

### 3. Семиуровневая модель взаимодействия открытых систем

Сложность сетевых структур и разнообразие телекоммуникационных устройств, выпускаемых различными фирмами, привели к необходимости стандартизации как устройств, так и процедур обмена данными между пользователями. Международная организация стандартов (*International Standards Organization – ISO*) создала эталонную модель взаимодействия **открытых систем** (*Open System Interconnection reference model – OSI*), которая определяет концепцию и методологию создания сетей передачи данных. Модель описывает стандартные правила функционирования устройств и программных средств при обмене данными между узлами (компьютерами) в *открытой системе*. *Открытая система* состоит из программно-аппаратных средств, способных взаимодействовать между собой при использовании стандартных правил и устройств сопряжения (интерфейсов).

Модель *ISO/OSI* включает семь уровней. На [рис. 1.5](#) показана модель взаимодействия двух устройств: узла **источника** (source) и узла **назначения** (destination). Совокупность правил, по которым происходит обмен данными между программно-аппаратными средствами, находящимися на одном уровне, называется **протоколом**. Набор протоколов называется стеком протоколов и задается определенным стандартом. Взаимодействие между уровнями определяется стандартными **интерфейсами**.



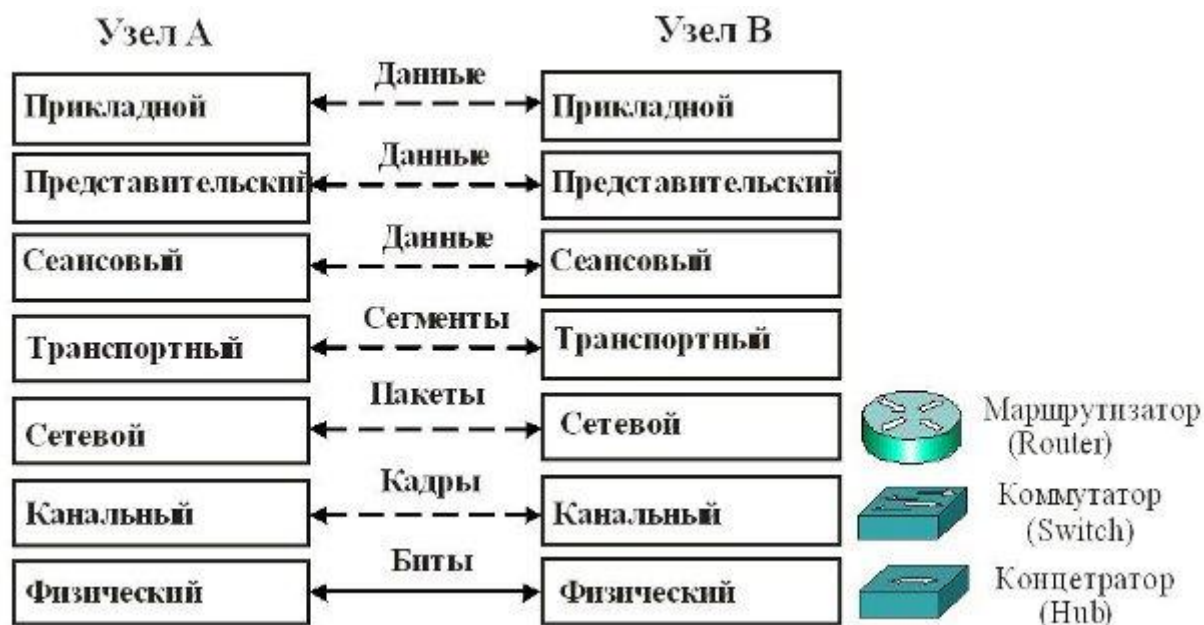
**Рис. 1.5.** Семиуровневая модель ISO/OSI

Взаимодействие соответствующих уровней является **виртуальным**, за исключением физического уровня, на котором происходит обмен данными по кабелям, соединяющим компьютеры. На [рис. 1.5](#) приведены также примеры протоколов,

управляющих взаимодействием узлов на различных уровнях модели *OSI*. Взаимодействие уровней между собой внутри узла происходит через межуровневый **интерфейс**, и каждый нижележащий уровень предоставляет услуги вышележащему.

Виртуальный обмен между соответствующими уровнями узлов А и В (рис. 1.6) происходит определенными единицами информации. На трех верхних уровнях – это **сообщения** или **данные (Data)**, на транспортном уровне – **сегменты (Segment)**, на сетевом уровне – **пакеты (Packet)**, на канальном уровне – **кадры (Frame)** и на физическом – последовательность битов.

Для каждой *сетевой технологии* существуют свои протоколы и свои технические средства, часть из которых имеет условные обозначения, приведенные на рис. 1.5. Данные обозначения введены фирмой Cisco и стали общепринятыми. Среди технических средств физического уровня следует отметить кабели, разъемы, **повторители сигналов (repeater)**, многопортовые *повторители* или **концентраторы (hub)**, **преобразователи среды (transceiver)**, например, преобразователи электрических сигналов в оптические и наоборот. На канальном уровне – это **мосты (bridge)**, **коммутаторы (switch)**. На сетевом уровне – **маршрутизаторы (router)**. Сетевые карты или адаптеры (*Network Interface Card – NIC*) функционируют как на канальном, так и на физическом уровне, что обусловлено *сетевой технологией* и *средой передачи данных*.



**Рис. 1.6.** Устройства и единицы информации соответствующих уровней

При передаче данных от источника к узлу назначения подготовленные на прикладном уровне передаваемые данные последовательно проходят от самого верхнего,

Прикладного уровня 7 узла источника информации до самого нижнего – Физического уровня 1, затем передаются по физической среде узлу назначения, где последовательно проходят от нижнего уровня 1 до уровня 7.

Самый верхний, **Прикладной уровень (Application Layer) 7** оперирует наиболее общей единицей данных – сообщением. На этом уровне реализуется управление общим доступом к сети, потоком данных, сетевыми службами, такими, как **FTP, TFTP, HTTP, SMTP, SNMP** и др.

**Представительский уровень (Presentation Layer) 6** изменяет форму представления данных. Например, передаваемые с уровня 7 данные преобразуются в общепринятый формат *ASCII*. При приеме данных происходит *обратный* процесс. На уровне 6 также происходит шифрация и сжатие данных.

**Сеансовый уровень (Session Layer) 5** устанавливает *сеанс* связи двух конечных узлов (компьютеров), определяет, какой *компьютер* является передатчиком, а какой приемником, задает для передающей стороны время передачи.

**Транспортный уровень (Transport Layer) 4** делит большое сообщение узла источника информации на части, при этом добавляет заголовок и формирует **сегменты** определенного объема, а короткие сообщения может объединять в один сегмент. В узле назначения происходит *обратный* процесс. В заголовке сегмента задаются **номера порта** источника и назначения, которые адресуют службы верхнего прикладного уровня для обработки данного сегмента. Кроме того, *транспортный уровень* обеспечивает надежную доставку пакетов. При обнаружении потерь и ошибок на этом уровне формируется *запрос* повторной передачи, при этом используется протокол **TCP**. Когда необходимость проверки правильности доставленного сообщения отсутствует, то используется более простой и быстрый протокол *дейтаграмм* пользователя (*User Datagram Protocol – UDP*).

**Сетевой уровень (Network Layer) 3** адресует сообщение, задавая единице передаваемых данных (**пакету**) **логические сетевые адреса** узла назначения и узла источника (**IP-адреса**), определяет **маршрут**, по которому будет отправлен **пакет данных**, транслирует логические *сетевые адреса* в физические, а на приемной стороне – *физические адреса* в логические. Сетевые *логические адреса* принадлежат пользователям.

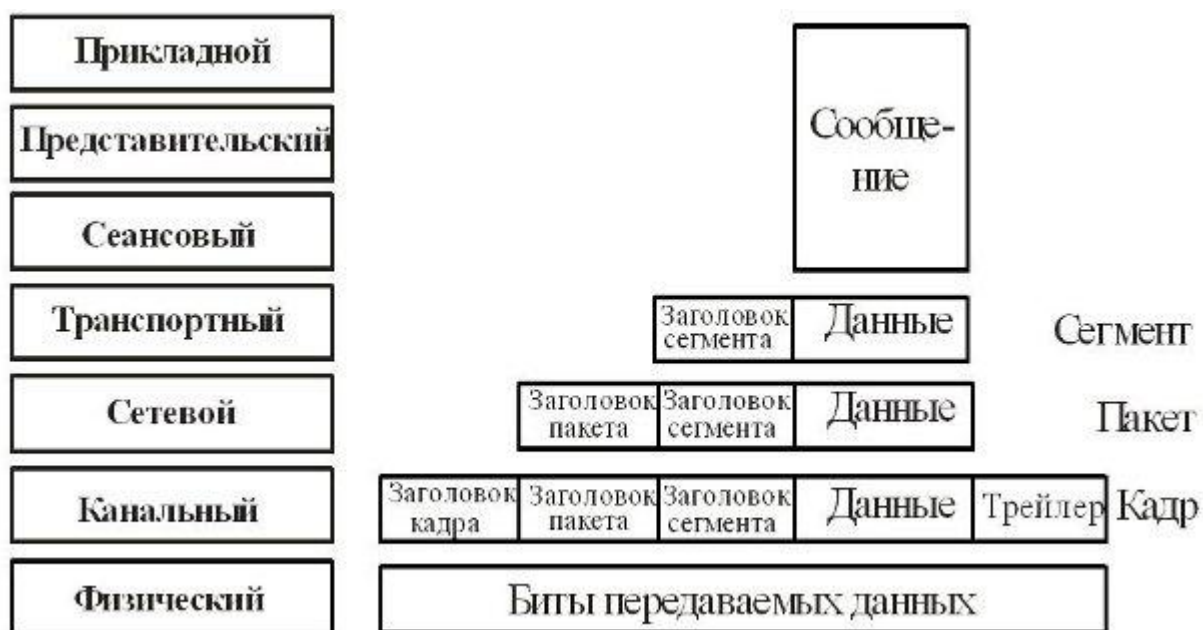
**Канальный уровень (Data Link) 2** формирует из пакетов **кадры** данных (*frames*). На этом уровне задаются *физические адреса* устройства-отправителя и устройства-получателя данных. Например, **физический адрес** устройства может быть прописан в *ПЗУ сетевой карты* компьютера. На этом же уровне к передаваемым данным добавляется *контрольная сумма*, определяемая с помощью алгоритма *циклического кода*.

На приемной стороне по *контрольной сумме* определяют и по возможности исправляют ошибки.

**Физический уровень (Physical) 1** осуществляет передачу потока битов по соответствующей физической среде (электрический или оптический *кабель, радиоканал*) через соответствующий *интерфейс*. На этом уровне производится *кодирование* данных, синхронизация передаваемых битов информации.

Протоколы трех верхних уровней являются сетезависимыми, три нижних уровня являются сетезависимыми. *Связь* между тремя верхними и тремя нижними уровнями происходит на транспортном уровне.

Важным процессом при передаче данных является **инкапсуляция** (*encapsulation*) данных. Передаваемое сообщение, сформированное приложением, проходит три верхних сетезависимых уровня и поступает на *транспортный уровень*, где делится на части и каждая часть инкапсулируется (помещается) в сегмент данных (рис. 1.7). В заголовке сегмента содержится номер протокола прикладного уровня, с помощью которого подготовлено сообщение, и номер протокола, который будет обрабатывать данный сегмент.



**Рис. 1.7.** Инкапсуляция данных

На сетевом уровне сегмент инкапсулируется в **пакет** данных, заголовок (**header**) которого содержит, кроме прочего, сетевые (логические) адреса отправителя информации (источника) – *Source Address (SA)* и получателя (назначения) – *Destination Address (DA)*. В данном курсе – это *IP-адреса*.

На канальном уровне пакет инкапсулируется в **кадр** или **фрейм** данных, заголовок которого содержит *физические адреса* узла передатчика и приемника, а также другую

информацию. Кроме того, на этом уровне добавляется **трейлер** (концевик) кадра, содержащий информацию, необходимую для проверки правильности принятой информации. Таким образом, происходит обрамление данных заголовками со служебной информацией, т. е. **инкапсуляция** данных.

Название информационных единиц на каждом уровне, их размер и другие параметры инкапсуляции задаются согласно протоколу единиц данных (*Protocol Data Unit – PDU*). Итак, на трех верхних уровнях – это **сообщение (Data)**, на Транспортном уровне 4 – **сегмент (Segment)**, на Сетевом уровне 3 – **пакет (Packet)**, на Канальном уровне 2 – **кадр (Frame)**, на Физическом Уровне 1 – **последовательность бит**.

Помимо семиуровневой *OSI* модели на практике применяется четырехуровневая модель *TCP/IP* (рис. 1.8).



**Рис. 1.8.** Модели OSI и TCP/IP

*Прикладной* уровень модели *TCP/IP* по названию совпадает с названием модели *OSI*, но по функциям гораздо шире, поскольку охватывает три верхних сетенезависимых уровня (прикладной, представительский и сеансовый). *Транспортный* уровень обеих моделей и по названию, и по функциям одинаков. *Сетевой* уровень модели *OSI* соответствует межсетевому (**Internet**) уровню модели *TCP/IP*, а два нижних уровня (канальный и физический) представлены объединенным уровнем доступа к сети (**Network Access**).

Ниже в [таблице 1.1](#) приведены обобщенные сведения об основной информации, добавляемой в заголовках сообщений на разных уровнях *OSI*-модели.

Таблица 1.1. Основная информация в заголовках сообщений

Физический уровень	Канальный уровень	Сетевой уровень	Транспортный уровень	Верхние уровни
Частотно-временные параметры	Физические адреса источника и назначения	Логические адреса источника и назначения	Номера портов источника и назначения	Сопряжение пользователей с сетью
синхронизация	назначения	назначения	назначения	сетью

На транспортном уровне в заголовке сегмента задаются *номера портов* приложений источника и назначения. *Номера портов* адресуют приложения или сервисы прикладного уровня, которые создавали сообщение и будут его обрабатывать на приемной стороне. Например, *сервер* электронной почты с номерами портов 25 и 110 позволяет посылать *e-mail* сообщения и принимать их, *номер порта* 80 адресует *веб-сервер*.

Для обмена сообщениями помимо номеров портов на сетевом уровне в заголовке пакета необходимо задать *логические адреса* источника и назначения. К *логическим адресам* относятся, например, *IP*-адреса пользователей. В документации, используемой в настоящее время, версии *IPv4* адреса *IP* отображаются в десятичной форме в виде четырех групп чисел. Каждая *группа* может содержать числа от 0 до 255. Группы разделены между собой точками, например 192.168.10.21, 172.16.250.17, 10.1.10.122.

В дополнение к *логическим адресам* на канальном уровне в заголовке кадра задаются *физические адреса* устройства-источника и устройства-назначения. Наиболее широко распространенной *сетевой технологией* канального уровня в настоящее время является *Ethernet* или ее модификации (*Fast Ethernet*, *Gigabit Ethernet*, *10Gigabit Ethernet*). При этом в качестве физических адресов используются *MAC*-адреса (*Media Access Control*). В документации *MAC*-адреса представлены в виде 12 шестнадцатеричных чисел, например, 00-05-A8-69-CD-F1. Тот же *адрес* может

быть представлен и в несколько другой форме 00:05:A8:69:CD:F1 или 0005.A869.CDF1. *MAC*-адреса компьютеров прошиты в *ПЗУ* сетевой карты.

Таким образом, тройная система адресации позволяет адресовать устройства, пользователей и *программное обеспечение* приложений.

Поскольку на трех нижних уровнях модели *OSI* функционируют аппаратно-программные средства, обработка сообщения проводится с высокой скоростью. На верхних же уровнях функционируют программные средства, что увеличивает время обработки (задержку). В вышеприведенных примерах (рис. 1.5, рис. 1.6) два конечных узла взаимодействовали непосредственно между собой. Поэтому сформированное на узле-источнике сообщение последовательно проходило все семь уровней с 7 по 1, на что тратилось много времени. В реальных сетях сообщение от одного конечного узла до другого проходит через целый ряд промежуточных устройств, таких как коммутаторы и маршрутизаторы. Поэтому для снижения времени задержки (повышения быстродействия)

на промежуточных устройствах сообщение обрабатывается средствами только трех или даже двух нижних уровней (рис. 1.9).



**Рис. 1.9.** Передача сообщения по сети

Таким образом, *Транспортный уровень*, обеспечивающий *надёжность* передачи данных, функционирует только на конечных узлах, что снижает задержку *передачи сообщения* по всей сети от одного конечного узла до другого. В приведенном примере (рис. 1.9) *протокол IP* функционирует на всех узлах сети, а *стек протоколов TCP/IP* – только на конечных узлах.

### Краткие итоги

1. Телекоммуникационная сеть образуется совокупностью *абонентов* и узлов связи, соединенных линиями (каналами) связи.
2. Различают *сети: с коммутацией каналов*, когда телекоммуникационные узлы выполняют функции *коммутаторов*, и с *коммутацией пакетов (сообщений)*, когда телекоммуникационные узлы выполняют функции маршрутизаторов.
3. Для создания маршрута в разветвленной сети необходимо задавать адреса источника и *получателя сообщения*. Различают *физические* и *логические адреса*.
4. *Сети передачи данных с коммутацией пакетов* подразделяются на локальные и глобальные.
5. Сети технологии IP являются *дейтаграммными*, когда отсутствует предварительное соединение конечных узлов и нет подтверждения приема сообщения.
6. Высокую надёжность обеспечивает *протокол управления передачей TCP*.

7. Эталонная модель взаимодействия открытых систем ISO/OSI определяет концепцию и методологию создания сетей передачи данных и включает в себя семь уровней.

8. Виртуальный обмен между соответствующими уровнями конечных узлов происходит определенными единицами информации. На трех верхних уровнях – это сообщения или данные. На транспортном уровне – сегменты, на сетевом уровне – пакеты, на канальном уровне – кадры и на физическом – последовательность битов.

9. Технические средства физического уровня представлены кабелями, разъемами, повторителями сигналов, многопортовыми повторителями или концентраторами (*hub*), преобразователями среды (*transceiver*). На канальном уровне – это мосты (*bridge*) и коммутаторы (*switch*). На сетевом уровне – маршрутизаторы (*router*). Сетевые карты или адаптеры (*Network Interface Card – NIC*) функционируют на канальном и на физическом уровнях.

10. Обрамление единиц информации заголовками со служебной информацией называется инкапсуляцией.

11. Тройная система адресации (*логические адреса, физические адреса, номера портов*) позволяет адресовать устройства, пользователей и программное обеспечение приложений.

### **Вопросы**

1. Что собой представляют телекоммуникационные сети?
2. Чем отличаются сети с коммутацией каналов от сетей с коммутацией сообщений?
3. Какие функции выполняет маршрутизатор?
4. Что собой представляет метрика протокола маршрутизации?
5. В чем различие коммутации пакетов или сообщений?
6. В чем различие между локальными и глобальными сетями передачи данных?
7. Каковы основные функции Уровня 1 модели OSI?
8. Каковы основные функции Уровня 2 модели OSI?
9. Каковы основные функции Уровня 3 модели OSI?
10. Каковы основные функции Уровня 4 модели OSI?
11. Каковы основные функции Уровня 5 модели OSI?
12. Каковы основные функции Уровня 6 модели OSI?
13. Каковы основные функции Уровня 7 модели OSI?
14. Что собой представляет инкапсуляция данных?

15. Какие устройства функционируют на Уровне 3 модели *OSI*?
16. Какие устройства функционируют на Уровне 2 модели *OSI*?
17. Какие устройства функционируют на Уровне 1 модели *OSI*?
18. Перечислите уровни модели TCP/IP.
19. Какие три системы адресации используются в *сетевых технологиях*?
20. На каком уровне модели *OSI* задаются IP-адреса?

**Упражнения**

1. Изобразите эталонную *модель взаимодействия открытых систем ISO/OSI*.
2. Сравните функции уровней моделей *OSI* и TCP/IP.