Layered Architecture

ICT 2156

Contents

- Layers
- Protocols and services
- ISO/OSI Reference Model
- Overview of TCP/IP architecture
- Addresses

Layers

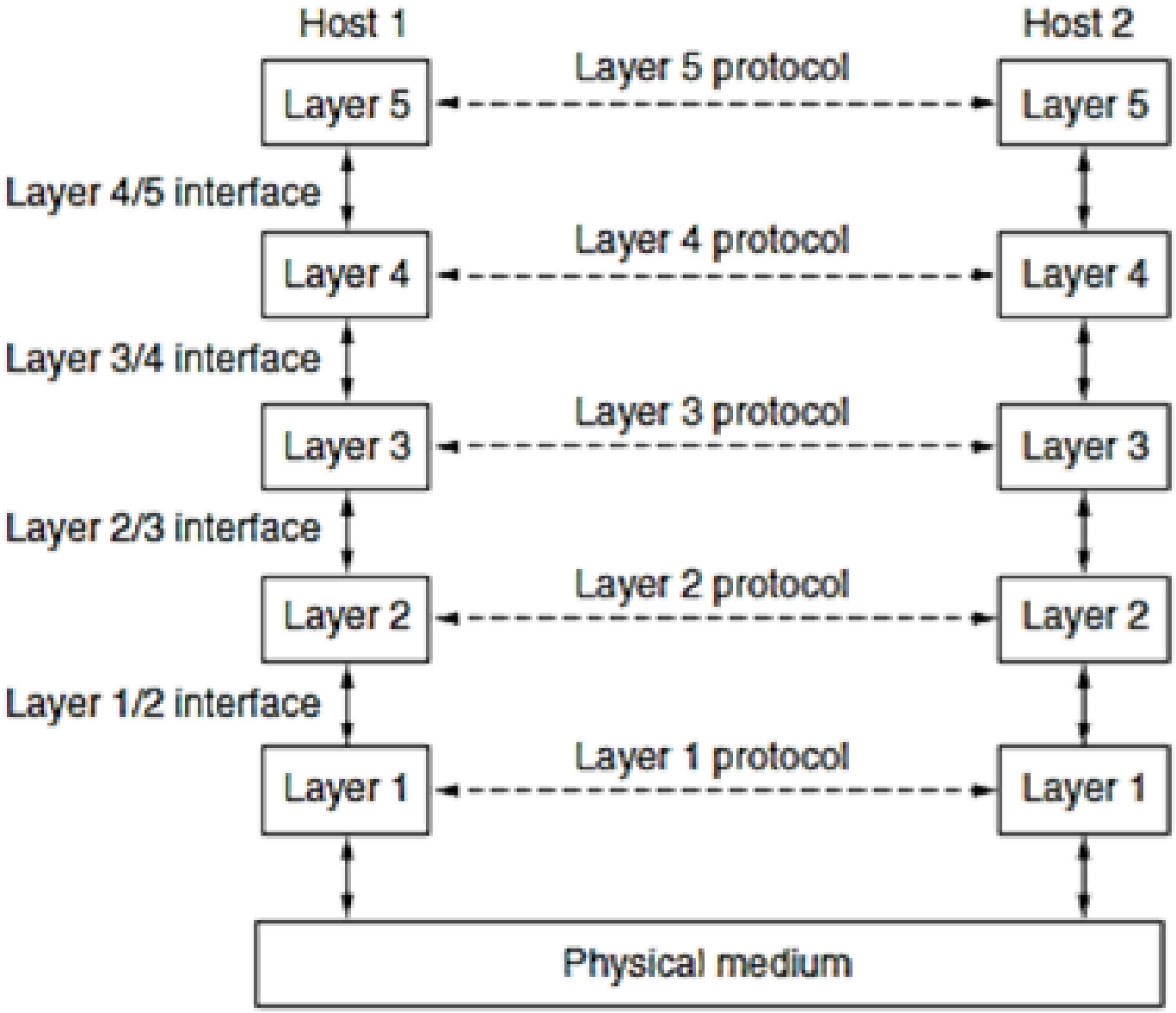
- Why Layers?
  - To reduce the design complexity, most networks are organized as a stack of **layers** or **levels**, each one built upon the one below it to enable communications.
- What is it?
  - The grouping of the communication functions into **related and manageable** sets is called **Layering** and the sets are called as **Layers**.
  - The number of layers, the name of each layer, the contents of each layer, and the function of each layer differ from network to network.

Layers

- Purpose of each layer?
  - To offer certain services to the higher layers while **shielding** those higher layers from the implementation details.
- When layer *n* on one machine carries on a conversation with layer *n* on another machine, the rules and conventions used in this conversation are collectively known as the **layer n protocol**.

Protocols

- A **protocol** is an agreement between the communicating parties on how communication is to proceed.
- It defines:
  - what is communicated,
  - how it is communicated, and
  - when it is communicated.
- **Peers**: Entities comprising the corresponding layers on different machines.
- Peers communicate with each other by means of protocols.
- Through which medium, actual communication occurs?


Host 1
Host 2
Layer 5
Layer 5 protocol
Layer 5
Layer 4/5 interface
Layer 4
Layer 4 protocol
Layer 4
Layer 3/4 interface
Layer 3
Layer 3 protocol
Layer 3
Layer 2/3 interface
Layer 2
Layer 2 protocol
Layer 2
Layer 1/2 interface
Layer 1
Layer 1 protocol
Layer 1
Physical medium

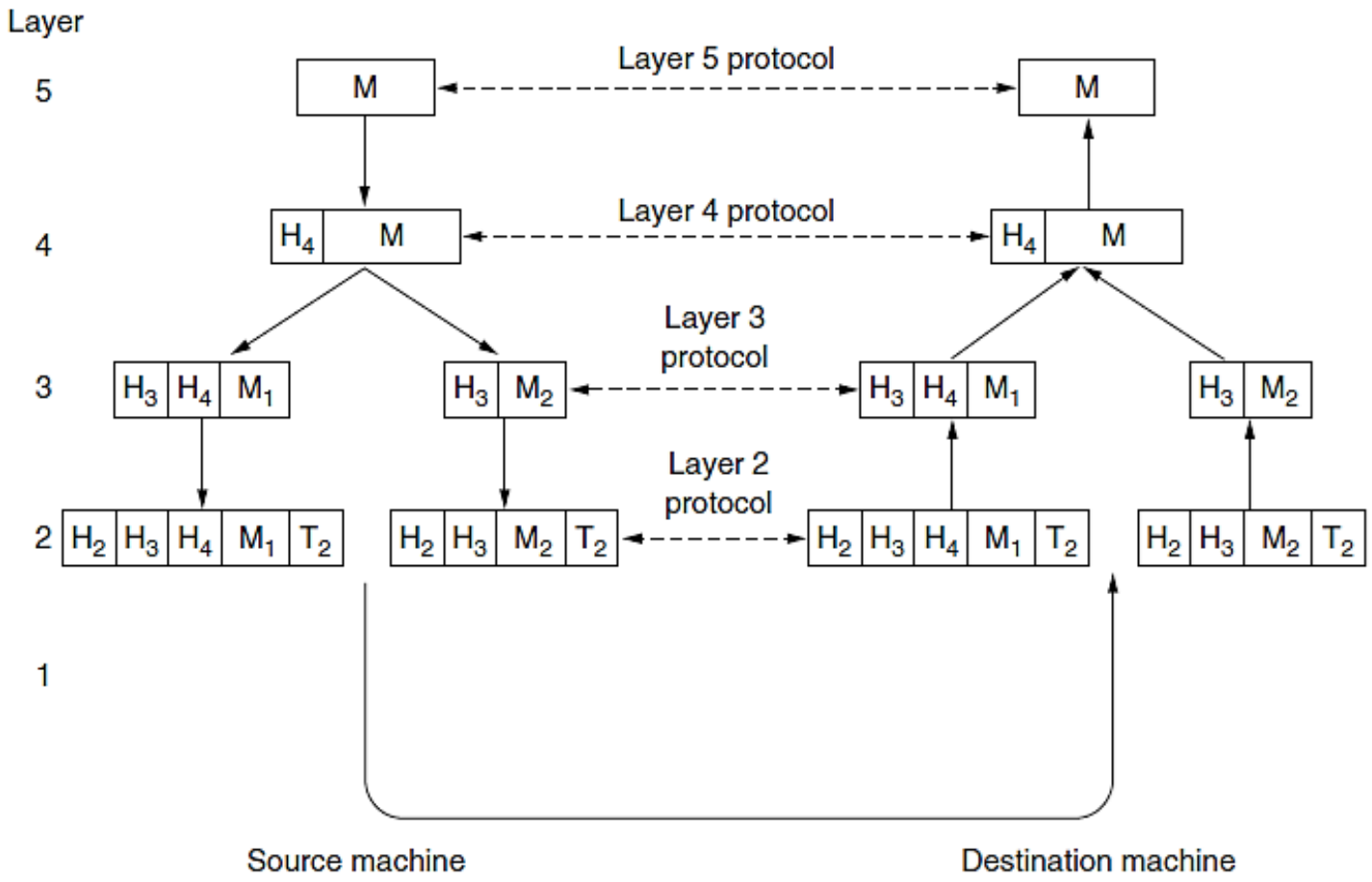
Interfaces

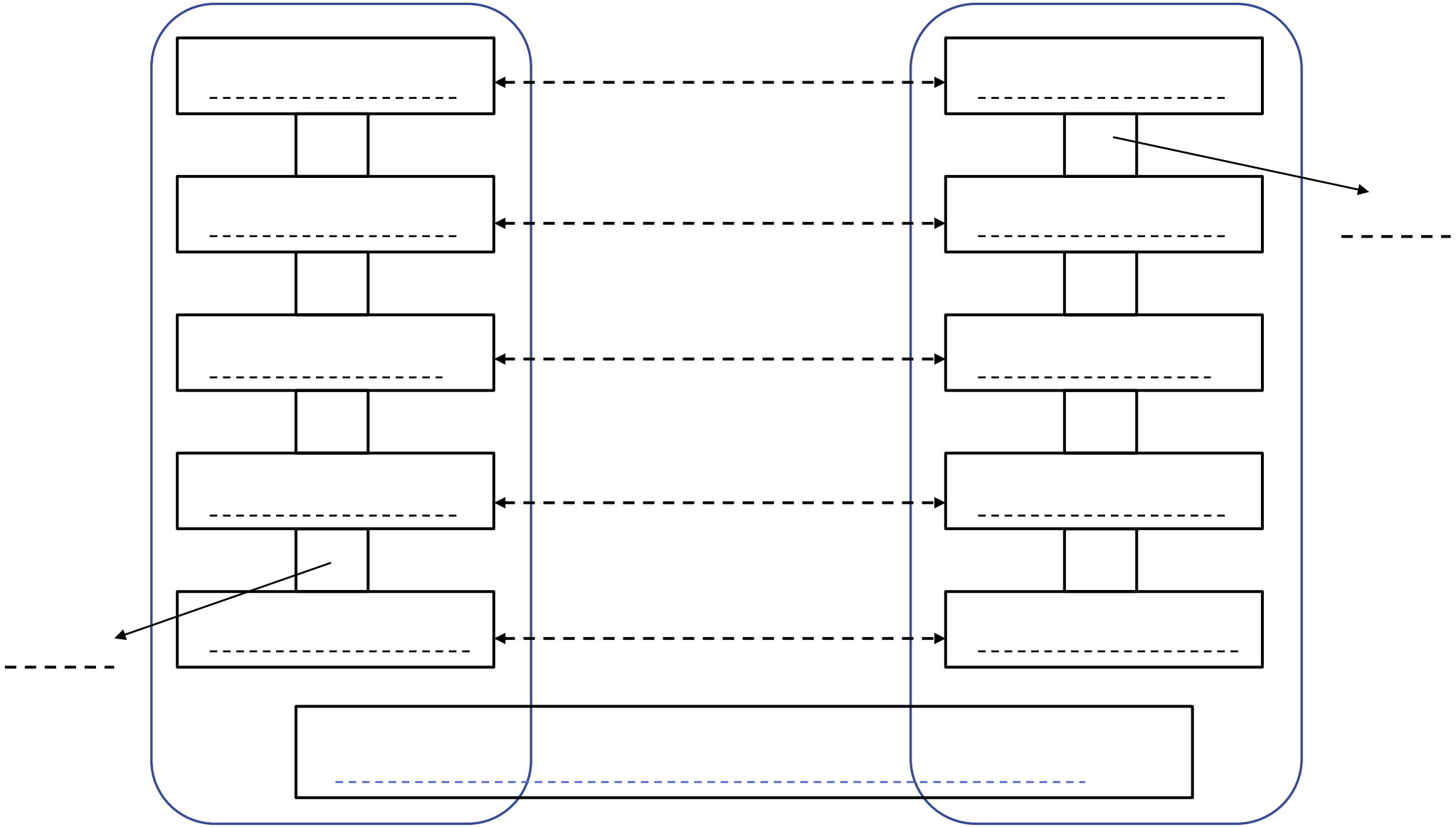
- The **interaction between layers** is done through the definition of the **interface** between layers through which a **service** is requested and results are conveyed.
- The interface defines which **primitive operations and services** the lower layer makes available to the upper one.
- The **abstraction** at the layers is possible through a clearly defined service and interface.
- Clear-cut interface:
  - minimizes the amount of information that is passed between layers
  - Makes it simple to replace one layer with a completely different protocol or implementation.

Terminology

- **Services**: Set of primitives (operations) that a layer provides to the layer above it.
- A set of layers and protocols is called a **network architecture.**
- A list of the protocols used by a certain system, one protocol per layer, is called a **protocol stack**.

Flow of information from one source to destination





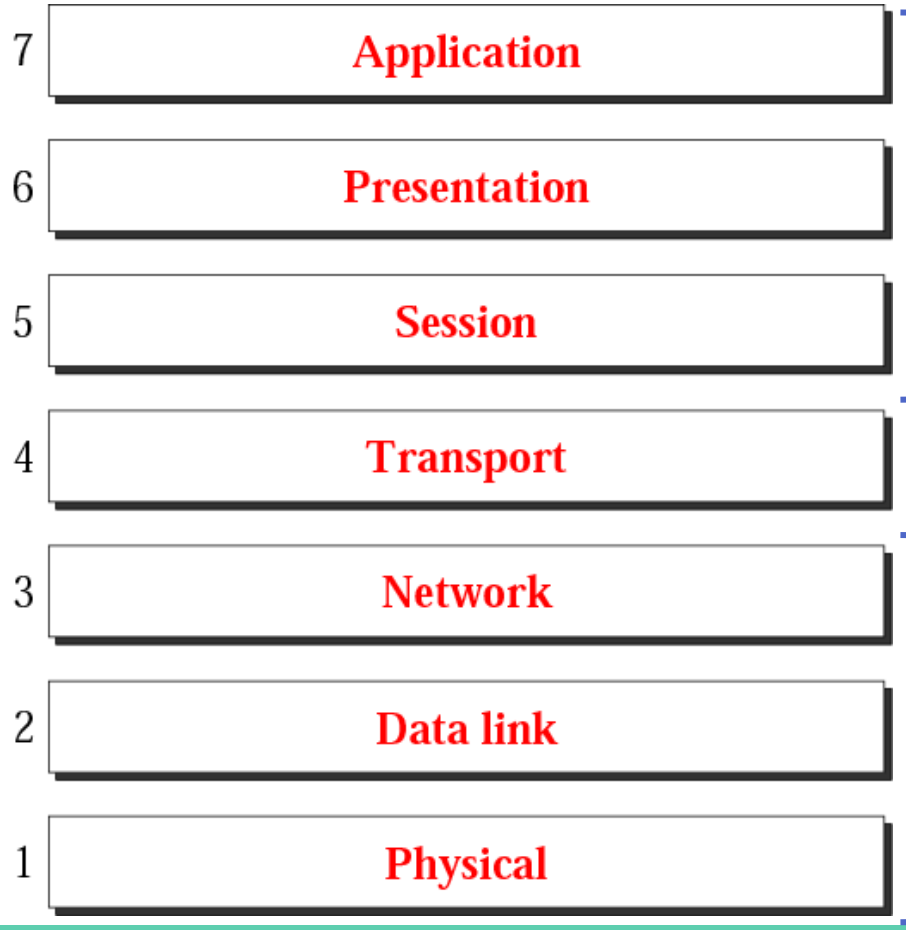
The OSI (Open Systems Interconnection) Model

- **Open system**
  - It is a set of protocols that allows any two different systems to communicate regardless of their underlying architecture.
- **Purpose of the OSI model**
  - To show how to facilitate communication between different systems without requiring changes to the logic of the underlying hardware and software.

❑The OSI model is **NOT a protocol**.

❑It is a model/**conceptual framework** for understanding and designing a network architecture that is flexible, robust, and interoperable.

The OSI Model

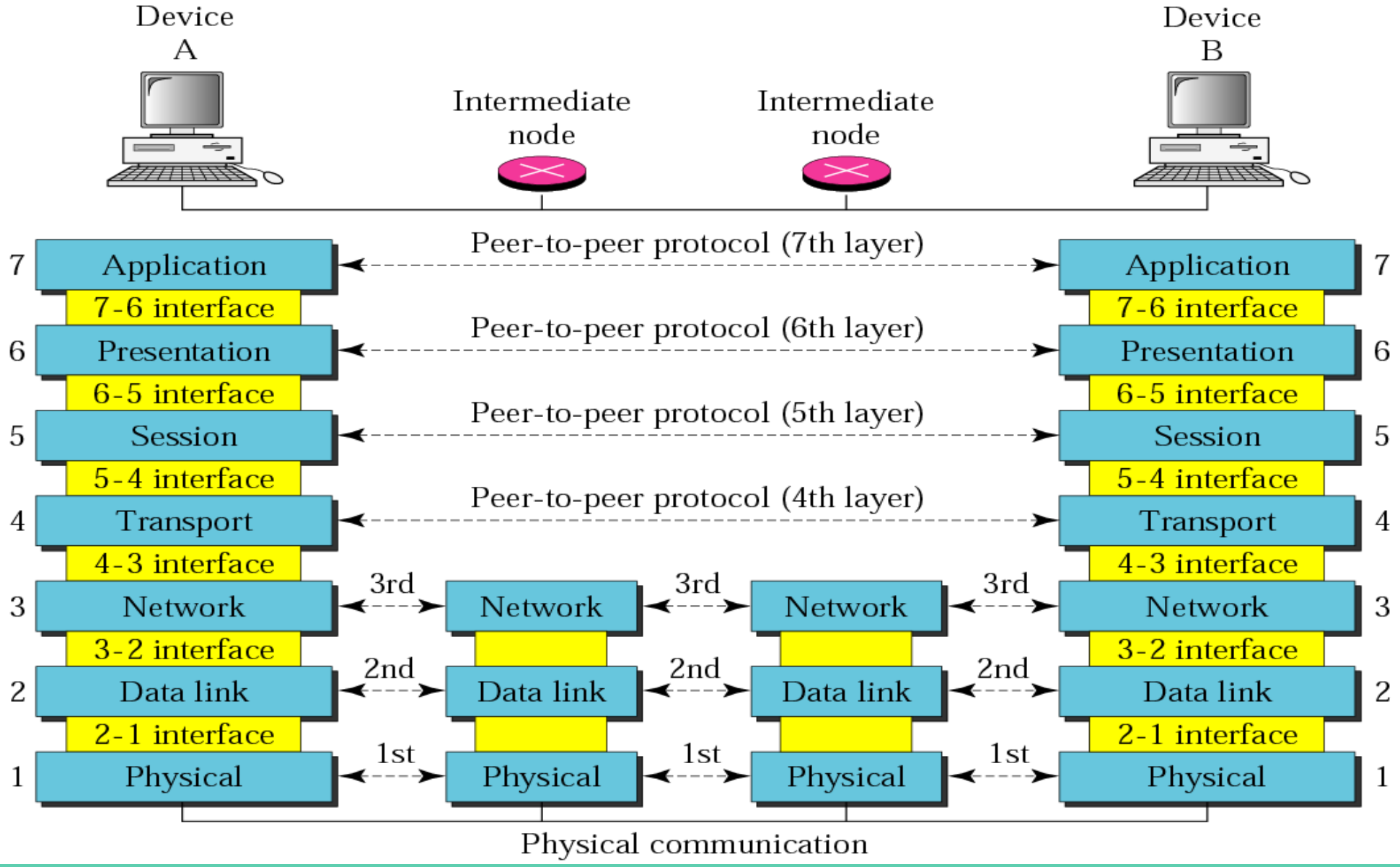


User Support Layers:
They allow interoperability among unrelated software systems.

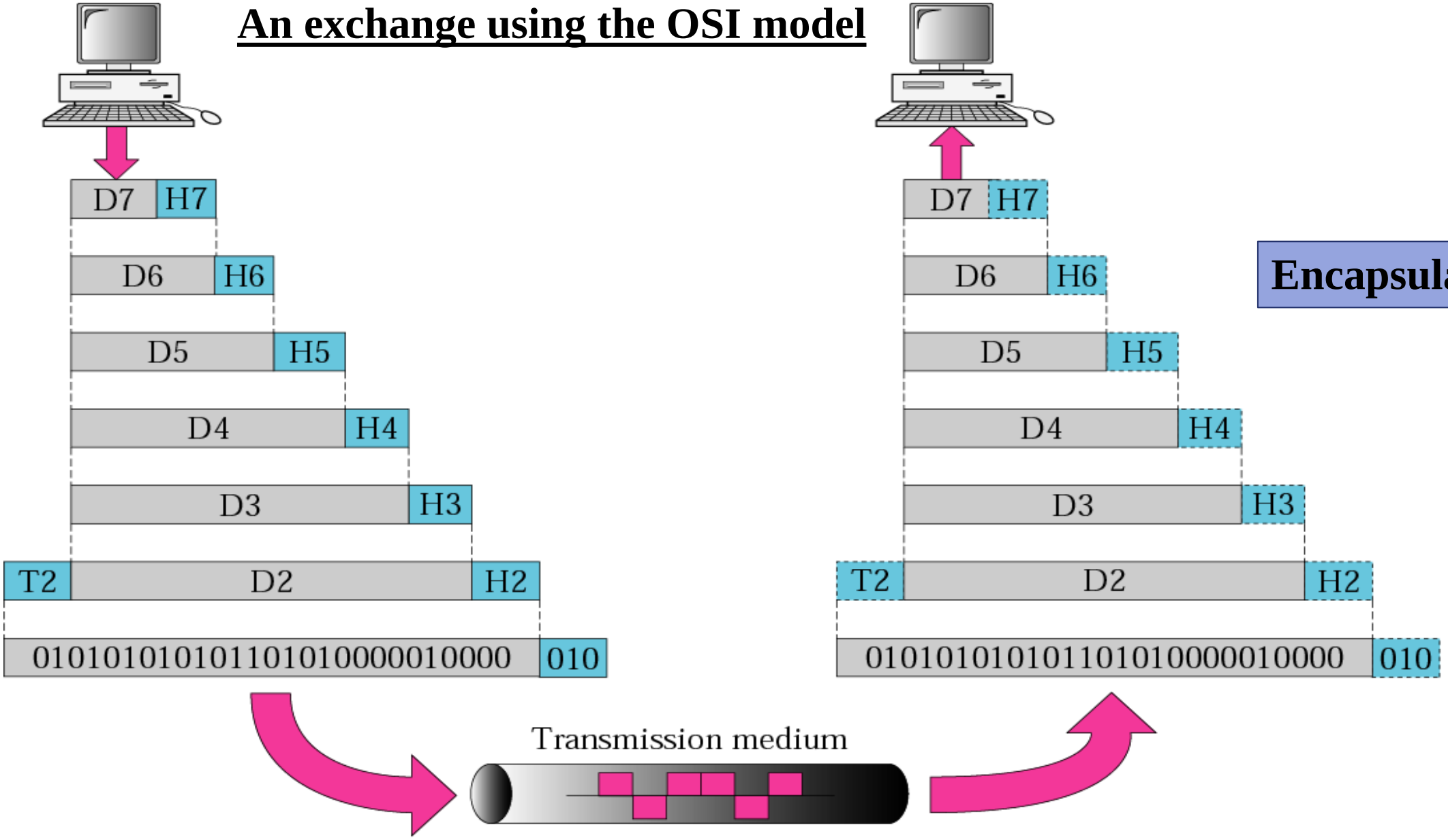
Links the two subgroups.

Network Support Layers:
The physical aspects of moving data from one device to another

OSI layers


Device A
Device B
Intermediate node
Intermediate node
Peer-to-peer protocol (7th layer)
Peer-to-peer protocol (6th layer)
Peer-to-peer protocol (5th layer)
Peer-to-peer protocol (4th layer)
7 Application
7-6 interface
6 Presentation
6-5 interface
5 Session
5-4 interface
4 Transport
4-3 interface
3 Network
3-2 interface
2 Data link
2-1 interface
1 Physical
3rd
2nd
1st
Physical communication

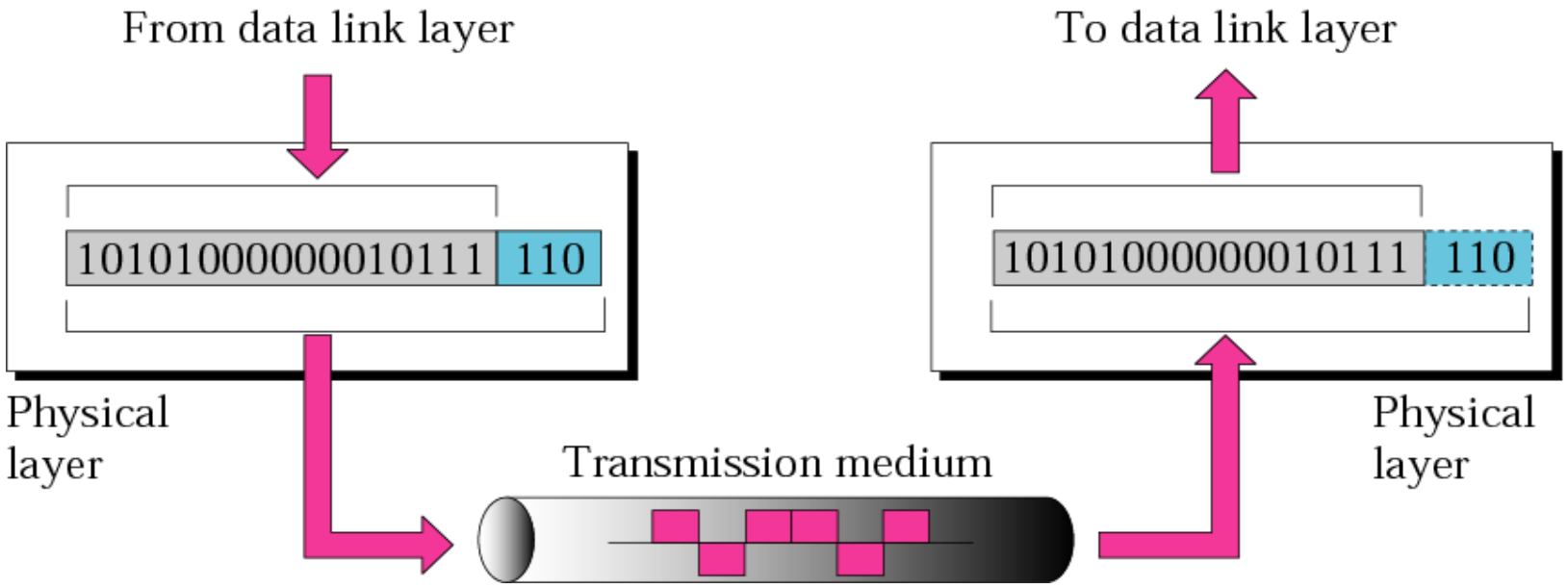
An exchange using the OSI model



Physical Layer

The physical layer is responsible for moving *individual bits* from one hop (node) to the next.

- Coordinates the functions required to carry a bit stream over a physical medium.



Physical Layer Functionalities

- Physical characteristics of interfaces and media.
- Representation of bits: Encoding
- Data rate. (The transmission rate), bit duration?
- Synchronization of bits : Clocks
- Line configuration.
  - point-to-point configuration
  - multipoint configuration
- Physical topology. (Mesh, Star, Ring, Bus)
- Transmission mode. (Simplex, Half-duplex, Full-duplex)

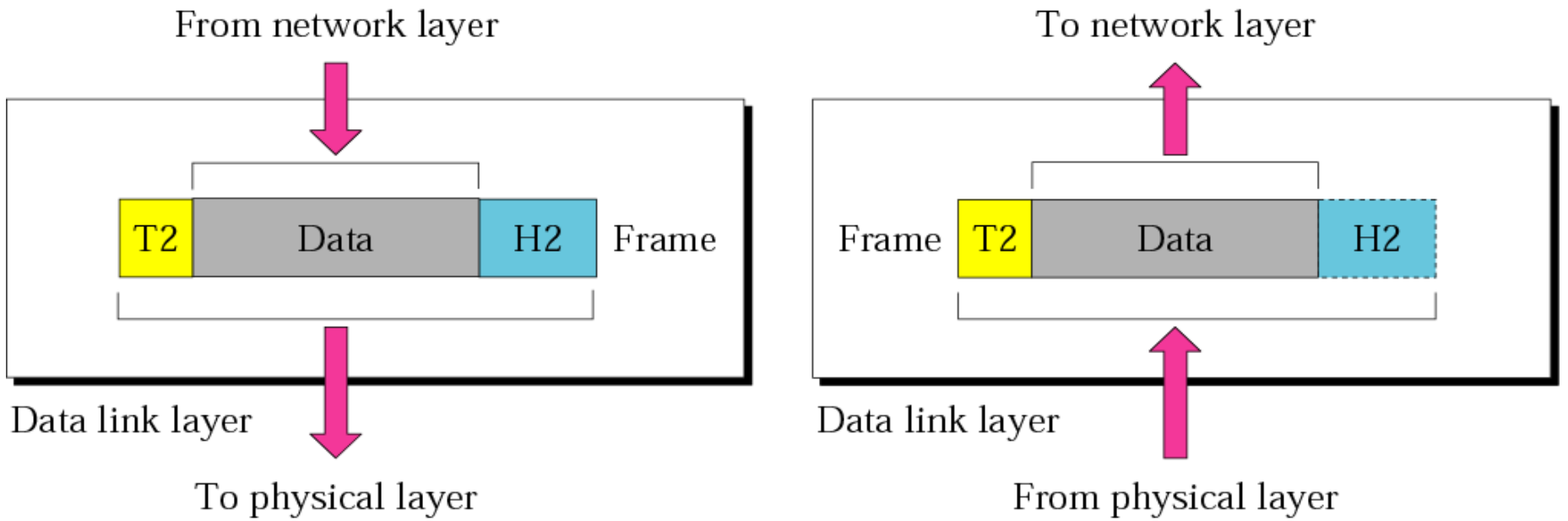
Physical Topology

- Mesh
- Star
- Ring
- Bus

Data Link Layer

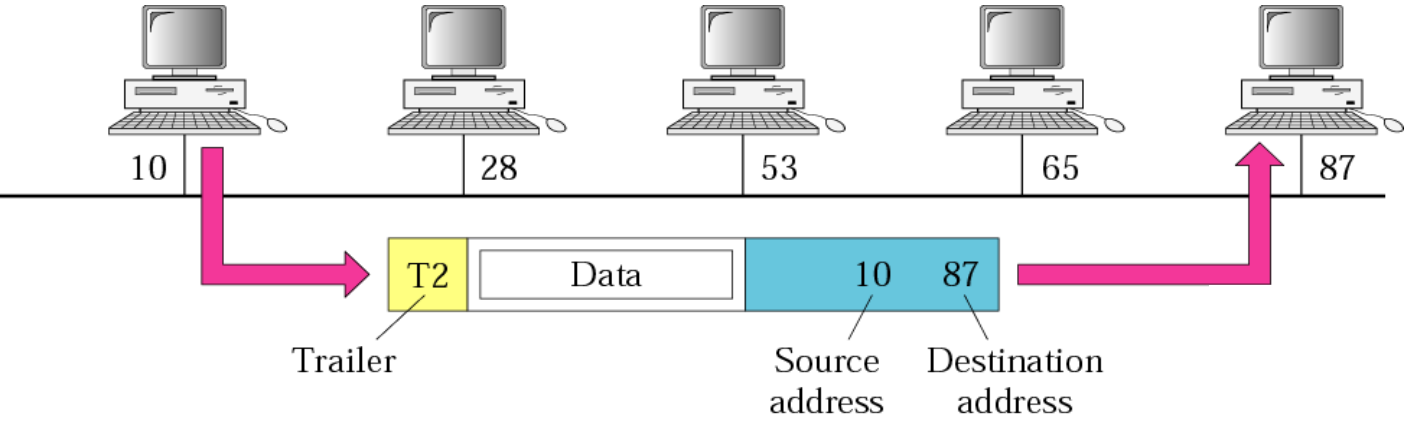
The data link layer is responsible for moving *frames* from one hop (node) to the next.

- Transforms the physical layer, a raw transmission facility, to a reliable link.
- It makes the physical layer appear error-free to the upper layer (network layer).

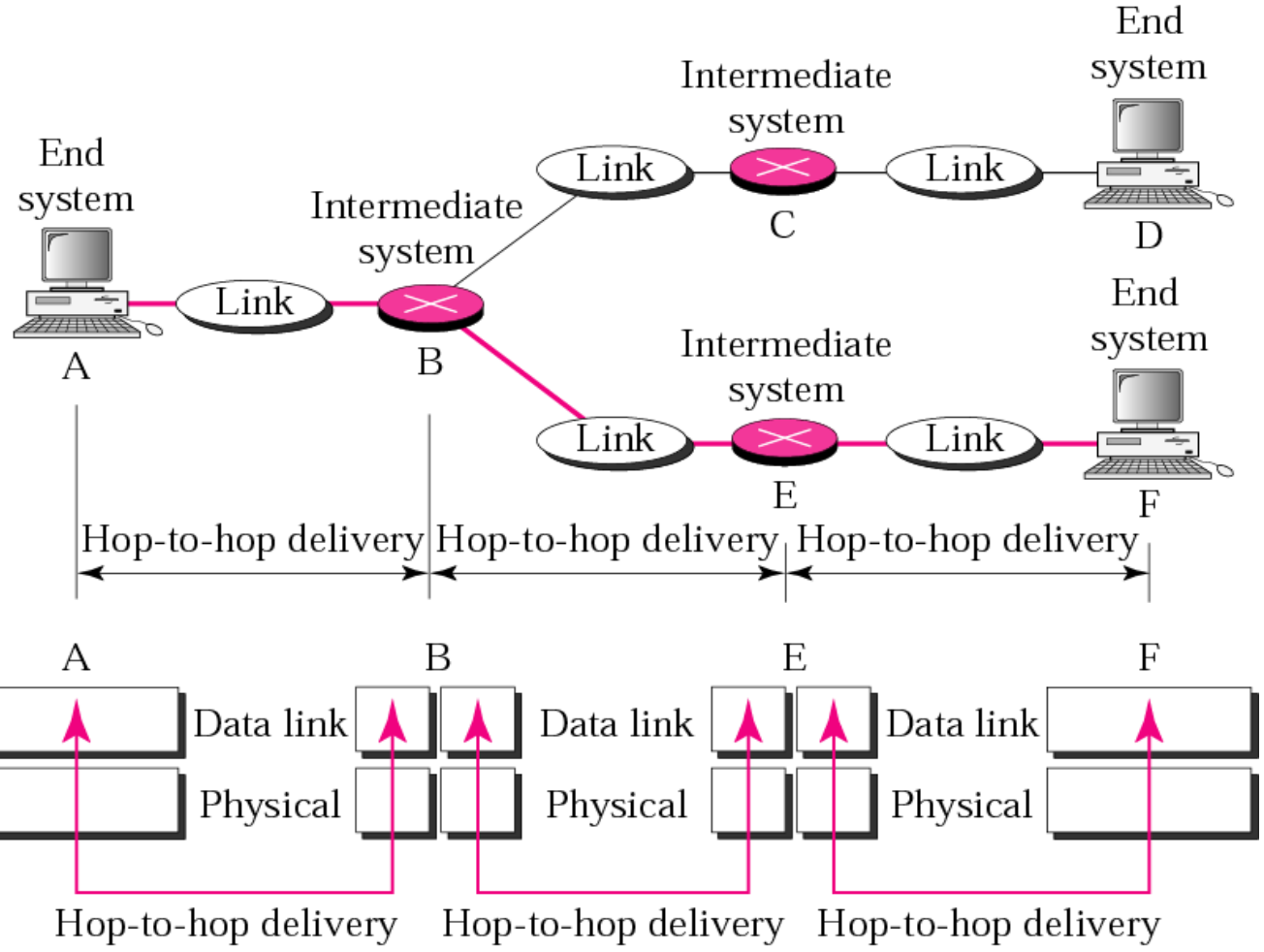


Data Link Layer Functionalities

- Framing.
- Physical addressing.
  - If the frame is intended for a system outside the sender's network, the receiver address is the address of the connecting device that connects the network to the next one.
- Flow control.
- Error control : Adds reliability to the physical layer.
  - Normally achieved through a **trailer** added to the end of the frame.
- Access control.



Physical addresses

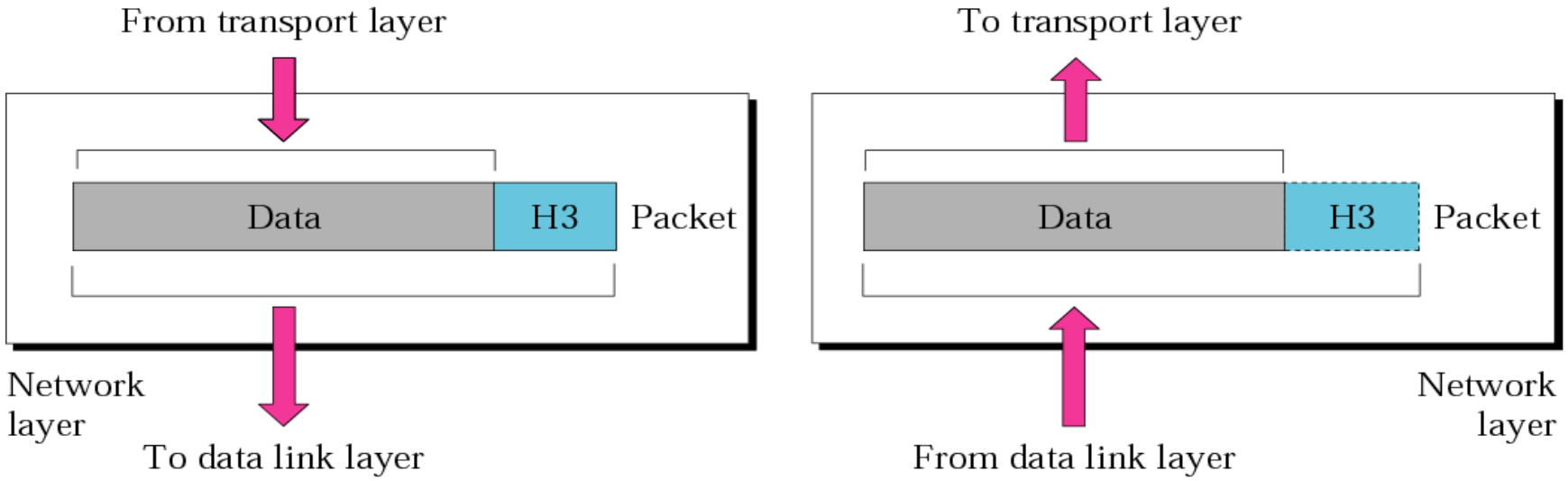


Data Link Layer: Hop-to-hop delivery

Network Layer

The network layer is responsible for the delivery of individual *packets* from the source host to the destination host.

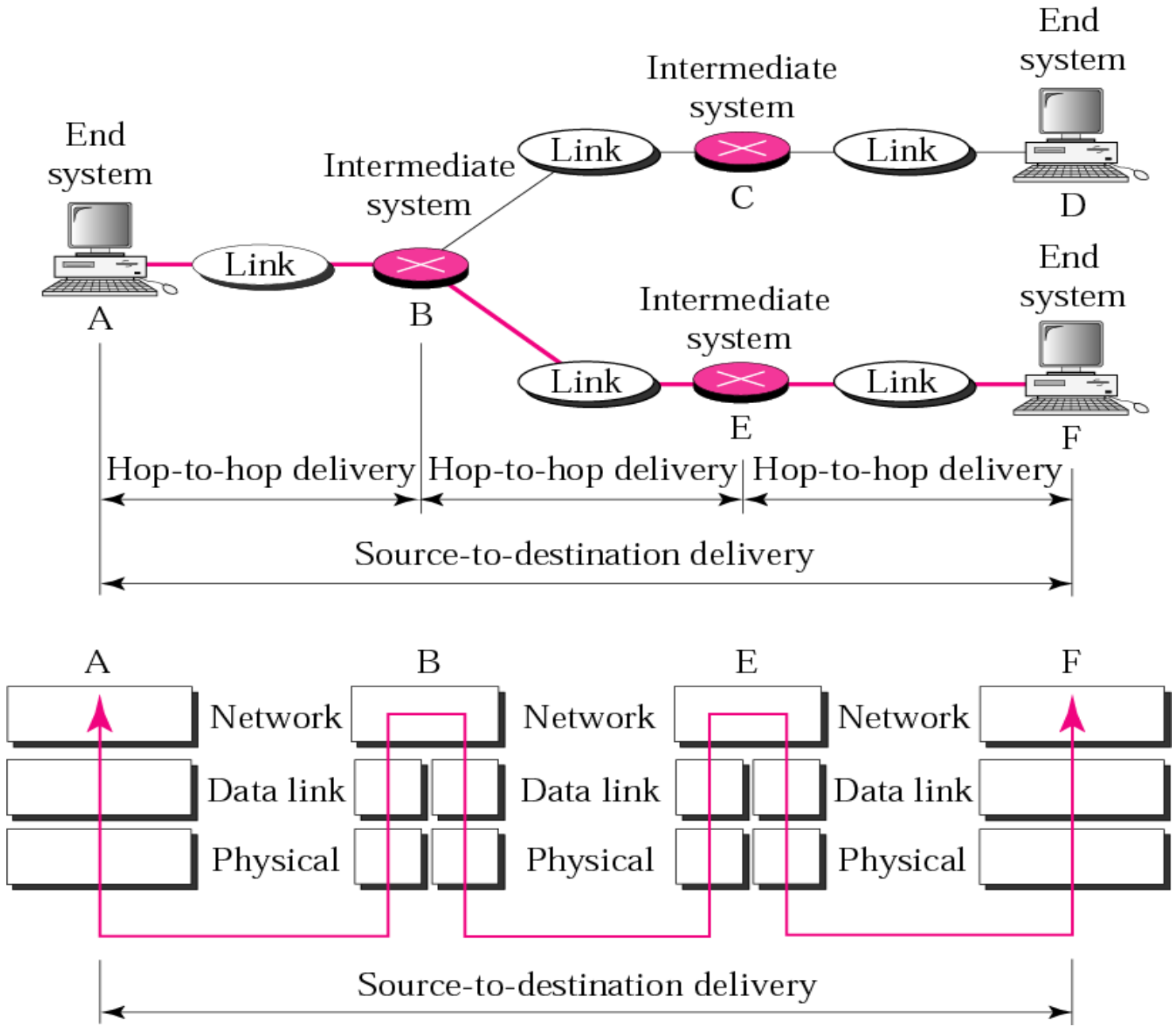
- Responsible for the **source-to-destination** delivery of a packet, possibly across multiple networks (links).
- Ensures that each packet gets from its point of origin to its final destination.
- If two systems are connected to the same link, there is usually no need for a network layer.



Network Layer Functionalities

- **Logical addressing.**
  - The physical addressing implemented by the data link layer handles the addressing problem locally.
  - If a packet passes the network boundary, we need another addressing system to help distinguish the source and destination systems.
  - NL adds a header to the packet: includes the **logical addresses** of the sender and receiver.
- **Routing.**

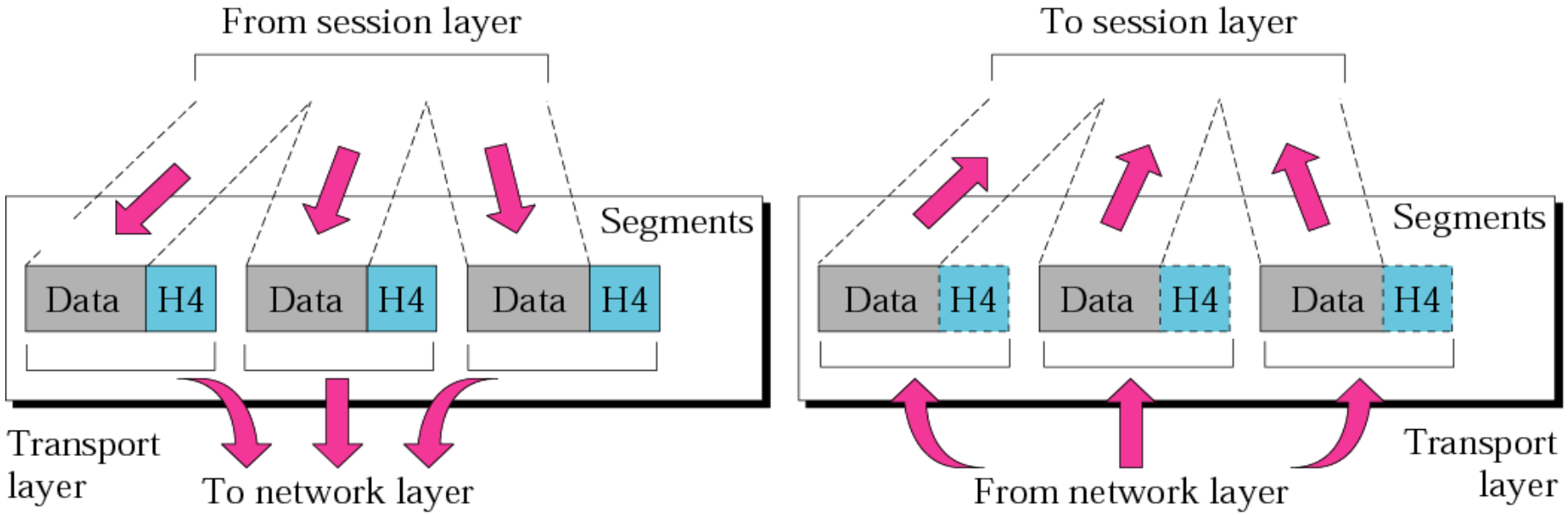
Network Layer:
Source-to-destination delivery



Transport Layer

The transport layer is responsible for the delivery of a *segment* from one process to another.

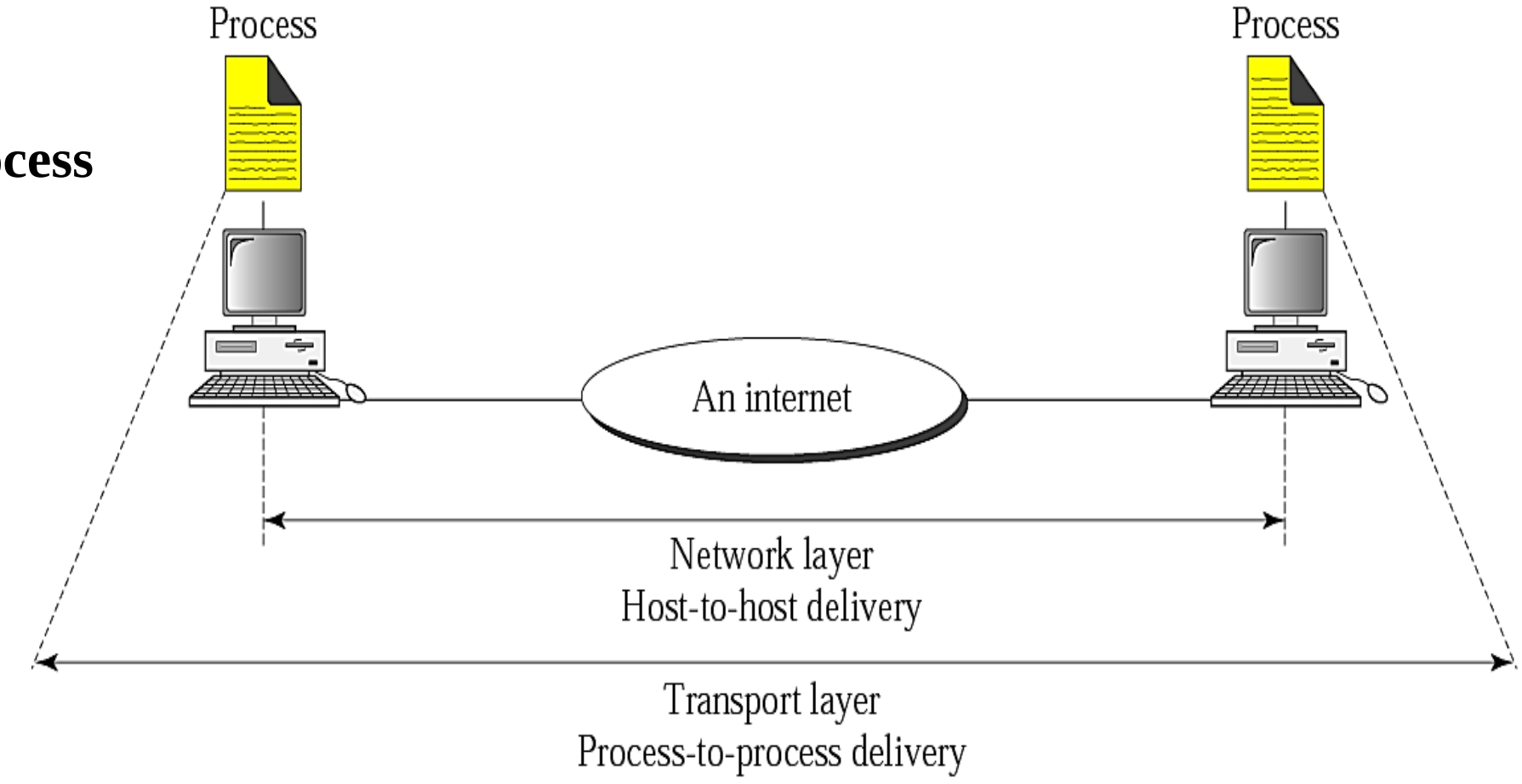
- Responsible for **process-to-process delivery** of the entire message.
- Ensures that the whole message arrives intact and in order, overseeing both error control and flow control at the source-to-destination level.



Transport Layer Functionalities

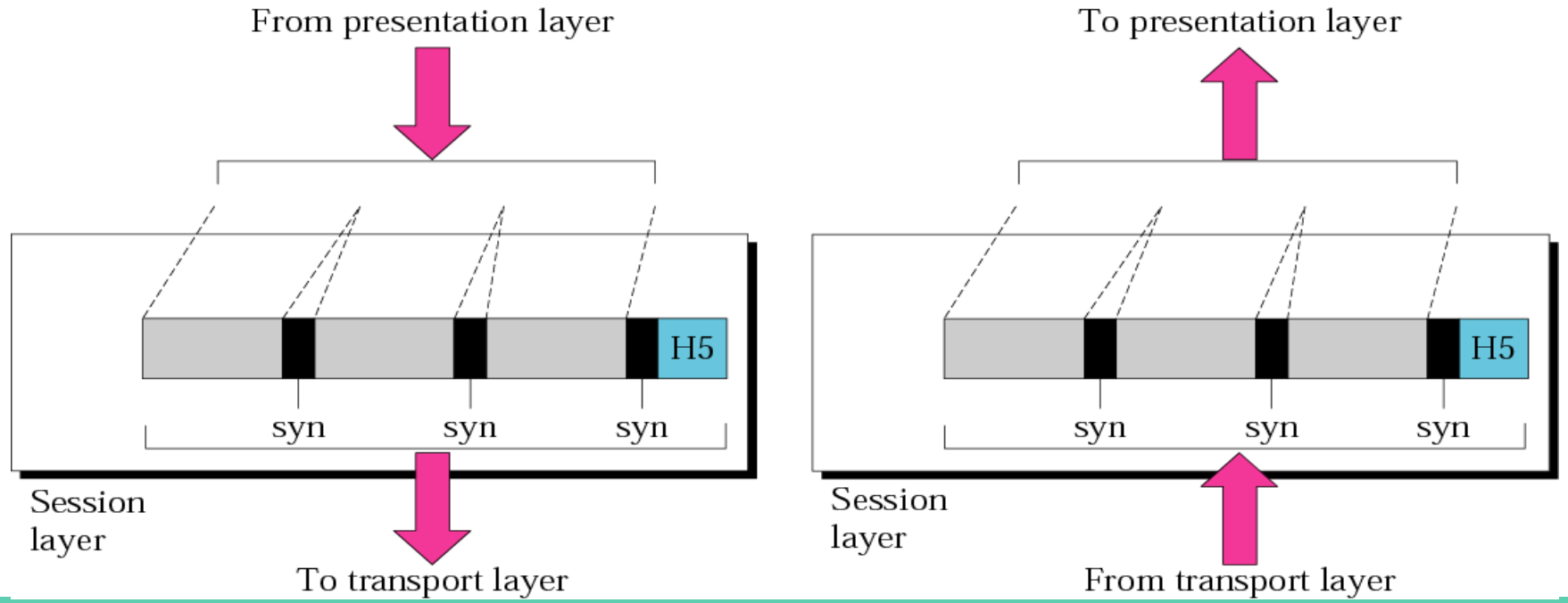
- Service-point addressing.
- Segmentation and reassembly. Sequence number.
- Connection control.
  - Connectionless or connection-oriented.
- Flow control.
  - End to end.
- Error control.
  - Process-to-process error control (damage, loss, or duplication).
  - Error correction is usually achieved through retransmission.

Transport Layer:
Reliable process-to-process delivery of a message



Session Layer

- The session layer is the **network dialog controller**.
- It establishes, maintains, and synchronizes the interaction between communicating systems.

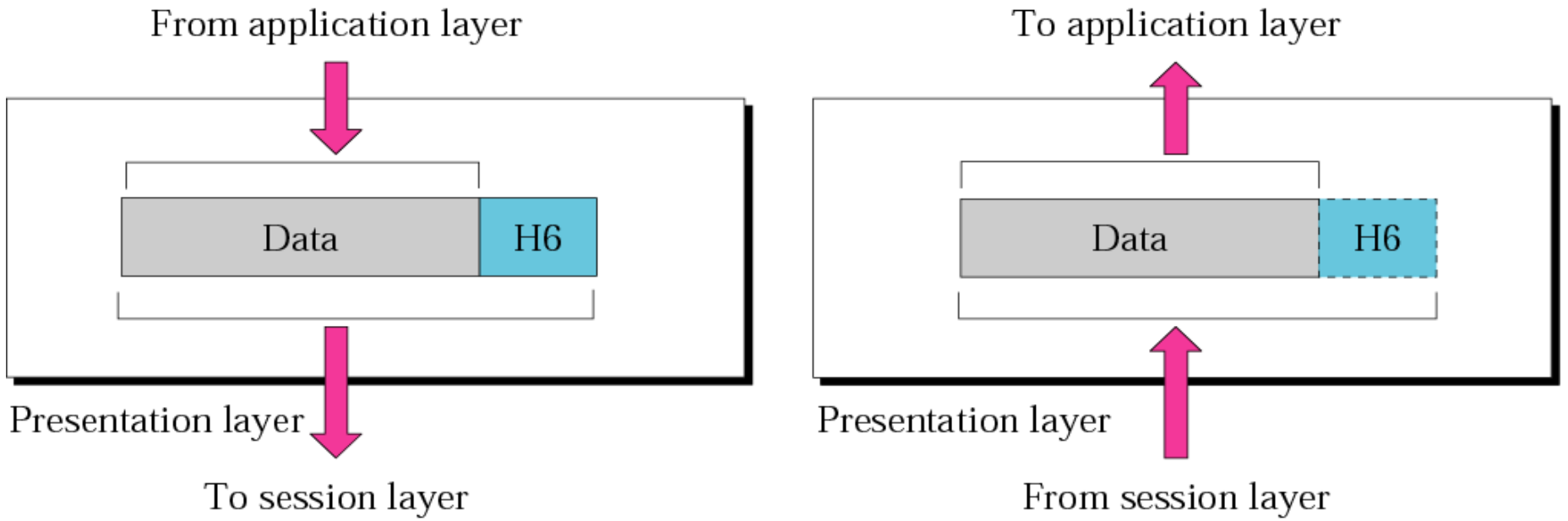


Session Layer Functionalities

- **Dialog control.**
- **Synchronization.**
  - Allows a process to add checkpoints **(synchronization points)** into a stream of data.
  - Example: Adding checkpoints after every 100 pages of a 2000-page file.

Presentation Layer

- Concerned with the **syntax and semantics** of the information exchanged between two systems.

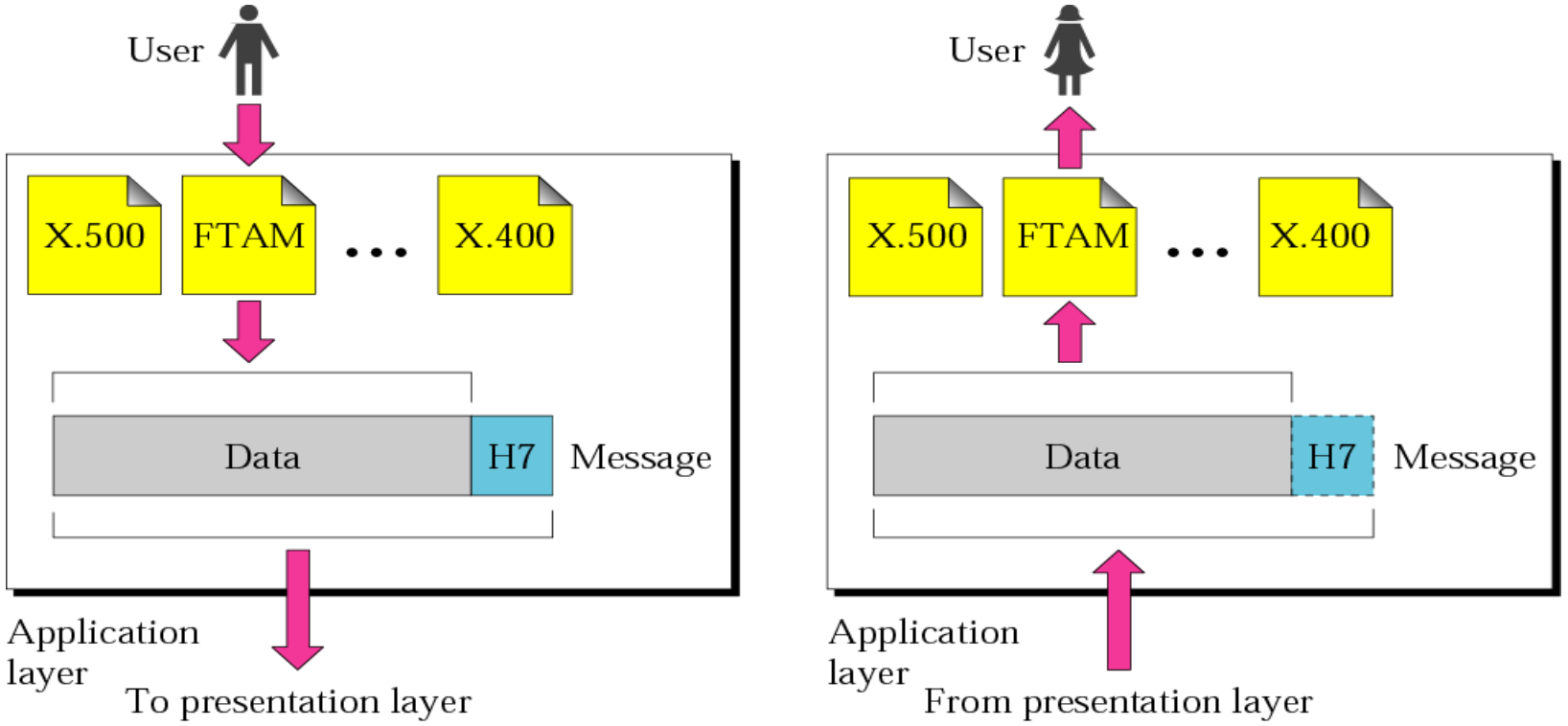


Presentation Layer Functionalities

- **Translation.**
  - Information to bit streams.
  - Responsible for interoperability between the different encoding methods used by different computers.
  - Sender-dependent format → Common format --------> Common format →receiver-dependent format.
- **Encryption.**
- **Compression.**

Application Layer

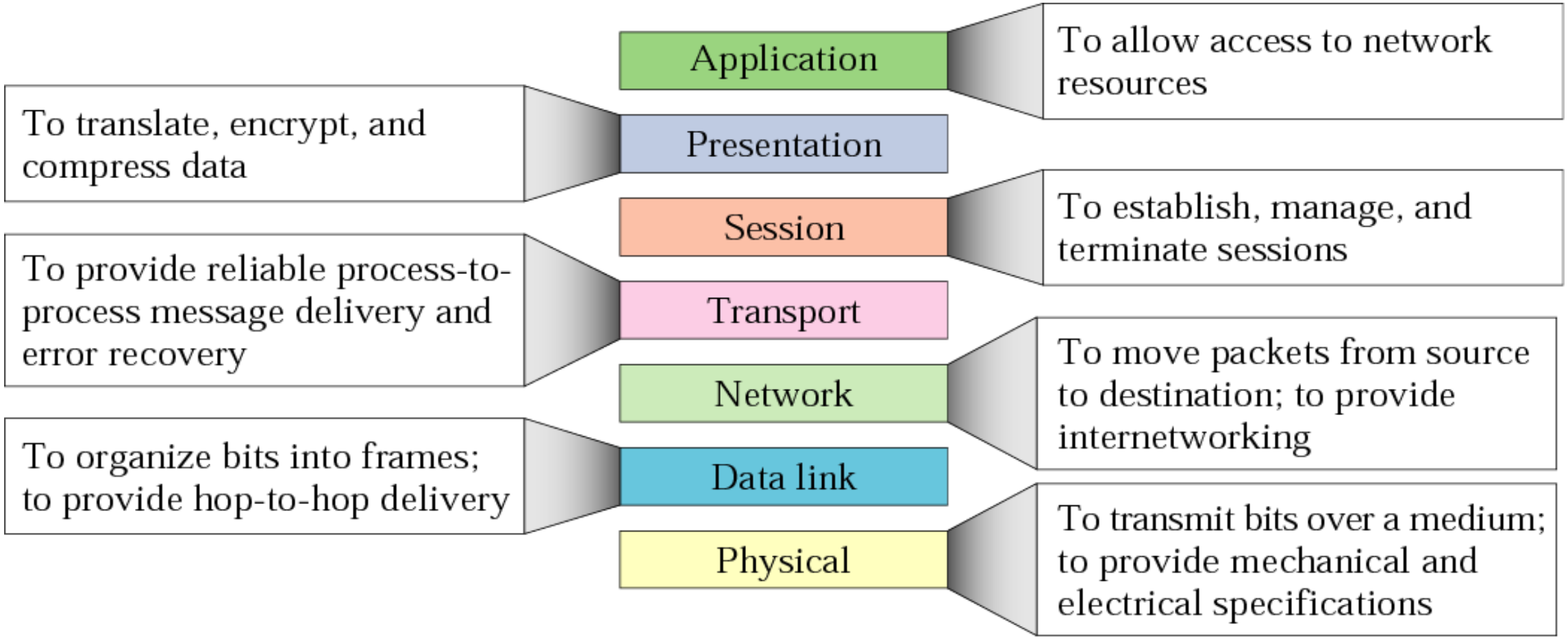
- The **application layer** enables the user, whether human or software, to access the network. The application layer provides the interfaces and services to access the network.



Application Layer Functionalities

- **Network virtual terminal.**
  - A network virtual terminal is a software version of a physical terminal and allows a user to log on to a remote host.
  - Application creates a software emulation of a terminal at the remote host.
  - The user's computer talks to the software terminal, which, in turn, talks to the host, and vice versa. The remote host believes it is communicating with one of its own terminals and allows you to log on.
- **File transfer, access, and management (FTAM).**
- **E-mail services.**
- **Directory services.**

Summary of the Layers



Addresses

- **Physical Address**: Also known as the link address.
  - It is address of the node as defined by the LAN or WAN.
  - Included in the frame by the data link layer.
  - Lowest level address.
- **Logical Address:** A universal addressing system in which each host can be identified uniquely regardless of the underlying physical network.
  - 32-bit address. No two publicly addressed and visible host on the internet can have same IP address.

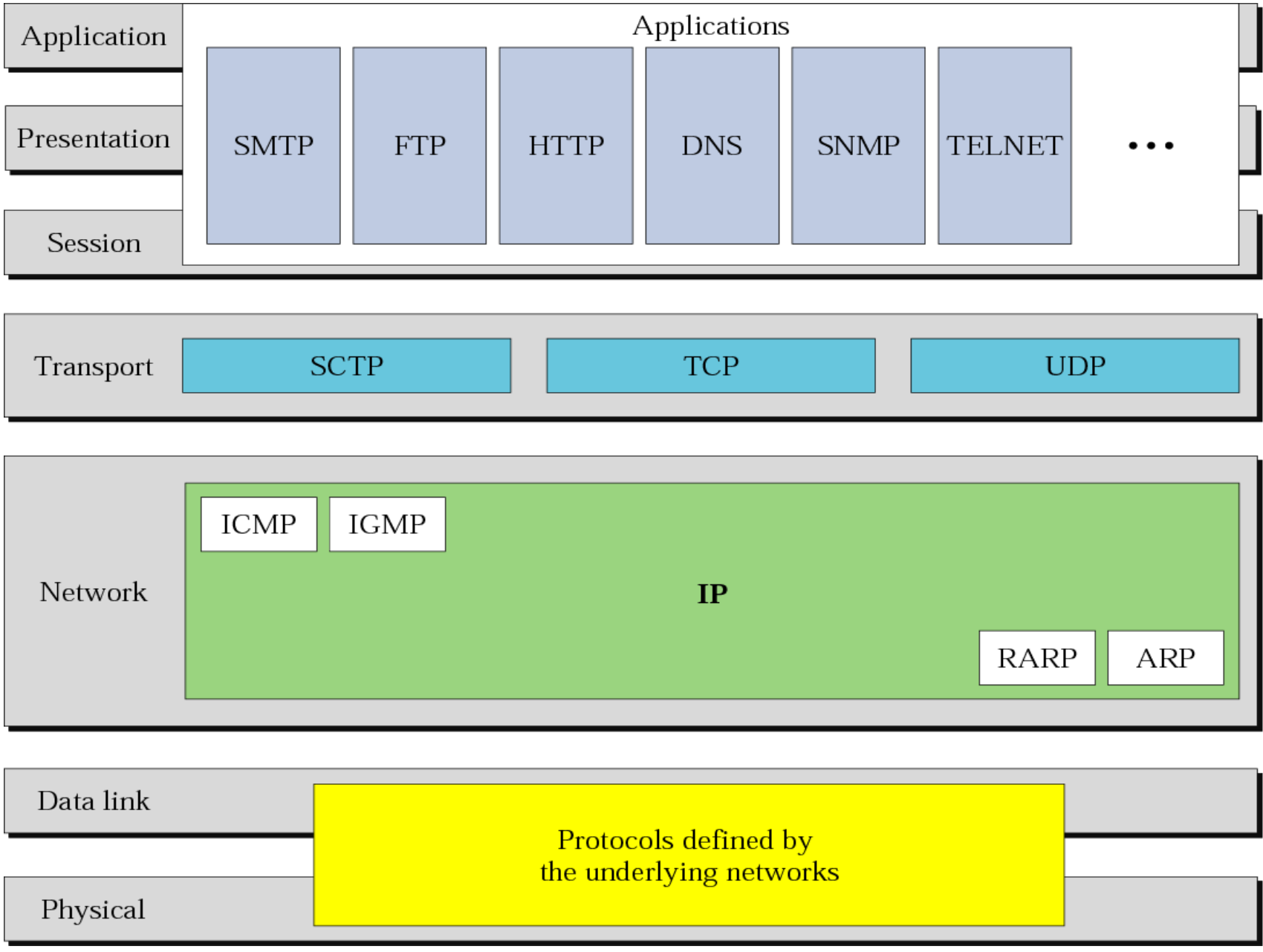
Addresses

- **Port Address:**
  - Objective of Internet communication is process communicating with another process.
  - Computer A can communicate with computer B using TELNET and at the same time computer A can communicate with C using FTP.
- **Application Specific Address**

The TCP/IP Protocol Suite

- The original TCP/IP protocol suite was defined as having four layers:
  - host-to-network (physical + data link)
  - Internet
  - Transport
  - Application
- When was the TCP/IP protocol suite developed: after OSI/ before OSI?

The TCP/IP Protocol Suite



OSI model and TCP/IP Protocol Suite

- Two layers, session and presentation, are missing from the TCP/IP protocol suite. These two layers were not added to the TCP/IP protocol suite after the publication of the OSI model. **Why?**
- TCP/IP is a **hierarchical** protocol made up of interactive modules, each of which provides a specific functionality, but the modules are **not necessarily interdependent.**
- Whereas the OSI model specifies which functions belong to each of its layers, the layers of the TCP/IP protocol suite contain **relatively independent protocols** that can be **mixed and matched**, depending on the needs of the system.

TCP/IP Protocol Suite

Layers	Protocol	Communication between	Unit of Communication	Address	
Physical	No specific protocol	2 nodes/hops	Single bit	Physical	'n' links, 'n' protocols
Data Link	No specific protocol	2 nodes/hops	Frames	Physical (Dest. Addr. needs to be correct. Why?)	Frames may be different in different links.
Network	Internet Protocol (IP)	Source to destination	Packets called Datagrams	Logical/IP	Datagrams take different route, no reordering of datagrams.
Transport	TCP, UDP, SCTP	Process to process	Segment (tens of datagrams)	Port/ Service-point	Reassembly
Application	Many protocols	End to end		Application-specific	

Application Layer Protocols

- TELNET: Terminal NETwork
- SMTP: Simple Mail Transfer Protocol
- FTP: File Transfer Protocol
- HTTP: HyperText Transfer Protocol
- DNS: Domain Name System

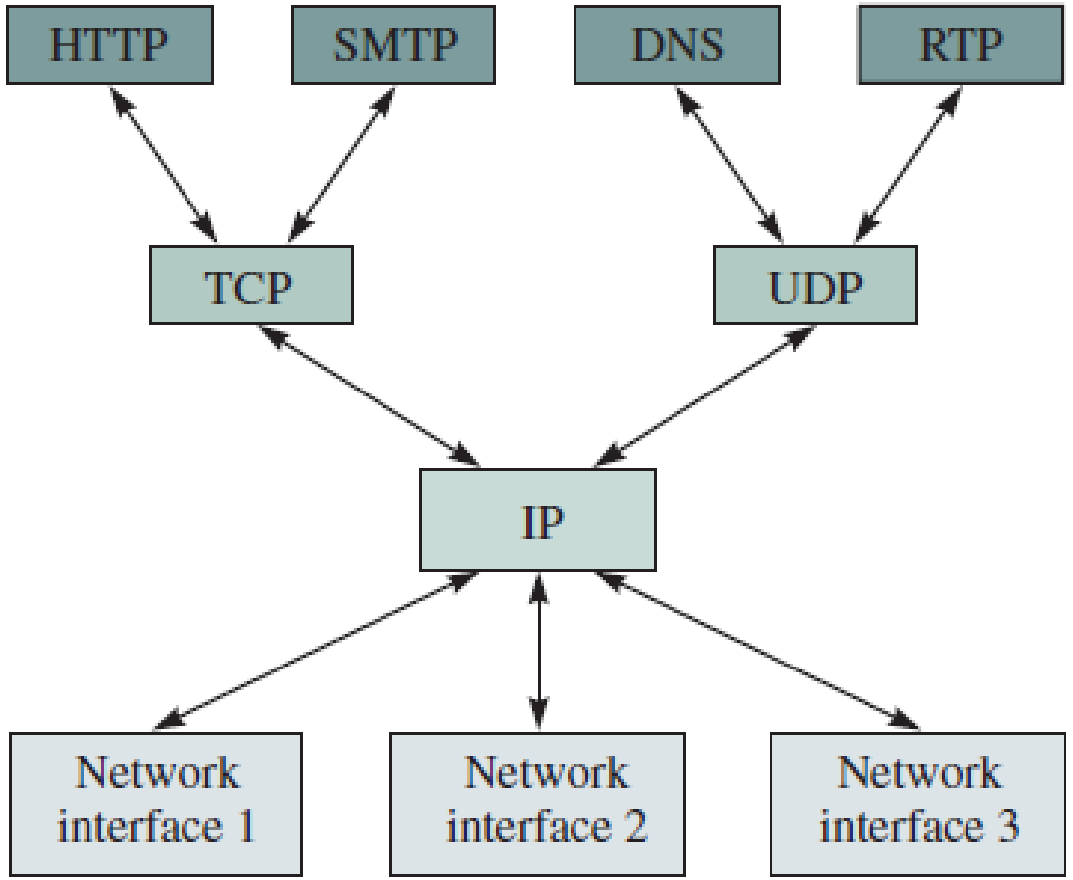
Transport Layer

- The application layer protocols are intended to run directly over the transport layer.
- They are two types of services provided:
  - Transmission Control Protocol (TCP) : Connection Oriented
  - User Datagram Protocol (UDP): Connectionless

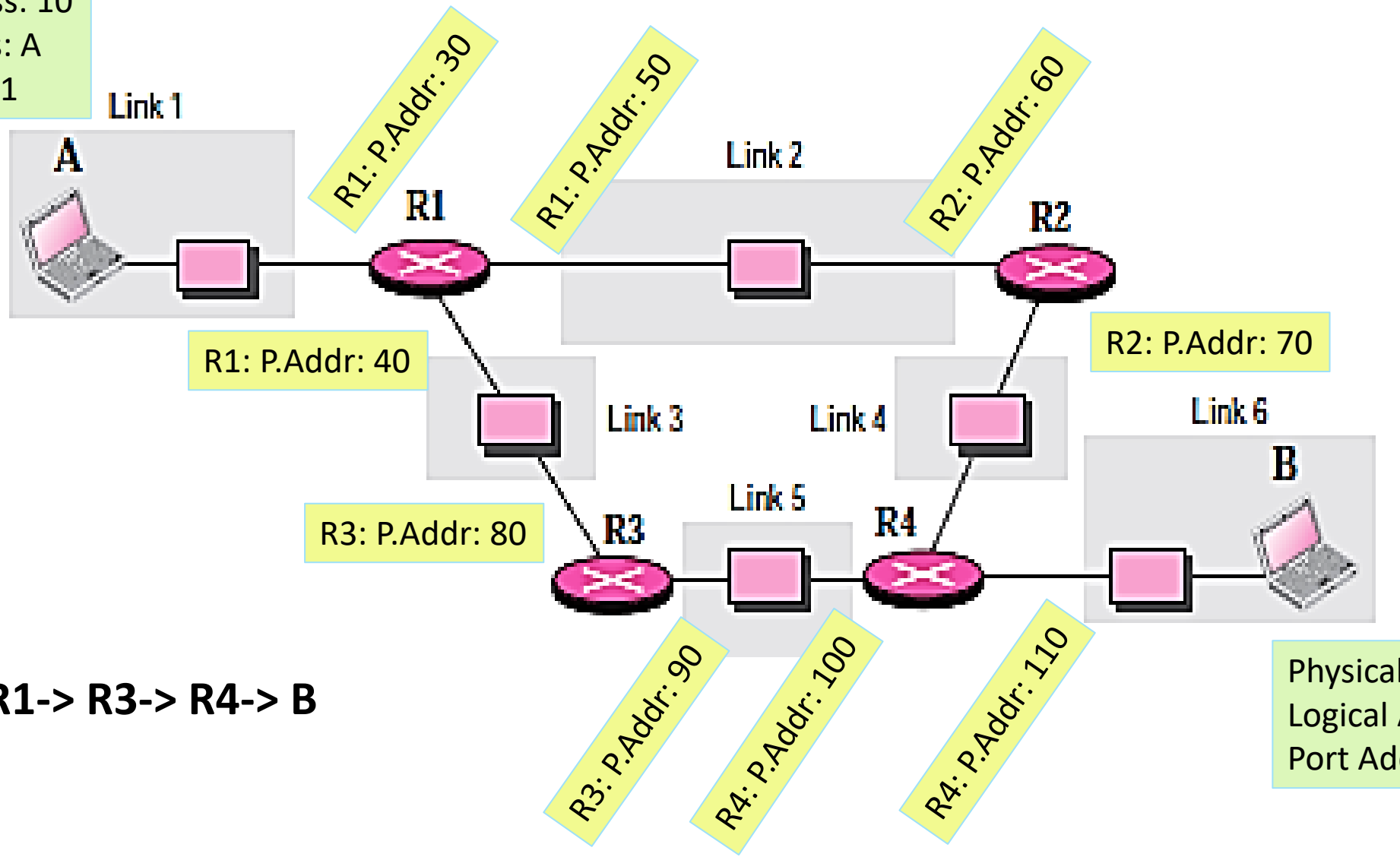
Internet Layer

- Main Protocol: IP
- Other Protocols: ICMP, IGMP, ARP, RARP

TCP/IP Protocol Graph

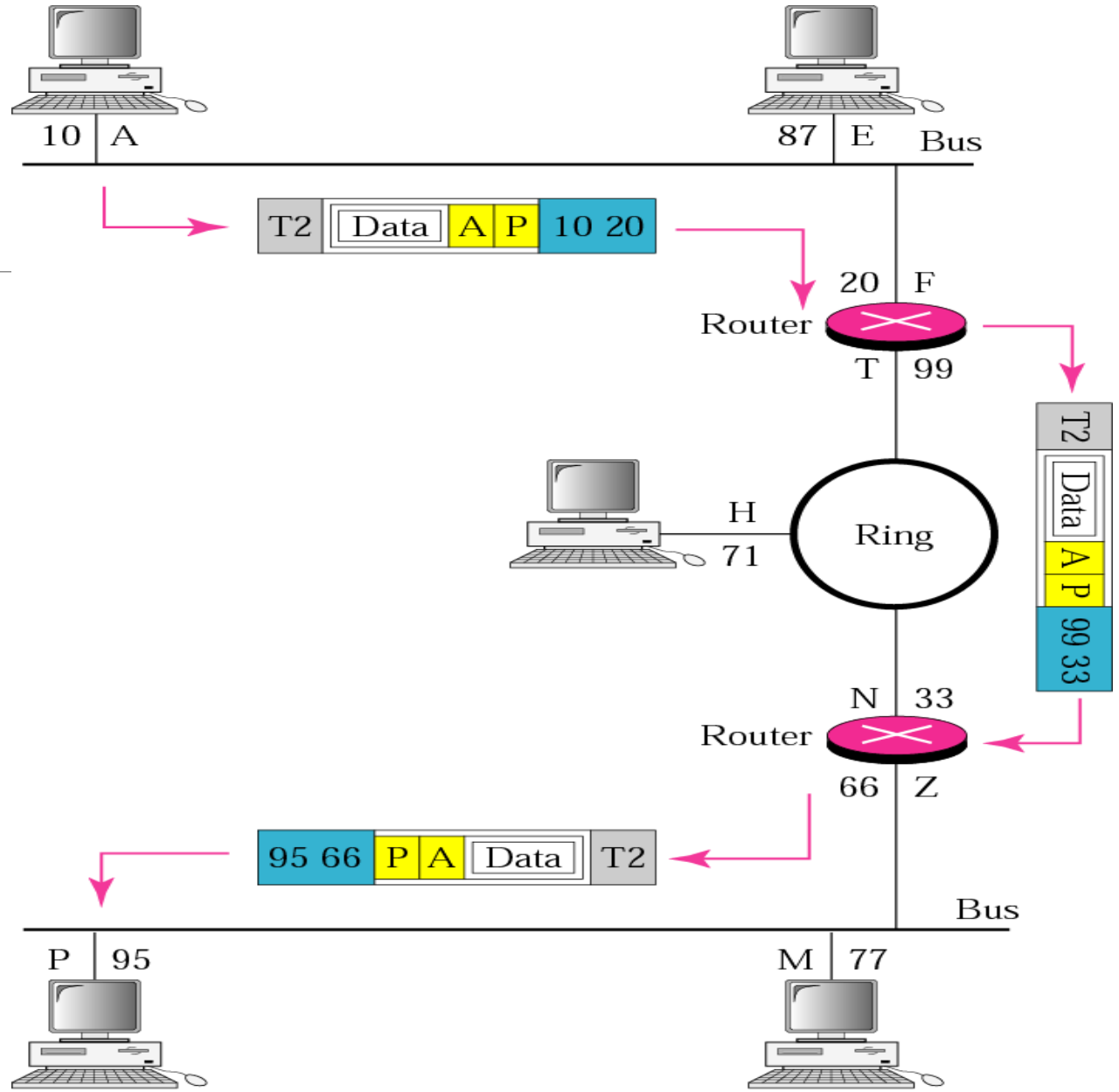


A Scenario



Route: A-> R1-> R3-> R4-> B

Scenario Problems



Problem 2

An internet path between two hosts involves a hop across network A, a packet-switching network, to a router and then another hop across packet-switching network B.

Suppose that packet switching network A carries the packet between the first host and the router over a two-hop path involving one intermediate packet switch. Suppose the second network is an Ethernet LAN.

Sketch the sequence of IP and non-IP packets and frames that are generated as an IP packet goes from host 1 to host 2.

Problem 3

- A 100-byte message is sent through a private internet using the TCP/IP protocol suite. If the protocol adds a 10-byte header at each layer, and a 10 byte trailer, what is the efficiency of the system?

Solution:

Efficiency of a system is defined as the ratio of the number of useful bytes to the number of total bytes.

Thus efficiency = actual size of message / total size (message + 3headers + 1 trailer)

Thus efficiency = 100/140 = 71.4%

Books

- Tannenbaum, “COMPUTER NETWORKS”, Prentice Hall of India [EE Edition], 4thedition, 2003. : chapter 1 – 1.3 (1.3.1, 1.3.2, 1.3.3)
- Behrouz A. Forouzan, “TCP/IP PROTOCOL SUITE”, Tata McGraw Hill, Third Edition, 2010 : Chapter 2 – 2.2, 2.3, 2.4
- Alberto Leon – Garcia, “Communication Networks”, Tata McGraw Hill, Second Edition, 2004. Chapter 2- 2.3