумма, стоповая комбинация.

Преамбула – обеспечивает предварительную настройку аппаратуры адаптера или другого сетевого устройства на прием и обработку пакета. Это поле может отсутствовать либо сводится к 1 единственному биту.

Идентификатор приемника – это фактически сетевой адрес принимающего узла. Он может быть индивидуальным, групповым или широковещательным. Именно он позволяет принимающей стороне определить, ему ли адресован пакет.

Идентификатор передатчика (отправителя) – это сетевой адрес отправителя пакета (обязательно индивидуальный). Этот адрес необходим прежде всего в 2 ситуациях:

А) работает передача с подтверждением

Б) узел получает пакеты от нескольких узлов – адрес отправителя в данной ситуации позволит корректно отсортировать пакеты.

Управляющая информация – информации, которая содержит в себе тип пакета, номер пакета, размер, формат, маршрут его доставки, способ преобразования пакета и т.д.

Данные – сама информация, может иметь переменную длину, имеются так называемые управляющие пакеты когда поле данных будет отсутствовать.

Контрольная сумма – числовой код, формируемых передатчиком по определенным правилам и содержащий в свернутом виде информацию о самом пакете или о данных.

Стоповая комбинация служит для информировании принимающей стороны об окончании пакета. Так же обеспечивает выход аппаратуры из состояния приема.

В процессе обмена информацией между принимающей и передающей стороной происходит обмен как информационными так и управляющими пакетами по определенным правилам (протоколам). В общем виде протокол обмена можно представить в виде схемы.

При реальном обмене данными используются либо многоуровневые протоколы либо набор протоколов именуемых стеком. Каждый из уровней (протокол) предполагает свою структуру пакета, а поэтому все пакеты более высоких уровней последовательно вкладываются в пакеты нижележащих уровней а именно в поле данные тем самым образуя многоуровневую структуру которая называется многоуровневой системой вложения пакетов. Обратный процесс называется декапсуляцией.

Базовая модель связи открытых систем OSI:

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

Интерфейс пользователя, прикладной интерфейс, область взаимодействия, физические средства соединения.

Каждый уровень модели OSI выполняет определенные задачи. В целом можно говорить, что модель OSI задает логику построения протоколов и их стеков, организующих и управляющих передачей. Если приложение может взять на себя функции некоторых верхних уровней (не больше 3х) то для обмена данными оно обращается напрямую к системным средствам, выполняющих функции оставшихся уровней модели OSI. Выделяют 2 варианта взаимодействия в рамках модели OSI, построенные на базе протоколов обеспечивающих механизм взаимодействия программных процессов на разных узлах (машинах); вертикальное – обеспечивается взаимодействием уровней на одной машине. Горизонтальное взаимодействие предполагает организацию логической или виртуальной связи, когда каждый уровень компьютера отправителя с соответствующим уровнем компьютера получателя как будто он связан напрямую. При реальной передаче информация на компьютере должна пройти через все уровни (с нижележащих) далее передается непосредственно по сети, а на компьютере получателя проходит все уровни но в обратной последовательности пока не дойдет до уровня с которого было отправлено. При подготовке информации для передачи по сети данные разбиваются на фрагменты, которые последовательно проходят через все уровни на них добавляется какая то служебная информация (возможно выполняются преобразования). Чем выше уровень тем более сложные задачи он решает. Отдельные уровни модели OSI в целом удобно рассматривать как группы программ предназначенных для выполнения конкретных функций. Обычно говорят что каждый уровень обеспечивает сервис для вышестоящего уровня, в свою очередь запрашивая сервис у нижестоящего. Отметим что верхние 3 уровня как правило запрашивают один и тот же сервис, а именно передача данных из одной сети в другую. Практическая реализация передачи данных и тд возложена на нижние уровни. Модель OSI для взаимодействия разнородных систем выполняет координирующие действия по:

1. Взаимодействию прикладных процессов
2. По формам представления данных
3. По единообразному хранению данных
4. По управлению сетевыми ресурсами
5. По безопасности данных и защите информации
6. По диагностики программы технических средств соединения

**Прикладной уровень**

Прикладной уровень обеспечивает прикладным процессам средство доступа к области взаимодействия. В действительности прикладной уровень это набор разнообразных протоколов, с помощью которых пользователь получает доступ к разделяемым ресурсам (например файлы и папки, контекстовые страницы) а так же организует совместную работу например по средствам электронной почты. Фактически можно говорить что одной из ключевых задач этого уровня является определение того как должен обрабатываться запрос от прикладной программы. Другой важнейшей задачей является установление (если это необходимо) различных параметров взаимодействия системы. Обычно говорят что прикладной уровень оперирует сообщениями (понятия пакета пока не может быть). Прикладной уровень может выполнять следующие функции:

1. Описание форм и методов взаимодействия прикладных процессов
2. Выполнение различных видов работ (управление заданиями, управление системой)
3. Идентификация пользователей по паролям, адресам, электронным подписям
4. Определение функционирующих абонентов и возможности доступа к новым прикладным процессам
5. Определение достаточности имеющихся ресурсов
6. Организация запросов на соединение с другими прикладными процессами
7. Передача заявок представительному уровни на необходимые методы описания информации
8. Выбор процедур планируемого диалога процессов
9. Управление данными, которыми обмениваются прикладные процессы и синхронизация взаимодействия прикладных процессов
10. Определение качества обслуживания (время доставки блоков данных, допустимой частоты ошибок)
11. Соглашение об исправлении ошибок и определении достоверности данных
12. Согласование ограничений, накладываемых на синтаксис (наборы символов, структура данных)

Перечисленные функции определяют виды сервиса которые прикладной уровень представляет прикладным процессам. Кроме того прикладной уровень им так же передает сервисы, предоставляемыми всеми остальными 6 уровнями модели OSI. Очень часто говорят, что прикладной уровень фактически должен предоставить в распоряжение пользователя информации в необходимом для них виде. К числу наиболее распространенных протоколов, которые выполняют функции прикладного уровня относятся: FTP, TFTP, X400 (email), telnet, SMTP, CMIP (info management), SLIP (IP for lines), SNMP (network management), FTAM (file transfer management), HTTP

**Уровень представления**

Данный уровень обеспечивает то что информация передаваемая прикладным уровнем будет понятным уровню представления другой машины. В случае необходимости представительский уровень выполняет преобразование форматов данных в некий общий формат представления, который будет понятен всем. Прикладные уровни разнородных машин преодолевают различия например синтаксические и другие. За основу общего представления данных положено единое для всех уровней модели система ASN.1. Это система служит для описания структуры файлов а так же может решать проблему безопасности через шифрование данных. Примером протокола выполняющего последнюю функцию является протокол SSL (secure socket layer) который обеспечивает секретный обмен сообщениями. Этот уровень так же при необходимости обеспечивает преобразование данных идущих с прикладного уровня в понятный для транспортного уровня формат. В целом прикладной уровень может выполнять следующие 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 реализующие функции уровня представления данных как правило являются составной частью протоколов 3х верхних уровней.

**Сеансовый уровень**

Это уровень, определяющий процедуру представления сеансов между пользователями или прикладными процессами (на разных машинах). Сеансовый уровень обеспечивает управление диалогом для того что бы фиксировать какая из сторон является активной в данный момент а так же предоставляет средство синхронизации. Последнее позволять контрольные точки в длинные передачи, что бы в случае отказа можно было вернуться к последней контрольной точке а не начинать передачу заново. Фактически можно сказать что сеансовый уровень координирует (управляет) прием, передачу и выдачу информации одного сеанса связи. Кроме того может содержать дополнительно функции управления диалогом, синхронизации и отмены связи в сеансе в следствие появления ошибок, координировать связь между 2мя прикладными программами, работающими на двух виртуальных машинах. Именно на сеансовом уровне определяется какой будет передача между 2мя прикладными процессами: полудуплексная (по очереди передача и прием) или дуплексная (одновременно передача и прием). При использовании полудуплексной передачи прикладному процессу выдается маркер, являющимся правом на передачу.

Сеансовый уровень выполняет следующие функции:

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

**Транспортный уровень**

Транспортный уровень предназначен для формирования и управлением передачи пакетов через коммуникационную сеть (каналы связи). Именно на транспортном уровне информация будет разбиваться на блоки к которым начнет добавляться служебная информация. Пакеты могут быть искажены, утеряны, хотя некоторые средства имеют средства обработки ошибок. В задачу транспортного уровня входят обеспечение требуемого уровня надежности передачи информации. Модель OSI выделяет 5 классов сервиса, предоставляемых транспортным уровнем. Эти виды сервиса отличаются качеством предоставляемых услуг, возможность восстановления прерванной связи, наличие средств мультиплексирования нескольких соединений, способности обнаружения и исправления ошибок, потеря пакета, дублирование. Считается что транспортный уровень гарантирует доставку блоков данных. Транспортный уровень контролирует очередность прохождения пакетов.

Функции транспортного уровня:

1. Управление передачей по сети в обеспечение целостности пакетов данных
2. Обнаружение ошибок, частичная их ликвидация (за счет использования избыточных кодов) и сообщение о неисправляемых ошибках
3. Восстановление передачи после отказа и неисправностей
4. Укрупнение пакетов или разделение данных по пакетам
5. Предоставление приоритетов при передаче пакетов (нормальная или срочная)
6. Подтверждение передачи
7. Ликвидация пакетов при тупиковых ситуациях в сети

Начиная с транспортного уровня все вышележащие протоколы реализуются программными средствами, обычно включаемыми в состав сетевой ОС.

**Сетевой уровень**

Сетевой уровень устанавливает сетевые соединения между 2мя системами и обеспечивает прокладку виртуальных каналов между ними. Кроме того сетевой уровень сообщает транспортному уровню о появившихся ошибках. Сообщение сетевого уровня принято называть пакетами, в них он главным образом добавляет адресную или маршрутную информацию. Прокладка наилучшего пути для передачи данных называется маршрутизацией и ее решением является главной задачей сетевого уровня. Эта проблема осложняется тем, что короткий путь не всегда наилучший. При выборе маршрута основным критерием является время доставки, оно зависит от пропускной способности канала, от интенсивности трафика, который носит динамический характер. Алгоритмы маршрутизации должны приспосабливаться к изменению нагрузки. Вывод маршрута так же может осуществляться исходя из других критериев, например надежности.

В целом считается что сетевой уровень регулирует доставку данных между сетями, а внутри сети данным процессом занимается канальный уровень. При этом при доставке данных между сетями важным является корректное определение номера сети и адреса получателя в этой сети. Сети как правило связываются между собой такими устройствами как маршрутизаторы. Они собирают информацию о топологии сетевых соединений, о состоянии сети и с учетом этого пересылают пакет в сеть назначения. Фактически маршрут следования пакета будет представлять собой последовательность маршрутизаторов, на котором он будет проходить. Прохождение одного маршрутизатора часто называют «транзитными передачами между системами» - haps. При этом любой из маршрутизаторов вправе изменить маршрут следования. Сетевой уровень так же отвечает за деление пользователей на группы (для групповых адресов) и за преобразование физических адресов (MAC – адреса) в сетевые (IP). Таким образом сетевой уровень выполняет функции:

1. Создание сетевых соединений и идентификация их портов
2. Обнаружение и исправление ошибок адресации, возникающих при передаче через коммуникационную сеть
3. Управление потоками пакетов (по адресам через определение маршрута). Упорядочение последовательности пакетов
4. Маршрутизация и коммутация

Наиболее часто используемые протоколы на сетевом уровне: IP, IPX (протокол межсетевого обмена пакетов, предназначенных для адресации и маршрутизации пакетов в сетях Novell), Х.25 (международный стандарт для глобальных коммуникаций с коммутацией.

Виды протоколов сетевого уровня:

1. Определение 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)
2. Протоколы обмена маршрутной информацией – маршрутизаторы собирают информацию о топологии межсетевых соединений, о состоянии сети, траффике в определенных точках сети, каналах и т.д. Пример протокола: Internet Control Management Protocol (ICMP), IPSecurity (IPSec) – данный протокол является составной частью IP протокола 6 версии, может быть использован с IP 4 версии.

**Канальный уровень**

Единицей информации канального уровня являются кадры. Кадр – логически организованная структура, в которую помещаются пакеты, выходящие из сетевого уровня. Это необходимо что бы учесть особенности используемых технологий и параметров среды передачи данных. Считается что одной из важнейших задач канального уровня является контроль за состоянием среды передачи данных и ее доступностью (логический контроль). Другой важнейшей задачей канального уровня является реализация механизмов обнаружения и исправления ошибок за счет использования различного рода контрольных сумм. Главная задача канального уровня – брать пакеты, поступающие с сетевого уровня и готовить их к передаче, укладывая в кадр соответствующего размера.

Канальный уровень делится на 2 подуровня:

1. LLC – управление логическим каналом осуществляет логический контроль связи. Подуровень LLC обеспечивает обслуживание сетевого уровня и связан с передачей и приемом пользовательских сообщений.
2. MAC – контроль доступа к среде. Подуровень MAC регулирует доступ к разделяемой физической среде и управляет доступом к каналу связи.

В локальных сетях используются коммутаторы, мосты, маршрутизаторы и сами компьютеры. В компьютерах функции канального уровня выполняются сетевыми адаптерами а если точнее, то их драйверами. В целом могут выполнятся следующие функции:

1. Организация канальных соединений и идентификация их портов
2. Организация и передача кадров на физический уровень
3. Обнаружение и исправление ошибок
4. Управление потоками данных
5. Обеспечение прозрачности логических каналов

Наиболее используемые протоколы и технологии на канальном уровне:

1. HDLC – протокол управления каналом передачи данных высокого уровня, для последовательных соединений.
2. Ethernet сетевая технология по стандарту IEEE 802.3 для сетей, использующая шинную топологию и коллективных доступ с прослушиванием несущей и обнаружением конфликтов.
3. Token Ring сетевая технология по стандарту IEEE 802.5, использующая кольцевую топологию и метод доступа к кольцу с передачей маркера.
4. FDDI – сетевая технология по стандарту IEEE 802.6, использующая оптоволоконный носитель
5. X.25 – международный стандарт для глобальных коммуникаций с коммутацией пакетов

**Физический уровень**

Сопряжение с физическими средствами соединения (совокупность физической среды, аппаратных и программных средств, обеспечивающая передачу сигналов между системами). Фактически физический уровень состоит из подуровня стыковки со средой и подуровня преобразования передачи. Первый из них обеспечивает сопряжение потока данных с используемым физическим каналом связи, а второй осуществляет преобразования связанные с применяемыми преобразованиями. Наиболее типичными преобразованиями является кодирование сигнала (под ним подразумевают преобразования бинарного потока (0 или 1) в физические сигналы нужного вида). Фактически можно говорить, что физический уровень обеспечивает физический интерфейс с каналом передачи данных, а так же описывает процедуры передачи сигнала, отправка и получение сигнала.

Функции физического канала:

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 (столкновение пакетов и кадров – коллизия, обрыв канала, отключение питания устройств, потеря механического контакта)

**Протоколы и стеки протоколов**

Спецификации стандартов

Спецификации IEEE802 определяют стандарты и правила использования физических компонентов сети. Эти компоненты будут относиться к физическому и канальному уровню модели OSI. Данные спецификации определяют механизм доступа адаптера к каналу связи, а так же механизмы передачи данных и управления ими. В соответствии с данными стандартами канальный уровень подразделяется на LLC и MAC.

Протоколы и стеки протоколов

Под стеком протоколов понимается согласованный набор протоколов разных уровней, достаточных для организации межсетевого взаимодействия. При этом для каждого уровня определяется набор функций-запросов, для взаимодействия с вышележащим уровнем, называемый интерфейсом. Существует несколько стеков протоколов, которые могут быть использованы в сетях. Наиболее популярные TCP/IP, IPX/SPX. В общем случае протоколы делят на сетевые, транспортные и прикладные. Сетевые протоколы решают вопросы с адресацией и маршрутизацией информации, контроль соответствующих ошибок, установление и контроль за правилами взаимодействия в сетевой среде. Транспортные протоколы предоставляют услуги по надежной транспортировке данных между компьютерами, а прикладные протоколы фактически отвечают за взаимодействие приложений.

Стек OSI – набор конкретных спецификаций протоколов, поддерживается американским правительством в рамках программ GOSIP. Фактически он полностью повторяет 7 уровней модели OSI.

Стек TCP/IP – с точки зрения системной архитектуры данный стек соответствует эталонной модели OSI с той разницей что функции нескольких уровней объединяются в один. Данный стек позволяет обмениваться данными по сети приложениям и службам, работающим практически на любой платформе.

Протокол IP обеспечивает обмен пакетами между узлами сети и является протоколом не устанавливающим соединения. Не ожидает подтверждения передачи. Основные функции:

1. При необходимости фрагментация пакетов (так же управляющая информация для этой операции)
2. Межсетевая адресация

В сетях используется 3 типа адресов: физический (MAC), сетевой (IP), символьный (DNS, NetBIOS).

**ФИЗИЧЕСКИЙ АДРЕС**

Международная организация по стандартизации предложила присваивать уникальный физический адрес каждому сетевому адаптеру на стадии производства. Количество адресов должно быть достаточно большим что бы удовлетворять потребностям. Поэтому был выбран 48-битный формат, который имеет структуру: I/G, U/L, OUI (идентификатор), OUA (сетевой адрес).

1. Младшие 24 разряда – организационно уникальный адрес – присваивают самим производителем адаптеров. Возможно порядка 16 млн. адресов.
2. Следующие 22 бита называемые организационно уникальными идентификаторами.

Суммарно эти 46 бит (идентификатор и адрес) – универсально управляющий адрес

1. Старшие 2 бита определяют тип адреса и способ его интерпретации. На тип адрес указывает старший бит IG (0 – индивидуальный, 1 – групповой). Второй бит UL (если он установлен в 0, то это означает, что последующие 46 разрядов устанавливаются непосредственно производителем адаптеров, если 1 – то адаптер считается адаптером локального использования и 46 бит устанавливает организация, использующая данный адаптер, данный адаптер не может быть использован в глобальных сетях). Для широковещательной передачи используется MAC адрес, который реально не существует: все 48 бит единицы. В реальности MAC адрес представлен в 16-тиричной системе счисления

IP 4 представляет собой 32 разрядное число, разбитое на 4х8 бит. В десятичной форме представление адреса каждый из октетов переводится в десятичное число. IP – адрес состоит из 2 логических частей: идентификатор сети (правила записи: в IP адресе 16 разрядов определяют номер сети, для того что бы записать Network ID необходимо оставшиеся разряды, отвечающие за номер узла заполнить нулями и перевести в десятичную систему), идентификатор хоста (оставшиеся 2 разряда). Существует способы определения разрядов:

1. По принципу классов – выделяют 5 классов
2. С помощью масок – если IP адрес, и имеется маска в какой-либо форме, то работает этот принцип, в другом случае принцип классов.

Определение идентификатора сети и хоста

IP: 192.168.89.16

Mask: 255.255.192.0

Переводим в двоичный код

Делаем операцию AND (где обе единицы – 1; остальное – 0)