COMPUTER NETWORKS

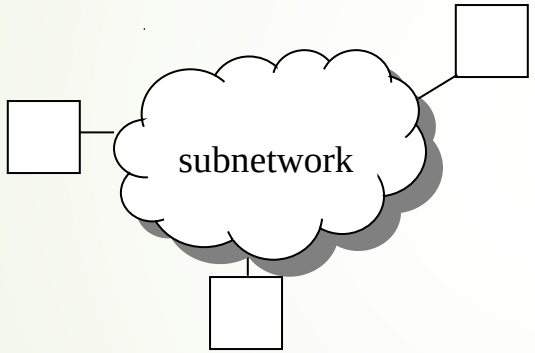
By
Dr. Reham Abobeah

Chapter One

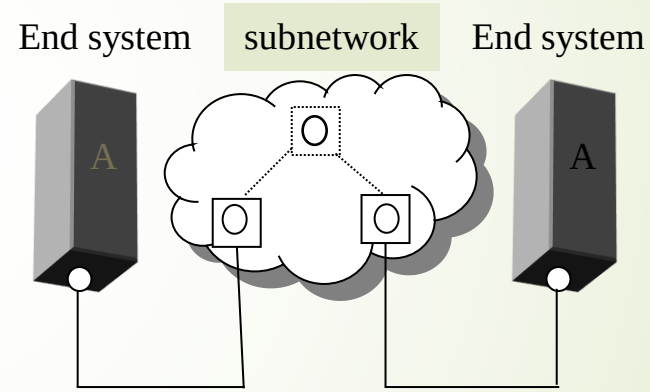
- Network Architecture

2.1 Topology of a Computer Network

A computer network consists of **end systems** which are sources and sinks of information, and which communicate through a **transit system**, (called subnetwork), interconnecting them.



(a)



(b)

A: application entity

: Node

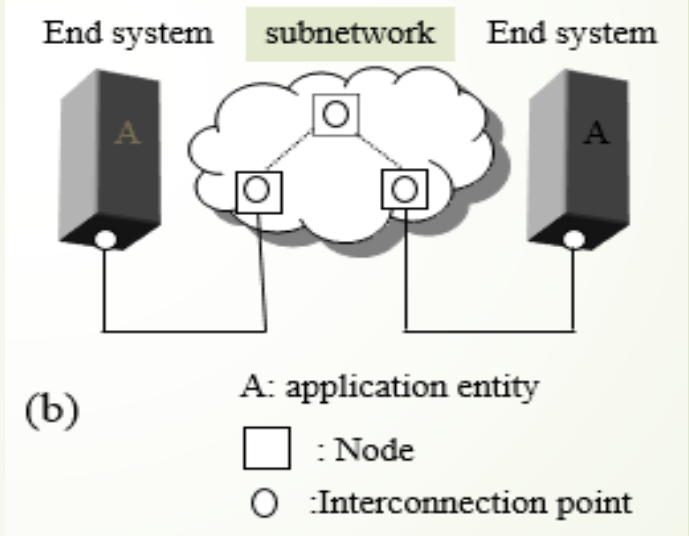
:Interconnection point

Computer Network

An end system consists of computers, terminals, software, and peripherals (e.g. disk drive, CD, HDD) forming an autonomous whole capable of performing information processing.

Each end system has an ***interconnection point*** through which it is physically connected to the transmission medium. ***The interconnection point has an address by which the end system is identified***.

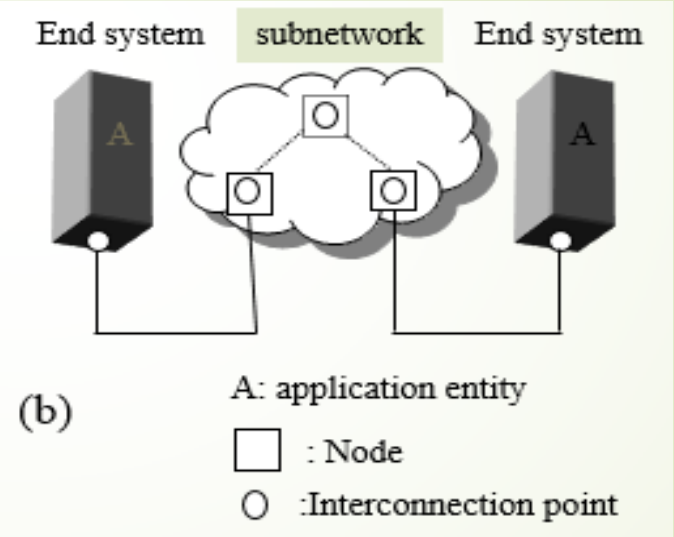
Each end system ***hosts one or more application entities***. Due to these application entities, the communication takes place between the end systems. They determine the subject and the duration of their communication.



The subnetwork, (without any application entity), performs all ***transmission and switching activities required for transporting messages between the end systems***.

Subnetwork consists of ***nodes that process the messages for routing them towards their destination***.

Nodes are interconnected to the end systems at the edges of the sub network through a **transmission medium** that carries electrical signals.



What is a protocol?

- It is a formal description of **message formats** and the **rules** that two computers must follow in order to exchange messages.
- This set of rules describes how data is transmitted over a network.

Why are protocols needed?

- Protocols are needed for communication between any two devices.
  - In what **format** will the messages be transmitted?
  - At what **speed** should messages be transmitted?
  - What to do if **errors** take place?
  - What to do if parts of a message are **lost**?

- **Communication** is based on the transfer of information having a wider scope.
- For establishing **meaningful communication,** functional requirements should be identified.
- A **man to man communication** is an example from which these functional requirements can be identified.

2.2 Elements of Meaningful Communication

- There are some basic elements of the communication process which **must be present in any communication** to make it **meaningful**. This is true for any type of communication, between human beings or between computers. These essential elements are:

- ***Common theme*** (i.e. the communicating entities should have common subject),
- ***Common language*** (i.e. the communicating entities should have the same language), and
- ***an orderly session*** (i.e. disciplined dialogue exchange which required flow control, error control, and synchronization).

2.3 Transport- Oriented Functions

In a computer network, the subnetwork provides means of transporting the messages. But some **additional functions must also be built into the end systems** for enabling **transport of the message through the subnetwork without any error**. These functions are assigned for:

- Interaction with the subnetwork,
- Quality of transport service,
- Conversion of signals, and
- Error control.

2.4 Components of a Computer Network

- There are **certain functional capabilities** that must be built into an **end system** for **meaningful communication**. These capabilities are:

- Authentication and login.
- Common syntax.
- Establishment of an orderly exchange of messages

  with markers for forward and backward synchronization.
- Establishing transport connection of required quality, flow control.
- Interacting with the subnetwork.
- Error control.
- Conversion of bits into electrical signals and vice versa.

2.5 Architecture of a Computer Network

- The architecture of a computer network, ***describes how the computer network has been assembled using its various components. It defines the specifications of the components and their interrelationship.***
- The architecture of a computer network specifies a complete set of **rules** for the connections and interactions of its physical and logical components for providing and utilizing communication services.
- The architecture of a computer network is **layered architecture**.
- A **computer network** is **portioned into** end systems interconnected using a subnetwork and communication process is decomposed into hierarchical functional layers.

2.6 Need for Standardization of Network Architecture

Standardization of network architecture makes us move from "closed systems" to "open systems". **Standardization** can solve many problems and solve a lot of effort required for developing **interfaces** for networking different architectures.

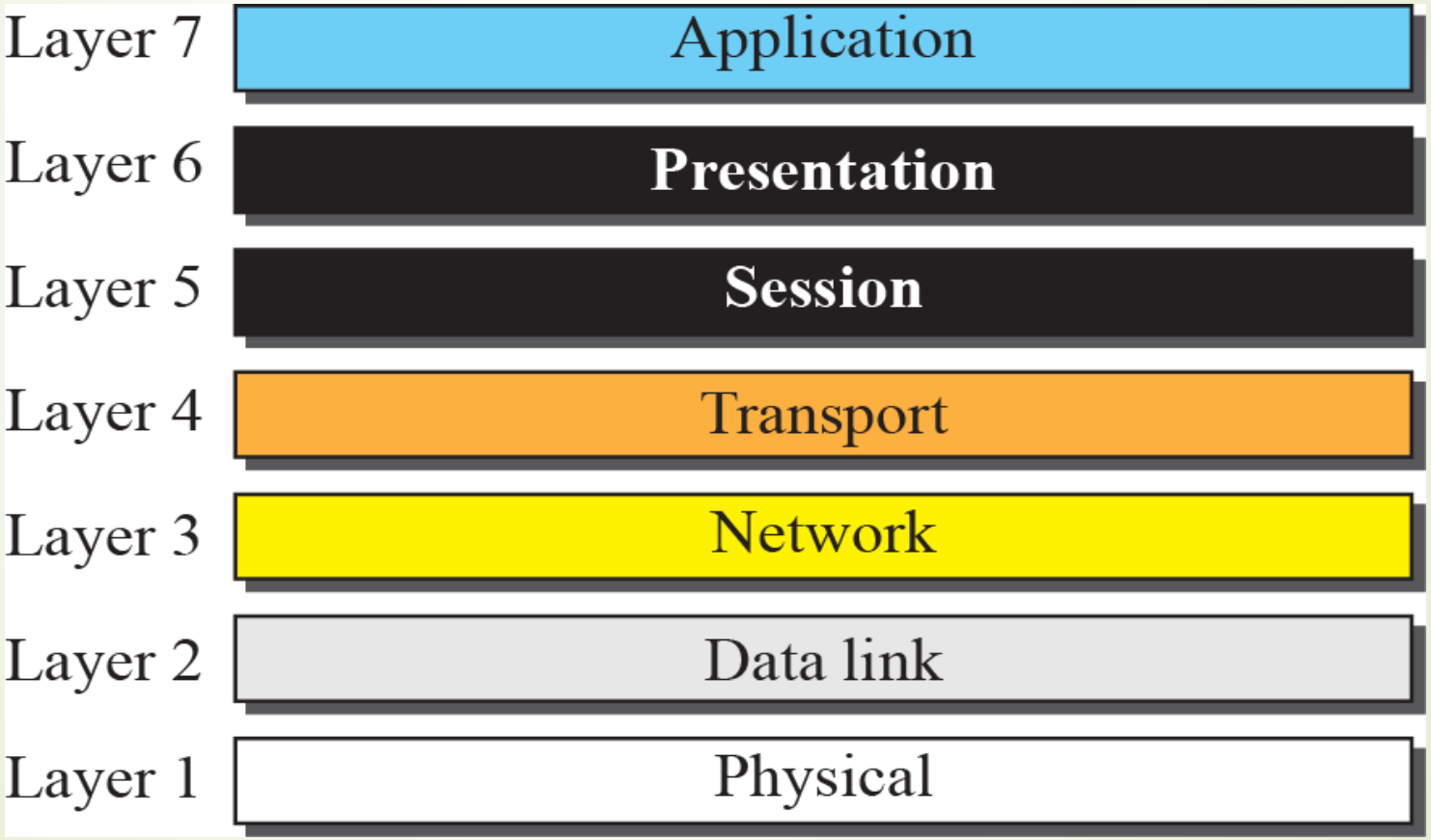
- There are several network architectures developed by manufacturers. **Some of the important network architectures are:**
- IBM's System network Architecture (SNA).
- Digital's Digital Network Architecture (DNA).
- Internet architecture.

2.7 Open System Interconnection / OSI Model

- **International standard organization (ISO)** established a committee in 1977 to develop an architecture for systems communication.
- **Open System Interconnection (OSI)** reference model is the result of this effort. It is the first step towards standardization.
- This model allows any **two different systems to communicate** regardless of their underlying architecture.

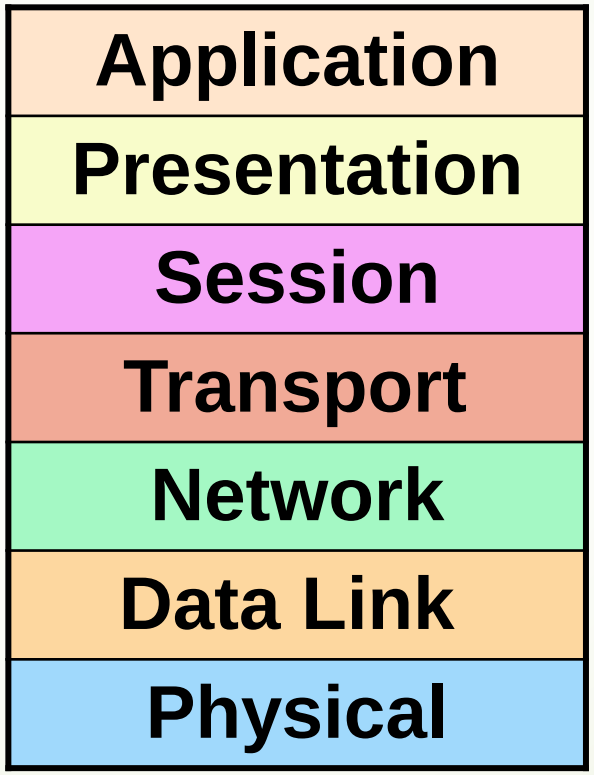
- The OSI model describes **how data flows** from one computer, through a network to another computer.
- The OSI model is not a protocol; it is a model for understanding and **designing a network architecture** that is flexible and robust.
- The OSI model consists of **seven separate but related layers**, each of which defines a part of the process of moving information across a network.
- OSI model, is a set of guidelines for implementing networking communications between computers.

Seven layers of the OSI model



Seven layers of the OSI model

All
People
Seem
To
Need
Data
Processing

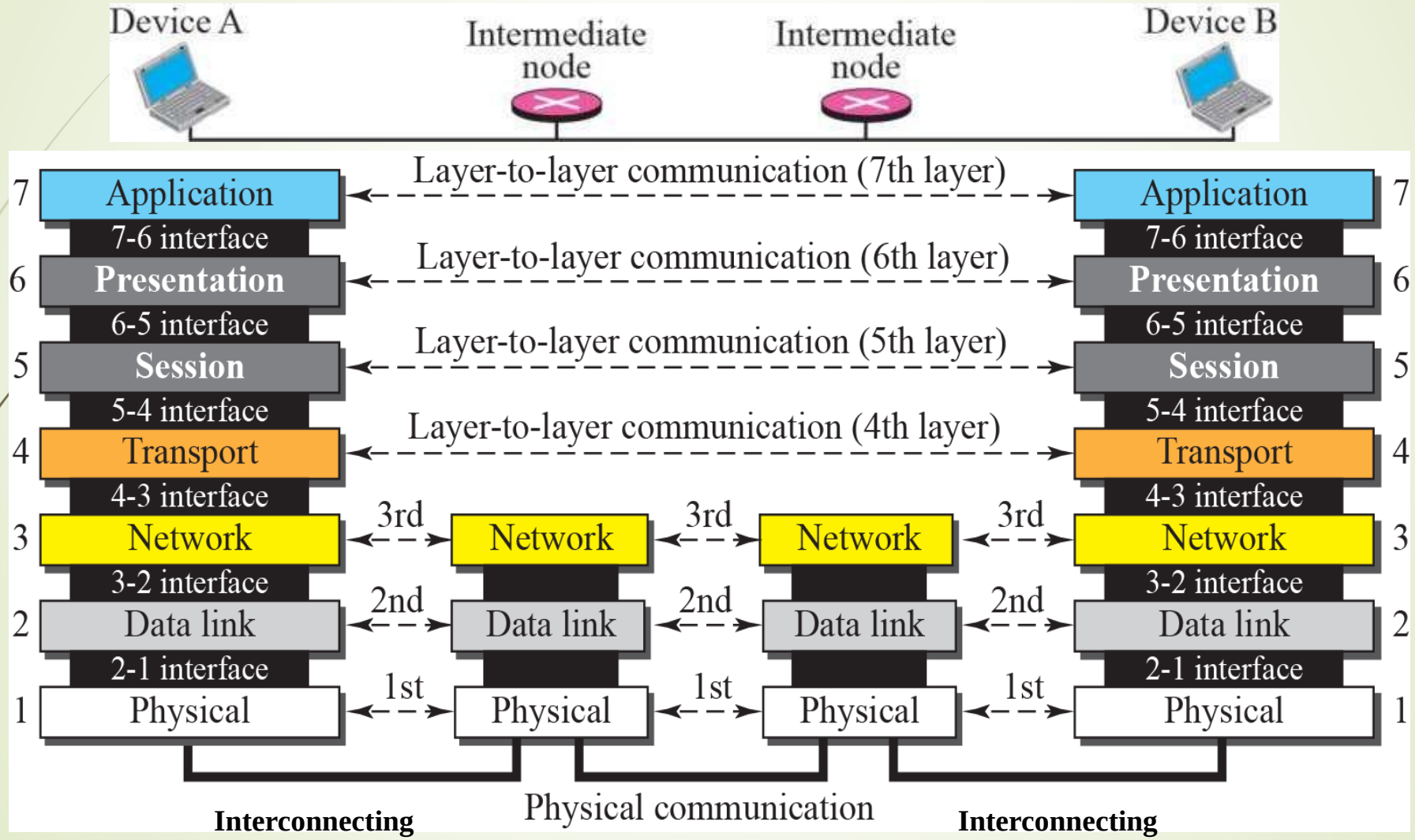


Please
Do
Not
Tell
Secret
Passwords
Anytime

Why are so many layers?

- To reduce the complexity, networks are organized as a stack of layers, one below the other.
- Each layer performs a specific task, and provides services to an adjacent layer.

2.8 Layered Architecture of the OSI Reference Model



- **Layers 1, 2, 3**, physical, data link and network are **network support** layers.
- **Layer 4**, the transport layer, **links the two subgroups**.
- **Layers 5,6,7**, session, presentation, and application are **user support layers**.

2.8.1 Application Layer

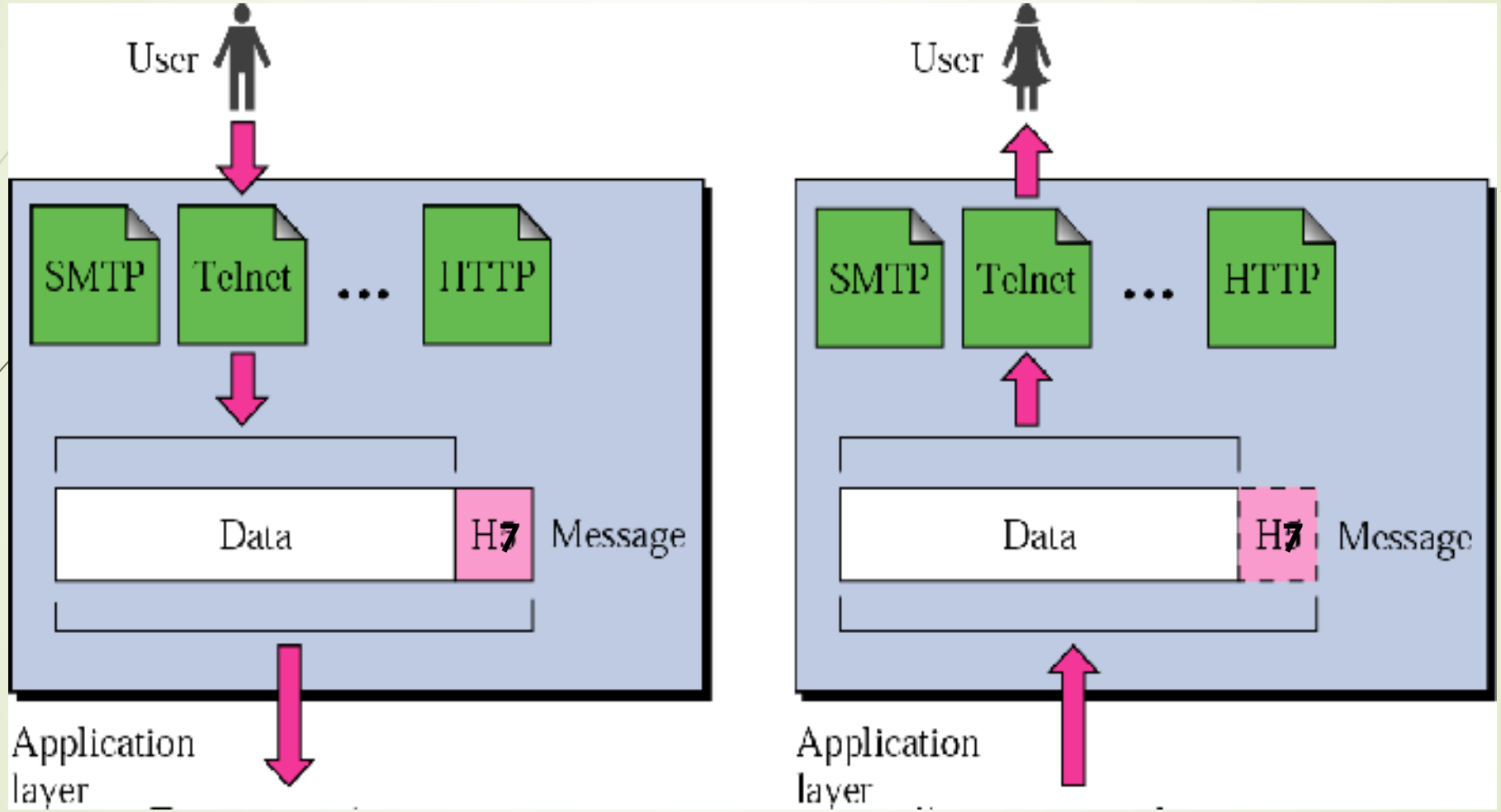
- It provides a means for application programs to access the OSI environment, in other word, the application layer is the **interface** for the distributed applications (programs).
- Some examples of the **applications programs**; Internet explorer, File transfer, Electronic mail, Terminal access to remote computer, and the World Wide Web (WWW).
- Each one of the above mentioned applications has a part of it that resides in the application layer, for example:

- **File transfer:** It uses File Transfer protocol (FTP).
- **Electronic mail:** It uses Simple Mail Transfer (SMT) protocol to send message.

- **Internet explorer:** It uses HTTP protocol of the application layer for exchanging messages.
- **Remote login:** It uses Telenet protocol for remote login.
- **World Wide Web (WWW):** It uses the application protocols mentioned above with some additional components.

- In addition, **the application layer provides such services as:**
- **Identification** of the intended communication partner (s) by name or by address.
- Establishment of **authority** to communicate.
- **Agreement** on privacy (**encryption**) mechanisms.
- **Authentication** of an intended communication partner.
- **Agreement** on responsibility for **error recovery**.

Application Layer



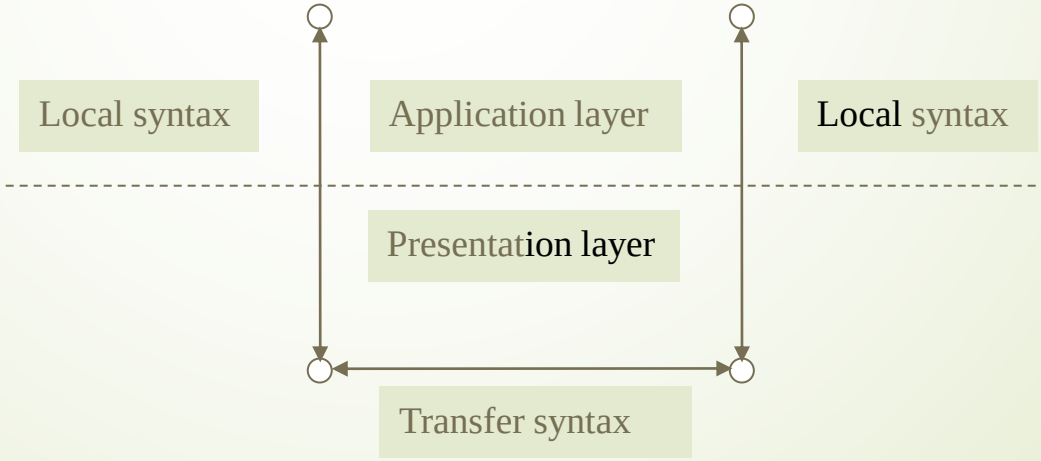
To Presentation Layer

From Presentation Layer

2.8.2 Presentation Layer

- *The **purpose of the presentation layer** is to present the information to the communicating application entities in a way that preserves the meaning while resolving the syntax (code and data format) differences.*

- **There are three syntactic versions of data being transferred**, syntax used by the application entity of the originator of the data, the syntax used the application entity of the recipient of data, and the "transfer" syntax used to transfer the data between the presentation entities.

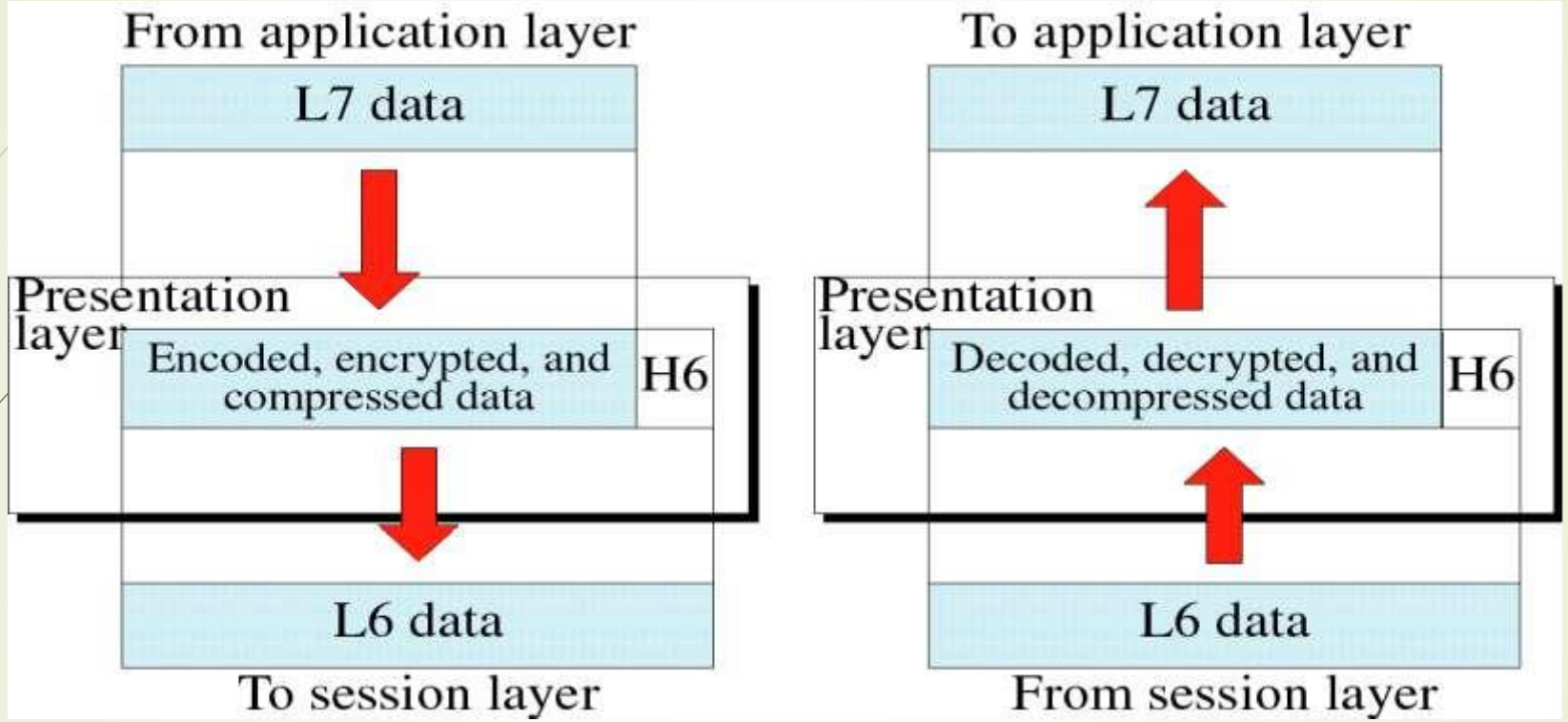


Local and transfer syntax of the presentation layer

- **The syntax may be same or different**. When they are not same, the presentation layer contains functions necessary to transform the transfer syntax to the required syntax used by the application entities preserving the meaning.
- There is no fixed transfer syntax, and is to be negotiated by the presentation entities.

- **Data Encryption--**transform data into an unintelligible format at the sending end -- for security and privacy purpose.

- **Data Compression**: it reduces the number of bits contained in the information.



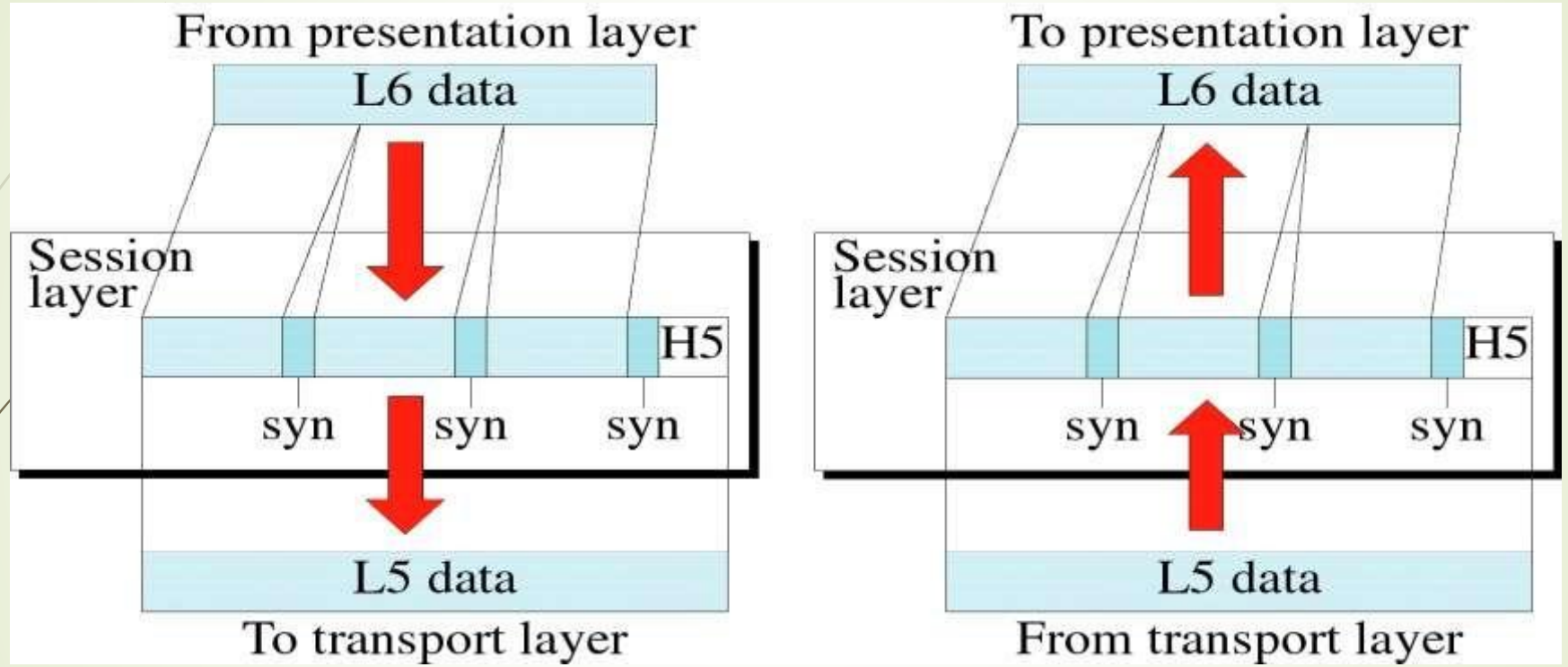
Presentation Layer

2.8.3 Session Layer

- **The main functions of the session layer are as follows −**

1. **Session organization:** The session layer provides functions which are necessary for setting up (opening) and clearing (terminating) a session between end-user application processes.
2. **Dialog Control :** This layer **allows two systems** to start communication with each other in either half-duplex or full-duplex mode of communication.
3. **Token Management:** This layer prevents two users to simultaneously attempt the same critical operation.
4. **Synchronization :** This layer allows a process to add synchronization points or checkpoints (e.g. –ve ack & +ve ack) in data streams for long communications, so that if a failure of some sort occurs between checkpoints, the session entity can retransmit all data since the last checkpoint.

Session Layer



- Synchronization – adds check points (synchronization points) into stream of data.

2.8.4 Transport Layer

- ***The transport layer is a true end-to-end layer, from source to destination***. In other words, a program on the source machine carries on conversation with a similar program on the destination machine, using the messages header and control messages

- ***In the lower layers (from the physical layer to the network layer***), the protocols are between each machine and its immediate neighbors, and not by the ultimate source and destination machines, which may be separated by many nodes.
- The transport layer uses some **standard protocol**s to enhance its functionalities are TCP(Transmission Control Protocol), UDP(User Datagram Protocol), DCCP(Datagram Congestion Control Protocol), etc.

❑ The basic functions of the transport layer are:

1. **Service-point addressing**

- Computers often run many programs at the same time. Due to this, source-to-destination delivery means **delivery from a specific job** (currently running program) on one computer to a specific job (currently running program) on the other system not only one computer to the next.
- For this reason, the **transport layer** added a specific type of address to its header, it is referred to as a service point address or **port address**.
- By this address, each packet reaches the correct computer and also the **transport layer** gets the complete message to **the correct process** on that computer.

2. **Segmentation and Reassembly:**

- Accept data from the session layer, **Split** it up into smaller units (transport data unit/segments), and number them to make a sequence for reassembly at the receiving side, then Pass these units to the network layer.

- Upon arriving at its destination system, message is reassembled correctly, identify and replaces packets that were lost in transmission.

3. **Error control** – make sure that the entire message arrives correctly at the session layer of the other end system– else retransmit.

4. Connection Control

- **Two types of transport service** are provided by the transport layer to the session layer, and ultimately, the users of the network. **These two types are concerned with the type of the transport connection:**

Connectionless transport service:

- ✓ The Transport Layer treats **each packet individually** and delivers it to the destination machine.
- ✓ In this type of transmission , the receiver does **not acknowledge** receipt of a packet.
- ✓ The transported messages have **no guarantee** about the order of delivery.

Connection-oriented transport service:

- ✓ The Transport Layer **creates a connection** with the Transport Layer at the destination machine before transmitting the packets to the destination.
- ❖ **To Create a connection following three steps are possible:**

  i. Connection establishment ii. Data transfer iii. Connection termination

- ✓ In this type, the receiving device sends an acknowledgment back to the source after a packet or group of packets is received.
- ✓ This ensures that data are **delivered error free, in sequence, with no loss or duplications**.

- ❖ When all the data are transmitted connection is terminated. Connectionless Service is less reliable than connection Oriented Service.

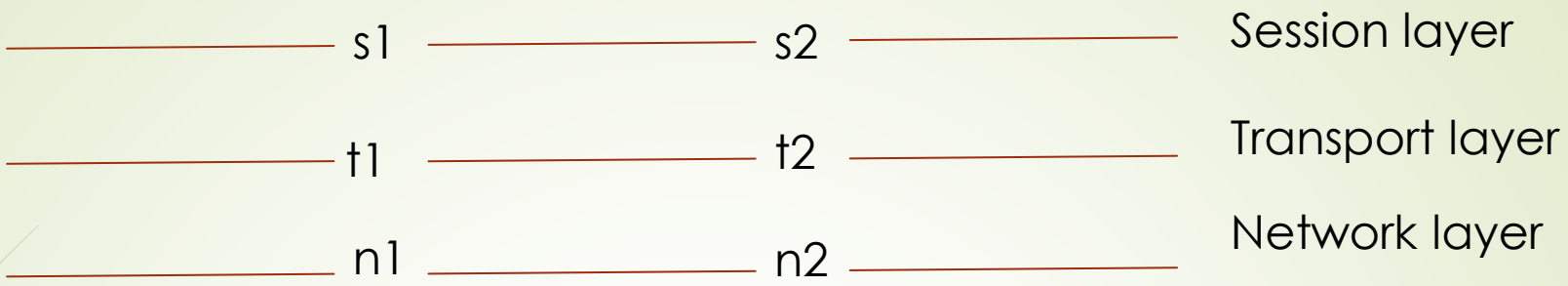
- Under normal conditions, the transport layer:

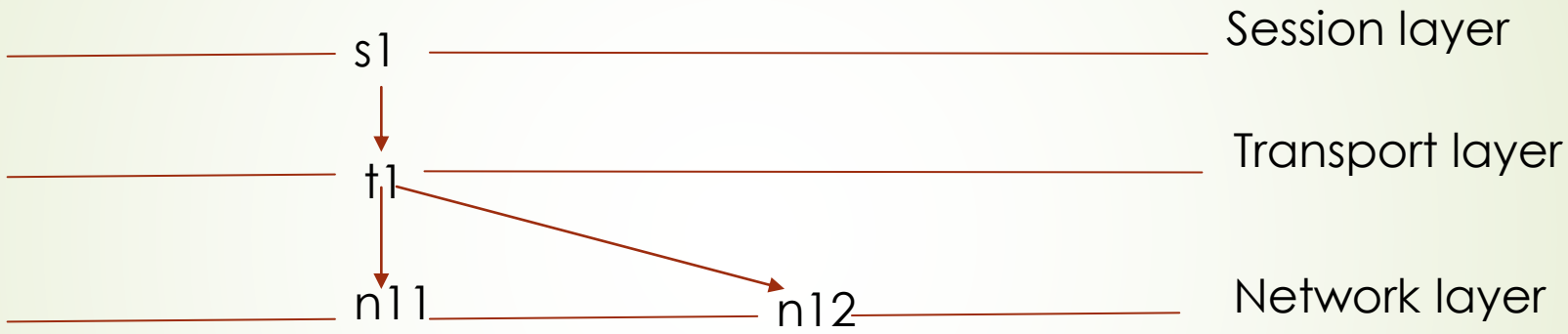
creates *a distinct network connection* for each *transport connection* required by the session layer.

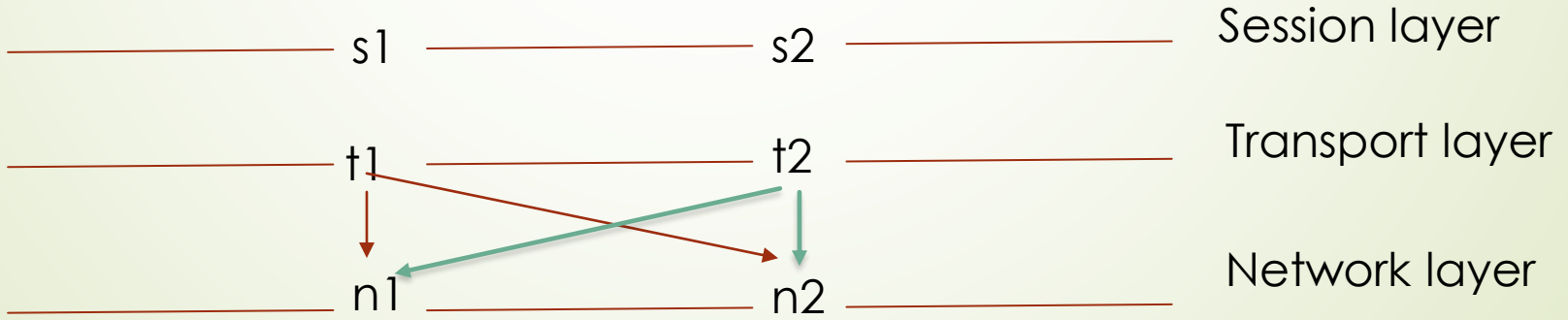
- If the transport connection requires *a high throughput*, the transport layer might create *multiple network connections*, dividing the data among the network connections to improve throughput.

On the other hand, if creating a network connection is expensive, the transport layer might *multiplex several transport connections onto the same network connection* to reduce the cost.

- ✓ **Network throughput** refers to how much data can be transferred from source to destination within a given timeframe. **Throughput** measures how many packets arrive at their destinations successfully. T**hroughput** capacity is measured in bits per second, but it can also be measured in data per second.







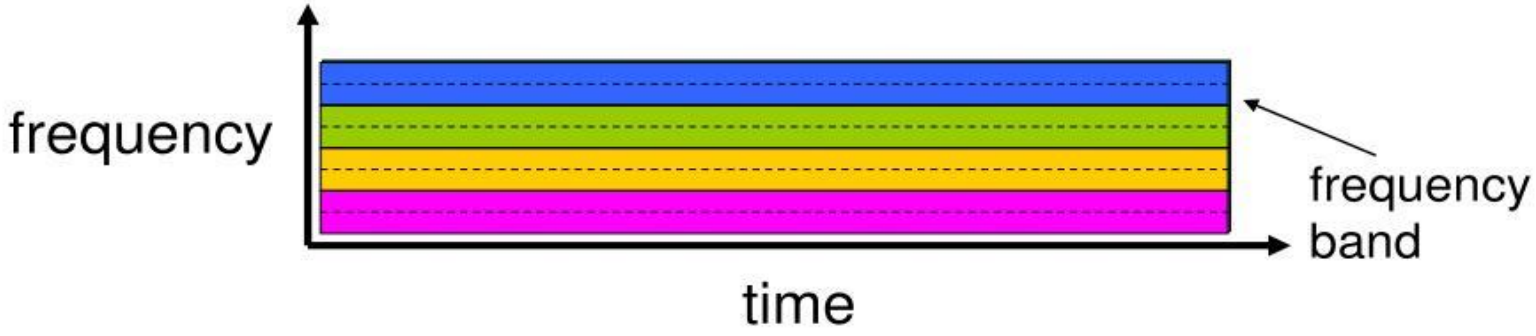
Multiplexing

Circuit switching: FDM and TDM

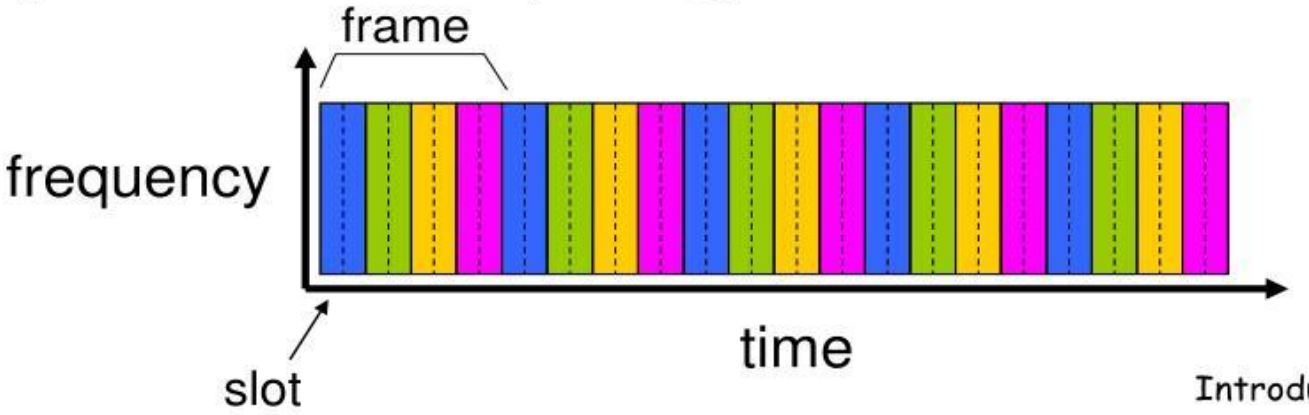
FDM (Frequency division multiplexing)

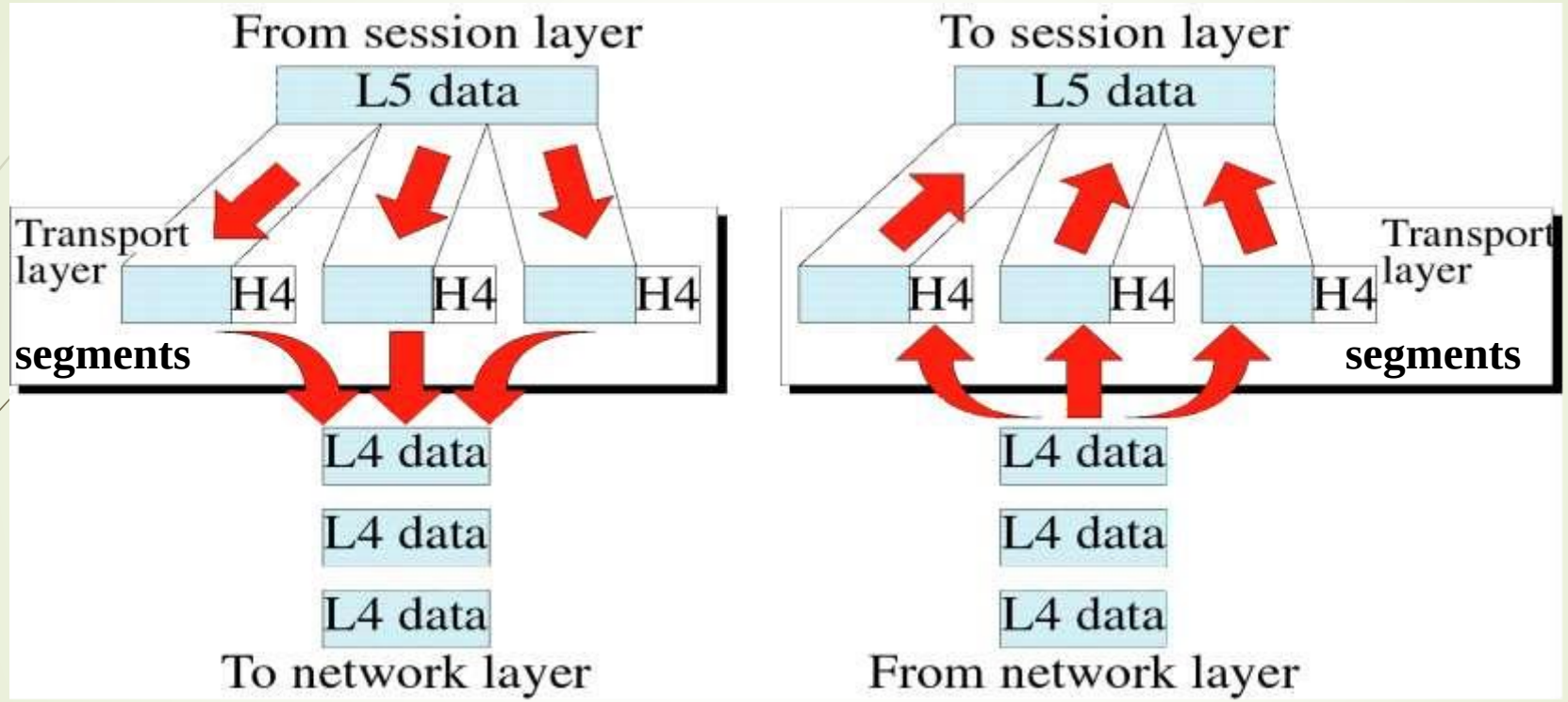
Example:

4 users



TDM (Time division multiplexing)





Transport Layer

2.8.5 Network Layer

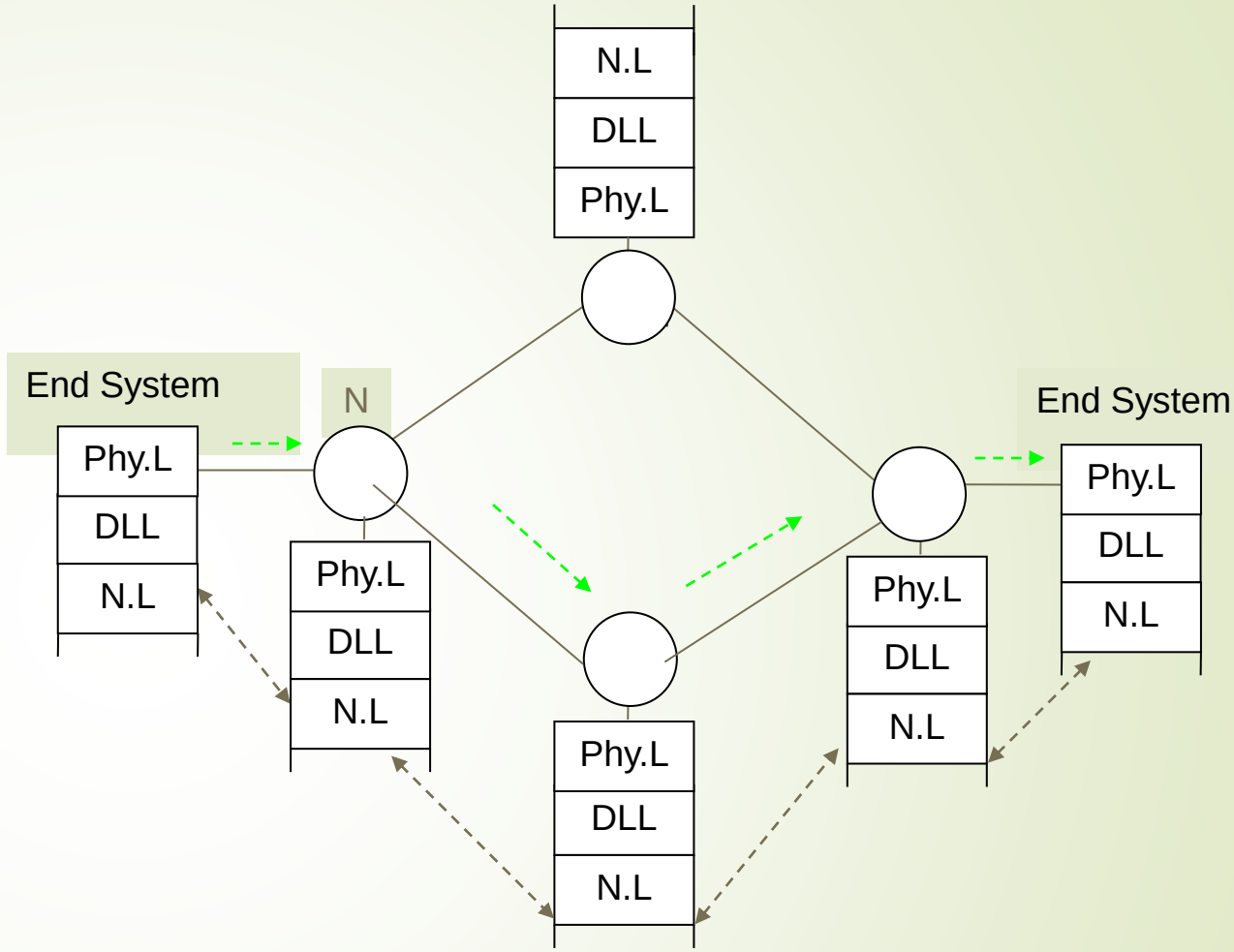
- The network layer provides the means to ***access the subnetwork for routing the message to the destination end system***.

- Network layer **relives the higher layers** (from transport layer to the application layer) of need to know anything about the underlying data transmission and switching technologies used to connect end systems.

- The network layer of an end system **interacts with the network layer of the access node** of the subnetwork.

- This interaction represents the dialogue between an end system and the subnetwork to **specify the destination address** and to request certain network facilities, such as priority.

- An **access node of a subnetwork** facing the end system must support the three lower layers of the OSI model.
- The packets that are created by the end system **pass through one or more network nodes** that act as **relays** between the end systems.
- The network nodes implement **layers 1 through 3** of the architecture.
- **Routing decisions** at each node are taken by the network layer.



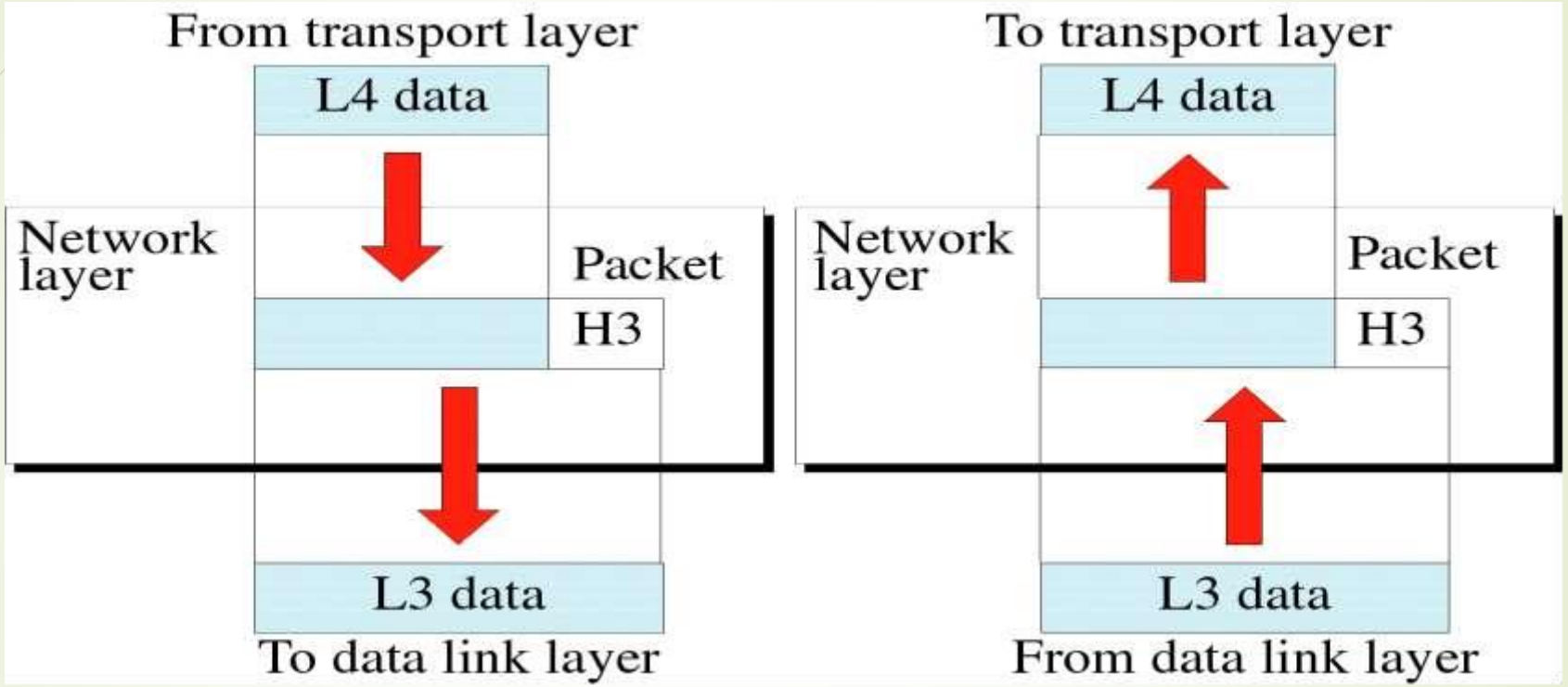
Routing of Packets by the Network Layer

- **Logical addressing**:- Adds Logical addresses of sender and Receiver.

- **Routing:-** Provide mechanism to transmit data over independent networks that are linked together.

  - When a packet has to travel from one network to another (different) one, to get to its destination, many problems can rise:
    - The addressing and protocols used by the second network may be different from the first one, hence the second one may not accept the packet at all.

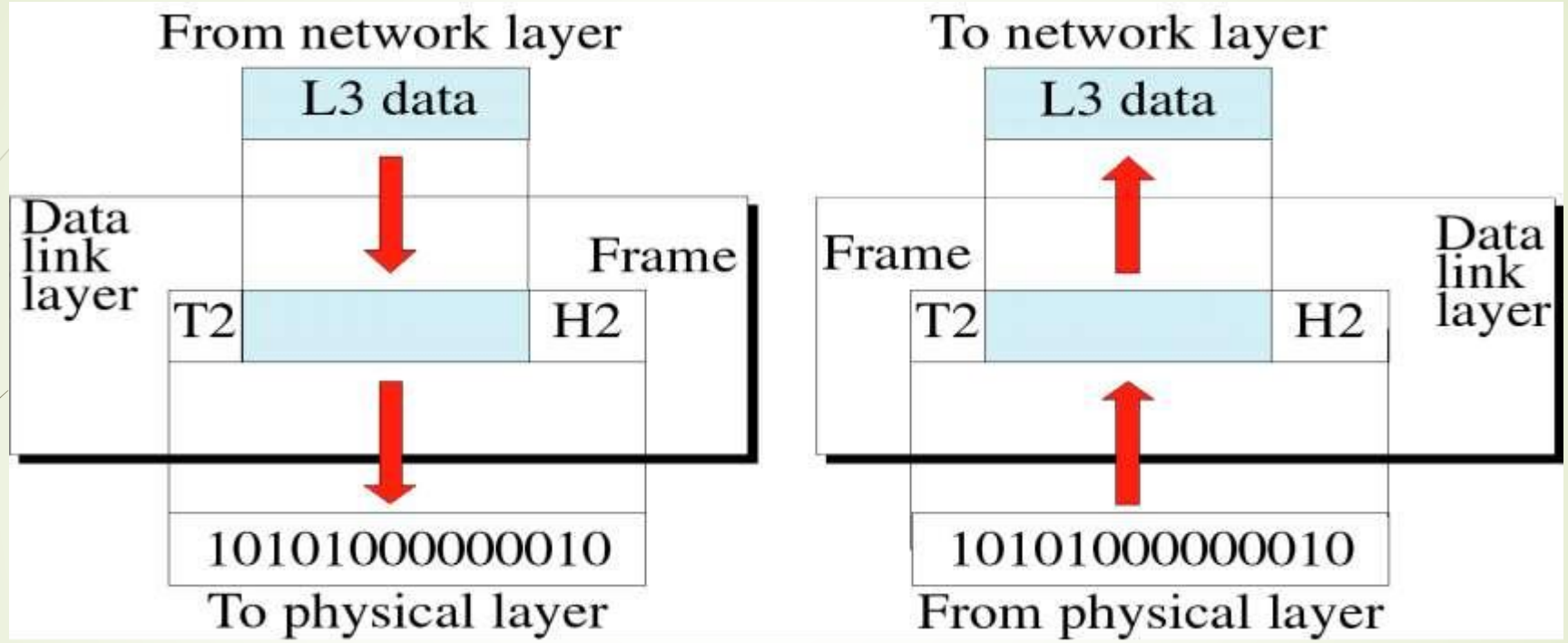
- It is up to the network layer to overcome all these problems to allow **heterogeneous networks** to be interconnected through using some sort of interconnecting techniques.
- Also, the **congestion control** is carried out by the network layer.



Network Layer

2.8.6 Data Link Layer

- **Framing**
  - At the sender, DLL breaks the incoming data packet, from the network layer, into up **data frames**, and transmits the frames sequentially using the physical layer.
  - At the receiver, it takes **a stream of bits transmission facility**, from the physical layer, and transform it into **a frame** that appears free of errors to the network layer.
- **Physical Addressing:-** Add a header to the frame to define the physical address of the source and the destination machines.
- **Flow Control**:- It is the traffic regulatory mechanism implemented by DLL - control rate at which data is transmitted –so as not to flood the slow receiver by the fast sender.
- **Error Control**:- It is a mechanism through which DLL can detect and retransmit damaged or lost frames. It also uses a mechanism to recognize duplicate frames. **Error control** is normally achieved through a **trailer** added to the end of the frame.
- **Access control**. In LANs, end systems are having a **common transmission medium**. **Medium Access Control (MAC) function** is carried out by the data link layer to determine which user can use the media for its transmission.



Data Link Layer

2.8.7 Physical Layer

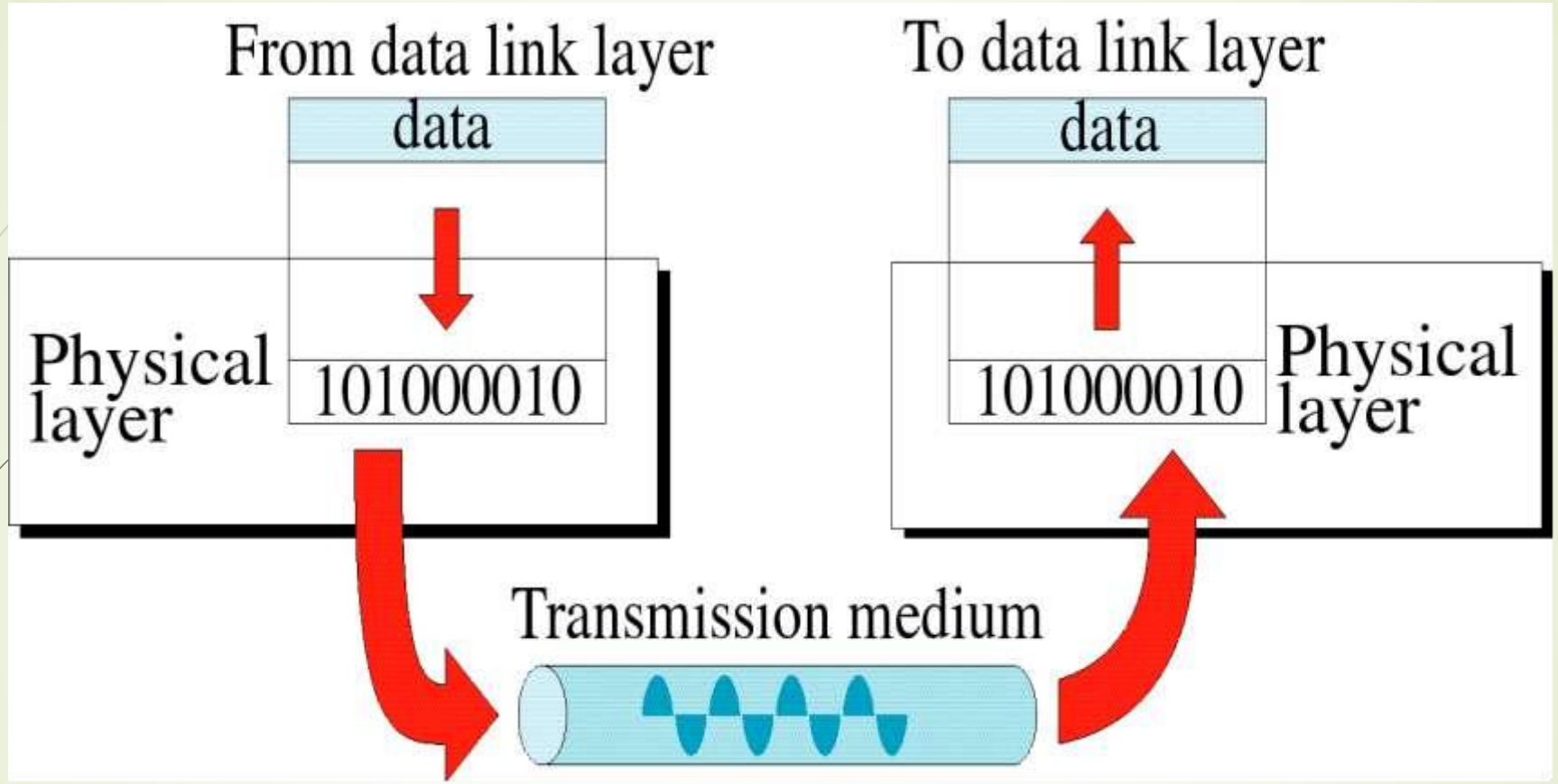
The physical layer is concerned with **transmitting raw bits** over a transmission medium.

- ✓ Typical questions here are how many **volts** should be used to **represent a "1"** and how many volts for a "0", how many microseconds a bit lasts, how many pins the network connector has and what each pin is used for, and what is the type of the used transmission medium.

❑ **To the end systems, the physical layer carries out the following functions:**

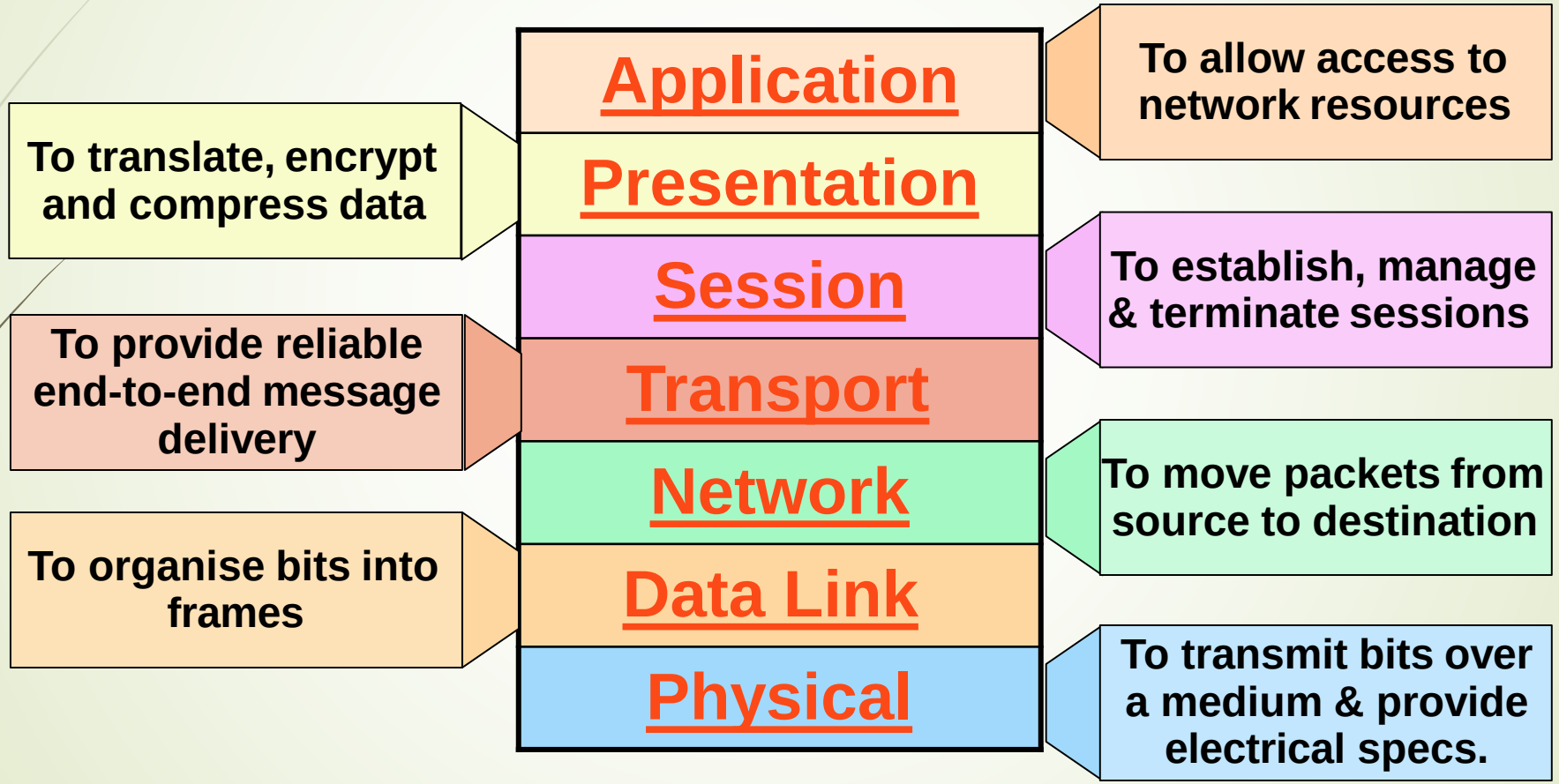
➨**Bit representation** – encode bits into electrical or optical signals having characteristics suitable for transmission over the medium.

- ➨ **Transmission rate** – The number of bits sent each second
- ➨ **Physical characteristics** of transmission media
- ➨ **Synchronizing** the sender and receiver clocks
- ➨ **Transmission mode** – simplex, half-duplex, full duplex
- ➨ **Physical Topology** – how devices are connected – ring, star, mesh, bus topology.



Physical Layer

Summary of Functions of Layers



Any Questions?