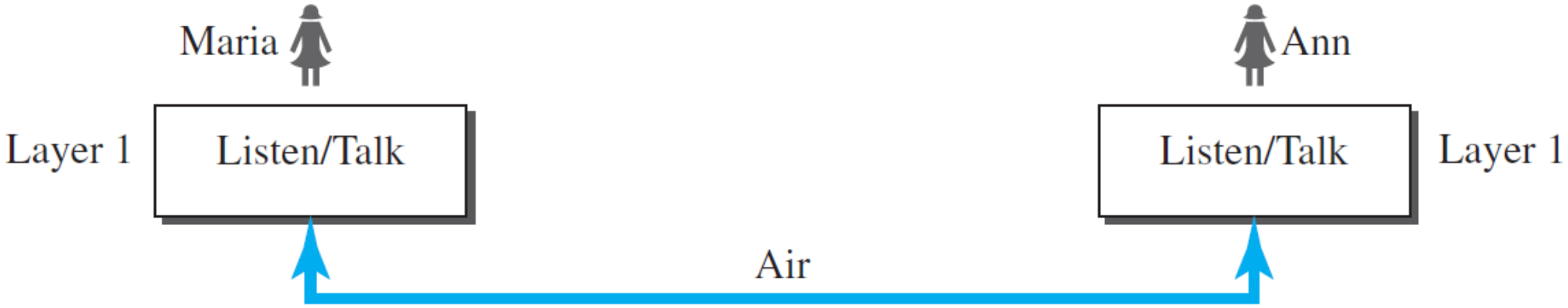
Layered Protocols

Outline

Layered Architecture

- Protocol Layering
- Principles
- Need of Layering
- Internet Protocol Stack

Protocol Layering



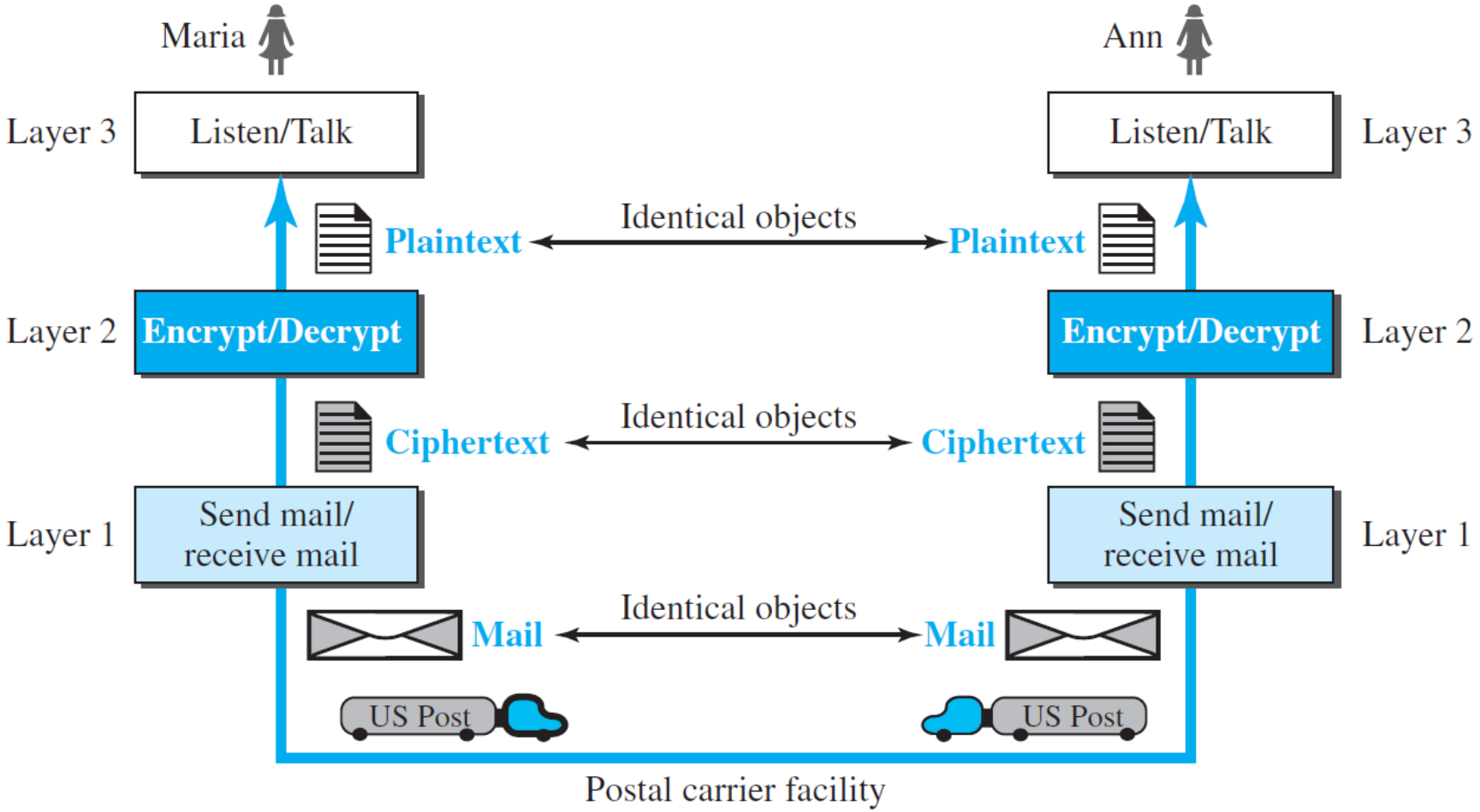
FIRST SCENARIO:

- **Simple Communication** : Only one **layer** i.e. **Face to Face**, in **same Language**.
- **Again a set of rules needs to be followed.**
  - They should **greet** each other.
  - Second, they their **vocabulary** to the **level of their friendship** (formal / informal)
  - Third, same **Language**.
  - Fourth, the conversation should be a **dialog**, not a monolog
  - Fifth, they should **exchange some nice words** when they **leave**.
- A **friendly protocol** is there.

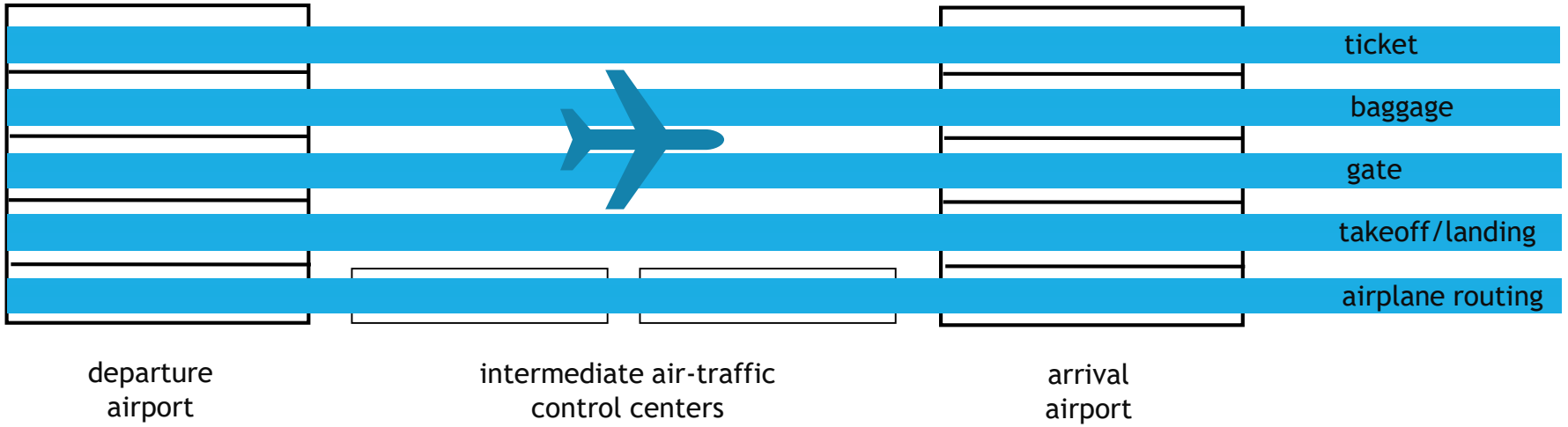
SECOND SCENARIO: Classroom Teaching

Protocol Layering … (contd.)

A Modular Approach



A Layered Approach



layers: each layer implements a **service**

- via its **own internal-layer actions**
- **relying on services** provided by **layer below**

Protocol Layering:

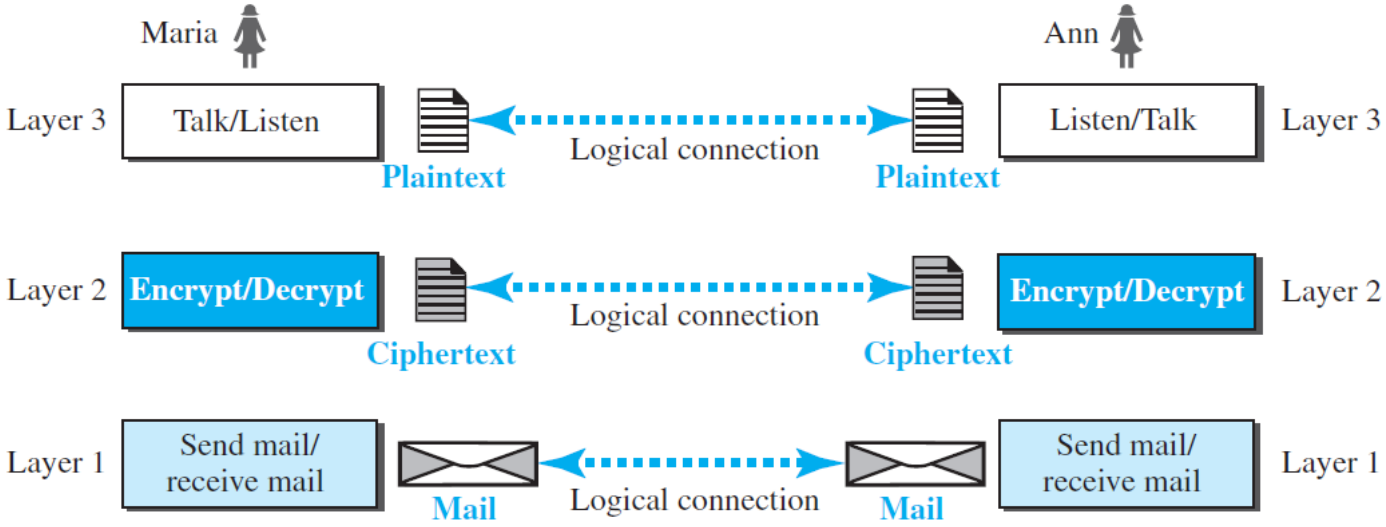
Principles & Logical Connection

First Principle

- *For Bidirectional Communication* : **Each layer** is able to **perform two opposite tasks**, one in each direction
  - **Example :** 3rd layer task is to listen (in one direction) and *talk* (in the other direction).
  - 2nd layer needs to be able to encrypt and decrypt.
  - 1st layer needs to send and receive mail.

Second Principle

- *Follow in protocol layering* : That the **two objects under each layer** at **both sites** should be **identical**.
  - **Example :** The object under layer 3 at both sites should be a plaintext letter.
  - The object under layer 2 at both sites should be a ciphertext letter.
  - The object under layer 1 at both sites should be a piece of mail.



Need of Layering

Dealing with Complex Systems:

- Explicit structure allows identification,
- Relationship of complex system's pieces

Layered Model

- Modularization eases maintenance,
- Updating of system

Internet Protocol Stack

Application:

Supporting network applications
- FTP, SMTP, HTTP

Transport:

Process-process data transfer
- TCP, UDP

Network:

Routing of datagrams from source to destination
- IP, routing protocols

Link:

Data transfer between neighboring network elements
- Ethernet, 802.111 (WiFi), PPP

Physical:

Bits “on the wire”

OSI Model

OSI Model

OSI Reference Model

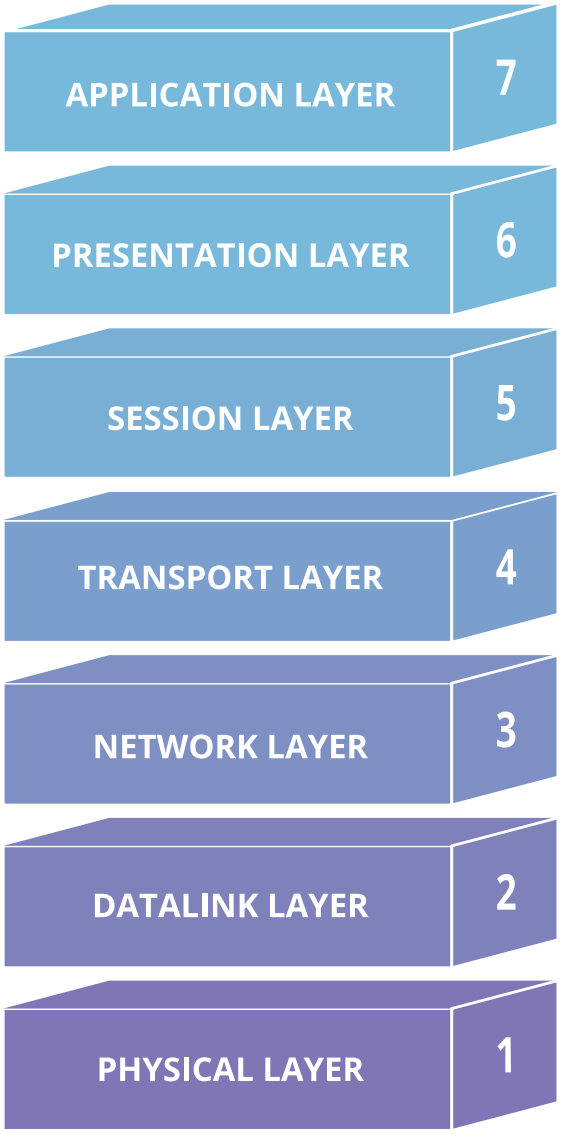
- Proposed in **1984** by **International Standards Organization (ISO)**
- Internationally standardized network architecture.
- **Conceptual model** that **enables diverse communication systems** to **communicate** using **standard protocols**

OSI = "Open Systems Interconnection"

- Deals with open systems,
- i.e. *systems open for communications with other systems*.
  - Any two systems which ***conform*** to the ***reference model*** and ***associated standards***

Model has **7 layers**.

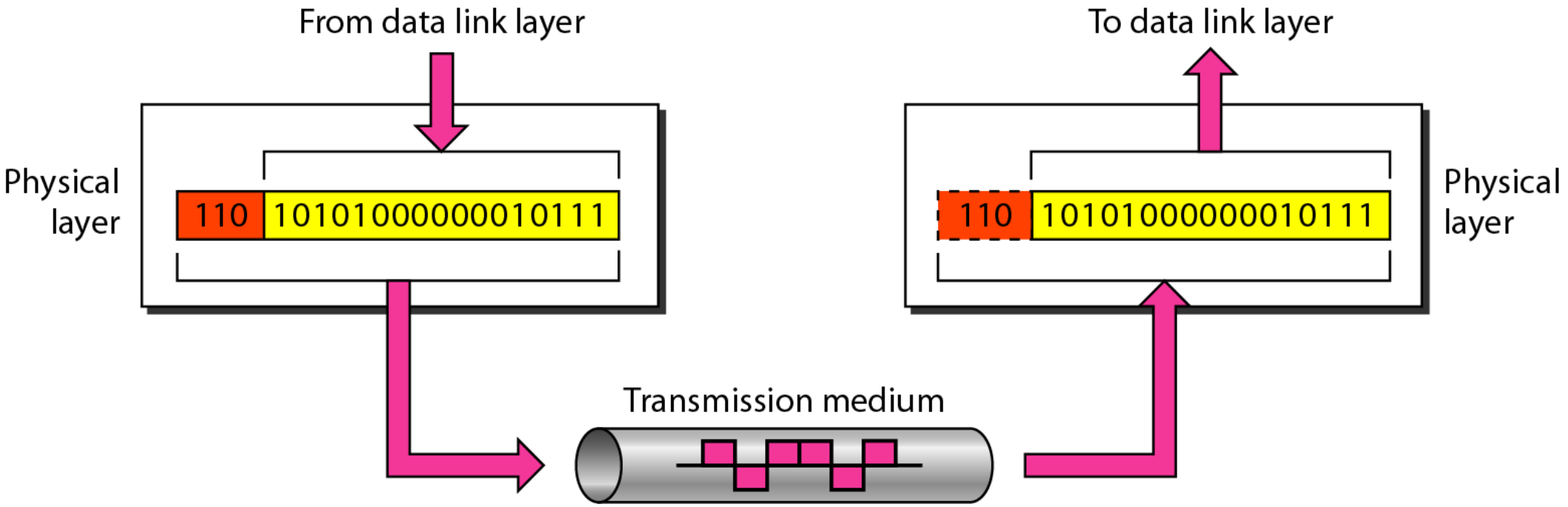
7 Layers of OSI Model



- Human-computer interaction layer, where applications can access the network services
- Ensures that data is in a usable format and is where data encryption occurs
- Maintains connections and is responsible for controlling ports and sessions
- Transmits data using transmission protocols including TCP and UDP
- Decides which physical path the data will take
- Defines the format of data on the network
- Transmits raw bit stream over the physical medium

OSI Model: **Physical Layer**

The physical layer is responsible for movements of individual bits from one hop (node) to the next

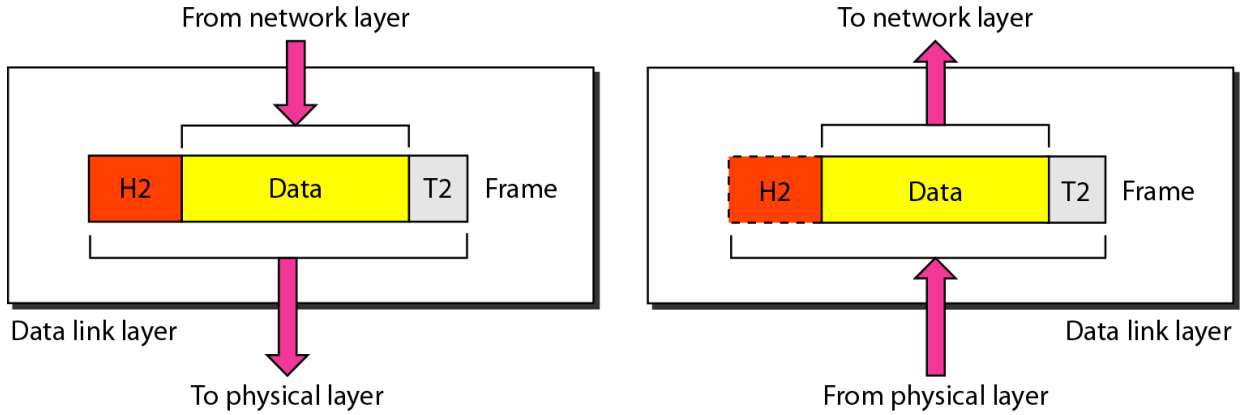


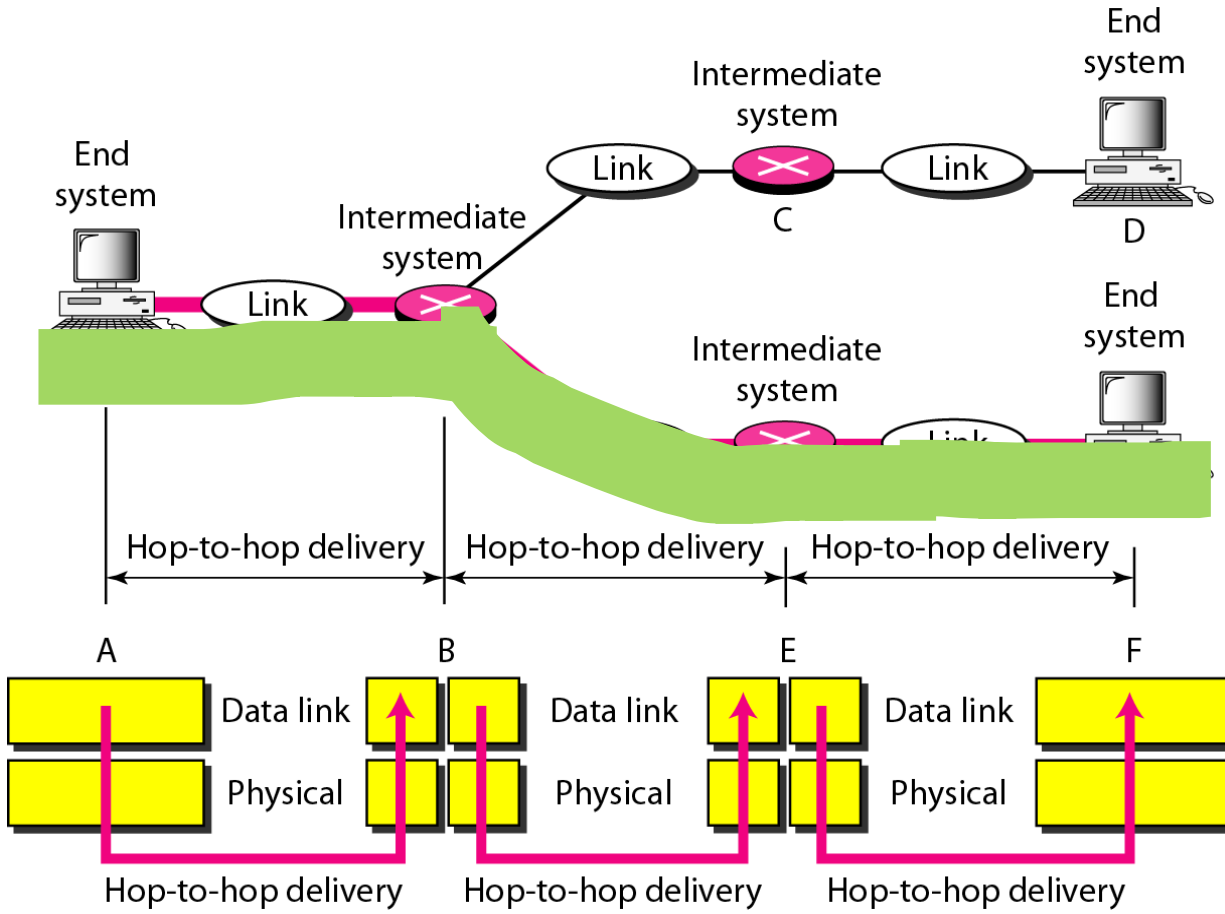
We cannot transfer raw bits on the medium. We need to convert that to a signal.

OSI Model: Data Link Layer

Facilitates **data transfer** between **two devices** on the **SAME network**

- Takes packets from the ***network layer*** and breaks ***them into smaller pieces*** called ***frames***.
  - It takes a datagram and encapsulates it in a packet called a frame.
- **Responsible** for **moving** the **packet** through the **link**.





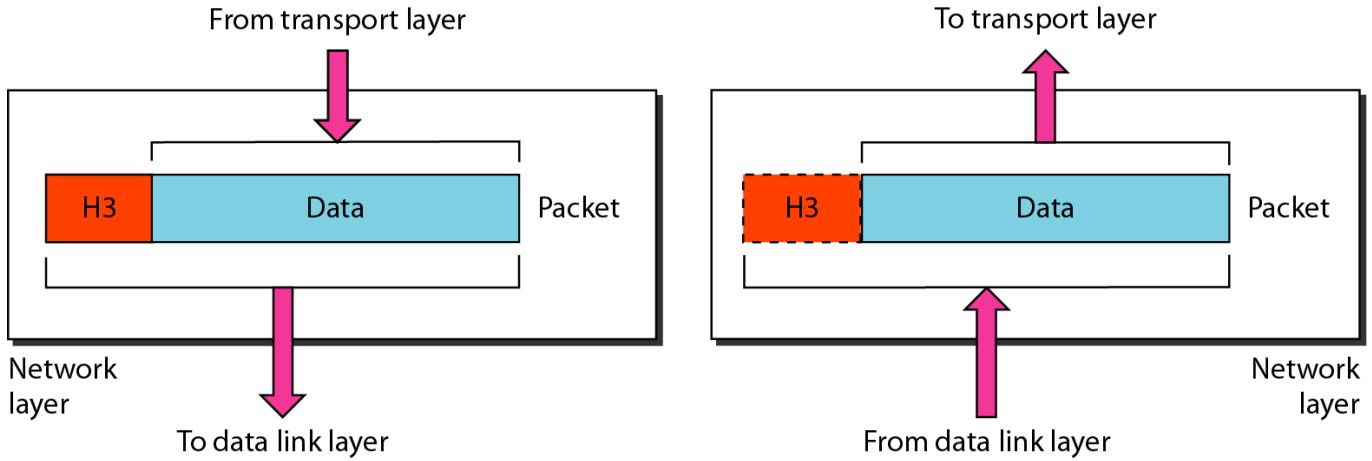
OSI Model: **Data Link Layer**

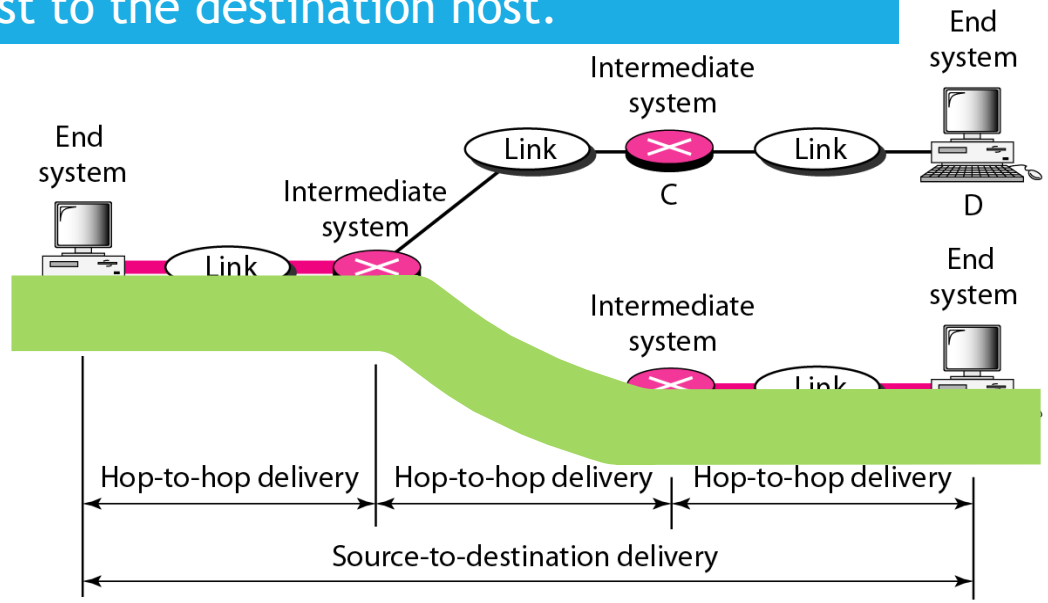
- ***Breaks*** the outgoing ***data*** into ***frames*** and ***reassemble*** the ***received frames***
- Attempts to provide ***reliable communication*** over the ***physical layer interface***.
- Some link-layer protocols provide ***complete error detection*** and ***correction***, some provide only ***error correction***.
- ***Create*** and ***detect*** frame ***boundaries***.
- ***Handle errors*** by implementing an ***acknowledgement*** and ***retransmission*** scheme.
- Implement ***flow control***
- Supports ***points-to-point*** as well as ***broadcast*** communication.
- Supports ***simplex***, ***half-duplex*** or ***full-duplex*** communication.

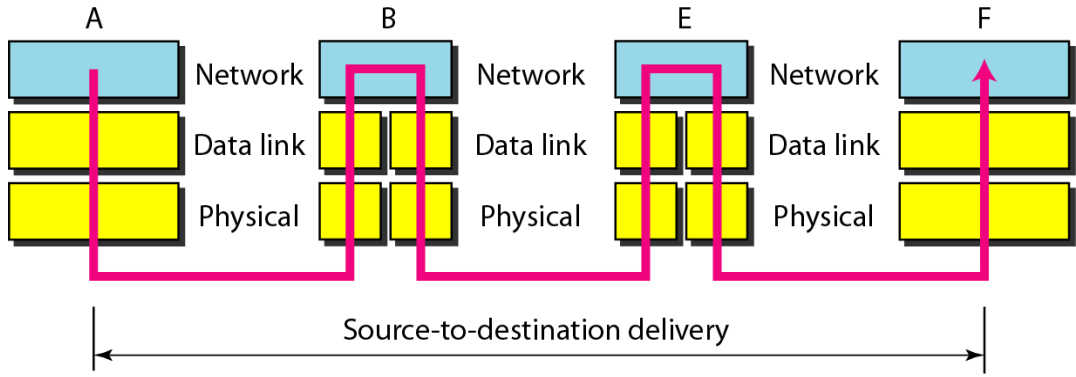
OSI Model: Network Layer

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

- Responsible for facilitating data transfer between two different networks.
- Responsible for creating a connection between the source computer and the destination computer.
- Communication at the network layer is host-to-host or between two different end systems on different networks.







OSI Model: Network Layer

- Network layer in the Internet includes the ***main protocol***, ***Internet Protocol (IP),***
  - that defines the **format of the packet**, called a **datagram** at the network layer.
- IP also defines the ***format*** and the ***structure*** of ***addresses used*** in this layer.
- IP is also responsible for ***routing a packet from its source to its destination***,
  - Defines the ***most optimum path*** the ***packet should take*** from the source to the destination
  - Achieved by each router forwarding the datagram to the next router in its path.
- IP is a ***connectionless protocol*** that provides ***no flow control***, ***no error control***, and ***no congestion control*** services.
- Defines ***logical addressing*** so that any ***endpoint*** can be ***identified***.
- Facilitates ***interconnection*** between ***heterogeneous networks*** (Internetworking).
- The network layer also defines ***how to fragment a packet into smaller packets*** to ***accommodate different media***.

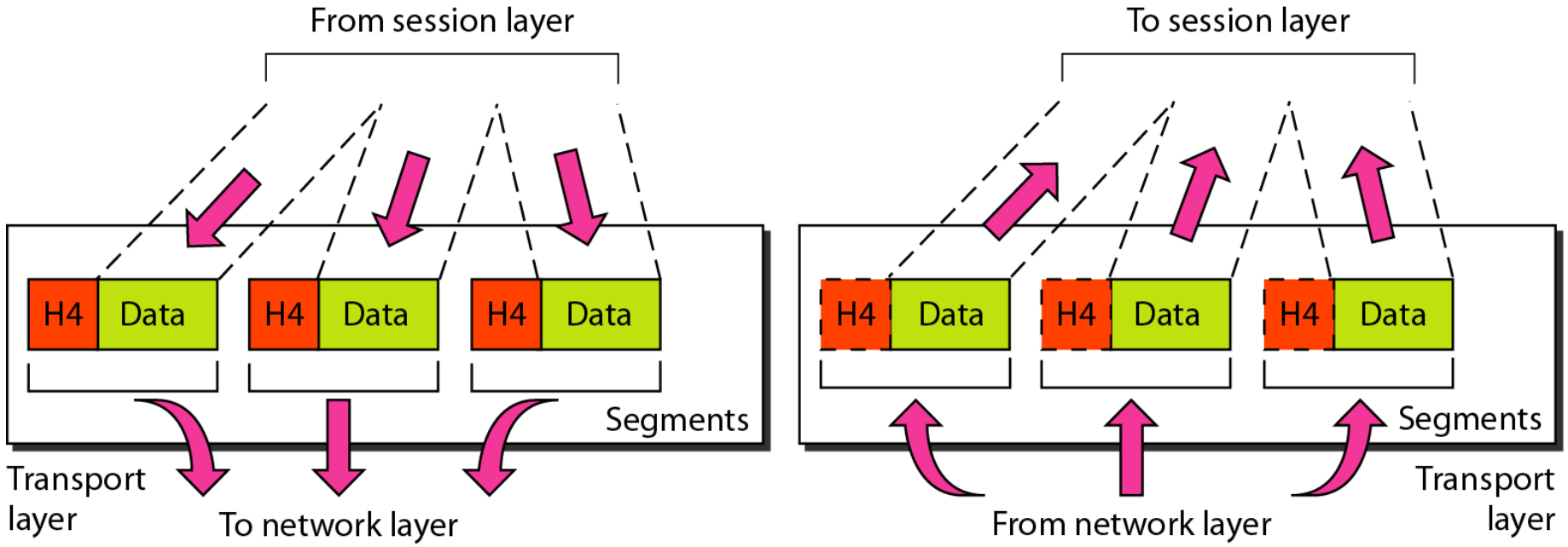
OSI Model: Transport Layer

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

The data in the transport layer is referred to as **Segments**.

It is responsible for the **End to End Delivery** of the **complete message.**

Provides the **acknowledgement** of the **successful data transmission** and **re-transmits** the data if an error is found.



OSI Model: Transport Layer

Responsible for **end-to-end communication between the two devices**.

Includes **taking data** from the **session layer** and **breaking** it up **into chunks** called **segments** before sending it to Network layer.

On the **receiving device** is **responsible** for **reassembling** the **segments** into **data** the session layer can consume.

Responsible for **flow control** and **error control**.

- Flow control ***determines an optimal speed of transmission*** to ensure that a sender with a fast connection doesn't overflood a receiver with a slow connection.

Performs **error control on the receiving end** by **ensuring that the data received is complete**, and requesting a retransmission if it isn't.

The functions of the transport layer are :

- **Segmentation and Reassembly**
- **Service Point Addressing**
  - To deliver the message to correct process, transport layer header includes a **port address**.
  - Thus by specifying this address, transport layer makes sure that the message is delivered to the correct process.

OSI Model: Transport Layer

At sender's side:

- Transport layer receives the formatted data from the upper layers, performs Segmentation and also implements Flow & Error control to ensure proper data transmission.
- It also adds ***Source and Destination port number*** in its ***header*** and forwards the segmented data to the Network Layer.
- **Note:** The sender need to know the port number associated with the receiver's application.
  - Generally, this destination port number is configured, either by default or manually.
  - For example, when a **web application** makes a request to a **web server**, it typically uses **port number 80**, because this is the default port assigned to web applications.

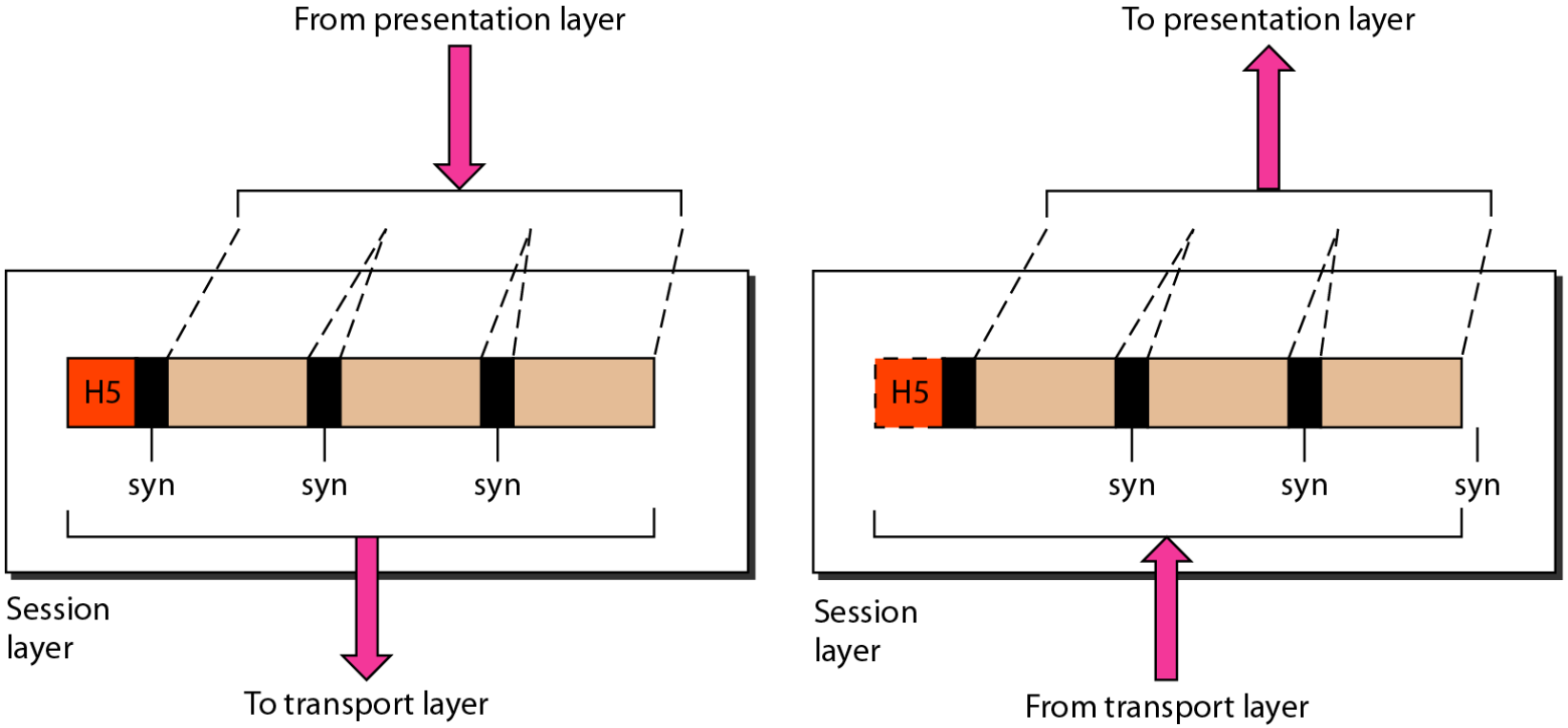
At receiver's side:

- Transport Layer ***reads*** the ***port number*** from its ***header*** and ***forwards*** the ***Data*** which it has received to the ***respective application***.
- It also performs ***sequencing*** and ***reassembling*** of the ***segmented data***.

OSI Model: **Session Layer**

The session layer is responsible for dialog control and synchronization.

It is responsible for establishment of **connection**, **maintenance** of **sessions**, **authentication** and also ensures **security**.



OSI Model: Session Layer

The functions of the session layer are :

Session establishment, maintenance and termination:

- The layer allows the two processes to **establish, use and terminate** a connection.
- **Synchronization :**
  - Allows a **process** to **add checkpoints** which are considered as **synchronization points** into the data.
  - These synchronization point **help to identify the error** so that the **data is re-synchronized properly**,
- Provides mechanism **for controlling the dialogue between the two end systems**.
  - Defines how to **start, control and end conversations** (called sessions) between applications.
- Requests for a **logical connection** to be **established** on an **end-user's request**.
- Any necessary **log-on or password validation** is also handled by this layer.
- Session layer can also provide **check-pointing mechanism** such that **if a failure of some sort occurs between checkpoints**, all data can be **retransmitted from the last checkpoint**.

OSI Model: Presentation Layer

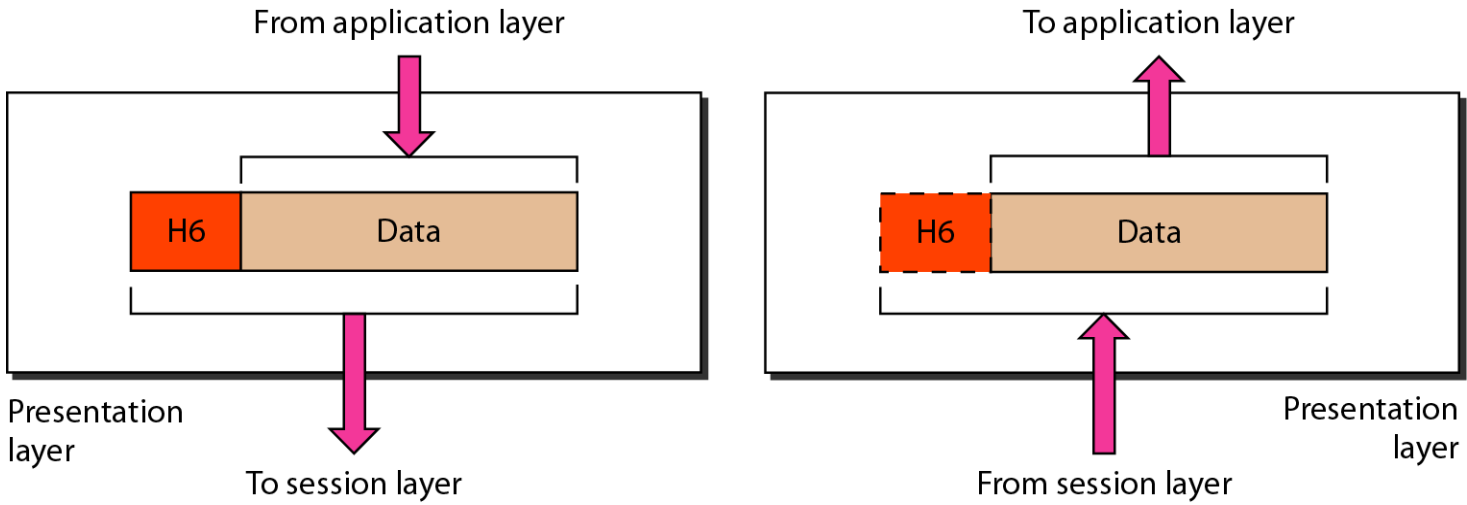
The presentation layer is responsible for translation, compression, and encryption.

Presentation layer : **Translation layer**.

- *The data from the application layer is extracted here and manipulated as per the required format to transmit over the network.*

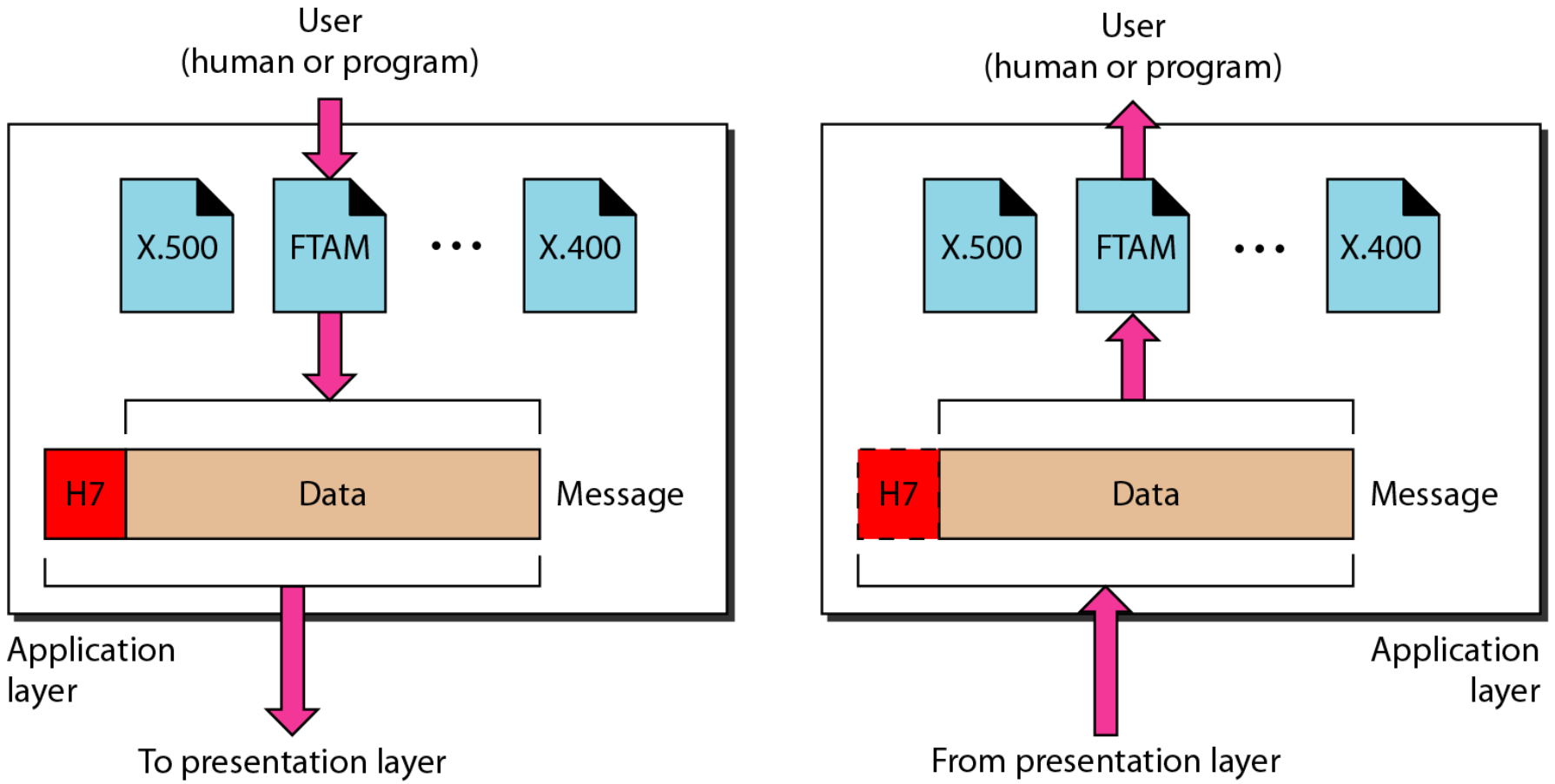
Functions:

- **Translation** : For example, ASCII.
- **Encryption**/ **Decryption** :
  - Data encryption encodes the data.
  - Encrypted data is known as the cipher text
  - Decrypted data is known as plain text.
  - A **key value** is used for encrypting as well as decrypting data.
- **Compression**:
  - Reduces the number of bits that need to be transmitted on the network.



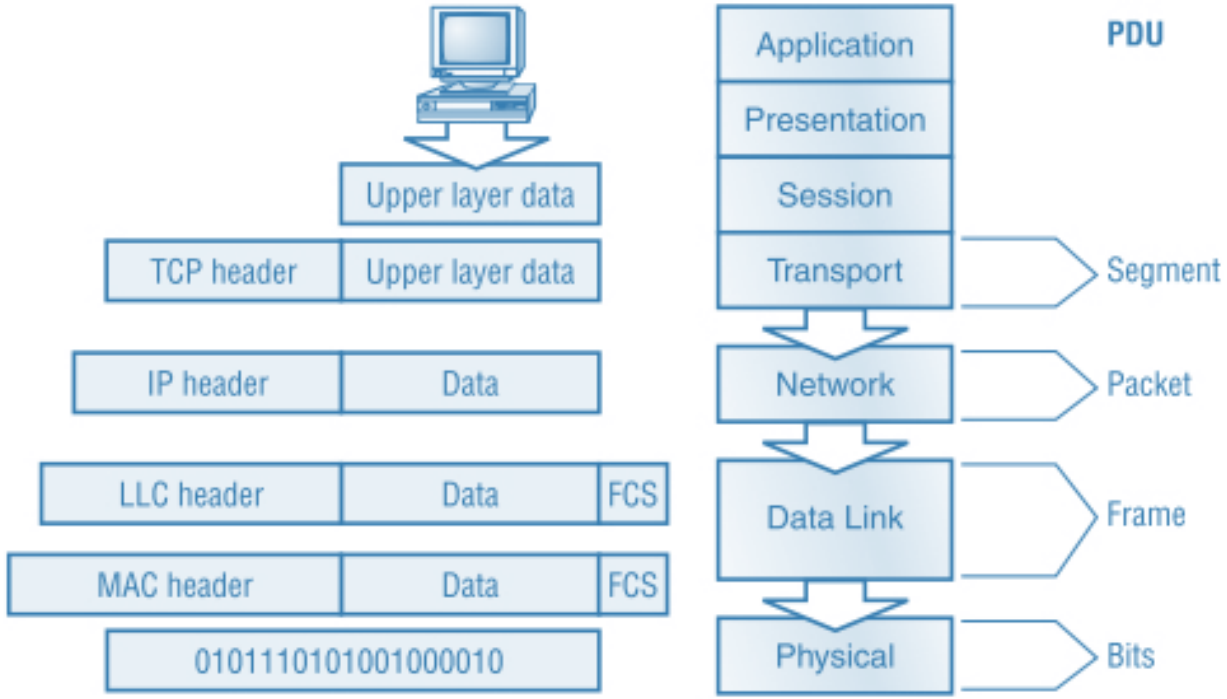
OSI Model: **Application Layer**

The application layer is responsible for providing services to the user.

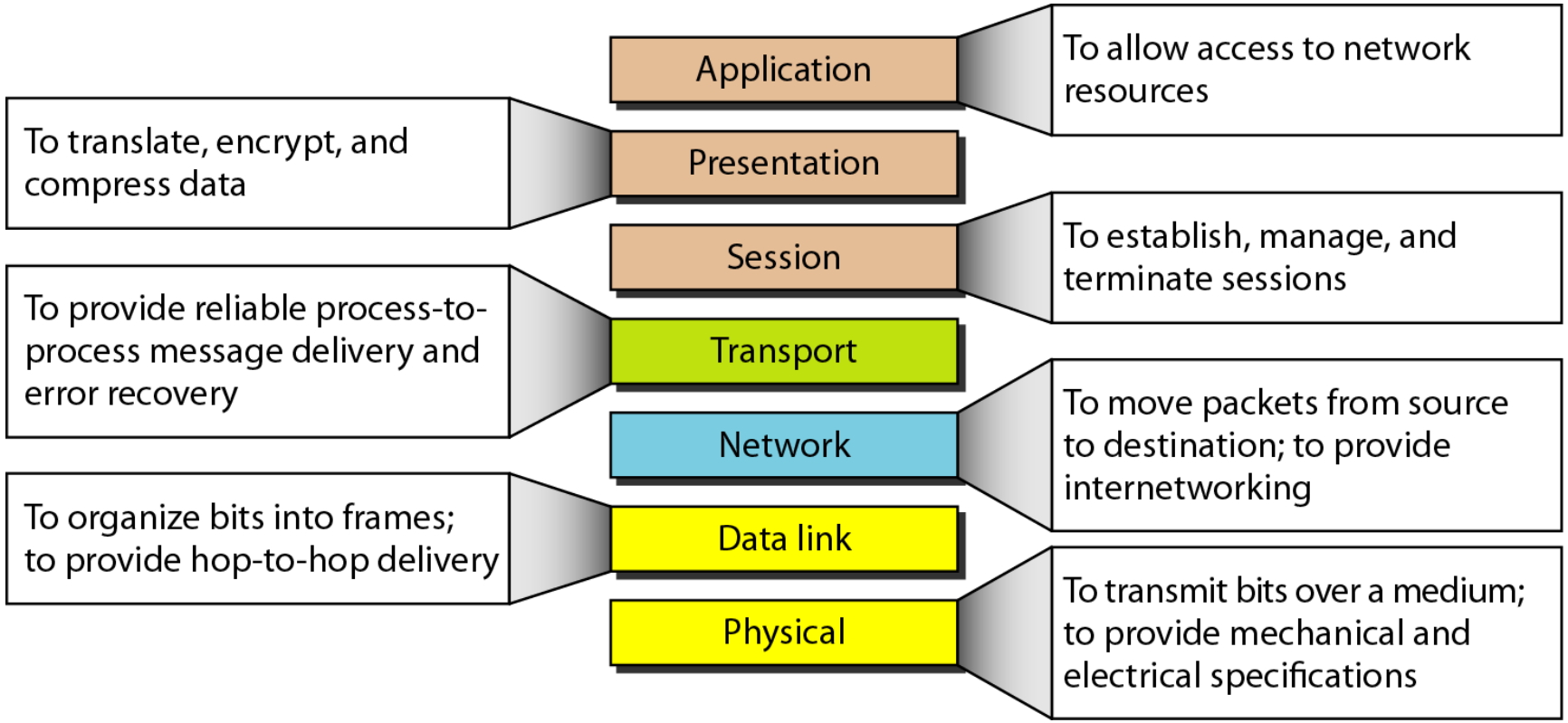


OSI in Action

- **A message begins:**
  - At the top application layer and
  - Moves down the OSI layers to the bottom physical layer.
- **As the message descends,**
  - Each successive OSI model layer adds a header to it.
- **Each header is layer-specific information**
  - Explains what functions the layer carried out.
- **Conversely, at the receiving end:**
  - Headers are striped from the message as it travels up the corresponding layers.



OSI Model: **Summary**



TCP/IP Model

Outline

TCP/IP Model

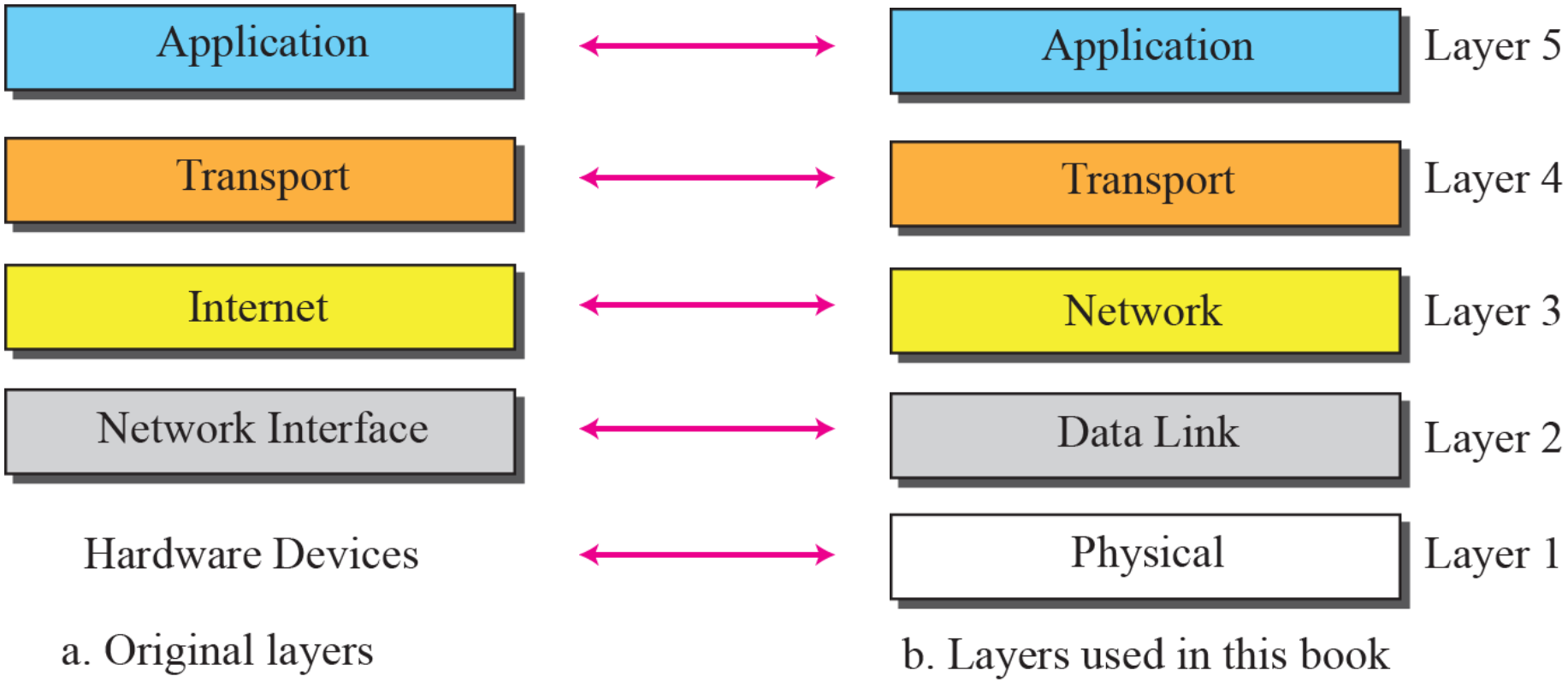
- Physical Layer
- Data Link Layer
- Network Layer
- Transport Layer
- Application Layer

Features of TCP/IP Model

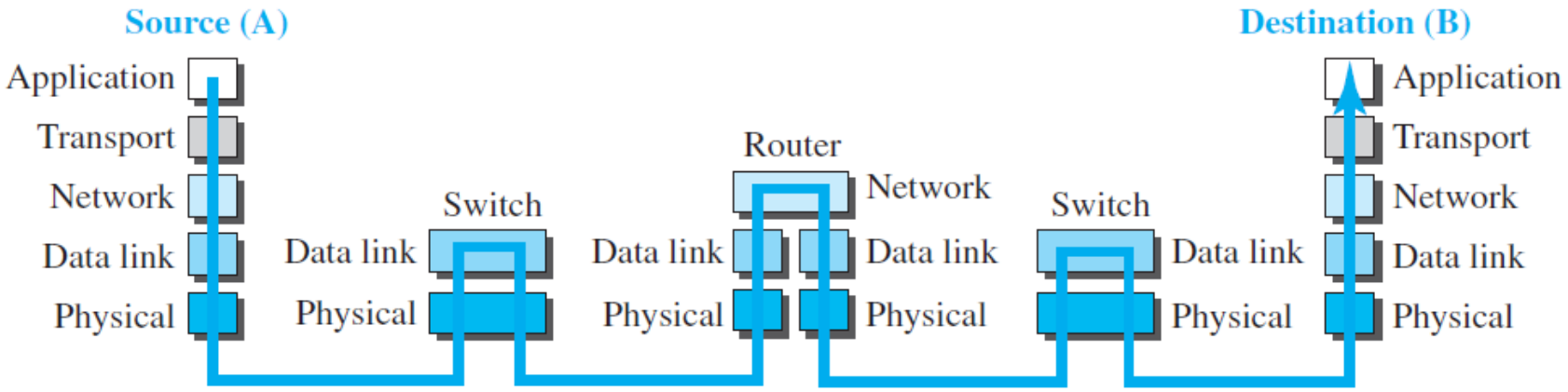
Comparison with OSI Model

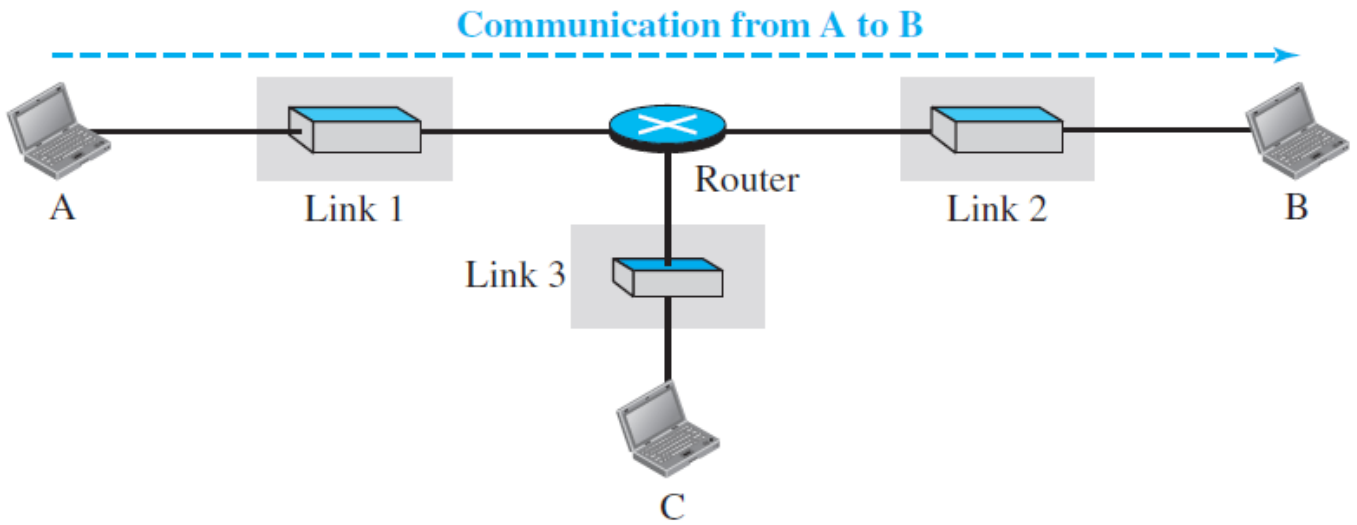
Addressing

TCP/IP Model


Application
Transport
Internet
Network Interface
Hardware Devices
a. Original layers
Application
Transport
Network
Data Link
Physical
Layer 5
Layer 4
Layer 3
Layer 2
Layer 1
b. Layers used in this book

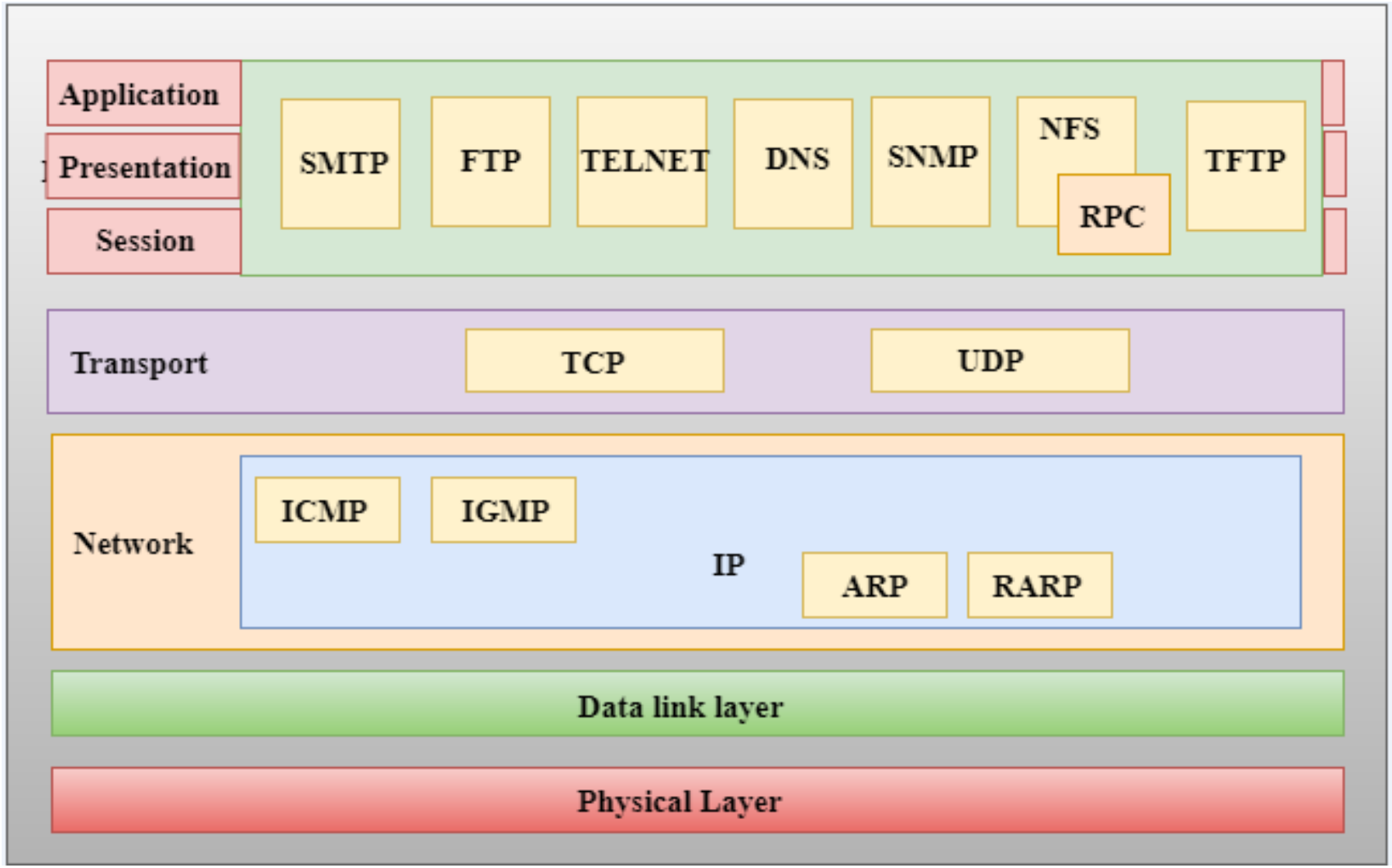
TCP/IP Model … (contd.)





- The unit of communication:
  - At the network layer is a datagram.
  - At transport layer is a segment, user datagram, or a packet, depending on the specific protocol used in this layer.
  - At application layer is a message.

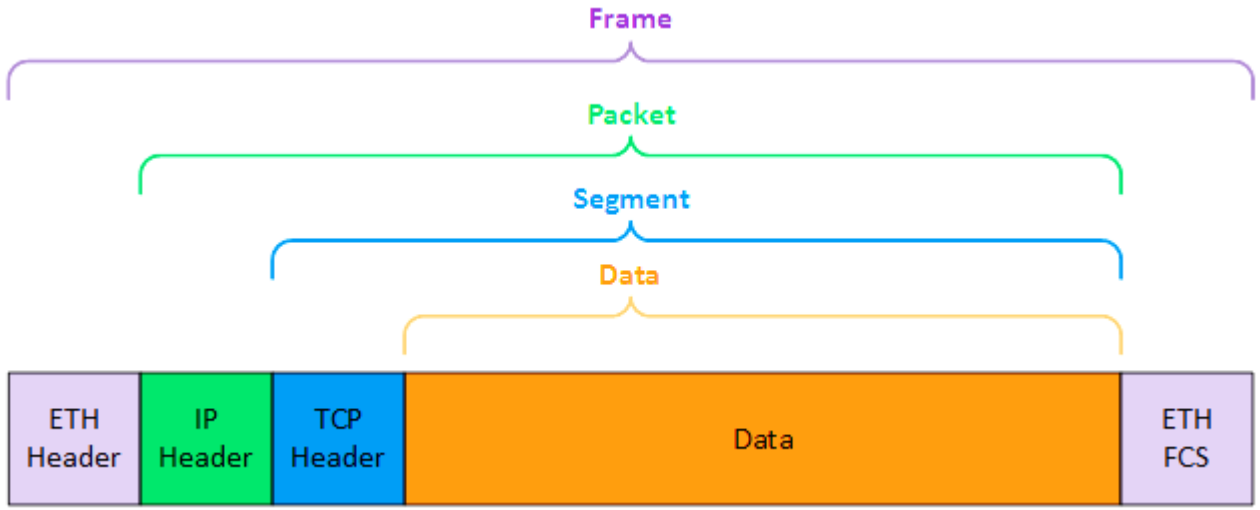
TCP/IP Model : **Layers**



Protocol Data Unit

PDU are named according to the protocols of the TCP/IP Suite

- Application Layer - Data
- Transport Layer - Segment
- Network Layer – Packet
- Data Link Layer - Frame
- Physical Layer - Bits



TCP/IP Model : **Layers**

Physical Layer

- Specifies the characteristics of the hardware to be used for the network.
- Specifies the physical characteristics of the communications media.

Data Link Layer

- Format the IP datagram into a **frame**.
- Supports all the standard and proprietary protocols.
  - Protocols attach a header and a footer to "frame" the datagram.
  - Each link-layer protocol may provide a different service.
  - Like: Error Detection and Error Correction,
- Then the data-link layer passes the frame to the physical layer.

TCP/IP Model : Network Layer

- Also known as the internet layer.
  - **Accepts and delivers packets for the network**.
- This layer includes the powerful **IP**, **ARP** and **ICMP**.

IP Protocol

- IP is responsible for the following:
  - **IP addressing -** The IP addressing conventions are part of the IP protocol.
  - **Host-to-host communications** – Determines the path a packet must take, based on the receiving host's IP address.
  - **Packet formatting** – Assembles packets into units that are known as IP datagrams.
  - **Fragmentation -** If a packet is too large for transmission over the network media, IP breaks the packet into smaller fragments. On the receiving host then reconstructs the fragments into the original packet.

ARP Protocol

- ARP assists IP in directing datagrams to the appropriate receiving host by mapping

ICMP Protocol

- Internet Control Message Protocol (ICMP) detects and reports network error conditions.
- ICMP reports on the following:
  - Dropped packets – Packets that arrive too fast to be processed
  - Connectivity failure – A destination host that cannot be reached)
  - Redirection – Redirecting a sending host to use another router

TCP/IP Model : Transport Layer

Ensure that packets arrive in sequence and without error,

- By acknowledgments of data reception, and retransmitting lost packets.

This type of communication is known as "end-to-end."

TCP Protocol

- TCP is a **connection-oriented protocol**
  - It first **establishes a logical connection** between **transport layers** at **two hosts before transferring data**.
- TCP provides
  - **Flow control :** Matching the sending data rate of the source host
  - **Error control** : Guarantee that the segments arrive at the destination without error and resending the corrupted ones
  - **Congestion control** to reduce the loss of segments due to congestion in the network.

UDP Protocol

- User Datagram Protocol (UDP), is a **connectionless** protocol
  - That *transmits user datagrams without first creating a logical connection*.
- In UDP, each user datagram is an independent entity
- Does not provide flow, error, or congestion control.

TCP/IP Model : Application Layer

The **topmost layer** in the TCP/IP model.

Allows the **user to interact with the application**.

Responsible for handling high-level protocols

Every application cannot be placed inside the application layer

- Except those who interact with the communication system.
- For example: **Text editor**.

Main protocols used in the application layer:

HTTP: (Hypertext transfer protocol)
- Allows us to access the data over the world wide web.

SNMP: (Simple Network Management Protocol)
- Framework used for managing the devices on the internet by using the TCP/IP protocol suite.

SMTP: (Simple mail transfer protocol)
- This protocol is used to send the data to another e-mail address.

DNS: (Domain Name Server)
- An IP address is used to identify the connection of a host to the internet uniquely.
- Prefer to use the names instead of addresses. So, the system that maps the name to the address is known as Domain Name System.

TELNET: (Terminal Network)
- Establishes the connection between the local computer and remote.

FTP: (File Transfer Protocol)
- FTP is a standard internet protocol used for transmitting the files from one computer to another computer.

TCP/IP Model : **Features**

The key features of the TCP/IP model are as follows:

Supports flexible architecture:

- **We can connect two devices with totally different architecture using the TCP/IP model.**

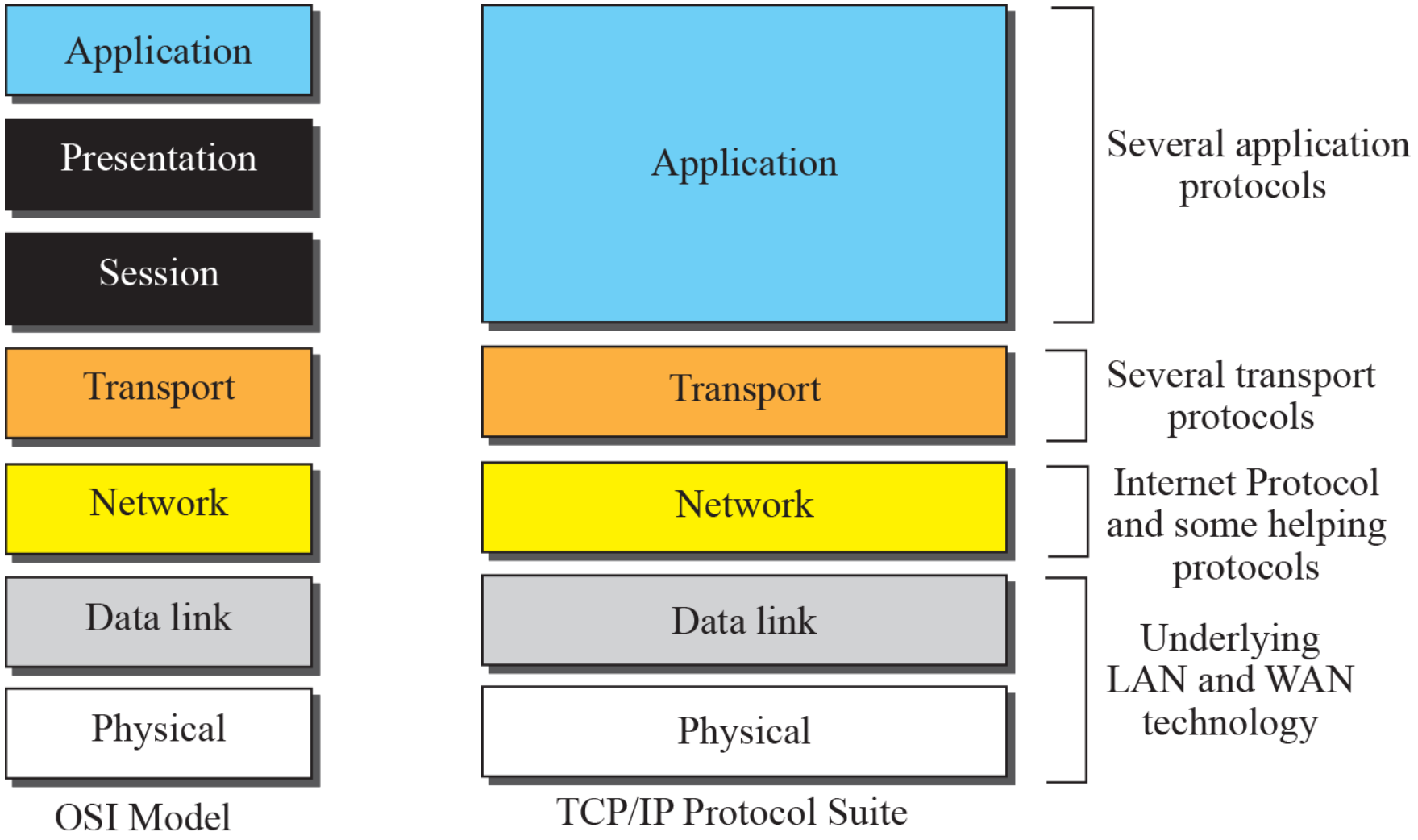
End-node verification:

- **The end-nodes(source and destination) can be verified, and connection can be made for the safe and successful transmission of data.**

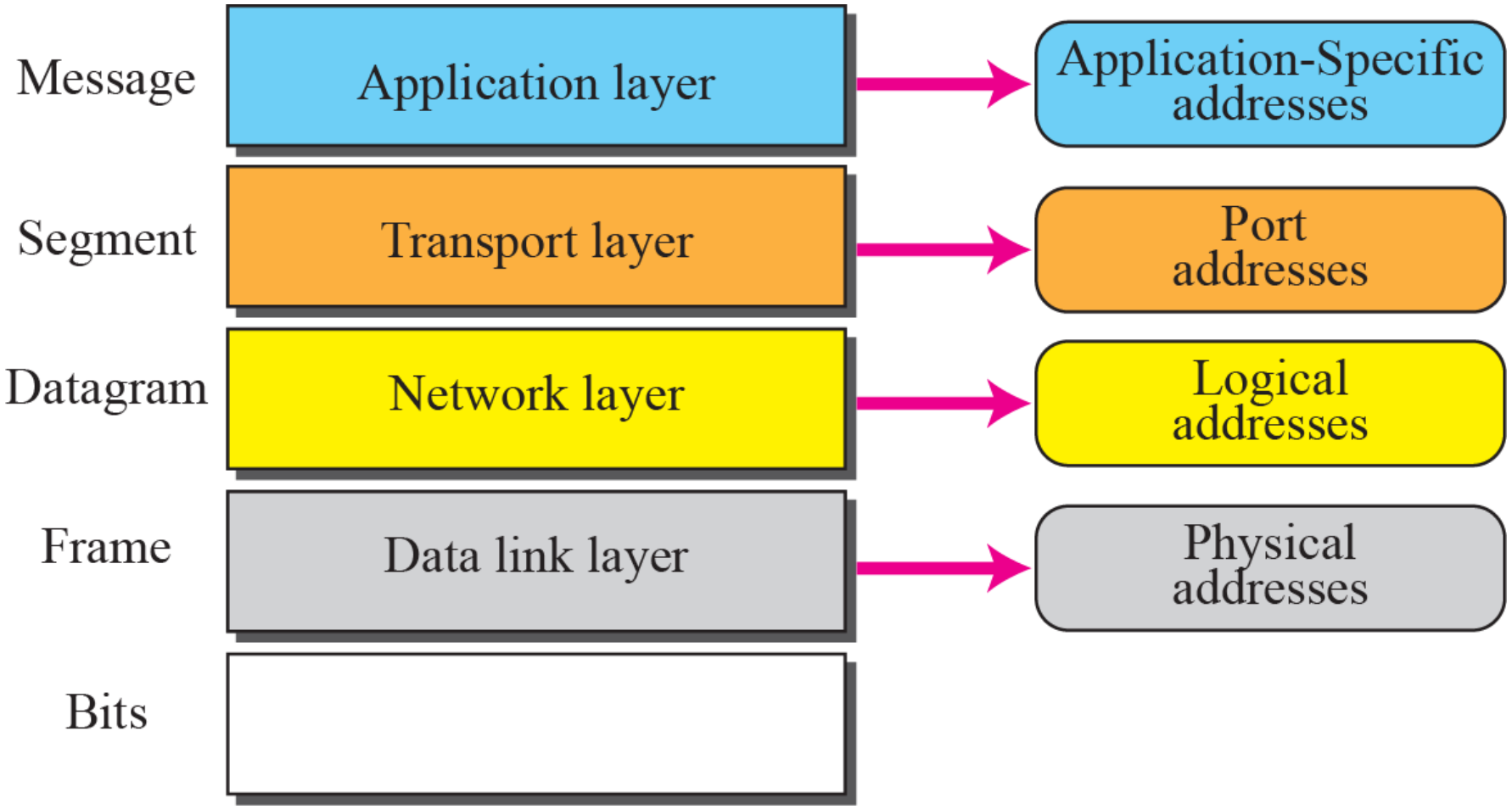
Dynamic Routing:

- **TCP/IP model facilitates the dynamic routing of the data packets through the shortest and safest path.**
- **Due to dynamic routing, the path taken by the data packet can not be predicted, and thus it improves data security.**

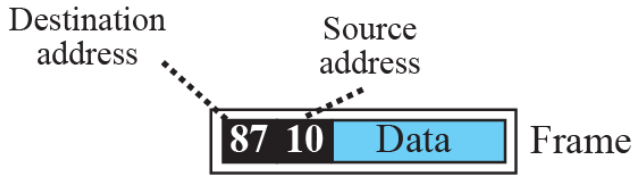
OSI & TCP/IP Model : **Comparison**

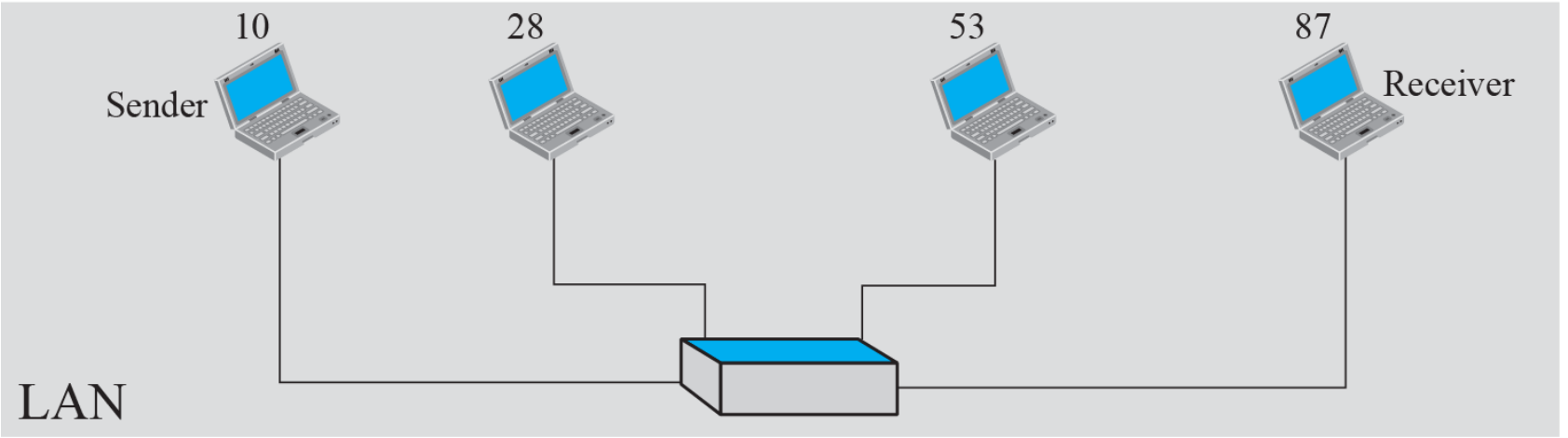


TCP/IP Model : **Addressing**



TCP/IP Model : Physical Addressing

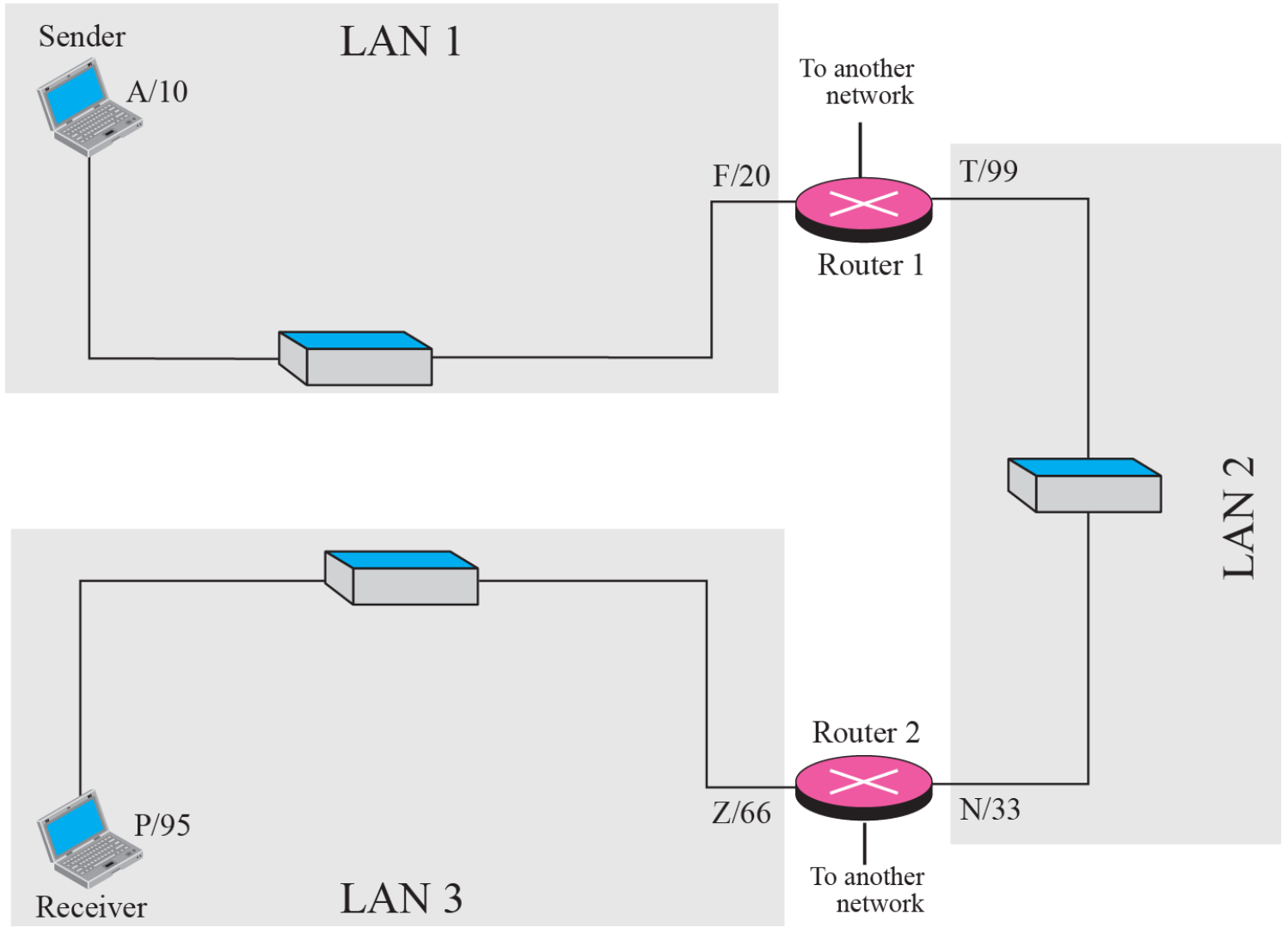




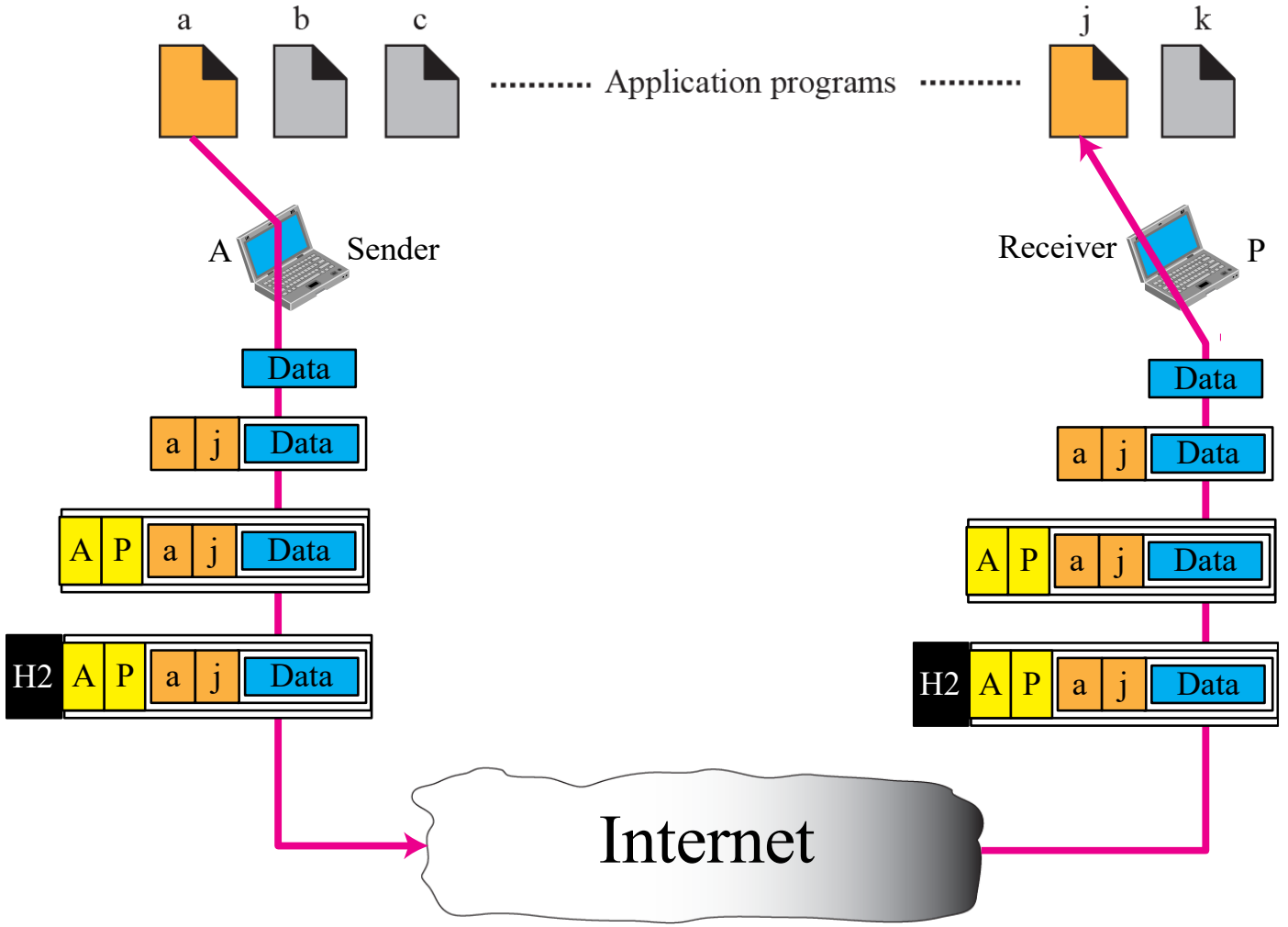
07:01:02:01:2C:4B

A 6-byte (12 hexadecimal digits) physical address

TCP/IP Model : Logical Addressing



TCP/IP Model : Port Addressing



Example : 753

A 16-bit port address represented as one single number

Note:

The physical addresses change from hop to hop, but the logical and port addresses usually remain the same.

Reference

Forouzan, A. Behrouz. *Data Communications & Networking*. 5th Edition. Tata McGraw-Hill Education.

Chapter 2 *Network Models*

Topics: 2.1, 2.2, 2.3

Thank You!