

# Сетевые модели. Понятие открытой системы

Компьютерной сетью предоставляются **сервисы**  
*по совместному использованию ресурсов:*

- файлов
- принтеров
- модемов
- факсов
- баз данных, знаний
- серверов приложений и др.

# Понятие открытой системы

Теоретическую основу современных информационных сетей определяет **Базовая эталонная модель Международной организации стандартов (ISO) – стандарт ISO 7498.**

ЭТО основа методологии организации взаимодействия прикладных процессов выполняемых в различных узлах сети.

**Определение.** Систему, удовлетворяющую требованиям стандартов ISO, именуют открытой системой

# Модель Взаимосвязи Открытых Систем - (Open System Interconnection - OSI)

При разработке стандарта эталонной модели ВОС решались следующие задачи:

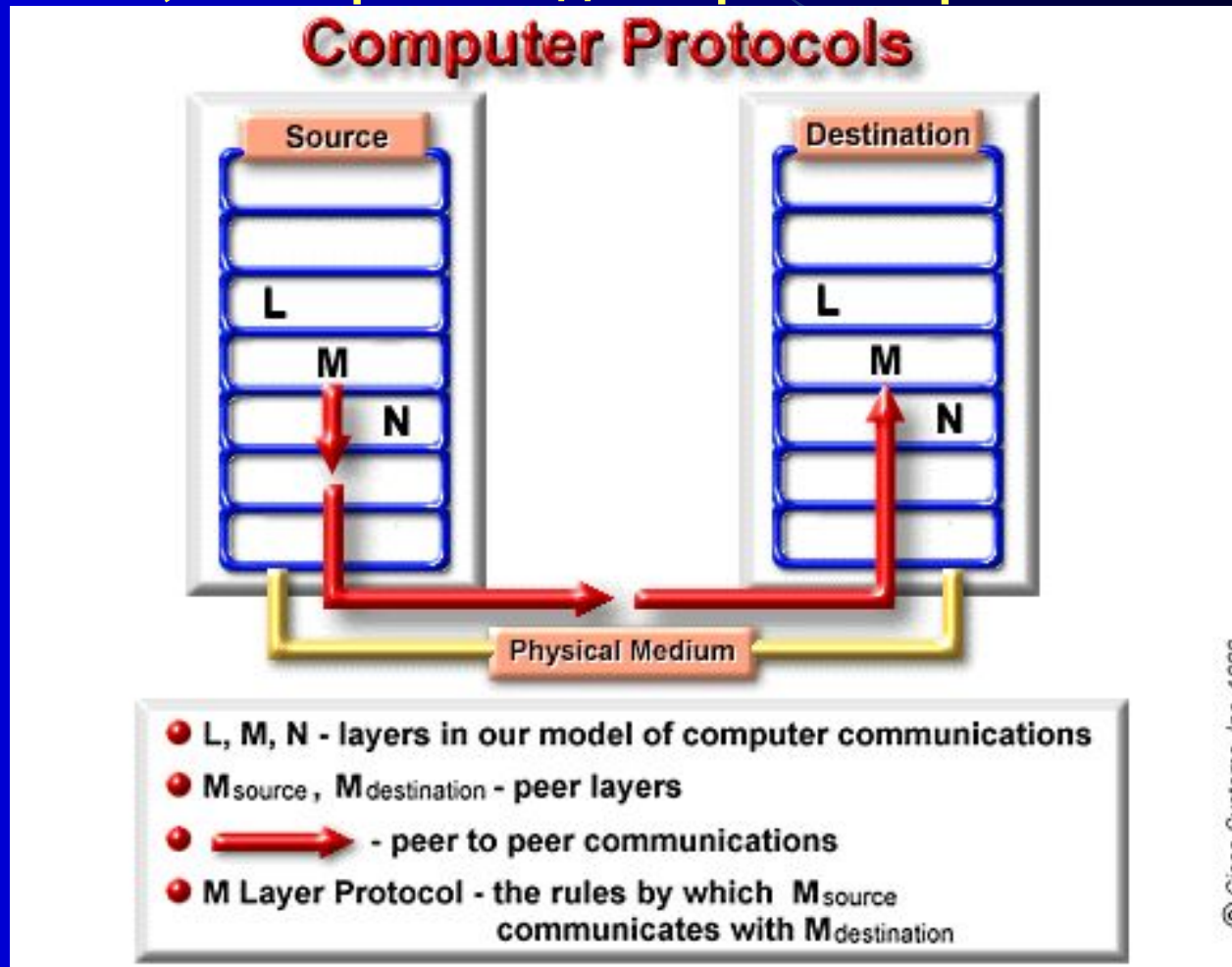
- унификация описания работы отдельной системы;
- определение интерфейса для обмена информацией между системами;
- унификация обмена данными между различными информационными системами;
- устранение технических препятствий для связи систем

Все задачи, которые необходимо решить для организации взаимодействия между объектами информационной системы, разделены на семь отдельных процедур или уровней

# Модель OSI - принцип слоистой архитектуры

В модели OSI несколько уровней протоколов образует так называемый

стек протоколов, в котором каждый протокол работает на своем уровне



# Уровневая структура модели OSI

- все функции сети разделены на уровни, которые взаимодействуют между собой через интерфейс
- такая слоистая структура позволяет модифицировать и даже заменять любой уровень, не затрагивая все остальные
- деление на уровни дает возможность максимально упростить реализацию каждого из них.

## Why a Layered Network Model?



- Reduces complexity
- Standardizes interfaces
- Facilitates modular engineering
- Ensures interoperable technology
- Accelerates evolution
- Simplifies teaching and learning

# АКТИВНОСТИ

**Назначение каждого уровня - обеспечить надлежащий сервис для вышележащего слоя.**

- **Активные элементы уровня называют активностями**
- **Активности могут быть программными и аппаратными**
- **Активности одного и того же уровня на разных машинах называются равнозначными активностями**
- **Активности уровня  $n+1$  являются пользователями сервиса, создаваемого активностями уровня  $n$ , которые называются поставщиками сервиса**

**Доступ к сервису осуществляется через точки доступа к сервису - SAPs (service access points)**

**Каждая точка доступа к сервису имеет уникальный адрес**

# СЕРВИС С СОЕДИНЕНИЕМ И БЕЗ СОЕДИНЕНИЯ

**Сервис с соединением** предполагает, что между получателем и отправителем сначала устанавливается соединение, и только потом доставляется сервис. Пример - телефонная сеть.

**Сервис без соединения** действует подобно почтовой службе. Каждое сообщение имеет адрес получателя. В надлежащих точках оно маршрутизируется по нужному маршруту. Независимо от других сообщений. При таком сервисе вполне возможно, что сообщение позже посланное придет раньше

Формально сервис определяется набором примитивных операций (или примитивов), с помощью которых пользователь или какая-либо активность получала доступ к сервису



# ПРОТОКОЛ И ИНТЕРФЕЙС

**Протокол** - это набор формализованных правил, определяющих последовательность, формат и назначение сообщений (пакетов, кадров), которыми обмениваются равнозначные активности (сетевые компоненты), лежащие на одном уровне, но в разных узлах сети.

Иерархически организованный набор протоколов, достаточный для взаимодействия устройств в сети, называется **стеком коммуникационных протоколов**.

Модули, реализующие протоколы соседних уровней и находящиеся в одном узле, также взаимодействуют друг с другом в соответствии с четкими правилами и с использованием стандартизованных форматов сообщений. Эти правила называют **интерфейсом**.

# ИНТЕРФЕЙС И ПРОТОКОЛ

Понятия "интерфейс" и "протокол", в сущности, обозначают одно и то же, а именно - **формализовано заданные процедуры взаимодействия компонент, решающих задачу связи компьютеров в сети.**

Понятие "протокол" чаще применяют при описании правил взаимодействия компонент одного уровня, расположенных на **разных узлах сети**

Понятие "интерфейс" - при описании правил взаимодействия компонентов соседних уровней, расположенных в пределах **одного узла.**

# СЕРВИС И ИНТЕФЕЙС

**Сервис** - это набор примитивов, который уровень предоставляет уровню над ним. Сервис определяет, какие операции данный уровень предназначен выполнить по поручению его пользователей, но он ничего не говорит о том, как эти операции реализованы.

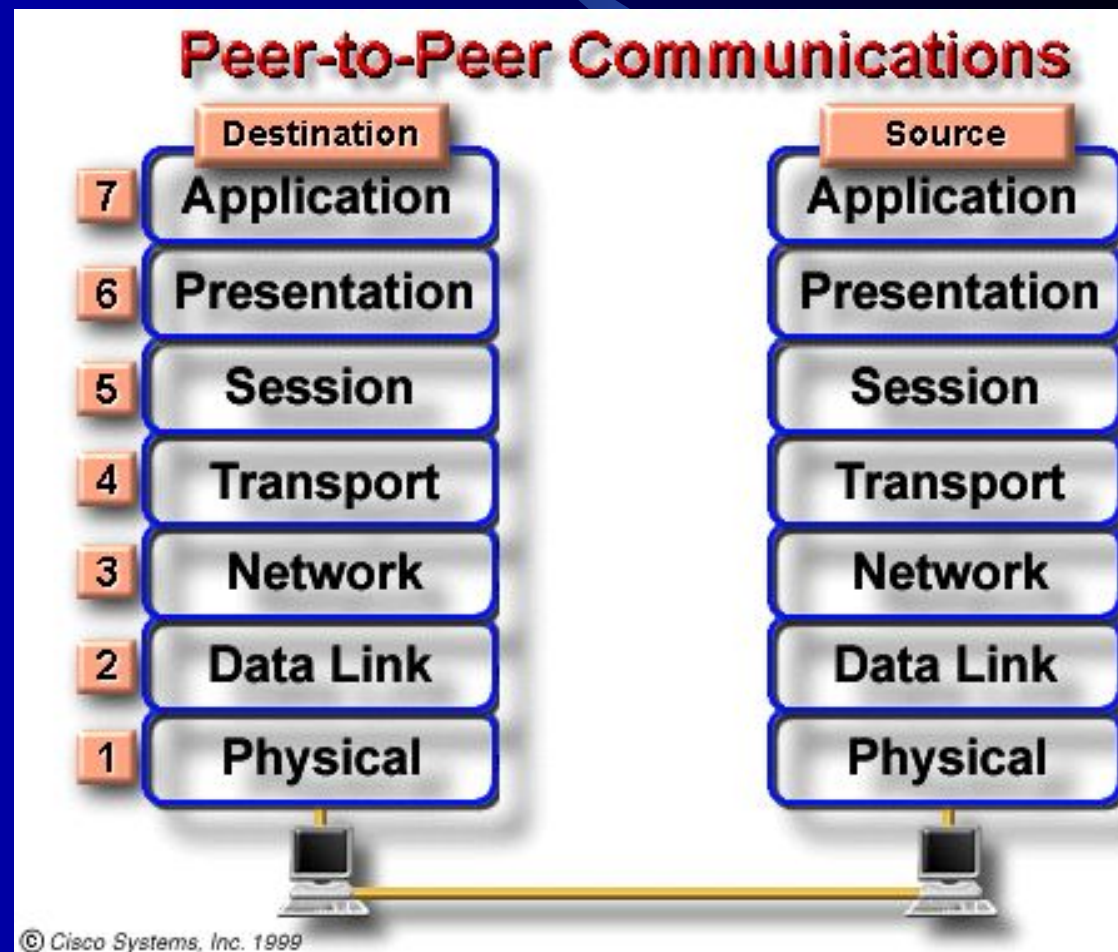
Сервис относится к **интерфейсу** между уровнями.

Нижележащий уровень является поставщиком сервиса, а вышележащий - пользователем услуг.

Определенный набор функций, выполняемых данным уровнем для выше лежащего уровня, а также форматы сообщений, которыми обмениваются два соседних уровня в ходе своего взаимодействия, называется **интерфейсом**

# ОПРЕДЕЛЕНИЕ МОДЕЛИ OSI

Семиуровневая Модель OSI - форма описания информационной системы, ее структуры, входящих в нее компонентов, а также правил и процедур взаимодействия элементов информационной системы в процессе работы



## Четыре базовых элемента:

- открытые системы;
- объекты (активности) уровня;
- соединения, связывающие объекты и позволяющие им обмениваться информацией;
- физические средства соединения.

Прикладной процесс благодаря существованию функций семи уровней получает разнообразные виды сервиса. Наряду с этим, уровневая организация обеспечивает относительную независимость подсистем

**Документы ISO для каждого уровня определяют:**

- **назначение уровня;**
- **сервис, предоставляемый данным уровнем расположенному над ним уровню;**
- **функции, выполняемые уровнем**
- **сервис, получаемый от расположенного под ним уровня.**

# APPLICATION LAYER - ПРИКЛАДНОЙ УРОВЕНЬ

Прикладной уровень обеспечивает доступ прикладных процессов пользователей к ресурсам и сервису информационной системы (сети).

Это могут быть программы, обеспечивающие:

- прием или передачу файлов
- управление работой сети
- доступ к базе данных
- передачу почтовых сообщений

Главная задача этого уровня - обеспечить удобный интерфейс для пользователя.

Примеры протоколов прикладного уровня: NCP - протоколы ядра в ОС Novell NetWare, SMB MS Windows NT, FTP и TFTP (Стек TCP/IP), X.400, NFS (UNIX).

На этом единица данных рассматривается как **Сообщение** (Message).

# PRESENTATION LAYER - УРОВЕНЬ ПРЕДСТАВЛЕНИЯ ДАННЫХ

Уровень представления определяет формат, используемый для обмена данными между узлами сети (его можно назвать переводчиком).

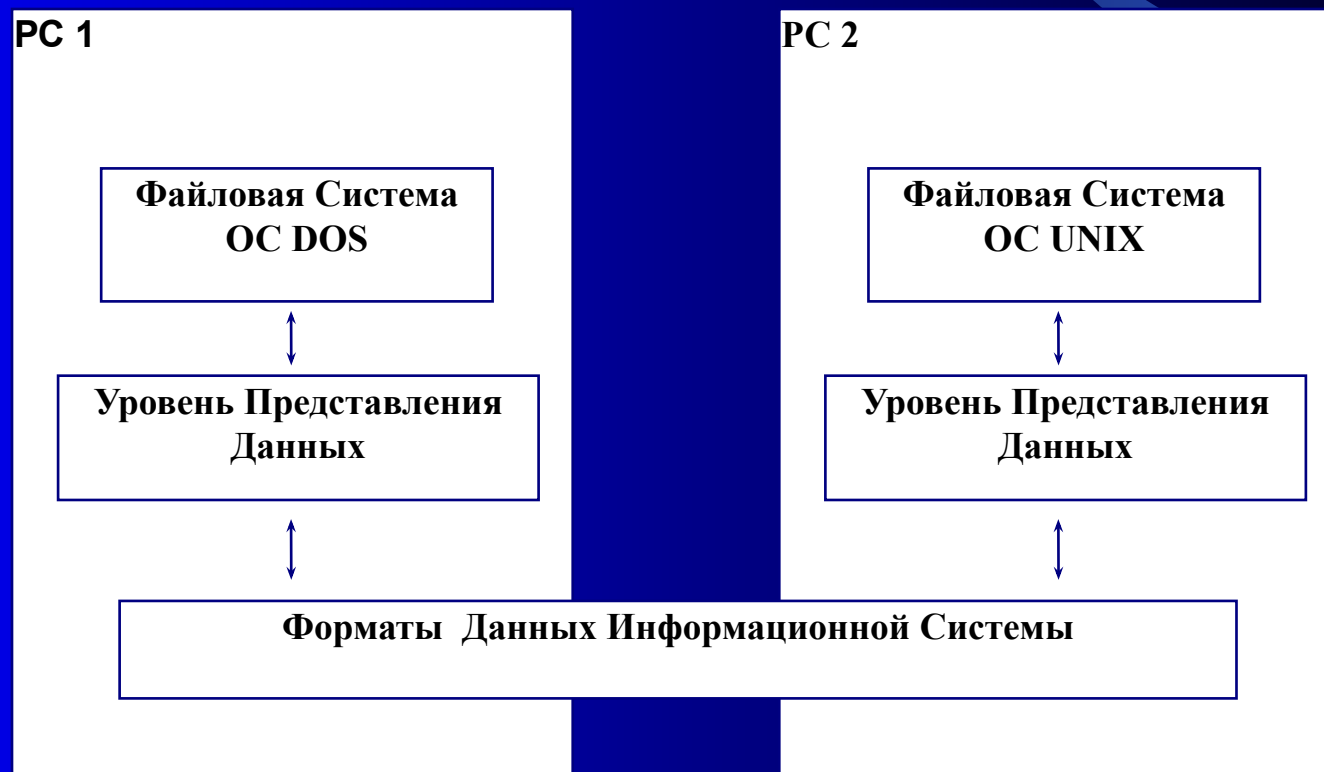
## Функции уровня:

- преобразование форматов данных;
- кодирование/декодирование данных, в том числе компрессию и декомпрессию данных;
- шифрование данных.



# PRESENTATION LAYER - УРОВЕНЬ ПРЕДСТАВЛЕНИЯ ДАННЫХ

Задача уровня представления данных заключается в том, чтобы при передаче информации преобразовать данные в формат, который используется в информационной системе



# SESSION LAYER - СЕАНСОВЫЙ УРОВЕНЬ

Сеансовый уровень определяет структуру управления взаимодействием абонентов сети, т.е. определяет и контролирует диалог между сетевыми объектами.

## Выполняет следующие функции:

- определяет начало и окончание сеанса связи (нормальное или аварийное);
- определяет время, длительность и режим сеанса связи;
- определяет точки синхронизации для промежуточного контроля и восстановления при передаче данных;
- распознавание имен и паролей;
- восстанавливает соединение после ошибок во время сеанса связи без потери данных.

Установление сеанса связи включает процедуры проверки пользовательского имени и пароля, определение прав доступа к тем или иным ресурсам системы

# TRANSPORT LAYER - ТРАНСПОРТНЫЙ УРОВЕНЬ

Транспортный уровень обеспечивает эффективную и надежную передачу данных между сеансовыми объектами (без ошибок, в нужной последовательности и без дублирования). На этом уровне сообщения переупаковываются – длинные разбиваются на несколько пакетов, а короткие объединяются в один.

## Функции транспортного уровня:

- устанавливает и разъединяет транспортные соединения;
  - контролирует последовательность передачи данных;
  - управляет потоком данных;
  - обнаруживает и обрабатывает ошибки передачи данных;
  - устанавливает соответствие между транспортными (логическими) и сетевыми адресами абонентов;
- позволяет мультиплексировать передаваемые сообщения или соединения.

# NETWORK LAYER - СЕТЕВОЙ УРОВЕНЬ

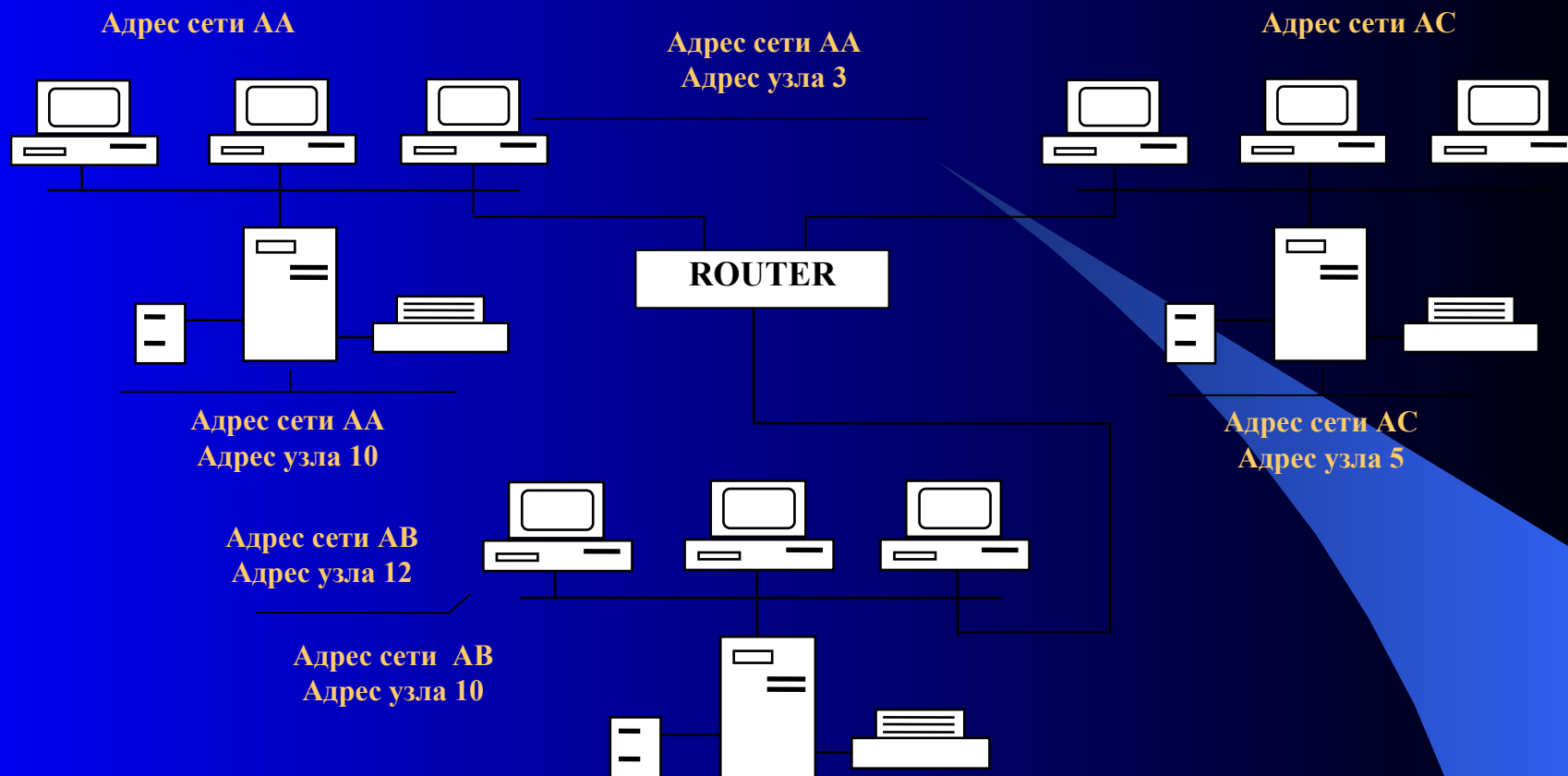
Этот уровень служит для образования единой транспортной системы, объединяющей несколько сетей с различными принципами передачи информации между конечными узлами.

## Функции уровня:

- устанавливает сетевые соединения;
- определяет маршрутизацию в сети и связь между сетями (интерсетевой протокол);
- обеспечивает независимость высших уровней от используемой для передачи информации физической среды

**Основная задача сетевого уровня -  
маршрутизация данных (передача данных между сетями )**

# NETWORK LAYER - СЕТЕВОЙ УРОВЕНЬ



Пример сети, состоящей из нескольких сетей (интерсеть)

# DATA LINK LAYER - КАНАЛЬНЫЙ УРОВЕНЬ

Канальный уровень предназначен для организации доступа к среде передачи данных, преобразования пакетов в кадры (frames) и передачи данных от сетевого уровня к физическому.

Уровень определяет:

- логическую топологию сети передачи данных;
- метод доступа к среде передачи данных;
- физическую адресацию;
- услуги по установлению соединений между станциями.

Уровень звена данных преобразовывает данные, полученные от сетевого уровня в кадр (**frame**), а затем в последовательность битов для передачи по линии связи (физическому уровню).

**Кадр** – это логически организованная структура, в которую можно помещать данные.

Канальный уровень также отвечает за контроль ошибок, управление потоком данных и физическую адресацию станций

# DATA LINK LAYER - КАНАЛЬНЫЙ УРОВЕНЬ



## Простой кадр данных

# PHYSICAL LAYER - ФИЗИЧЕСКИЙ УРОВЕНЬ

Физический уровень определяет механические и электрические характеристики передающей среды и интерфейсного оборудования.

Уровень определяет количество и назначение контактов на сетевых разъемах, в каком виде передаются биты, какие типы кабеля могут использоваться и т.п.

Функции на этом уровне обеспечивают установление, поддержку и разрыв физического соединения между узлами сети по запросу от канального уровня.

**На этом уровне посылка рассматривается как последовательность битов**