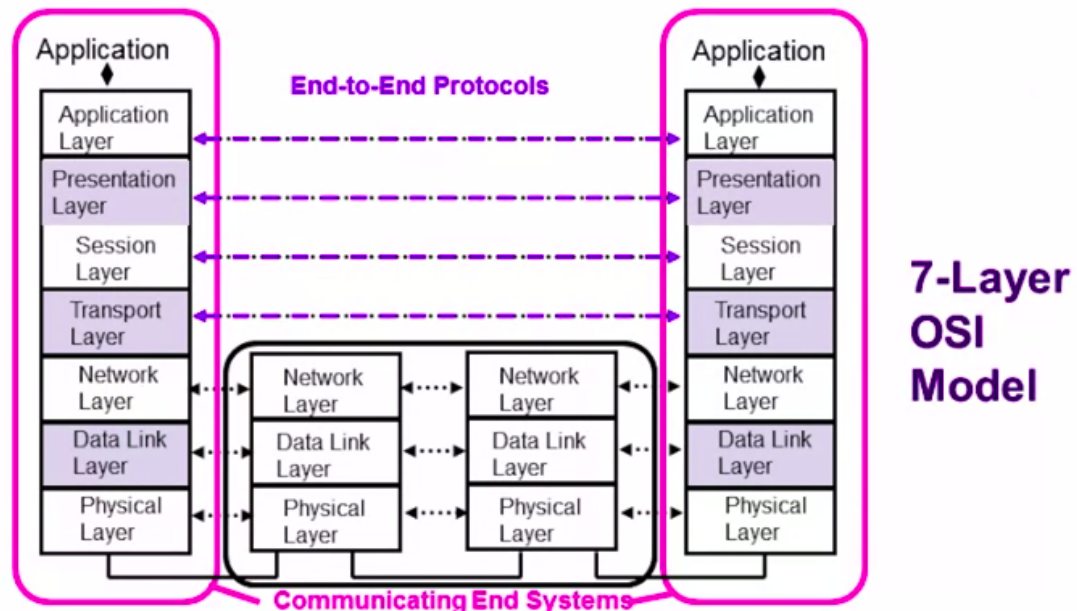


Layered Architecture

Layered Architecture in Networking

OSI Model



Why layering

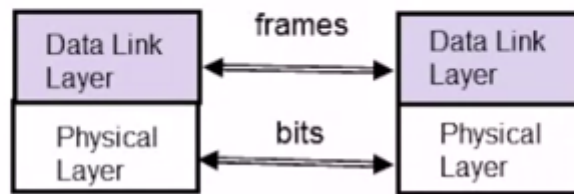
- Layering simplifies design, implementation, and testing by partitioning
- Protocol in each layer can be designed separately from those in other layers
- Protocol makes “calls” for services from layer below
- Layering provides flexibility for modifying and evolving protocols and services
- Non-layered architectures are costly, inflexible, and soon obsolete

Physical Layer

- Transfers bits across a link
- Definition & specification of the physical aspects
 - Mechanical: cable, plugs, pins...
 - Electrical/optical: modulation, signal strength, voltage levels
 - functional/procedural: how to activate, maintain, and deactivate physical links...
- Ethernet, DSL, cable modem, telephone modems...
- Twisted-pair cable, coaxial cable optical fiber, radio, ...

Data Layer

- Transfers *frames* across *direct* connections
 - Groups bits into frames
 - Detection of bit errors; Retransmission of frames
- Activation, maintenance of data link connections
- Medium access control for local area networks
- *Node-to-node* flow control



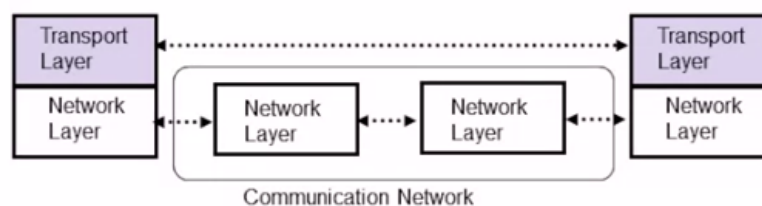
Network Layer

Ip layer operates at network layer

- Transfers *packets* across multiple links and/or multiple networks
 - *Addressing* must scale to large networks
 - Nodes execute *routing* algorithm to determine paths across the network
 - Routing protocol means the procedure used to select routing paths
 - *Forwarding* transfers packet across a node
 - *Congestion control* to deal with traffic surges
 - Most complex layer in the OSI reference model

Transport Layer

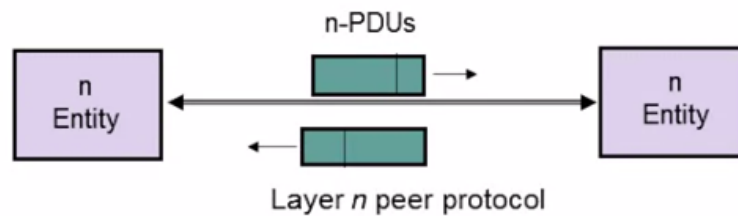
- Transfers *segments* end-to-end from process in a machine to process in another machine
 - *Reliable* stream transfer or quick-and-simple single-block transfer
 - Message segmentation and reassembly
 - Connection setup, maintenance, and release



Session and Presentation layered are grouped as application layer

Unified OSI view

- The machines at the same layer use a set of precise and unambiguous rules called the *layer- n protocol*.
- Layer- n peer processes communicate by exchanging *Protocol Data Units (PDUs)*



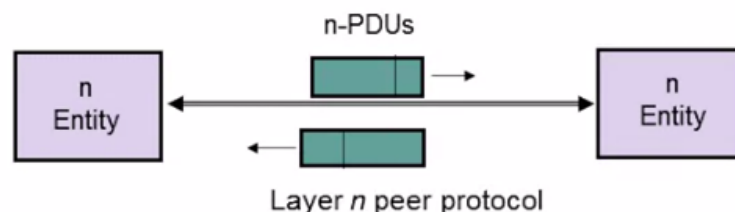
OSI Unified view : layers

- A layer is a set of related communication functions managed and grouped together
- Layer n in one machine interacts with layer n in another machine to provide a service to its upper layer $n + 1$
- The entities comprising the corresponding layers on different machines are called *peer processes*.

The processes at layer n are referred to as layer n entities.

OSI Unified view : protocols

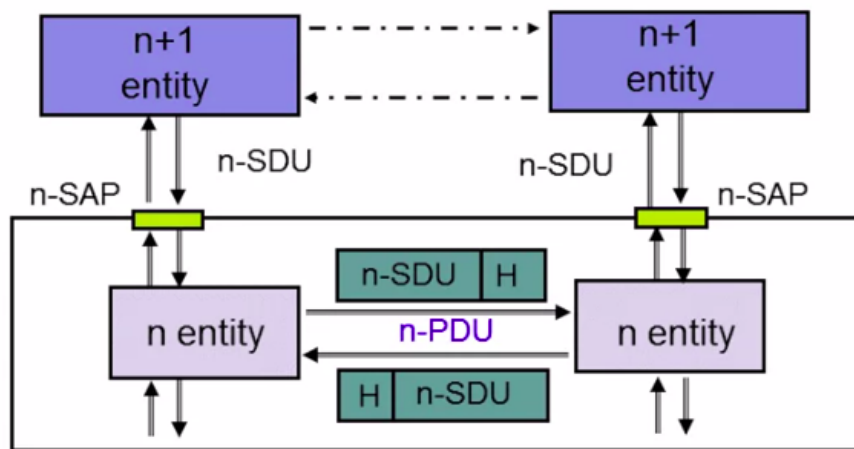
- The machines at the same layer use a set of precise and unambiguous rules called the *layer- n protocol*.
- Layer- n peer processes communicate by exchanging *Protocol Data Units (PDUs)*



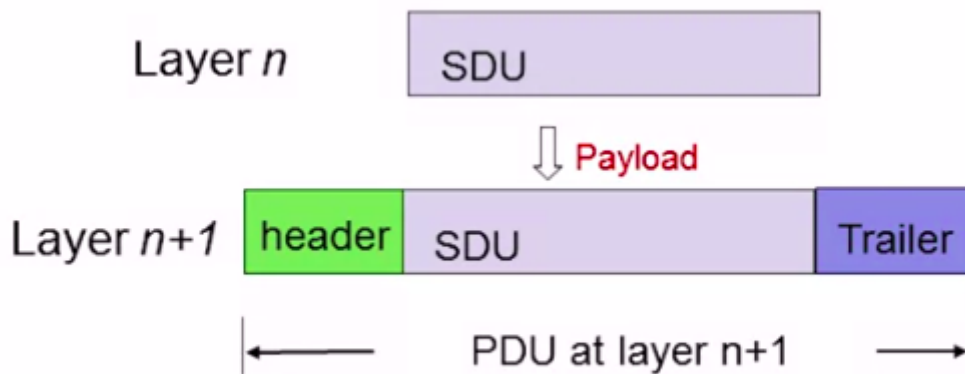
OSI Unified view : Services

- Communication between peer processes is virtual and actually indirect
- Layer $n+1$ transfers information by invoking the services provided by layer n
- Services are available at *Service Access Points (SAP's)*
- Each layer passes data & control information to the layer below it until the physical layer is reached and transfer occurs
- The data passed to the layer below is called a *Service Data Unit (SDU)*; SDU's are *encapsulated* in PDU's

Layers, Services & Protocols

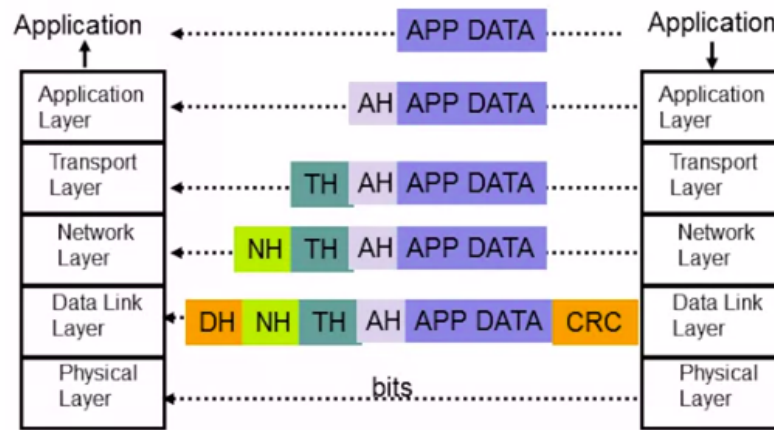


Encapsulation

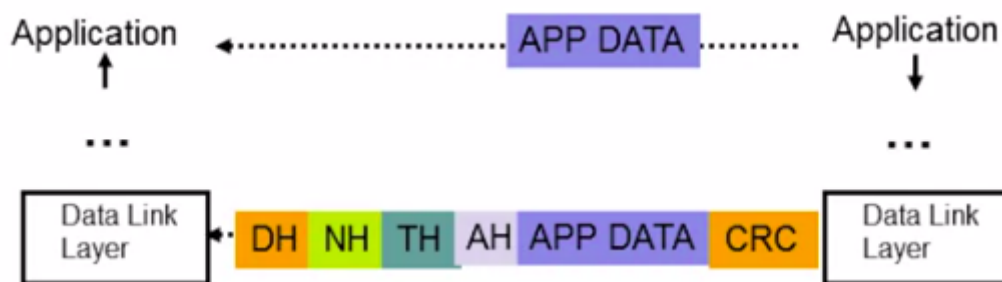


Header & Trailer

- Each protocol uses a header that carries addresses, sequence #...

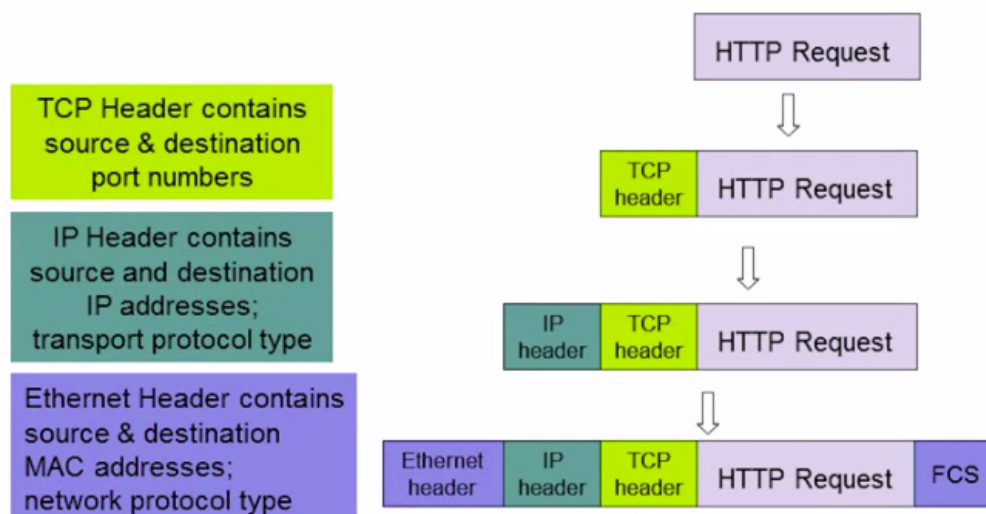


Bandwidth



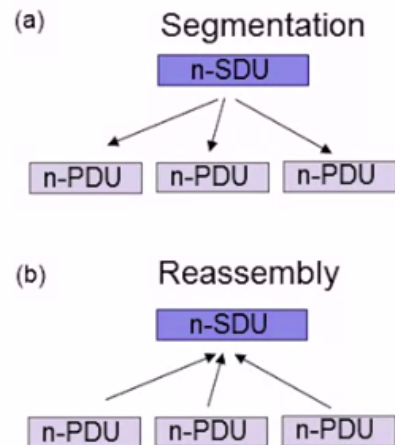
$$\text{Utilization} = \frac{\text{APP DATA}}{\text{APP DATA} + \text{HEADERS} + \text{CRC}}$$

Encapsulation in TCP/IP



Segmentation and Reassembly

- A layer may impose a limit on the size of a data block that it can transfer
- Thus a layer- n SDU may be too large to be handled as a single unit by layer- $(n-1)$
- Sender side: SDU is segmented into multiple PDUs
- Receiver side: SDU is reassembled from sequence of PDUs



Connection-Oriented and Connection-less service

λ Connection-Oriented λ Connectionless

λ Three-phases:

1. Connection setup between two SAPs to initialize state information
2. SDU transfer
3. Connection release

λ E.g. TCP, ATM

λ Immediate SDU transfer

λ No connection setup

λ E.g. UDP, IP