

Еталонен модел на
мрежите.

Характеристики на нивата.
Модел ТСР/ІР.

Създаване на йерархия от задачи

Организацията на взаимодействието между устройствата в мрежата е сложна задача.

Както е известно, за решаването на сложни задачи се използва универсален метод – *декомпозиция*, т.е. разбиване на сложна задача на няколко по-прости задачи – модули.

Декомпозицията се изразява в точното определяне на функциите на всеки модул, а също така и тяхното взаимодействие (интерфейси).

Като резултат се постига логическо опростяване на задачата и се появява възможност за автономно разработване и модифициране на отделните модули без да се променя останалата част на системата.

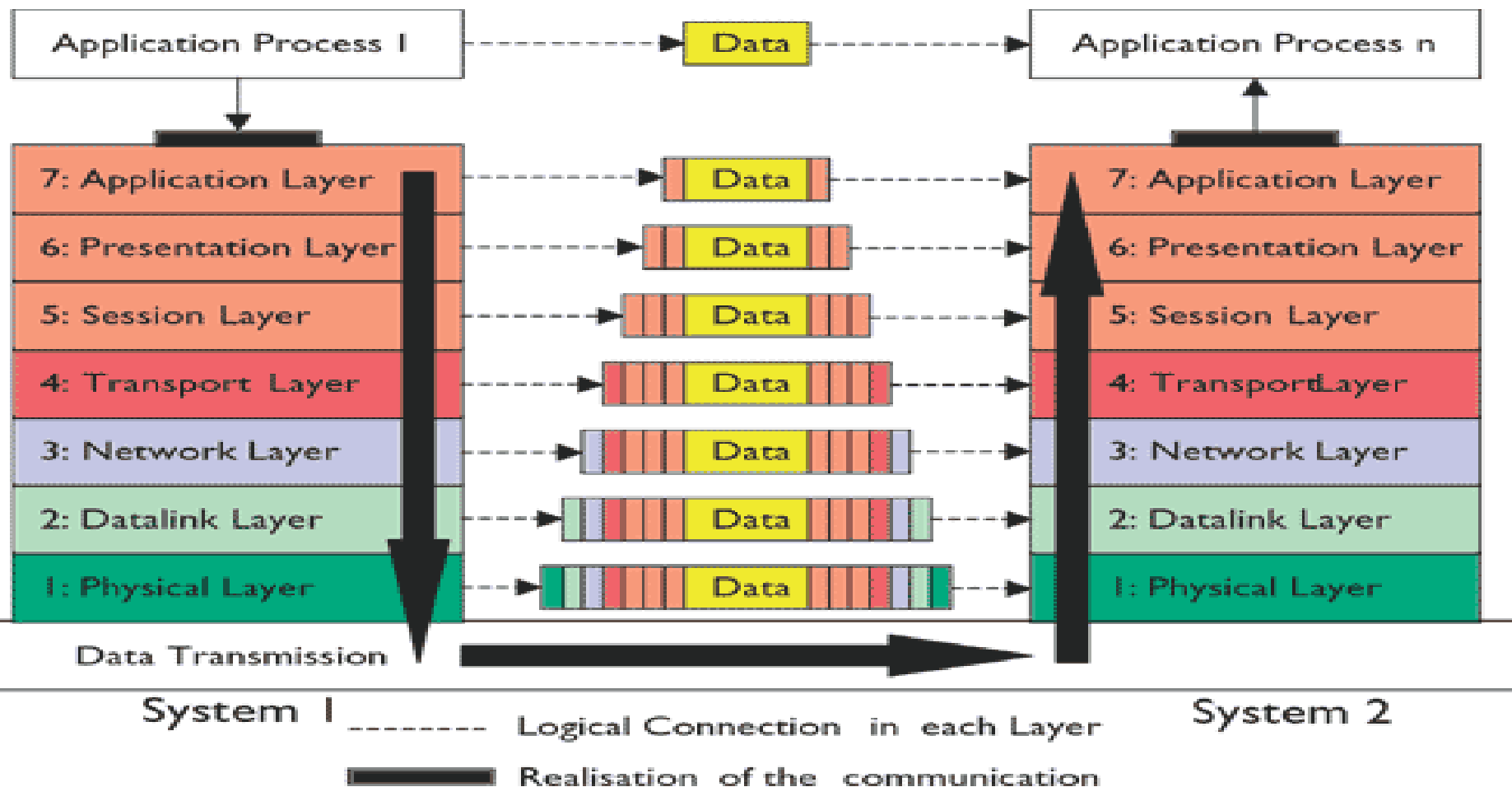
Многослойна архитектура на мрежите

Основен принцип в съвременните мрежови архитектури е принципът за **разслояване на функциите** по управление на връзките, като всеки слой ползва услугите, предоставени от по-долните слоеве, без да знае как са реализирани тези услуги. Това е принципът на **прозрачност**.

Слоят n на една машина взаимодейства със слой n (на същото ниво) на друга машина. Правилата, по които се осъществява това взаимодействие, се определят от **протокола на n -то ниво**.

Всеки слой n предава данни и контролна (служебна) информация (**header+trailer**) на непосредствено по-долния слой $n-1$, докато се достигне най-долния слой 1 , където се осъществява реалната комуникация между машините през физическата среда. В приемника получените данни се разпространяват в обратна посока - от слой 1 нагоре, като всеки слой премахва служебната информация, която се отнася до него. **Опаковане (влагане) и разопаковане (Encapsulation – decapsulation/ демултиплексиране)**.

Многослойна архитектура на мрежите



Данните+Контр. Инф. на слой n се наричат *протоколен блок от данни* (PDU). За слой $n-1$ PDU(n) са си обикновени данни. Чисто потребителските данни – *payload*.

Многослойна архитектура на мрежите

Всеки слой n предоставя интерфейс на слой $n+1$ - функциите и услугите, които слой n предоставя на слой $n+1$. Ясно да се знае какви функции изпълнява всеки слой.

Разслояването позволява да се промени изцяло реализацията на даден слой n , без да се променя реализацията на другите слоеве – достатъчно е да се запази множеството от услугите, които слой n осигурява на горния слой $n+1$. Прозрачност (*transparency*) и гъвкавост (*flexibility*).

Една **мрежова архитектура** се определя от **множеството на слоевете**, услугите които те предоставят и протоколите, по които се осъществява взаимодействие между слоевете на едно и също ниво.

Многослойна архитектура на мрежите

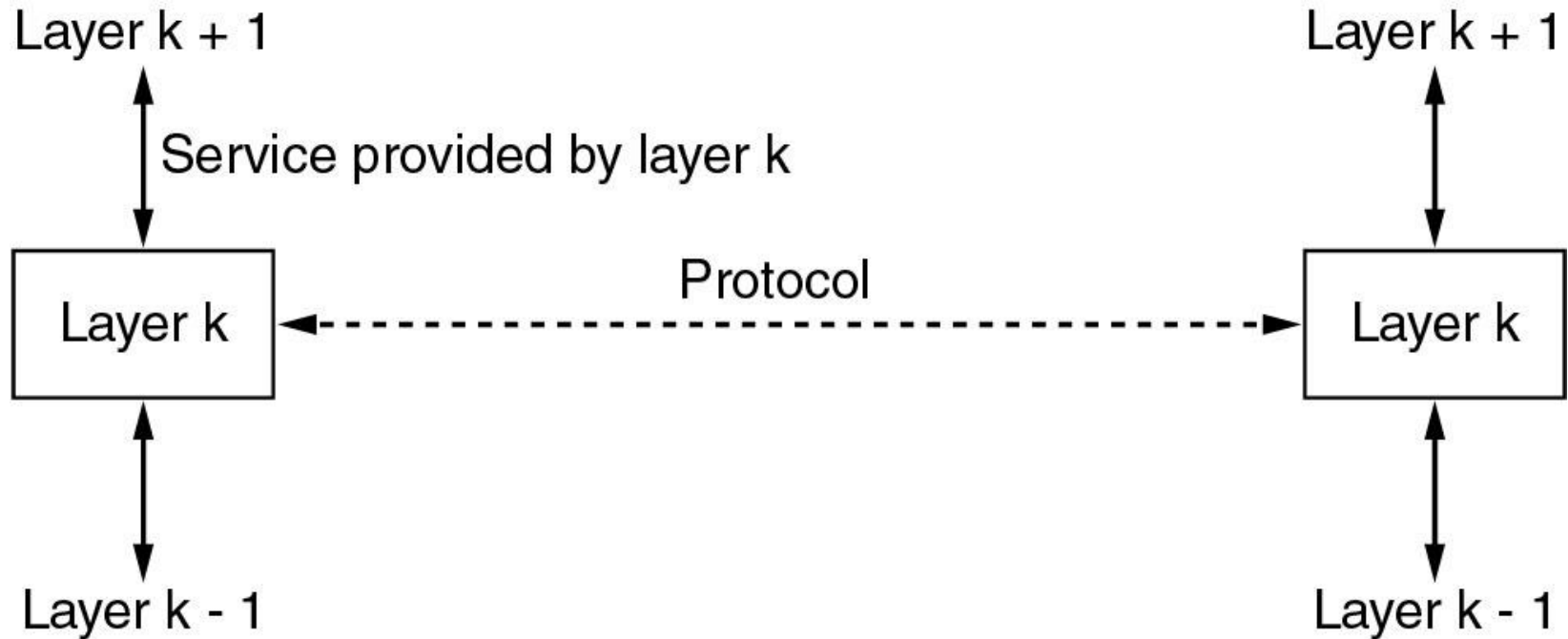
Реализацията на слоевете, както и интерфейсите между отделните слоеве не е задължително да са едни и същи на машините в една мрежа – достатъчно е всеки слой n да може да комуникира със съответния слой n по определения протокол и да предоставя съответните услуги на по-горния слой.

Списъкът от протоколи, използвани от една система, по един протокол за всеки слой се нарича **протоколен стек**.

В OSI модела базови са три понятия:

- **услуга** определя функционалността на слоя, без конкретика (не казва как);
- **интерфейс** определя за по-горележащите слоеве достъпа до услугата;
- **протокол** определя реализацията на услугата.

Протоколи и услуги



Протоколи и услуги на едно ниво: k .

Моделът OSI

Съвременните мрежови архитектури следват принципите на модела **OSI** (**Open Systems Interconnection**), създаден от Международната организация по стандартизация **ISO** (**International Standards Organization**) за връзка между отворени системи.

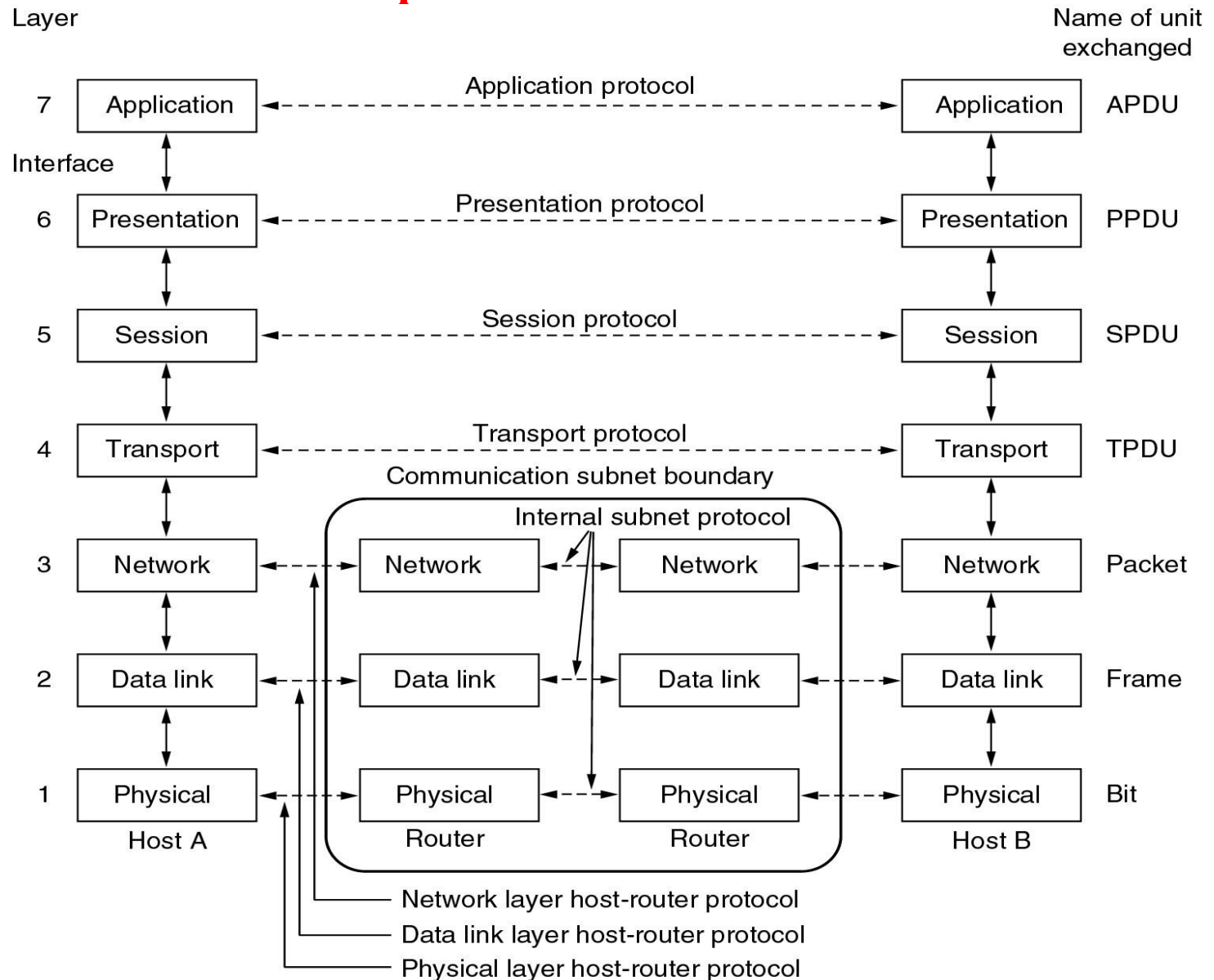
Отворена система е система, чиито ресурси могат да се използват от другите системи в мрежата.

OSI моделът е абстрактен модел на мрежова архитектура, който описва предназначението на слоевете, но не се обвързва с конкретен набор от протоколи. Поради това OSI моделът се нарича още **еталонен** (опорен) модел и всъщност дава препоръки (**Reference Model**).

В еталонния модел има седем слоя – физически, канален, мрежов, транспортен, сесиен, представителен, приложен.

OSI RM – хост машини и комуникационна мрежа

The OSI reference model.



Физически слой

Физическият слой (**physical layer**) има за задача да реализира предаването на битове през физическата среда.

Основна функция на физическия слой е да управлява кодирането и декодирането на сигналите, представлящи двоичните цифри 0 и 1. Той не се интересува от предназначението на битовете.

Физическият слой трябва да осигурява възможност на по-горния слой да активизира, поддържа и прекратява физическите съединения.

Обектите на този слой са хардуерните устройства, реализиращи предаването на 0-и и 1-ци през физическата среда – мрежови карти (NIC) и модули, модеми.

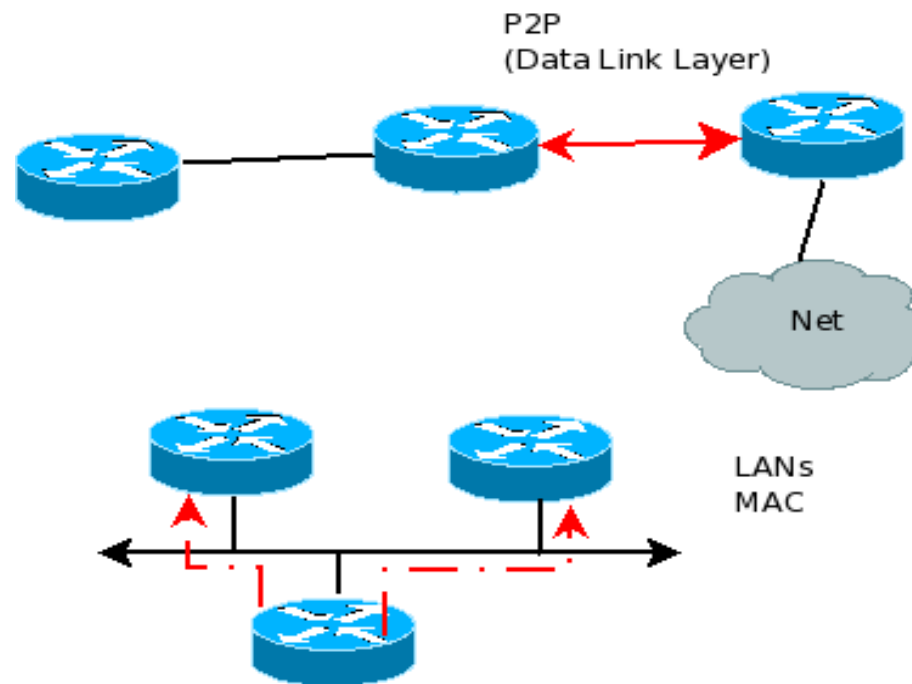
Канален слой

Основна функция на **каналния слой (data-link layer)** е управлението на канала от един възел до друг (точка-точка) според класическия модел, “точка-много точки” (напр. Frame Relay) или достъп до преносната среда (MAC) в LAN.

Друга възможна функция е (евентуално) откриването и (или) коригирането на грешки при предаването на данните.

Данните на канално ниво се обменят на порции (PDU), наречени **кадри (frames)**, обикновено с дължина от няколко стотин до няколко хиляди байта в зависимост от скоростта на линията.

Канален слой



Канален слой

При надеждна комуникация приемникът трябва да уведомява изпращача за всеки успешно получен кадър като му изпраща обратно потвърждаващ кадър.

Форматът на кадрите се определя от избрания протокол на канално ниво. Функциите на каналния слой обикновено се реализират смесено - апаратно и програмно. Колкото повече функции са реализирани софтуерно (контролерът е реализиран на дънната платка), толкова е по-ниска производителността.

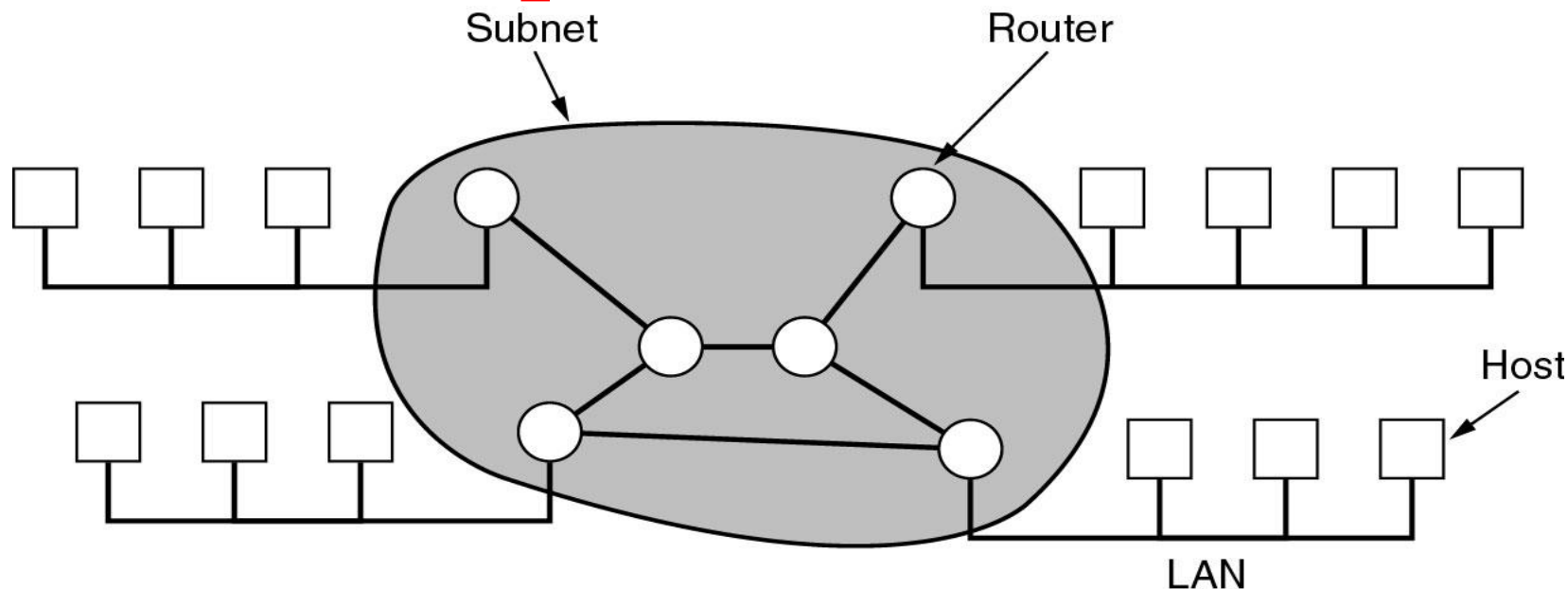
Мрежов слой

Мрежовият слой (network layer) отговаря за функционирането на комуникационната подмрежа.

Пакетите са с фиксирана големина в рамките на една мрежа.

При преминаване от една КМ в друга е възможно пакетът да се раздели на части – **фрагментира**, като накрая да може да се възстанови. Напр. преход: LAN-WAN-LAN

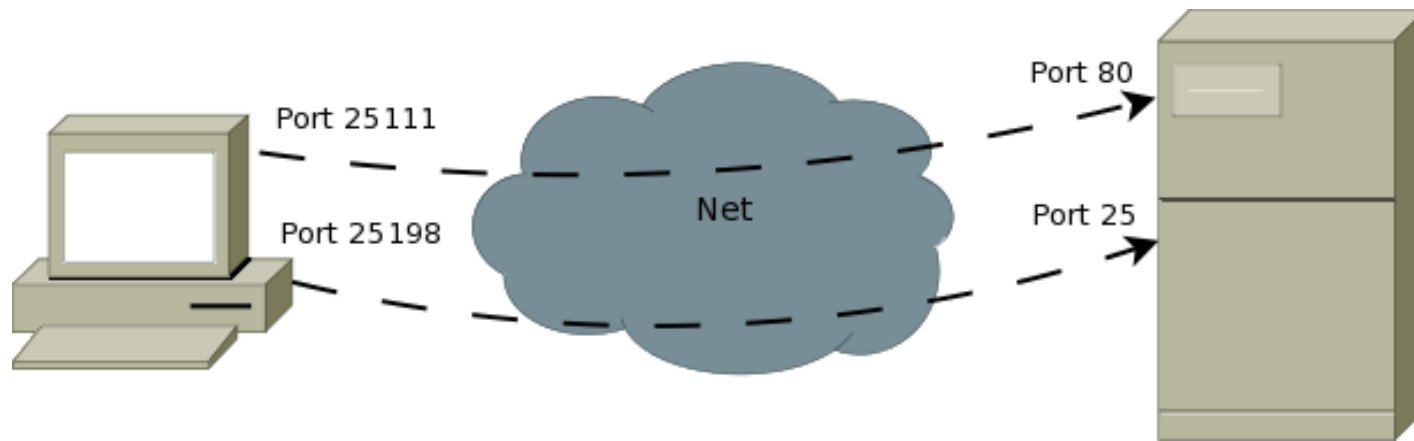
Мрежов слой



Основна задача на мрежовия слой е **маршрутизирането** на тези фрагменти, опаковани като **пакети (PDU за мрежов слой)**.

За системите, реализиращи възлите на комуникационната подмрежа (**маршрутизатори - routers**) този слой е последен. Функциите на мрежовия слой, както и на по-горните слоеве се реализират програмно.

Транспортен слой



Транспортният слой (transport layer) осигурява транспортирането на съобщения от източника до получателя. Той е най-ниският слой, който реализира връзка от тип “край-край” (end to end) между комуникиращите системи.

Изгражда се **програмен канал между портовете** на приложения, които си “говорят” през мрежата.

Транспортен слой

Транспортният слой осигурява на приложния слой (и по-горните слоеве) такава степен на надеждност, каквата е необходима.

В транспортния слой на изпращача съобщенията (евентуално могат) се разбиват на **сегменти** (PDU за транспортния слой) и се подават на мрежовия слой, където се опаковат като пакети, а в транспортния слой на получателя разопакованите от мрежовия слой сегменти се реасемблират.

Т.е. транспортният слой (евентуално може да) отговаря за целостта на обменяните съобщения, което включва откриване на загубени сегменти и тяхното повторно предаване.

Сесиен слой

Сесийният слой (**session layer**) е отговорен за диалога между две комуникиращи програми. Съобщения се обменят след като двата крайни абоната установят **сесия**.

Сесийният слой осигурява различни *режими на диалог*:
двупосочен едновременно диалог (**full duplex - FD**),
двупосочен алтернативен диалог (**half duplex - HD**),
еднопосочен диалог (**simplex**).

Освен това той предоставя възможност за прекъсване на диалога и последващо възстановяване от мястото на прекъсването.

При липсата на сесиен слой всяко съобщение се предава независимо от другите съобщения.

Представителен слой

Представителният слой (presentation layer) е най-ниският слой, който разглежда значението на предаваната информация.

Първата функция на този слой е да определи общ синтаксис за предаване на съобщенията.

Втората функция на слоя е да унифицира вътрешната структура на представените данни в съобщенията.

По този начин за по-горния приложен слой няма значение дали двете крайни системи използват различни представяния на данните.

UTF-8 (8-bit UCS/Unicode Transformation Format) представя всеки символ в Unicode стандарт, но е и обратно съвместим с ASCII. По тези причини е предпочитан за e-mail, уеб страници и др.

Може да използва криптиране на данните, компресия.

Приложен слой

Приложният слой (application layer) е най-горния слой, към който се свързват потребителските процеси в двата крайни абоната.

Някои потребителски процеси са интерактивни - взаимодействат си в голям период от време с кратки *съобщения* от тип *заявка-отговор* (request-reply).

Други потребителски процеси взаимодействат с малко на брой големи по обем порции от данни.

За двата вида процеси се предвиждат различни протоколи на приложния слой - например протокол **FTP** (file transfer protocol) за обмен на цели файлове, протокол **HTTP** (hyper text transfer protocol) за обмен на уеб страници и др.

Модел ТСП/ІР

Когато започват да се изграждат реални мрежи, използвайки OSI модела и съществуващите протоколи се вижда, че те не отговарят на изискваните спецификации за обслужване.

Въведен е за първи път през 1974 г. от V. Cerf и Kahn в ARPANET - първата компютърна мрежа, която прераства в Интернет. (Целта е била да позволи свързването на различни мрежи, да бъде жизнеспособна и гъвкава, да оцелее и в условията на ядрен апокалипсис.)

Мрежа с комутация на пакети, базирана на обслужване с неустановена връзка (connectionless - без предварително уговаряне на параметрите на връзката между източник и приемник).

Това е мрежовото ниво, имаме “best effort delivery” (негарантирана доставка). ІР протокол, ІР пакети.

*Докато комитетите на ISO са спорили за своите стандарти, се променя цялата концепция на мрежите и по цял свят се внедрява **TCP/IP**.*

Днес OSI постепенно се движи към състояние да се окаже на страниците на историята на компютрите.

Други разработени стекове:

Стек IPX/SPX

Това е оригинален стек от протоколи на фирмата Novell, разработен за собствената мрежова операционна система NetWare в началото на 80-те. Протоколите Internetwork Packet Exchange (IPX) и Sequenced Packet Exchange (SPX), които дават името на стека, са адаптация на протоколите XNS на фирмата Xerox, разпространени в много по-малка степен от IPX/SPX.

Стек NetBIOS/SMB

Фирмите Microsoft и IBM съвместно разработват стека протоколи NetBIOS/SMB.

Средствата NetBIOS се появяват през 1984 г. като мрежово разширение на стандартните функции на базовата система за вход/изход (BIOS) IBM PC за мрежовите програми PC Network на фирмата IBM, която на приложния слой се е използвала за реализация на мрежовите услуги на протокола SMB (Server Message Block).

*През 90-те години става ясно, че **конкуренцията при мрежите следва да се замени с тяхното обединение**. Възниква Интернет. От това печелят и производители и ISP (повече информационни ресурси, повече услуги, повече клиенти).*

Следва да се различават моделът OSI и стекът OSI. Моделът OSI е концептуална схема за взаимодействие на отворени системи, а стекът OSI е набор от конкретни протоколи. За разлика от протоколните стекове, стекът OSI напълно съответства на модела OSI.

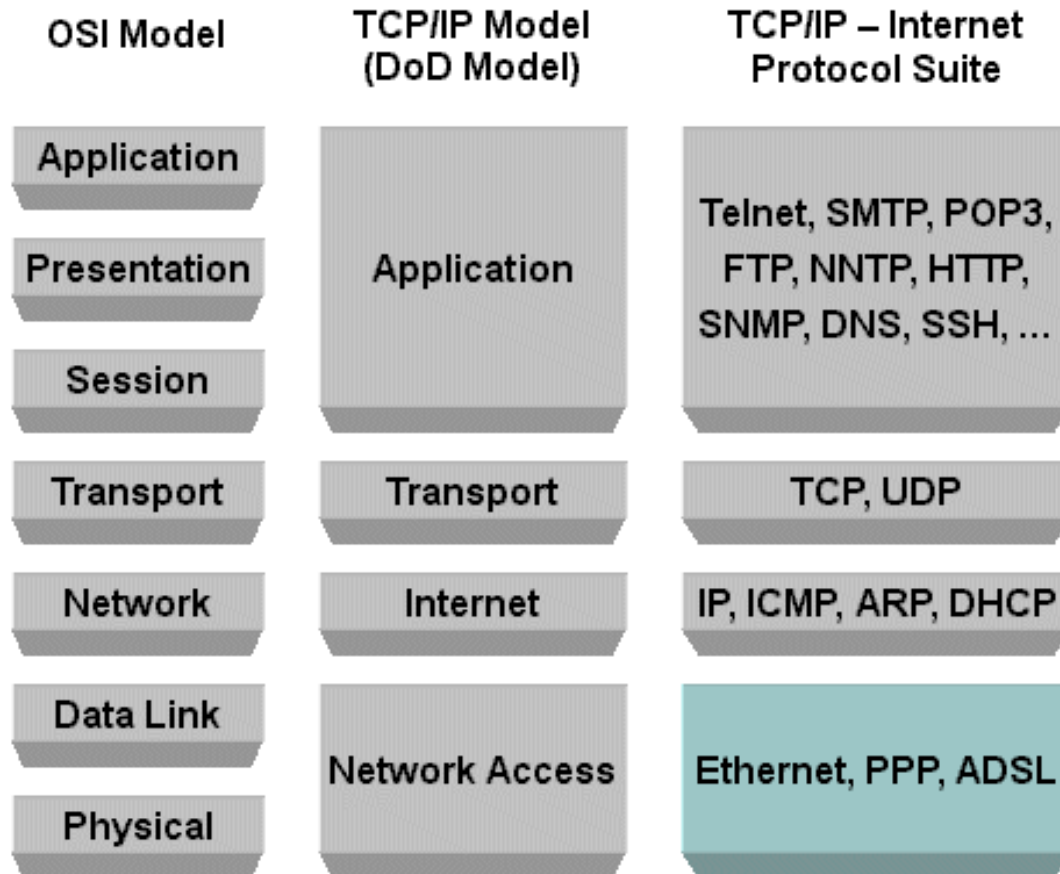
Протоколите на стека OSI са сложни и с нееднозначна спецификация на свойствата. Те са резултат от общата политики на разработчиците на стека, които са се стремили в своите протоколи да отчетат всички възможни варианти и всички съществуващи и появяващи се технологии. Също така са направени голям брой политически компромиси, неизбежни при приемането на международни стандарти по такъв злободневен въпрос, като построяване на отворени изчислителни мрежи.

Отворени, в смисъл - публикувани, общостъпни спецификации, съответстващи на стандартите и приети в резултат на достигнато съгласие след всестранно обсъждане от всички заинтересовани страни.

Стекът OSI е международен, независим от производителите стандарт.

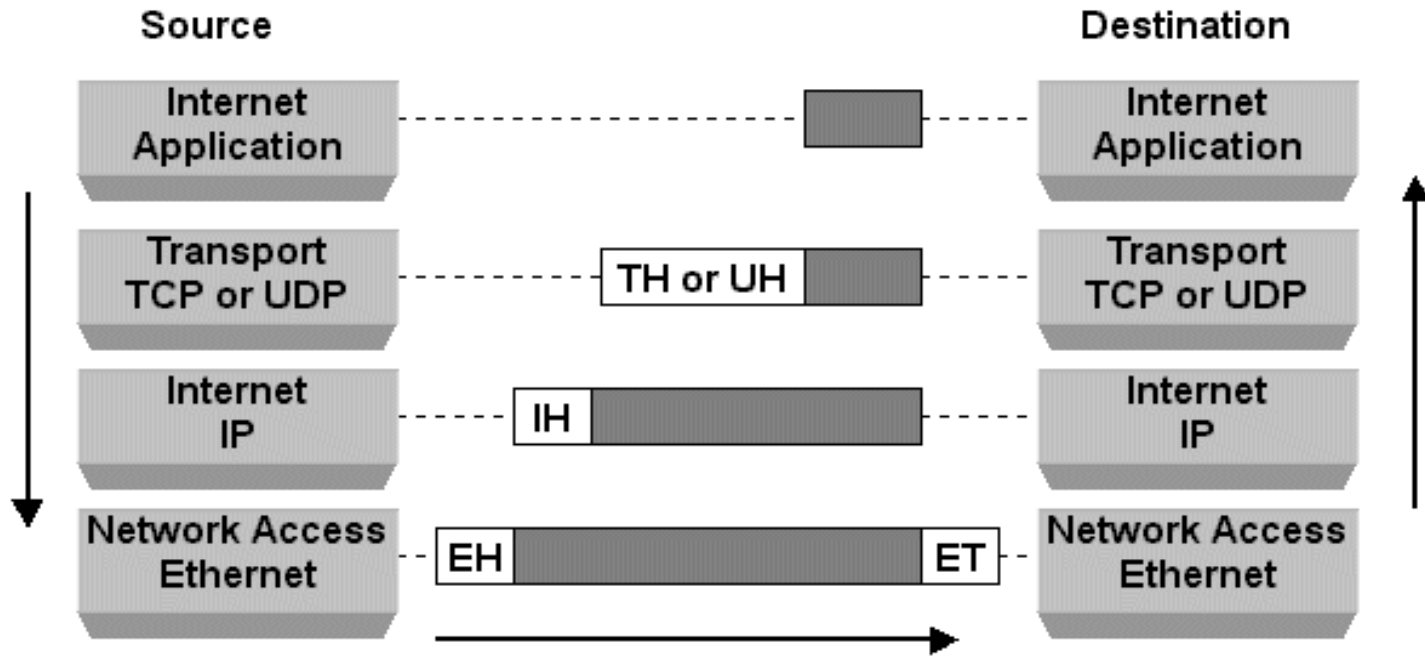
Един от най-големите производители, поддържащ OSI, е компанията AT&T. Нейната мрежа Stargroup се базира на този стек.

OSI vs. TCP/IP



DoD — *department of defense*

ТСР/ІР – мрежи, протоколи и услуги



Както Интернет, така и транспортният слой е подобен на OSI.

TCP (Transmission Control Protocol) е connection-oriented. Потокът от байтове да бъде доставен без грешка. Съобщението се разбива на сегменти.

UDP (User Datagram Protocol) е connectionless за обмен на звук, къси съобщения: NTP, TFTP, SNMP.

Сравнение на OSI и TCP/IP

Общи свойства: единен стек от независими протоколи, подобни функции.

Три основни свойства на OSI:

- Дефиниране на услуги;
- Дефиниране на интерфейси;
- Дефиниране на протоколи.

Основно предимство на OSI: прави разграничение между тези три свойства.

TCP/IP няма точно разграничение между трите.

Протоколите в OSI са по-добре обособени, отколкото в TCP/IP. Могат да бъдат заменяни по-лесно.

Сравнение на OSI и TCP/IP

OSI – преди да е създадена концепцията за протоколите – **достатъчно общ**. Липса на опит с конкретни обекти – недостатъчна функционалност.

Канален слой – за връзки “точка-точка”. С поява на LAN – broadcast мрежите – нов подслой.

Подслоеве да бъдат изменяни в зависимост от различията в конкретните мрежи.

OSI създателите – всяка държава по една OSI мрежа под управлението на правителството. Не е мислено за междумрежово свързване.

TCP/IP – първо се разработват протоколите. Моделът – реално описание на вече съществуващи протоколи. Т.е. пасват перфектно, без да е необходима да са напасвани към модела, както при OSI.

Сравнение на OSI и TCP/IP

TCP/IP не е приложим за описание на мрежи, които не поддържат TCP/IP. Но днес всички производители го поддържат. Такива със собствени протоколни стекове. Novell се отказва от SPX/IPX, Apple – от AppleTalk, Microsoft – от NetBIOS и др.

Т.е. TCP/IP става световен мрежов стандарт.

Други разлики:

- На мрежово ниво - TCP/IP е само connectionless; OSI – и connection oriented.
- На транспортно ниво - OSI е само connection oriented; а TCP/IP – и двете (TCP и UDP).

Проблеми на Интернет

- Ограничено адресно пространство
- Препълване на маршрутните таблици и ограничена пропускателна способност на каналите на опорните мрежи
- Ограничени възможности за интеграция на услугите и потоците от данни (глас, аудио, видео и други)
- Недостатъчно ефективни механизми за осигуряване на гарантирано качество на услугите
- Лоша защита на информацията

Идва ли краят на ТСП/ІР?

Интернет може да стане по-бърза и по-сигурна, като се изостави концепцията за пакети и корекцията на грешките, която забавя трафика заради повторните предавания.

Това предполагат учени от Aalborg University, Дания, в сътрудничество с MIT и Caltech (California Institute of Technology).

Те искат да заменят сегашния модел със система от линейни уравнения.

Методиката

Методиката се базира на мрежовото кодиране и декодиране, по-точно RLNC (**Random Linear Network Coding** – Случайно линейно мрежово кодиране)

Все едно, колите навлизат в кръстовище от всички посоки, без да се налага да се изчакват помежду си или да чакат да им светне “зелено”.

Естествено, без да стават катастрофи :)

4-минутно видео се “сваля” 5 пъти по-бързо отколкото по традиционната технология.

Как работи

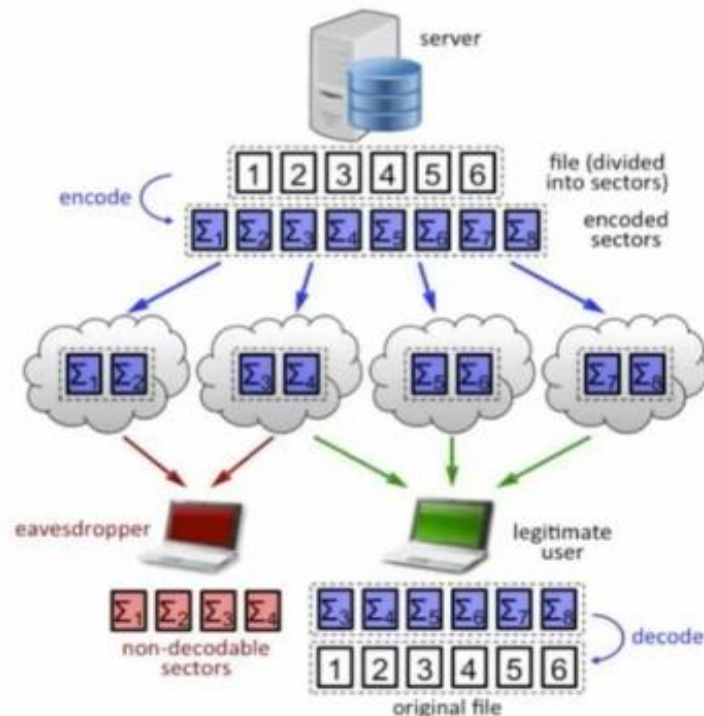
На съдържанието на пакета се гледа като на отделно число. Всеки възел в мрежата създава система от линейни уравнения от числата, извлечени от съдържанието на пакетите и множество от случайно генерирани коефициенти.

Ако си спомняте от гимназиалния курс по математика, необходими са N линейни уравнения, за да се намерят стойностите на N неизвестни.

Всеки кодиран пакет съдържа по едно уравнение. Т.е. на получателя му трябва N пакета (с различни коефициенти), за да може да декодира данните.

По-сигурна система. На “подслушвачите” ще им трябва да прихванат всички пакети, за да декодират информацията

Coding as a Measure Content of Protection



Противниците

На нас ни предстои да се занимаваме с модела ТСР/IP.
Рано е да се каже, че е за изхвърляне. Трябва да отбележим, че пакетите не е задължително да бъдат подредени (протоколът ТСР няма такова изискване). Сегментите, на които се разделя дадено съобщение, се номерират.

Благодарение на прозоречни механизми и др. техники, повторните предавания (ако се наложи) не забавят скоростта.

Нека имаме предвид и високото бързодействие на днешните интегрални схеми.

Все пак предлаганата технология сигурно ще намери приложение в **5G** мобилни мрежи, сателитни комуникации и **Internet of Things**.

Вече в Силиконовата долина

RLNC технологията е патентована и “опакована” в C++ софтуер от фирмата Steinwurf с марката Kodo. Steinwurf планира да я продава на хардуерни производители.

Steinwurf е основана от професор Frank Fitzek от Aalborg University и двама негови бивши студенти заедно с американските им колеги.

Компанията вече има офис и в Силиконовата долина, но управлението ѝ е все още в Aalborg.