

### **39. Модель OSI. Архитектура. Назначение. Уровни. Принципы. Взаимодействие канального и сетевого уровней при передаче данных по составной сети через маршрутизаторы. Инкапсуляция данных при передаче.**

Ссылки на материалы:

1. [http://infocisco.ru/network\\_model\\_osi.html](http://infocisco.ru/network_model_osi.html)
2. [http://rz6hpi.narod.ru/net/cisco/cisco/cv\\_307.html](http://rz6hpi.narod.ru/net/cisco/cisco/cv_307.html)
3. [http://infocisco.ru/network\\_model\\_encapsulation\\_pdu.html](http://infocisco.ru/network_model_encapsulation_pdu.html)
4. [http://citforum.ru/nets/ip/glava\\_1.shtml](http://citforum.ru/nets/ip/glava_1.shtml)

Сетевая модель OSI (базовая эталонная модель взаимодействия открытых систем, [англ.](#) Open Systems Interconnection Basic Reference Model) — абстрактная [сетевая модель](#) для коммуникаций и разработки [сетевых протоколов](#). Представляет уровневый подход к [сети](#). Каждый уровень обслуживает свою часть процесса взаимодействия. Благодаря такой структуре совместная работа  [сетевого оборудования](#) и [программного обеспечения](#) становится гораздо проще и прозрачнее.

В настоящее время основным используемым семейством протоколов является [TCP/IP](#), разработка которого не была связана с моделью OSI. За все время существования (мета)модели OSI она не была реализована технически, но методологически ее создание и является ее реализацией<sup>[[источник?](#)]</sup> (т.с. Ding an Sich) и ее использование стало триумфом методологии и стратегии проектирования<sup>[[источник?](#)]</sup>. В технических проектах используется некоторое подмножество модели OSI. Но масштабные проекты, например [HL7](#), [XML](#), реализуемые опять же в стиле OSI, используют полный профиль OSI<sup>[[источник?](#)]</sup>.

Критики OSI утверждают, что история модели OSI является типичным примером неудачного и оторванного от жизни проекта, дурной "философии".

С организационной точки зрения OSI - вершина айсберга плавающего в океане стандартов и организаций национальных и международных стандартизации.

#### **Уровни модели OSI**

Модель состоит из 7-ми уровней, расположенных вертикально друг над другом. Каждый уровень может взаимодействовать только со своими соседями и выполнять отведённые только ему функции.

#### **Прикладной (Приложений) уровень (Application layer)**

Верхний (7-й) уровень модели, обеспечивает взаимодействие сети и пользователя. Уровень разрешает приложениям пользователя иметь доступ к сетевым службам, таким как обработчик запросов к базам данных, доступ к файлам, пересылке электронной почты. Также отвечает за передачу служебной информации, предоставляет приложениям информацию об ошибках и формирует запросы к уровню представления. Пример: HTTP, POP3, SMTP.

#### **Уровень представления(Представительский) (Presentation layer)**

Этот уровень отвечает за преобразование протоколов и кодирование/декодирование данных. Запросы приложений, полученные с уровня приложений, он преобразует в формат для передачи по сети, а полученные из сети данные преобразует в формат, понятный приложениям. На этом уровне может осуществляться сжатие/распаковка или кодирование/декодирование данных, а также перенаправление запросов другому сетевому ресурсу, если они не могут быть обработаны локально.

Уровень 6 (представлений) эталонной модели OSI обычно представляет собой промежуточный протокол для информации из соседних уровней. Это позволяет осуществлять обмен между приложениями на разнородных компьютерных системах прозрачным для приложений образом. Уровень представлений обеспечивает форматирование и преобразование кода. Форматирование кода используется для того, чтобы гарантировать приложению поступление информации для обработки, которая имела бы для него смысл. При необходимости этот уровень может выполнять перевод из одного формата данных в другой. Уровень представлений имеет дело не только с форматами и представлением данных, он также занимается структурами данных, которые используются программами. Таким образом, уровень 6 обеспечивает организацию данных при их пересылке. Чтобы понять, как это работает, представим, что имеются две системы. Одна использует для представления данных расширенный двоичный код обмена информацией EBCDIC, например, это может быть мэйнфрейм компании IBM, а другая — американский стандартный код обмена информацией ASCII (его используют большинство других производителей компьютеров). Если этим двум системам необходимо обменяться информацией, то нужен уровень представлений, который выполнит преобразование и осуществит перевод между двумя различными форматами. Другой функцией, выполняемой на уровне представлений, является шифрование данных, которое применяется в тех случаях, когда необходимо защитить передаваемую информацию от приема несанкционированными получателями. Чтобы решить эту задачу, процессы и коды, находящиеся на уровне представлений, должны выполнить преобразование данных. На этом уровне существуют и другие подпрограммы, которые сжимают тексты и преобразовывают графические изображения в битовые потоки, так что они могут передаваться по сети. Стандарты уровня представлений также определяют способы представления графических изображений. Для этих целей может использоваться формат PICT — формат изображений, применяемый для передачи графики QuickDraw между программами для компьютеров Macintosh и PowerPC. Другим форматом представлений является тэгированный формат файлов изображений TIFF, который обычно используется для растровых изображений с высоким разрешением. Следующим стандартом уровня представлений, который может использоваться для графических изображений, является стандарт, разработанный Объединенной экспертной группой по фотографии (Joint Photographic Expert Group); в повседневном пользовании этот стандарт называют просто JPEG. Существует другая группа стандартов уровня представлений, которая определяет представление звука и кинофрагментов. Сюда входят интерфейс электронных музыкальных инструментов MIDI (Musical Instrument Digital Interface) для цифрового представления музыки, разработанный Экспертной группой по кинематографии стандарт MPEG, используемый для сжатия и кодирования видеороликов на компакт-дисках, хранения в оцифрованном виде и передачи со скоростями до 1,5 Мбит/с, и QuickTime — стандарт, описывающий звуковые и видео элементы для программ, выполняемых на компьютерах Macintosh и PowerPC.

#### Сеансовый уровень (Session layer)

5-й уровень модели отвечает за поддержание сеанса связи, позволяя приложениям взаимодействовать между собой длительное время. Уровень управляет созданием/завершением сеанса, обменом информацией, синхронизацией задач, определением права на передачу данных и поддержанием сеанса в периоды неактивности приложений. Синхронизация передачи обеспечивается помещением в поток данных контрольных точек, начиная с которых возобновляется процесс при нарушении взаимодействия.

#### Транспортный уровень (Transport layer)

4-й уровень модели предназначен для доставки данных без ошибок, потерь и дублирования в той последовательности, как они были переданы. При этом неважно, какие данные передаются, откуда и куда, то есть он предоставляет сам механизм передачи. Блоки данных он разделяет на фрагменты, размер которых зависит от протокола, короткие объединяет в один, а длинные разбивает. Протоколы этого уровня предназначены для взаимодействия типа точка-точка. Пример: TCP, UDP. Существует множество классов протоколов транспортного уровня, начиная от протоколов, предоставляющих только основные транспортные функции (например, функции передачи данных без подтверждения приема), и заканчивая протоколами, которые гарантируют доставку в пункт назначения нескольких пакетов данных в надлежащей последовательности, мультиплексируют несколько потоков данных, обеспечивают механизм управления потоками данных и гарантируют достоверность принятых данных. Некоторые протоколы сетевого уровня, называемые протоколами без установки соединения, не гарантируют, что данные доставляются по назначению в том порядке, в котором они были посланы устройством-источником. Некоторые транспортные уровни справляются с этим, собирая данные в нужной последовательности до передачи их на сеансовый уровень. Мультиплексирование (multiplexing) данных означает, что транспортный уровень способен одновременно обрабатывать несколько потоков данных (потоки могут поступать и от различных приложений) между двумя системами. Механизм управления потоком данных — это механизм, позволяющий регулировать количество данных, передаваемых от одной системы к другой. Протоколы транспортного уровня часто имеют функцию контроля доставки данных, заставляя принимающую данные систему отправлять подтверждения передающей стороне о приеме данных.

#### Сетевой уровень (Network layer)

3-й уровень сетевой модели OSI предназначен для определения пути передачи данных. Отвечает за трансляцию логических адресов и имён в физические, определение кратчайших маршрутов, коммутацию и маршрутизацию, отслеживание неполадок и заторов в сети. На этом уровне работает такое сетевое устройство, как [маршрутизатор](#). Протоколы сетевого уровня маршрутизируют данные от источника к получателю и могут быть разделены на два класса: протоколы с установкой соединения и без него. Описать работу протоколов с установкой соединения можно на примере работы обычного телефона. Протоколы этого класса начинают передачу данных с вызова или установки маршрута следования пакетов от источника к получателю. После чего начинают последовательную передачу данных и затем по окончании передачи разрывают связь. Протоколы без установки соединения, которые посылают данные, содержащие полную адресную информацию в каждом пакете, работают аналогично почтовой системе. Каждое письмо или пакет содержит адрес отправителя и получателя. Далее каждый промежуточный почтамт или сетевое устройство считывает адресную информацию и принимает решение о маршрутизации данных. Письмо или пакет данных передается от одного промежуточного устройства к другому до тех пор, пока не будет доставлено получателю. Протоколы без установки соединения не гарантируют поступление информации получателю в том порядке, в котором она была отправлена. За установку данных в соответствующем порядке при использовании сетевых протоколов без установки соединения отвечают транспортные протоколы.

#### Канальный уровень (Data Link layer)

Этот уровень предназначен для обеспечения взаимодействия сетей на физическом уровне и контроля за ошибками, которые могут возникнуть. Полученные с физического уровня данные он упаковывает во [фреймы](#), проверяет на целостность, если нужно

исправляет ошибки (посылает повторный запрос поврежденного кадра) и отправляет на сетевой уровень. Канальный уровень может взаимодействовать с одним или несколькими физическими уровнями, контролируя и управляя этим взаимодействием. Спецификация IEEE 802 разделяет этот уровень на 2 подуровня — [MAC](#) (Media Access Control) регулирует доступ к разделяемой физической среде, [LLC](#) (Logical Link Control) обеспечивает обслуживание сетевого уровня. На этом уровне работают [коммутаторы](#), [мосты](#).

В программировании этот уровень представляет [драйвер](#) сетевой платы, в [операционных системах](#) имеется программный интерфейс взаимодействия канального и сетевого уровней между собой, это не новый уровень, а просто реализация модели для конкретной ОС. Примеры таких интерфейсов: [ODI](#), [NDIS](#)

### Физический уровень (Physical layer)

Самый нижний уровень модели предназначен непосредственно для передачи потока данных. Осуществляет передачу электрических или оптических сигналов в кабель или в радиоэфир и соответственно их приём и преобразование в биты данных в соответствии с [методами кодирования цифровых сигналов](#). Другими словами, осуществляет интерфейс между сетевым носителем и сетевым устройством. На этом уровне работают [концентраторы](#) ([хабы](#)), повторители ([ретрансляторы](#)) сигнала и медиаконверторы. Функции физического уровня реализуются на всех устройствах, подключенных к сети. Со стороны компьютера функции физического уровня выполняются сетевым адаптером или последовательным портом. К физическому уровню относятся физические, электрические и механические интерфейсы между двумя системами. Физический уровень определяет такие свойства среды сети передачи данных как оптоволокно, витая пара, коаксиальный кабель, спутниковый канал передач данных и т.п. Стандартными типами сетевых интерфейсов, относящимися к физическому уровню, являются: V.35, RS-232C, RJ-11, RJ-45, разъемы AUI и BNC.

### Взаимодействие уровней

Уровни взаимодействуют сверху вниз и снизу вверх посредством интерфейсов и могут еще взаимодействовать с таким же уровнем другой системы с помощью протоколов. Подробнее можно посмотреть на рисунке.



Модель OSI и реальные протоколы

Семиуровневая модель OSI является теоретической, и содержит ряд недоработок. Были попытки строить сети в точном соответствии с моделью OSI, но созданные таким образом сети были дорогими, ненадёжными и неудобными в эксплуатации. Реальные сетевые протоколы, используемые в существующих сетях, вынуждены отклоняться от неё, обеспечивая непредусмотренные возможности, поэтому привязка некоторых из них к уровням OSI является несколько условной: некоторые протоколы занимают несколько уровней модели OSI, функции обеспечения надёжности реализованы на нескольких уровнях модели OSI.

Основная недоработка OSI — непродуманный транспортный уровень. На нём OSI позволяет обмен данными между приложениями (вводя понятие порта — идентификатора приложения), однако, возможность обмена простыми датаграммами (по типу UDP) в OSI не предусмотрена — транспортный уровень должен образовывать соединения, обеспечивать доставку, управлять потоком и т.п. (по типу TCP). Реальные же протоколы реализуют такую возможность.

## Семейство [TCP/IP](#)

Семейство TCP/IP имеет два транспортных протокола: TCP, полностью соответствующий OSI, обеспечивающий проверку получения данных, и UDP, отвечающий транспортному уровню только наличием порта, обеспечивающий обмен датаграммами между приложениями, не гарантирующий получения данных. (В семействе TCP/IP есть ещё около двухсот протоколов, самым известным из которых является [ICMP](#); но они служебные (используются для внутренних нужд обеспечения работы сети), а не транспортные.)

## Семейство IPX/SPX

В семействе IPX/SPX порты (называемые «сокеты» или «гнезда») появляются в протоколе сетевого уровня IPX, обеспечивая обмен датаграммами между приложениями (операционная система резервирует часть сокетов для себя). Протокол SPX, в свою очередь, дополняет IPX всеми остальными возможностями транспортного уровня в полном соответствии с OSI.

В качестве адреса хоста IPX использует идентификатор, образованный из четырёхбайтного номера сети (назначаемого [маршрутизаторами](#)) и MAC-адреса сетевого адаптера.

## Модель DOD

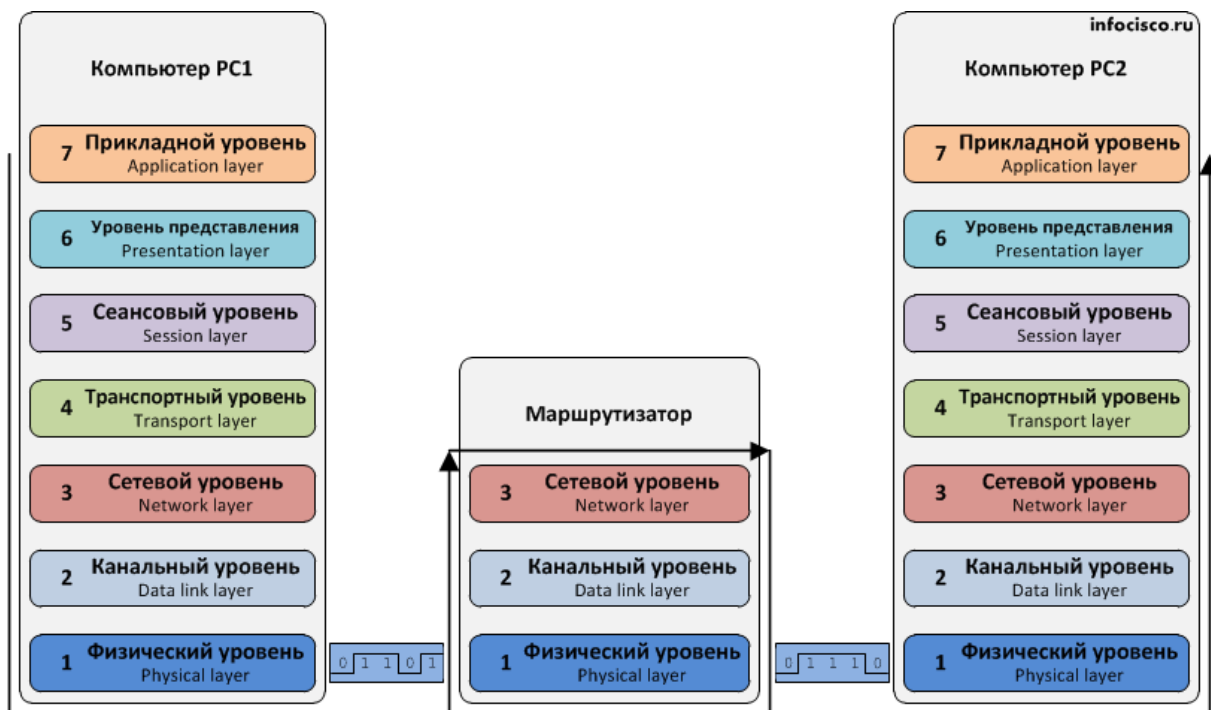
Стек протоколов TCP/IP, использующий упрощённую четырёхуровневую модель OSI.

## Инкапсуляция

Инкапсуляция – это процесс передачи данных с верхнего уровня приложений вниз (по стеку протоколов) к физическому уровню, чтобы быть переданными по сетевой физической среде (витая пара, оптическое волокно, Wi-Fi, и др.). Причём на каждом уровне различные протоколы добавляют к передающимся данным свою информацию.

Напомню, что сетевая модель OSI состоит из 7 уровней (уровень приложений, уровень представления, сеансовый, транспортный, сетевой, канальный и физический). Все сетевые устройства работают согласно модели OSI, только некоторые используют все 7 уровней, а другие меньше. Это позволяет обрабатывать поступающие данные в несколько раз быстрее.

Например, Ваш компьютер использует все 7 уровней, маршрутизатор – 3 нижних уровня, коммутатор – только 2 нижних уровня.



На рисунке Вы видите взаимодействие двух компьютеров, между которыми находится маршрутизатор. Компьютерами PC1 и PC2 могут быть как домашние компьютеры, так и сервера. Маршрутизатор, как и говорилось выше, работает только на трех уровнях модели, их (трех уровней) достаточно, чтобы проложить маршрут в любой сети.