

COMPUTER NETWORKS

23CSPC303

Course outcomes

1. Explain the components and layered architecture of the TCP/IP protocol suite.
2. Apply data link layer techniques in network communication.
3. Demonstrate the application of routing protocols to facilitate network layer functionalities.
4. Apply transport layer protocols for TCP/UDP services, connections, and flow control mechanisms.
5. Illustrate application layer protocols in real word application

40 hours

Modules

Module 1: Introduction and Physical layer	No. of Hrs: 8
Introduction: Data Communications, Networks, Network Types, Network Models: Protocol Layering, TCP/IP Protocol suite, The OSI model, Introduction to Physical Layer: signals, signal impairment, multiplexing. Switching: Packet Switching and its types.	
Text Book 1: Chapter 1, Chapter 2, Chapter 3, Chapter 8	
Module 2: Data Link Layer	No. of Hrs: 9
Data Link Layer: Framing, Error Detection and Correction: Introduction, Block Coding, Cyclic Codes. Data link control: DLC Services: Framing, Flow Control, Error Control, Connectionless and Connection Oriented, Data link layer protocols, High Level Data Link Control. Media Access Control: Random Access, Controlled Access. Check Sum and Point to Point Protocol, Ethernet: Standard Ethernet.	
Text Book 1: Chapter 10, Chapter 11, Chapter 12, Chapter 13.2	
Module 3: Network Layer	No. of Hrs: 8
Network layer Services, performance, IPv4 Address, IPv4 Datagram, IPv6 Datagram, Introduction to Routing Algorithms, Unicast Routing Protocols: DVR, LSR, PVR, Unicast Routing protocols: RIP, OSPF, Multicasting Routing-MOSPF	
Text Book 1: Chapter 18, Chapter 19.1, Chapter 20, Chapter 21.3.2	

Modules

Module 4: Transport Layer	No. of Hrs: 8
User Datagram Protocol: UDP Services, applications, Transmission Control Protocol: TCP services, features, segments, TCP connections, flow control, Error control, Congestion control.	
Text Book 1: Chapter 24	
Module 5: Application Layer	No. of Hrs: 9
Introduction, Client-Server Programming, Socket interface programming. Standard Client-Server Protocols: World Wide Web and HTTP, FTP, Electronic Mail, Domain Name System(DNS), Secure Shell (SSH),	
Text Book 1: Chapter 25.1,25.2 Chapter 26	

Learning Resources

Text Book:

1. Behrouz A. Forouzan, Data Communications and Networking with TCP-IP Protocol Suite, 5th Edition, Tata McGraw-Hill, 2022.

Reference Books:

1. Larry L. Peterson and Bruce S. Davie: Computer Networks – A Systems Approach, 4th Edition, Elsevier, 2019.
2. Nader F. Mir: Computer and Communication Networks, 2nd Edition, Pearson Education, 2015
3. William Stallings, Data and Computer Communication 10th Edition, Pearson Education, Inc., 2014.

Web links:

1. Computer Networks and Internet Protocol:
<https://www.digimat.in/nptel/courses/video/106105183/L01.html>
2. Computer Networks: Crash Course: <https://www.youtube.com/watch?v=3QhU9jd03a0>
3. Computer networks: <https://nptel.ac.in/courses/106105080>

Module 1

- **Module 1: Introduction and Physical layer Introduction**

Data Communications, Networks, Network Types, Network Models: Protocol Layering, TCP/IP Protocol suite, The OSI model, Introduction to Physical Layer: signals, signal impairment, multiplexing. Switching: Packet Switching and its types. Text Book 1: Chapter 1, Chapter 2, Chapter 3, Chapter 8

1.1 Introduction- Data communications

Data: Small units of information that are transmitted between devices.

Communication is an act of sending or receiving data.

Why Data communication?

Data communication is important because **it allows for the exchange of information between devices and systems, which enables a variety of applications and services.**

Introduction- Data communications

- Data communications are the **exchange of data between two devices** via some **form of transmission medium** (wired/Wireless).
- For data communications to occur, the **communicating devices must be part of a communication system** made up of a **combination of hardware (physical equipment) and software (programs)**.
- The effectiveness of a data communications system depends on **four fundamental characteristics: delivery, accuracy, timeliness, and jitter**.

Introduction- Data communications

1. **Delivery.** The system must **deliver data to the correct destination.** Data must be received by the **intended device or user and only by that device or user.**

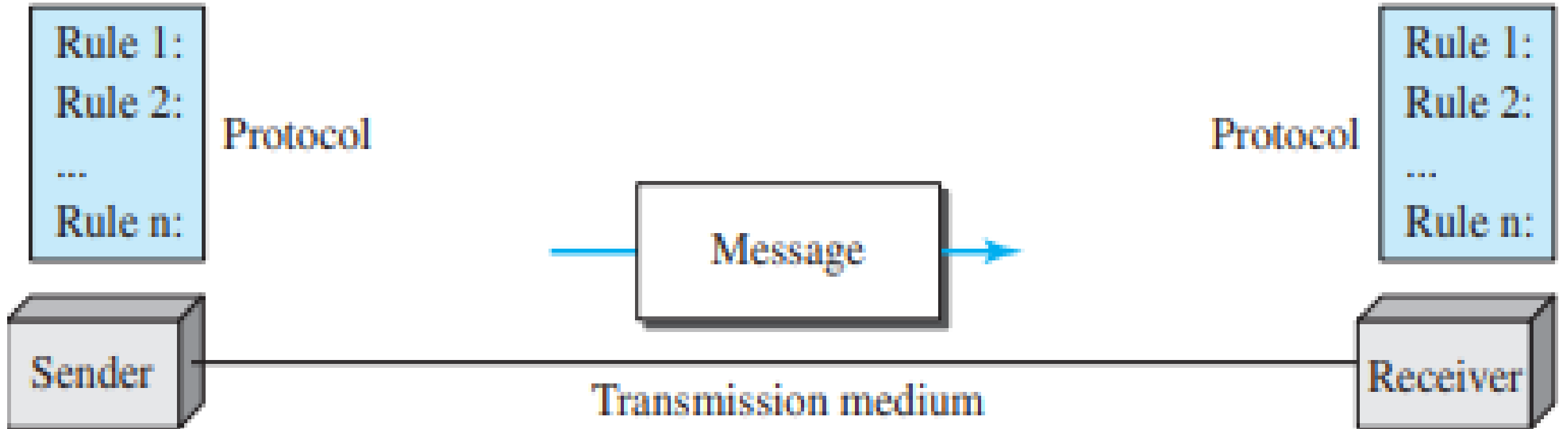
2. **Accuracy.** The system must **deliver the data accurately.** Data that have been altered in transmission and left uncorrected are unusable.

3. **Timeliness.** The system must deliver **data timely; late data is useless.** For video and audio, timely delivery means transmitting data as produced, in the same order, and without significant delay, known as real-time transmission.

4. **Jitter.** Jitter refers to the **variation in the packet arrival time. It is the Is the gap between packet arrivals.**

1.1 Components

A data communications system has five components.



1.1 Components

A data communications system has five components.

1. **Message.** The message is the information (data) to be communicated. Popular forms of information include text, numbers, pictures, audio, and video.
2. **Sender.** The sender is the device that sends the data message. It can be a computer, workstation, telephone handset, video camera, and so on.
3. **Receiver.** The receiver is the device that receives the message. It can be a computer, workstation, telephone handset, television, and so on.

1.1 Components

A data communications system has five components.

4. Transmission medium. The transmission medium is the physical path by which a message travels from sender to receiver. Some examples of transmission media include twisted-pair wire, coaxial cable, fiber-optic cable, and radio waves/electro magnetic waves.

5. Protocol. A protocol is a set of rules that govern data communications. It represents an agreement between the communicating devices.

1.1.2 Data Representation

Information today comes in **different forms such as text, numbers, images, audio, and video.**

1. **Text:** Represented as bit patterns using Unicode (represent over one million characters) or ASCII (represent only 256 characters), allowing for efficient storage and exchange of written communication.
2. **Numbers:** Directly converted to binary numbers for mathematical operations, enabling computers to perform calculations and data analysis.

1.1.2 Data Representation

3. Images: Composed of pixels, each assigned a bit pattern, with resolution affecting memory storage and image quality, allowing for digital visualization and sharing.

4. Audio: Continuous signal represented as bit patterns for recording, playback, and broadcasting, enabling digital music and voice communication.

5. Video: Combination of images or continuous signal represented as bit patterns to convey motion, allowing for digital video recording, playback, and streaming.

1.1.3 Data Flow

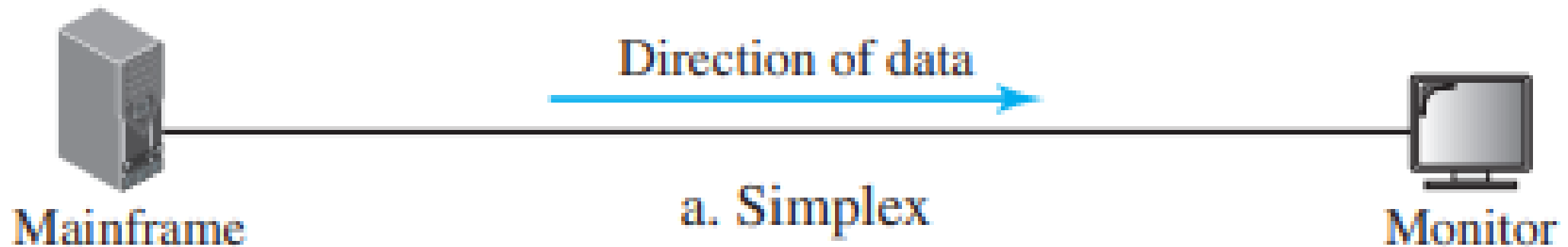
Communication between two devices can be

- **Simplex**
- **Half-duplex**
- **Full-duplex.**

1.1.3 Data Flow

Simplex:

In simplex mode, **communication is unidirectional**, allowing **only one device to transmit while the other receives**. **Keyboards and traditional monitors** exemplify this, with keyboards providing input and monitors accepting output. This mode uses the entire channel capacity to send data in one direction..



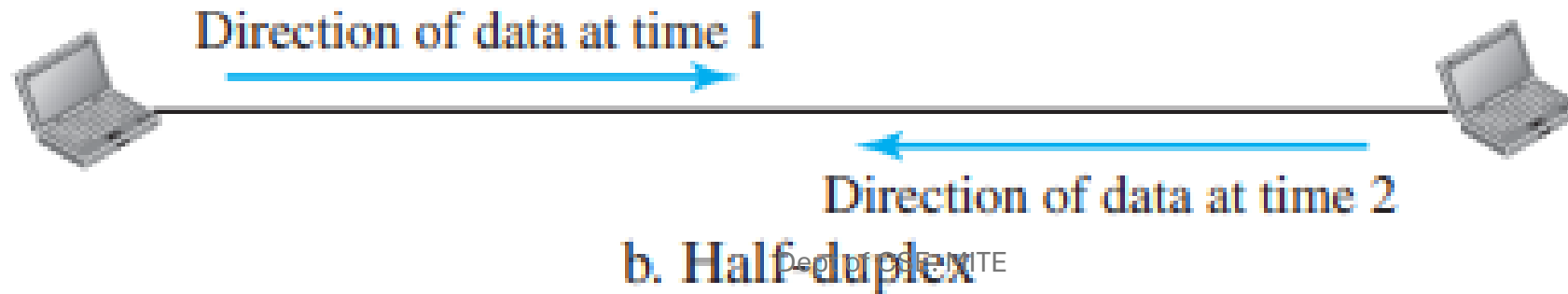
1.1.3 Data Flow

Half-Duplex:

In half-duplex mode, **each station can transmit and receive, but not simultaneously.** When **one device sends, the other can only receive**, and vice versa.

This mode takes over the **entire channel capacity for whichever device is transmitting.**

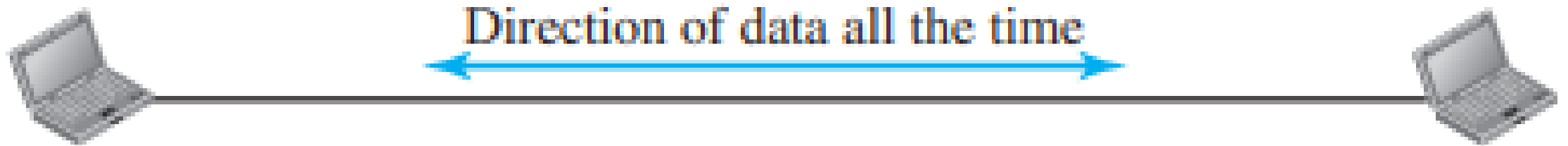
Walkie-talkies and Legacy Ethernet(bus topology) are half-duplex systems that utilize the channel's full capacity for each direction when communication in both directions simultaneously is unnecessary.



1.1.3 Data Flow

Full-Duplex:

In full-duplex mode, **both stations can transmit and receive simultaneously**, resembling a **two-way street with traffic flowing in both directions**. Signals in one direction share the link's capacity with signals going the other way. A common example is the **telephone network**, where **both parties can talk and listen at the same time**, requiring **constant communication in both directions** while dividing the channel's capacity.



1.2 Networks

A **network** is a set of devices (often referred to as nodes) **connected by communication links**. A node can be a computer, printer, or any other device capable of sending and/or receiving data generated by other nodes on the network.

Network Criteria A network must be able to meet a certain number of criteria. The most important of these are **performance, reliability, and security**.

1.2 Networks

Performance can be measured by transit time and response time, with transit time being the duration for a message to travel between devices and response time being the time between an inquiry and a response.

Network performance depends on factors like the number of users, transmission medium, hardware capabilities, and software efficiency, and is evaluated by throughput and delay, which are often.

1.2 Networks

- Two key performance indicators are throughput and delay.
 - **Throughput** refers to the **amount of data successfully transferred from one place to another** in a given period of time, usually measured in bits per second (bps).
 - **Delay** (also called latency) is the **time it takes for data to travel from the source to the destination**. It includes all the delays caused by the processing, transmission, and queuing of data across the network.
- Increasing throughput can sometimes cause delays due to network congestion.

1.2 Networks

Network reliability is the ability of a network to perform its required functions consistently over time without failures or interruptions. A reliable network ensures that data is transmitted accurately and promptly, and that the network remains operational even in the face of hardware or software issues.

Security: Network security issues include protecting data from unauthorized access, protecting data from damage and development, and implementing policies and procedures for recovery from breaches and data losses.

1.1.2 Physical Structures

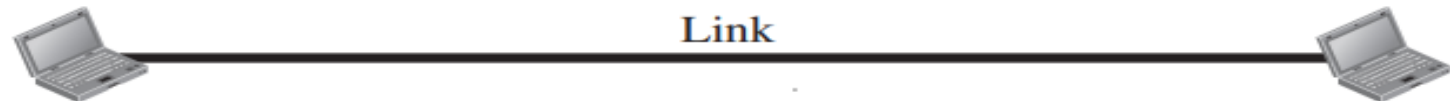
Types of Connections

A network **connects two or more devices through links**, which are **pathways for data transfer**. For simplicity, a link can be seen as a line between two points. Communication occurs when two devices are connected to the same link.

There are two connection types: **point-to-point and multipoint**

Types of Connections

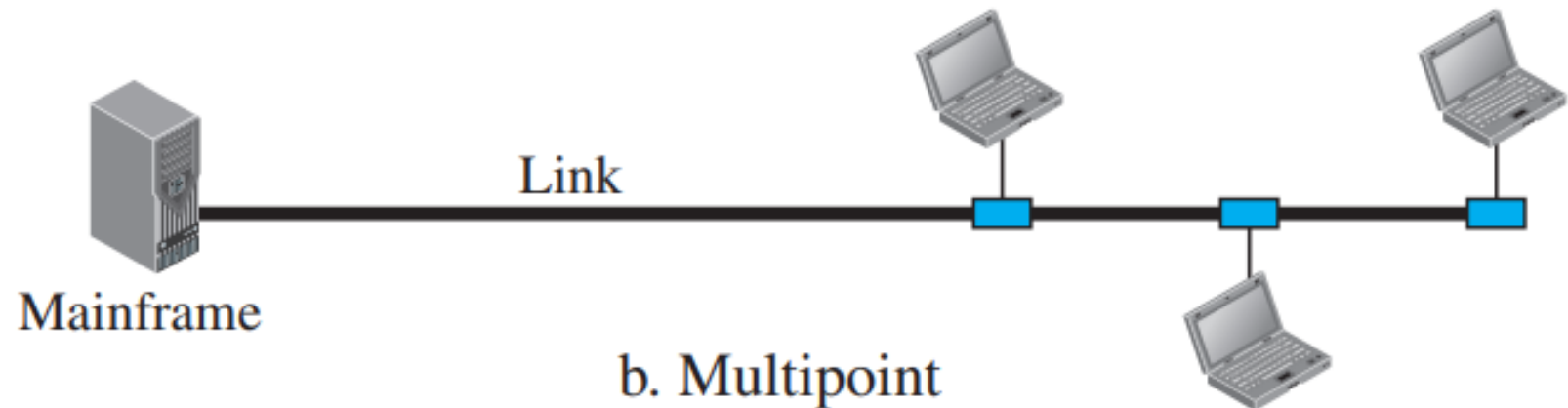
- **Point-to-Point Connection:** In this type of connection, a single link is used to connect two specific devices. The entire link is dedicated to those two devices, meaning no other devices can use that path.
- **An example** would be a direct connection between a computer and a printer via a USB cable.
- **Another example** is a remote control for a television. When you press a button on the remote, an infrared signal is sent directly from the remote to the TV, forming a point-to-point connection between the two devices.



a. Point-to-point

Types of Connections

- **Multipoint Connection:** In a multipoint connection, **more than two devices share the same link**. This means multiple devices can communicate over the same connection. A good **example** would be a Wi-Fi network at home. Several devices, such as phones, laptops, and tablets, can all connect to the same Wi-Fi router, sharing the same communication link to access the internet.



b. Multipoint

Physical Topology

- Physical topology describes **how a network is physically arranged**. Devices connect through links, and multiple links form a topology. It visually represents the connections between nodes.
- The four main topologies are **mesh, star, bus, and ring**.

Physical Topology

Mesh Topology

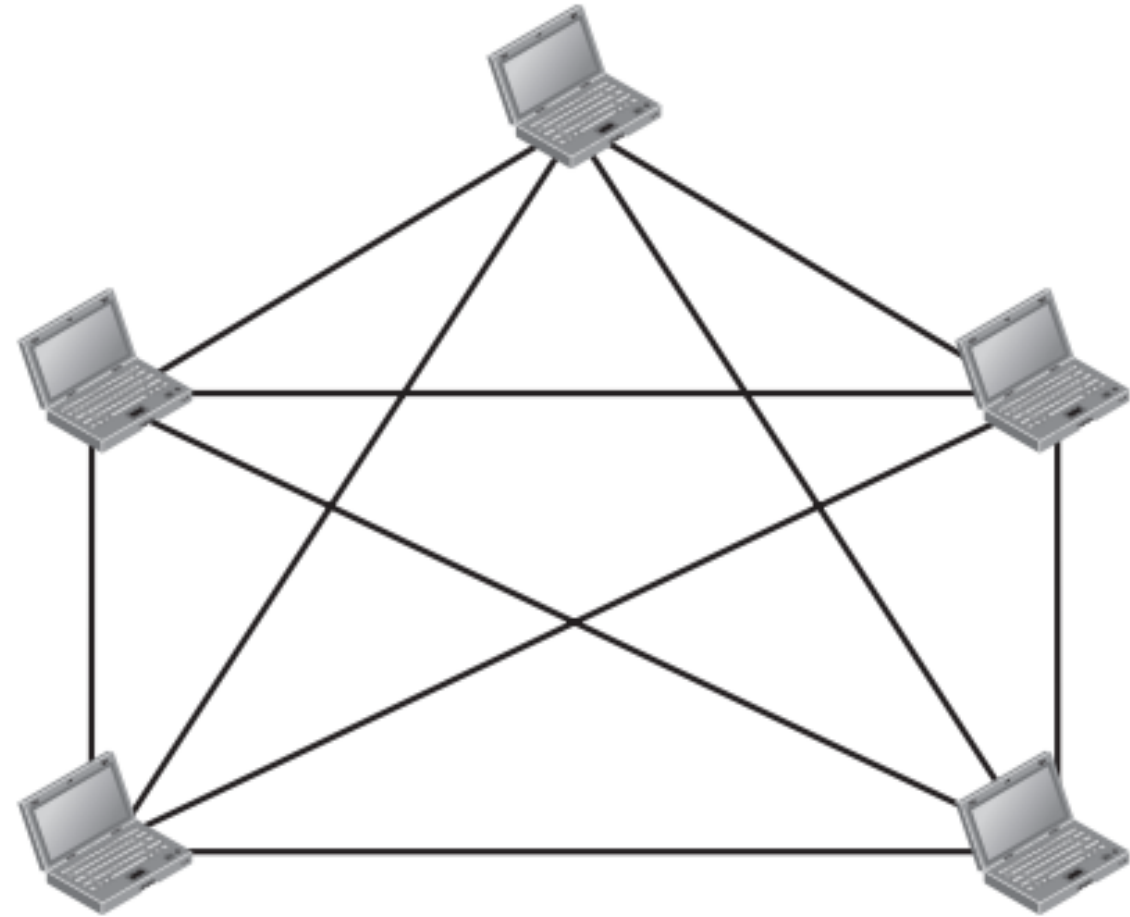
- In a mesh topology, each device has a direct, **point-to-point link to every other device, with each link carrying traffic only between those two devices.**
- For a fully connected network with n nodes, each node connects to $n-1$ others, requiring $n(n-1)/2$ duplex-mode links.
- Each device needs $n-1$ input/output ports to connect to the other nodes

Physical Topology

Mesh Topology

$n = 5$
10 links.

Example of a mesh topology is the connection of **telephone regional offices** in which each regional office needs to be connected to every other regional office.



Physical Topology

Mesh Topology

Advantages of Mesh Topology:

- 1.Dedicated links:** Each connection carries its own data, reducing traffic issues.
- 2.Robustness:** A single link failure does not affect the whole system.
- 3.Security:** Messages travel along dedicated lines, ensuring only the intended recipient sees them.
- 4.Fault detection:** It is easy to identify and isolate faults in the network.

Disadvantages of Mesh Topology:

- 1.High cabling requirements:** Every device must connect to all others, leading to complex installations.
- 2.Space issues:** The volume of wiring may exceed available space.
- 3.Costly hardware:** I/O ports and cabling for each connection can be expensive.

Physical Topology

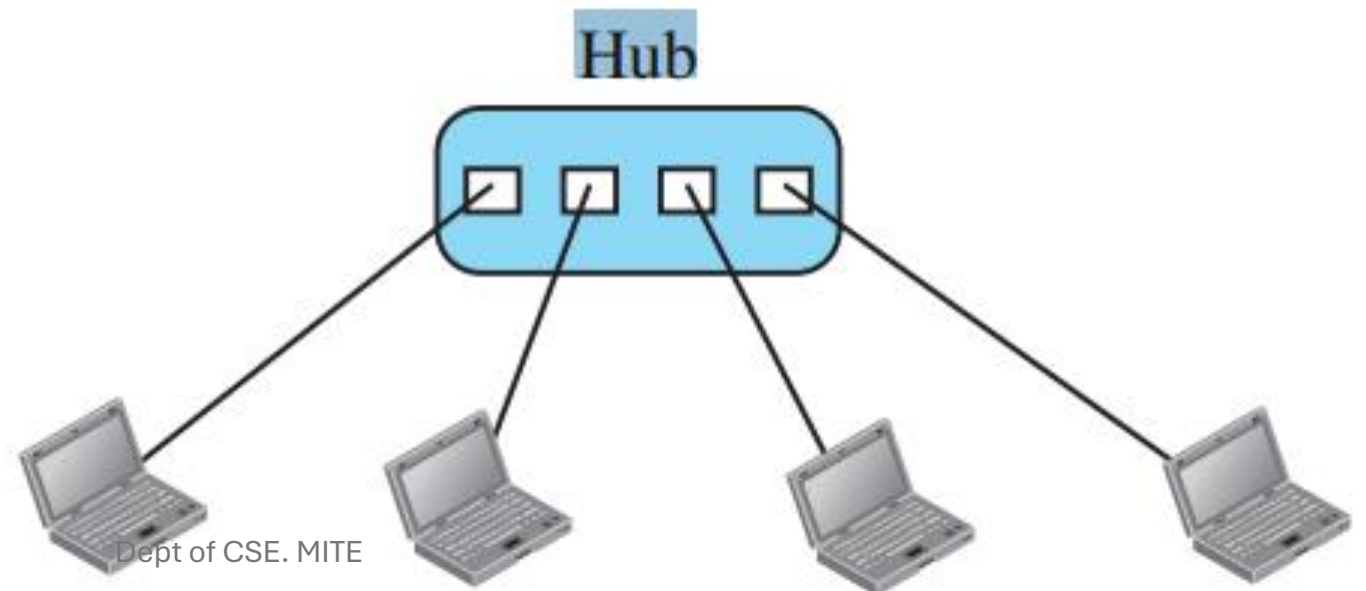
Star Topology

- In a **star topology**, all devices are connected directly to a central hub or controller, which manages the communication between devices.
- Devices do not communicate with each other directly; instead, they send data to the hub, which relays it to the intended recipient.
- This setup centralizes the network's traffic management in one hub, making it easier to control and manage.

Physical Topology

Star Topology

- If one device wants to send data to another, it sends the data to the controller, which then relays the data to the other connected device.
- A star topology is less expensive than a mesh topology.
- In a star, each device needs only one link and one I/O port to connect it to any number of others.



Physical Topology

Star Topology

Advantages of Star Topology:

- **Easy to set up and manage** since each device connects to a central hub.
- **Reliability:** If one connection fails, it doesn't affect the rest of the network.
- **Error:** Problems are easy to detect and fix because each device has its own connection.
- **Cost:** Requires less cabling than a mesh network.

Disadvantages of Star Topology:

- If the **hub fails**, the entire network stops working.
- Needs more cables than some other types of networks like bus or ring.
- The **network's performance depends on the hub**; if it's **overloaded**, the whole network can slow down.

Physical Topology

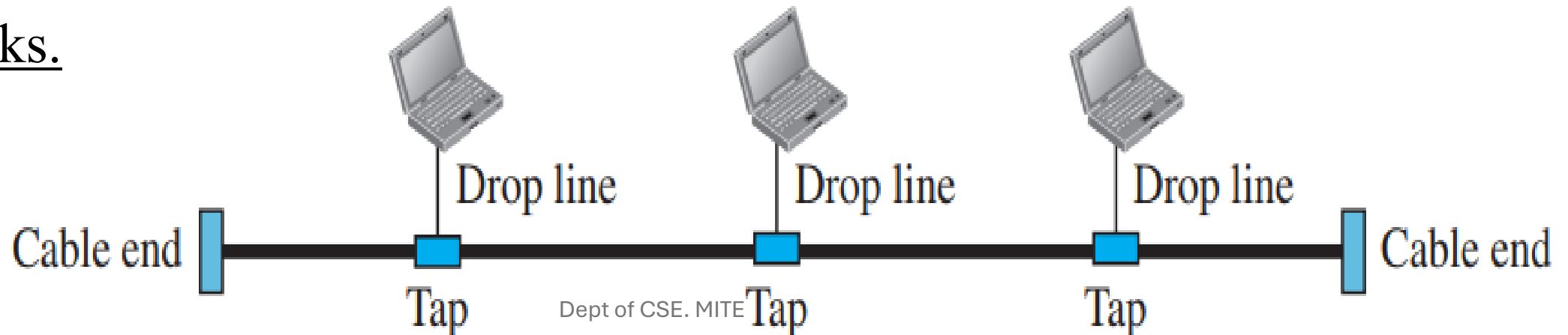
Bus Topology

- Bus topology is **multipoint**, unlike point-to-point connections.
- A single long cable, or **backbone**, connects all devices.
- Devices are linked to the main cable via **drop lines** and **taps**.
- **Drop line:** A connection from a device to the main cable.
- **Tap:** A **connector that attaches to the cable**, either by splicing or puncturing it.
- As signals travel along the backbone, they lose strength, limiting the number of taps and the distance between them.

Physical Topology

Bus Topology

- A **fault or break** in the bus cable **stops all transmission**.
- The damaged area reflects signals back in the direction of origin, creating noise in both directions.
- Bus topology was the one of the first topologies used in the design of early local area networks.



Physical Topology

Bus Topology

Advantages of Bus Topology:

- **Easy installation:** The backbone cable can be laid efficiently and connected to nodes with drop lines of varying lengths.
- **Less cabling:** Uses fewer cables compared to mesh or star topologies, reducing cost and complexity.
- **Efficient use of space:** Only the backbone cable runs through the facility, and drop lines connect to the nearest point on the backbone.

Disadvantages of Bus Topology:

Hard to Reconnect or Add Devices:

Adding new devices or fixing problems can be tricky.

Signal Degradation:

Connecting too many devices or having them too far apart can cause the signal quality to drop.

Fault Issues:

If the main cable gets damaged, it can stop all communication,

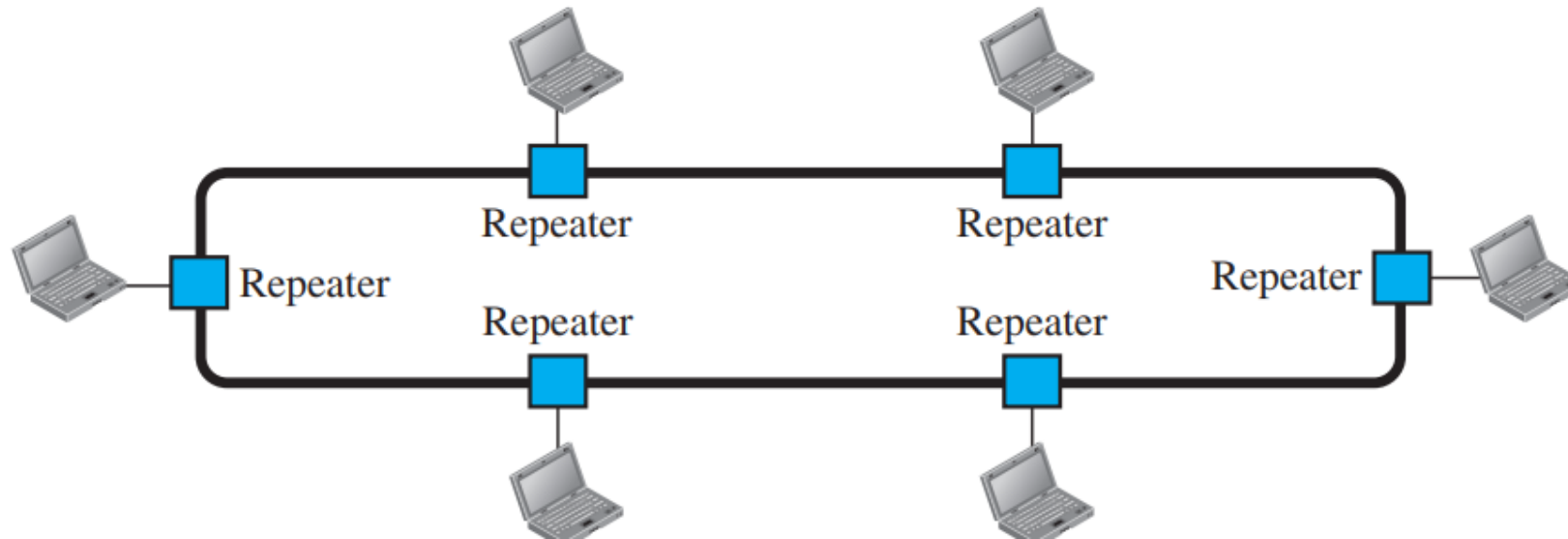
Physical Topology

Ring Topology

- In a **ring topology**, each device has a **dedicated point-to-point connection** with **only the two devices on either side of it**.
- A signal is passed along the ring in one direction, from device to device, until it reaches its destination. Each device in the ring incorporates a repeater.
- When a device receives a signal intended for another device, its repeater regenerates the bits and passes them along.

Physical Topology

Ring Topology



Token Ring: token moves around the ring sequentially and passes through each node one at a time, every node gets an equal chance to capture the token and transmit its data.

Physical Topology

Ring Topology

Advantages of Ring Topology:

- Simple installation
- Fewer Cables are needed.
- Minimizes the possibility of data collision i.e Unidirectional.
- An easy problem to solve.
- The access time is the same for every node
- Easy Setup and Fault Detection

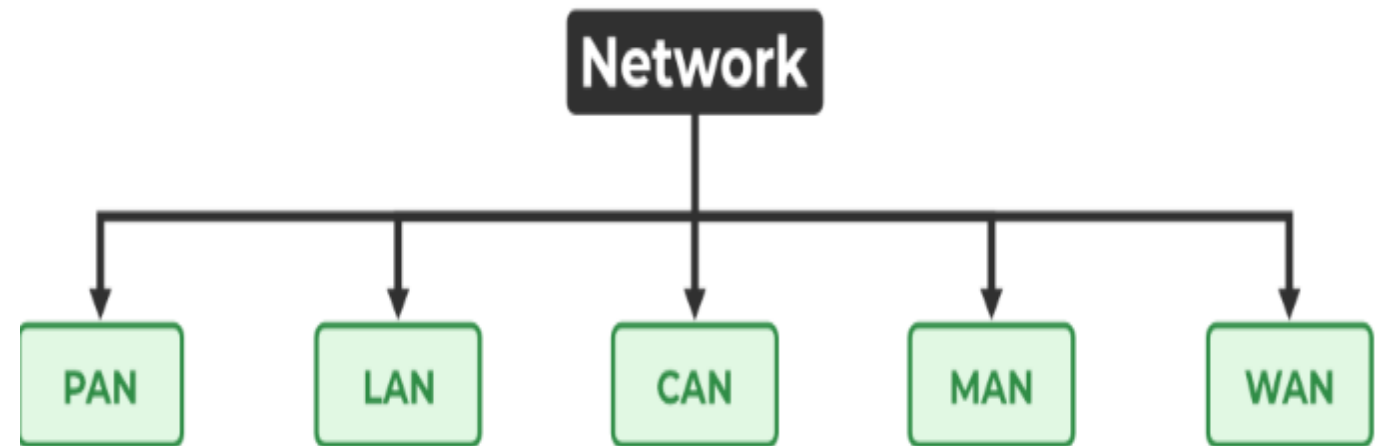
Disadvantages of Ring Topology:

- One broken connection can stop everything.
- More devices increase the delay.
- Adding/Changing the network can be tricky.
- Setting up is more expensive.
- Bigger networks can be complex to troubleshoot.

1.3 Network Types

There are mainly five types of Computer Networks

- Personal Area Network (PAN)
- Local Area Network (LAN)
- Campus Area Network (CAN)
- Metropolitan Area Network (MAN)
- Wide Area Network (WAN)



1.3 Network Types

- **Personal Area Network (PAN):** A network that connects devices within a very short range, typically within a few meters, like Bluetooth or USB connections.
- **Local Area Network (LAN):** A network that connects devices within a small geographical area, such as a home, office, or building.
- **Campus Area Network (CAN):** A network that connects multiple LANs across multiple buildings within a limited geographical area, such as a university or corporate campus.
- **Metropolitan Area Network (MAN):** A network that spans a city or metropolitan area, connecting multiple LANs or CANs across a wider region.
- **Wide Area Network (WAN):** A network that spans a large geographical area, such as a country or continent, often connecting multiple LANs and MANs across vast distances.

1.3 Network Types

1.3.1 Local Area Network (LAN)

- A Local Area Network (LAN) is a network that connects devices like computers and printers within a small area, such as an office, building, or campus.
- **Private and Local:** It's usually privately owned and covers a limited area, like a single office or an entire company.

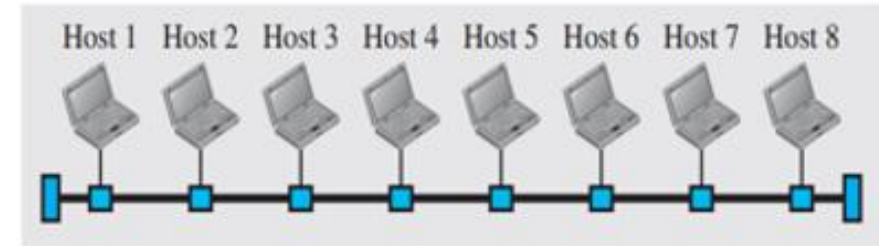
1.3 Network Types

1.3.1 Local Area Network (LAN)

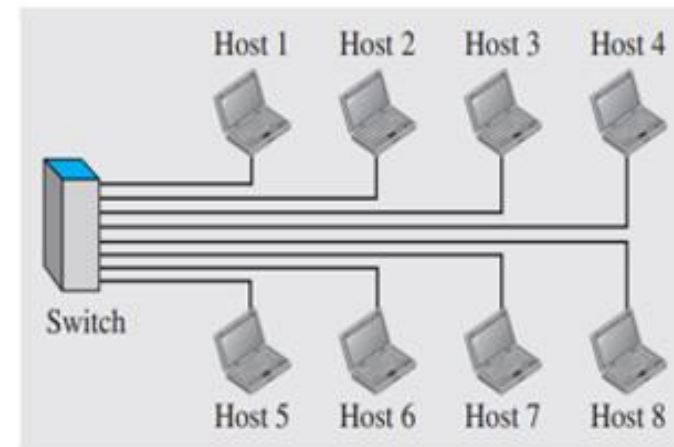
Unique Addresses: Each device in a LAN has a **unique address** so that data can be sent to the correct device.

Old vs. New: In the past, all devices were **connected by one cable**, and data was sent to everyone on the network. Now, most LANs use a smart switch that **only sends data to the intended device**, reducing unnecessary traffic.

Efficiency: The smart switch helps manage data flow and allows multiple devices to communicate at the same time without interference.

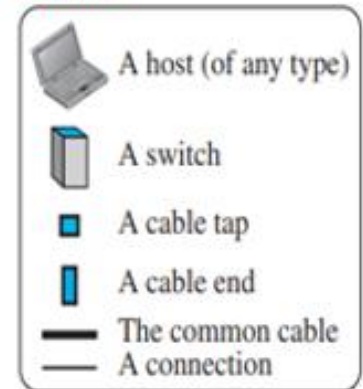


a. LAN with a common cable (past)



b. LAN with a switch (today)

Legend



1.3 Network Types

1.3.1 Local Area Network (LAN)

Advantages of a LAN

Privacy: LAN is a private network, thus no outside regulatory body controls it, giving it a privacy.

High Speed: LAN offers a much higher speed(around 100 mbps) and data transfer rate comparatively to WAN.

Supports different transmission mediums: LAN support a variety of communications transmission medium such as an Ethernet cable (coaxial cable, and twisted pair), fiber and wireless transmission.

Inexpensive and Simple: A LAN usually has low cost, installation, expansion and maintenance and LAN installation is relatively easy to use, good scalability.

1.3 Network Types

1.3.1 Local Area Network (LAN)

Disadvantages of a LAN

Short Range: LANs only work over short distances, like within a single building or campus. They can't connect devices far apart.

Security Risks: If not secured properly, unauthorized people might access sensitive information on the network.

Maintenance Costs: Keeping the LAN running smoothly can be expensive, with costs for equipment and tech support(continuous investment in both hardware and expertise.)

Network Traffic: Too many devices using the network at once can slow everything down.

1.3 Network Types

1.3.2 Wide Area Network (WAN)

A Wide Area Network (WAN) is a type of network that **connects devices over much larger distances than a Local Area Network (LAN).**

WANs cover large geographic areas, such as cities, countries, or even continents.

WANs are often used to connect multiple LANs together.

1.3 Network Types

1.3.2 Wide Area Network (WAN) and Local Area Network(LAN)

Area covered:

LAN: Covers a small area like an office or building.

WAN: Covers a much larger area, such as a city, country, or even globally.

Connections:

LAN: Connects devices directly, like computers and printers.

WAN: Connects networking equipment, such as switches, routers, or modems.

Ownership:

LAN: Usually owned by the organization using it.

WAN: Typically run by communication companies and leased to organizations.

1.3 Network Types

Internetwork

An **internetwork** (or **internet**) is formed when multiple networks are interconnected.

1. Network Connection: It connects two or more LANs or WANs to allow communication between them.

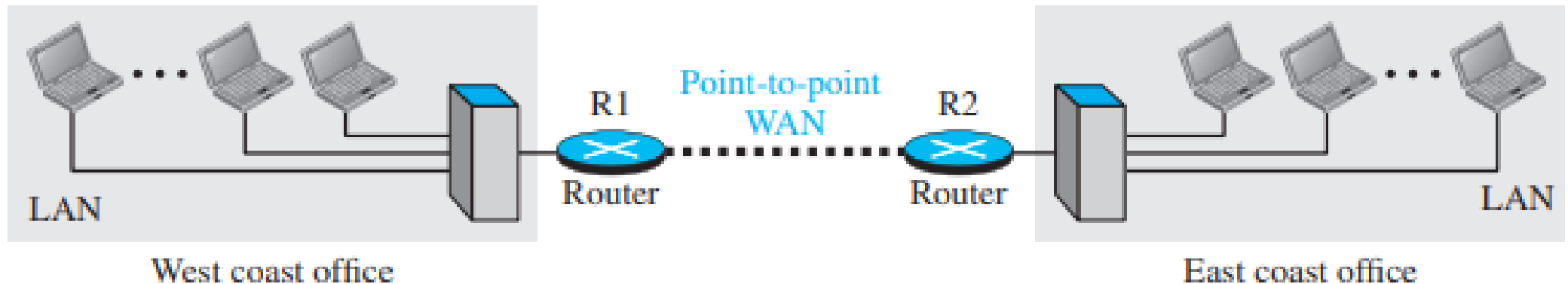
2. Private Internet: This type of network can be private, as seen in organizations connecting offices using a dedicated WAN.

1.3 Network Types

Internetwork

3. Improved Communication: Connecting networks enables communication between geographically separated offices or departments.

4. Service Providers: Companies often lease services (like WANs) from providers (e.g., telecom companies) to establish these connections.



1.3.3 Switching

A **switched network** is a type of communication network where devices are connected using switches that forward data between them.

In this network, switches direct data only to the device or network where it is intended, rather than broadcasting to all devices. This improves efficiency, security, and performance.

- **Key Functions of a Switch:**

- 1.**Data Forwarding:** Directs data packets to the appropriate device based on its **MAC address**(Media Access Control - 00:1A:2B:3C:4D:5E).

- 2.**Collision Reduction:** Minimizes data collisions by **creating individual data paths between devices.**

- 3.**Segmentation:** Divides a network into **smaller sections to improve performance and security.**

1.3.3 Switching

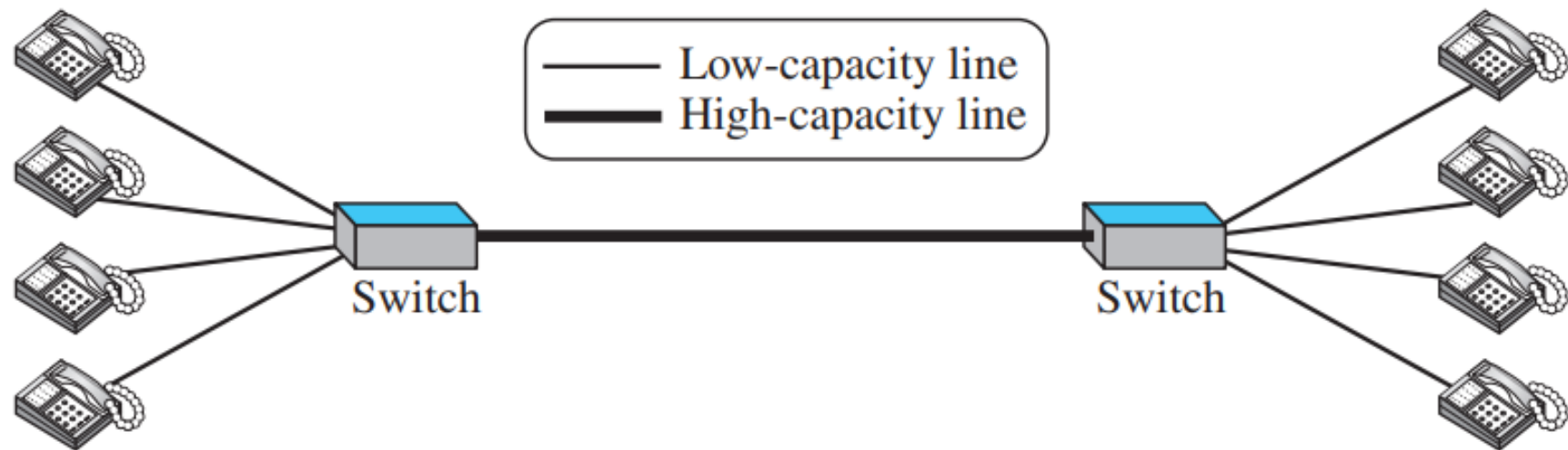
1. Circuit-switched

- In a circuit-switched network, a **dedicated connection**, called a **circuit**, is always available between the two end systems.
- End to end **connection is reserved** for duration of transmission, Once **transmission ends connection is terminated**.
- **Telephone sets are used as end systems** instead of **computers** because **circuit switching** was common in **telephone networks** historically, though modern networks often use **packet-switching**.
- **The telephones on each side are connected to a switch**, which **links a telephone on one side to a telephone on the other side**.

1.3.3 Switching

Circuit-switched

Figure 1.13 illustrates a simple switched network that connects **four telephones** to each end.



•**Circuit Switching:** A dedicated path is established before communication begins and remains open for the duration of the session.

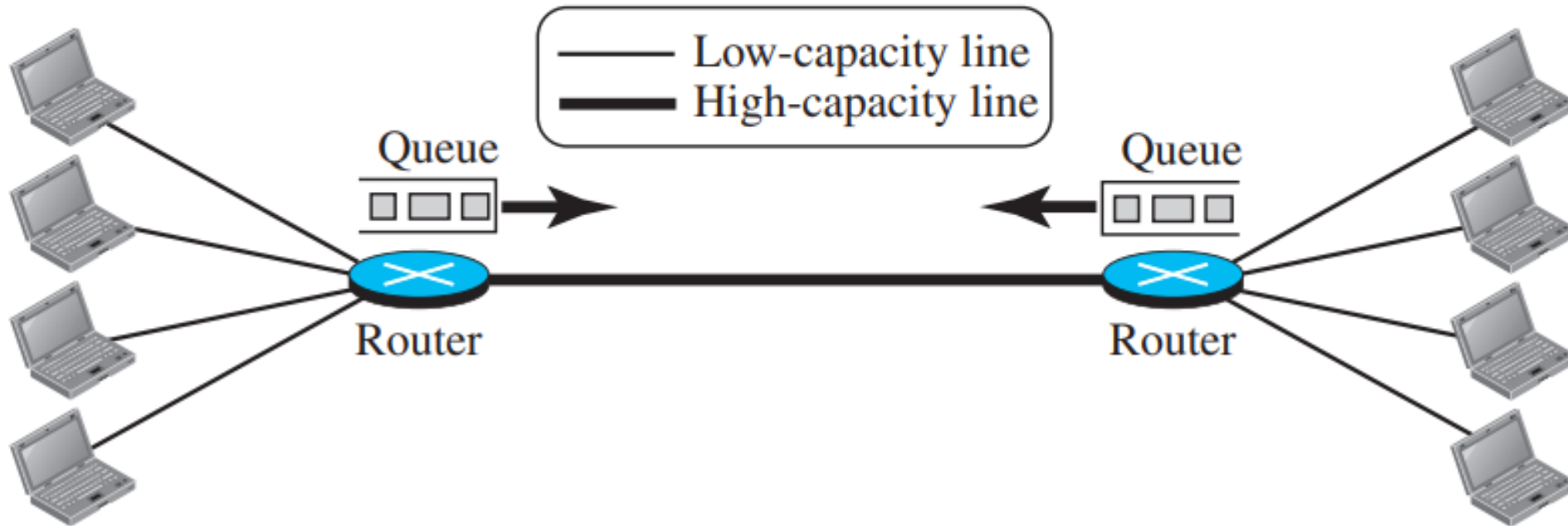
1.3.3 Switching

2. Packet-Switched Network

- In a computer network, communication between two ends is carried out in blocks of data known as packets.
- Unlike continuous communication in a telephone network, computers exchange individual data packets.
- This method allows switches to function for both storing and forwarding since packets are independent entities that can be stored and sent later.
- Each packet is assigned a **header with source/destination** address and **routed** through network.
- A **router** in a packet-switched network **is equipped with a queue that can store and forward packets.**

1.3.3 Switching

2. Packet-Switched Network

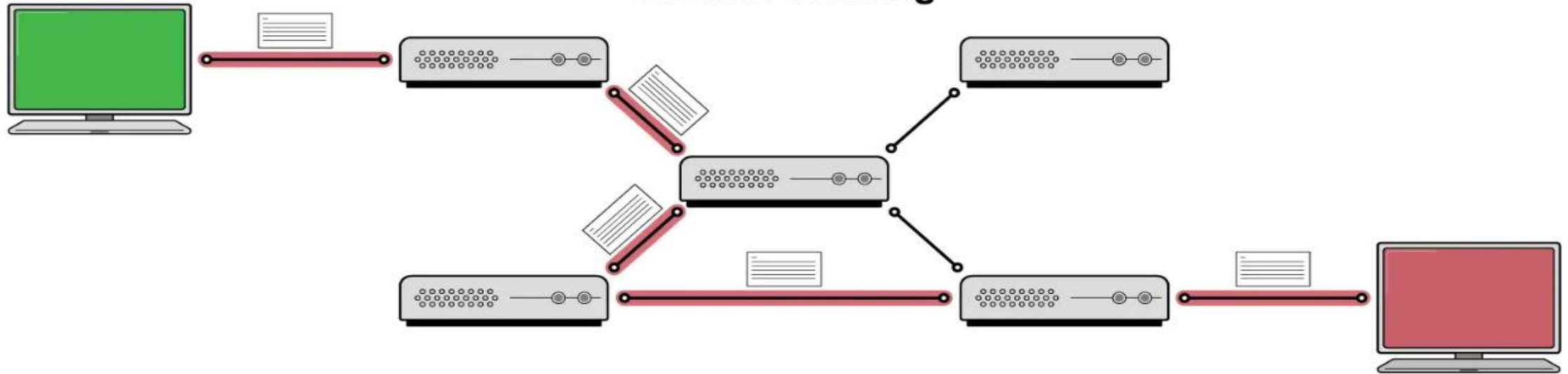


Packet Switching: No dedicated path. Data is sent in packets, and each packet is routed independently.

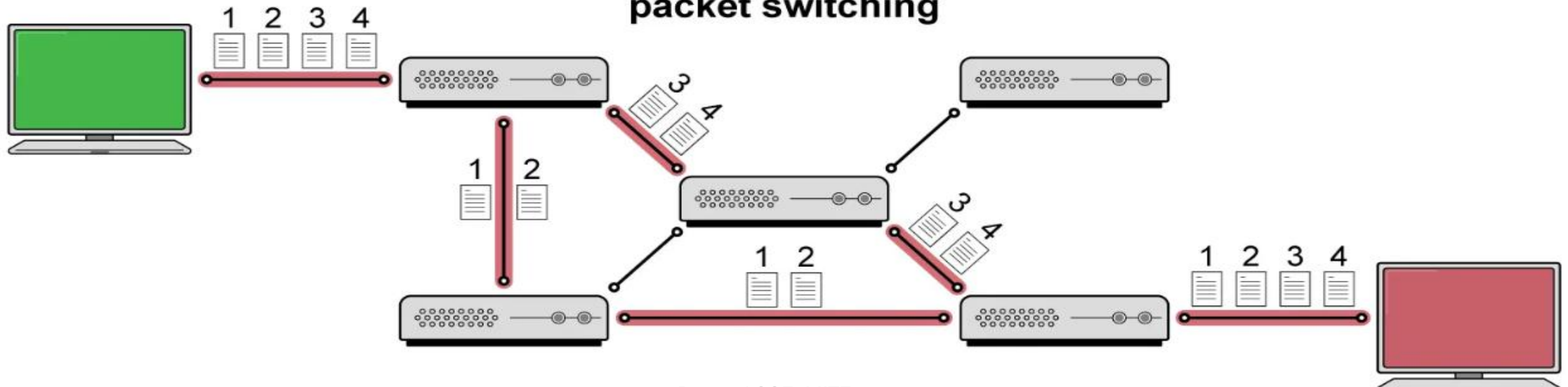
Switching networks

1.3.3 Switching

circuit switching



packet switching



1.3.4 The Internet

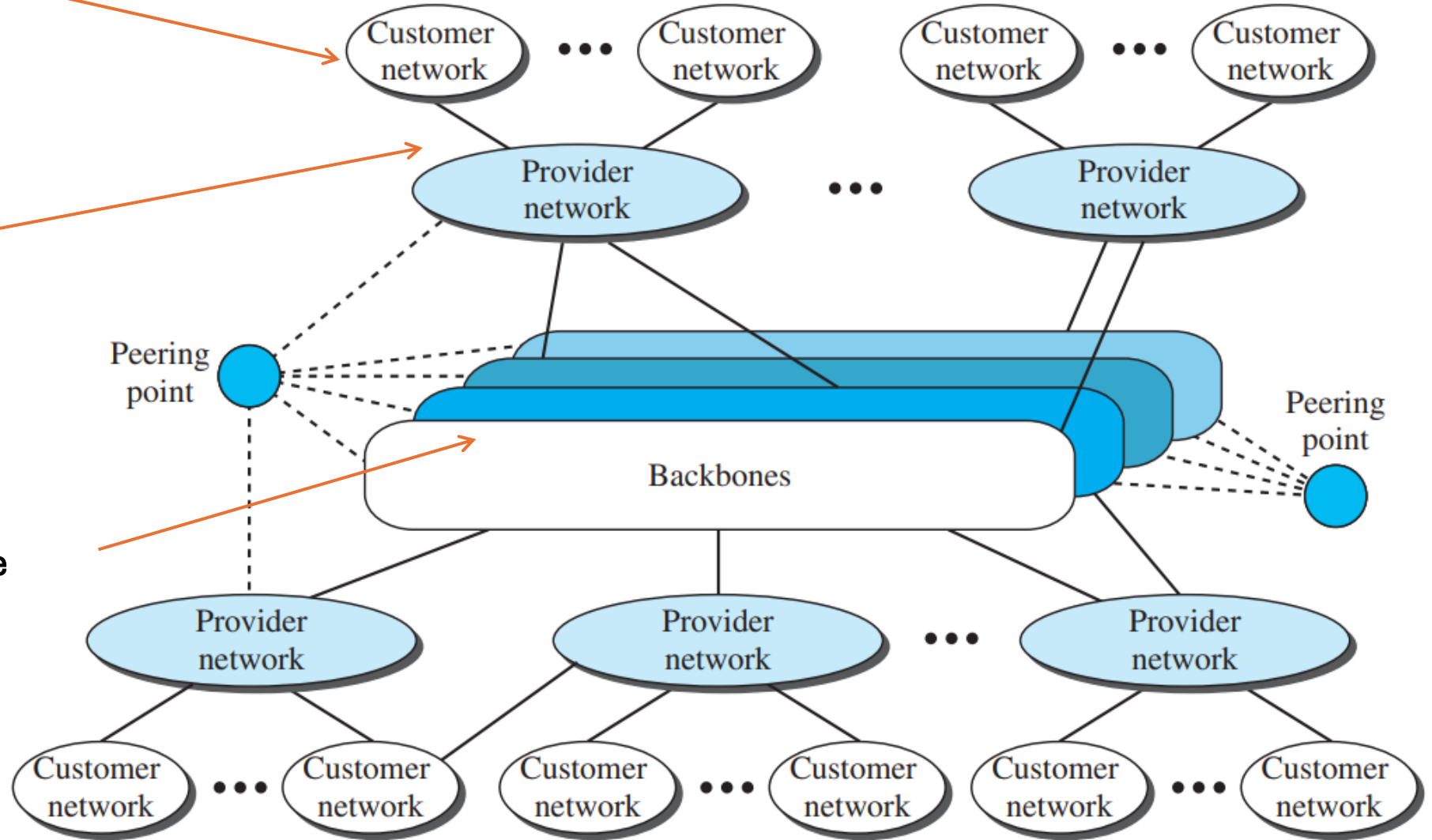
- **Definition of an internet (lowercase 'i'):** An internet refers to two or more networks that are capable of communicating with each other.
- **Definition of the Internet (uppercase 'I'):** The Internet is the most prominent internet, comprising thousands of interconnected networks.
- **Conceptual View of the Internet:** The Internet consists of backbones, provider networks, and customer networks

1.3.4 The Internet

Internet via a Wi-Fi router

Local or regional traffic (Airtel, Jio)

Global or long-distance national/international traffic (Tata communication, Zayo Group)



1.3.4 The Internet

This diagram illustrates the structure of the Internet, showing how different types of networks are interconnected to form the larger global system. Here's an explanation of the components:

1.Backbones: The core of the Internet consists of large, high-capacity networks owned by major communication companies. These backbones handle significant amounts of data and provide connectivity across vast distances.

2.Peering Points: The backbones are connected to each other through peering points, which are complex switching systems that allow data to be exchanged between different backbone networks.

1.3.4 The Internet

3. Provider Networks: These are smaller networks that connect to the backbones. They serve as intermediaries and provide services to other smaller networks or customers. Provider networks can be **national or regional ISPs** (Internet Service Providers).

4. Customer Networks: At the **edge of the Internet are customer networks**, which are local networks used by businesses, homes, or other organizations. These networks rely on the services provided by the provider networks and backbones to connect to the broader Internet.

1.3.4 The Internet

Accessing the Internet

The Internet is an internetwork that allows users to connect, but they need to be linked to an ISP through a physical connection, usually a point-to-point WAN.

1.Using Telephone Networks: Most homes and **small businesses have telephone service**, and these networks are often connected to the Internet.

a. Dial-up Service:

1. A **modem converts data to voice**, and the computer dials the ISP.
2. It is slow and can't be used for both **Internet and voice calls simultaneously**.
3. Suitable for **small residences**. PC → ISP (via phone call) → Website → ISP → PC

b. DSL Service: Digital Subscriber Line

1. Telephone companies offer faster Internet through DSL(traditional copper telephone lines).
2. It allows simultaneous use of the line for both voice and Internet.

1.3.4 The Internet

Accessing the Internet

2. Using Cable Networks:

Residents and small businesses can connect to the **Internet through upgraded cable TV networks**. This offers higher speeds, but the speed can vary depending on the number of users in the area.

3. Using Wireless Networks:

Households and small businesses can use a **mix of wired and wireless connections or access the Internet through wireless WANs**.

4. Direct Connection to the Internet:

Large organizations or corporations can act as **local ISPs by leasing high-speed WAN connections from carriers and linking directly to regional ISPs**, such as a university connecting its campuses to the Internet.

Chapter 2 Network Models

- Protocol Layering
- TCP/IP Protocol suite
- The OSI model

Network Models

A **network model** provides a framework that defines **how data is communicated across a network**. It standardizes the functions and **interactions between network components** (hardware and software), ensuring consistent data transmission and interoperability between different systems.

1. OSI Model (Open Systems Interconnection Model)

2. TCP/IP Model (Transmission Control Protocol/Internet Protocol Model)

Network Models

2.1 Protocol Layering:

- **A protocol defines the rules for communication between sender, receiver, and intermediaries.**
- **Complex communication tasks are divided into smaller tasks across multiple layers.**
- **Each layer has its own protocol, simplifying the overall communication process.**

Scenarios: Two scenarios to better understand the need for protocol layering

Single Layer Communication

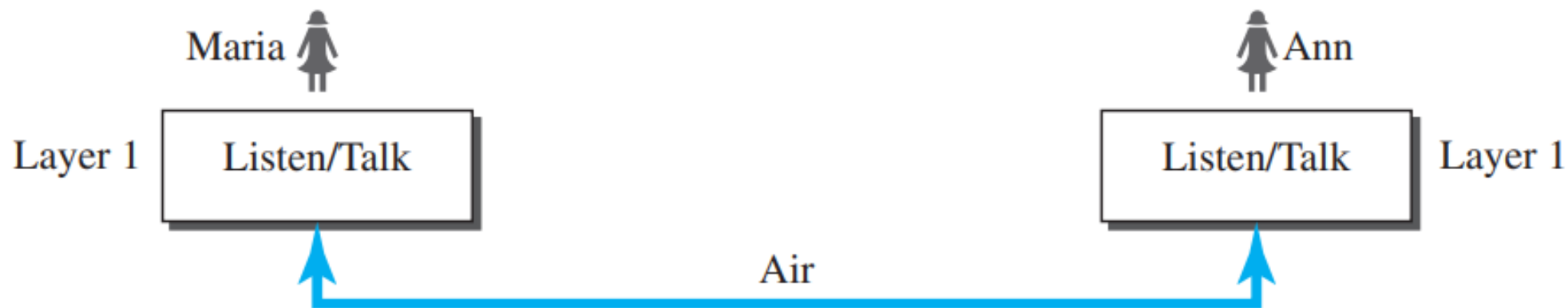
Multi-layer Communication

Network Models

2.1 Protocol Layering:

First Scenario (Single Layer Communication): Face-to-face communication between Maria and Ann.

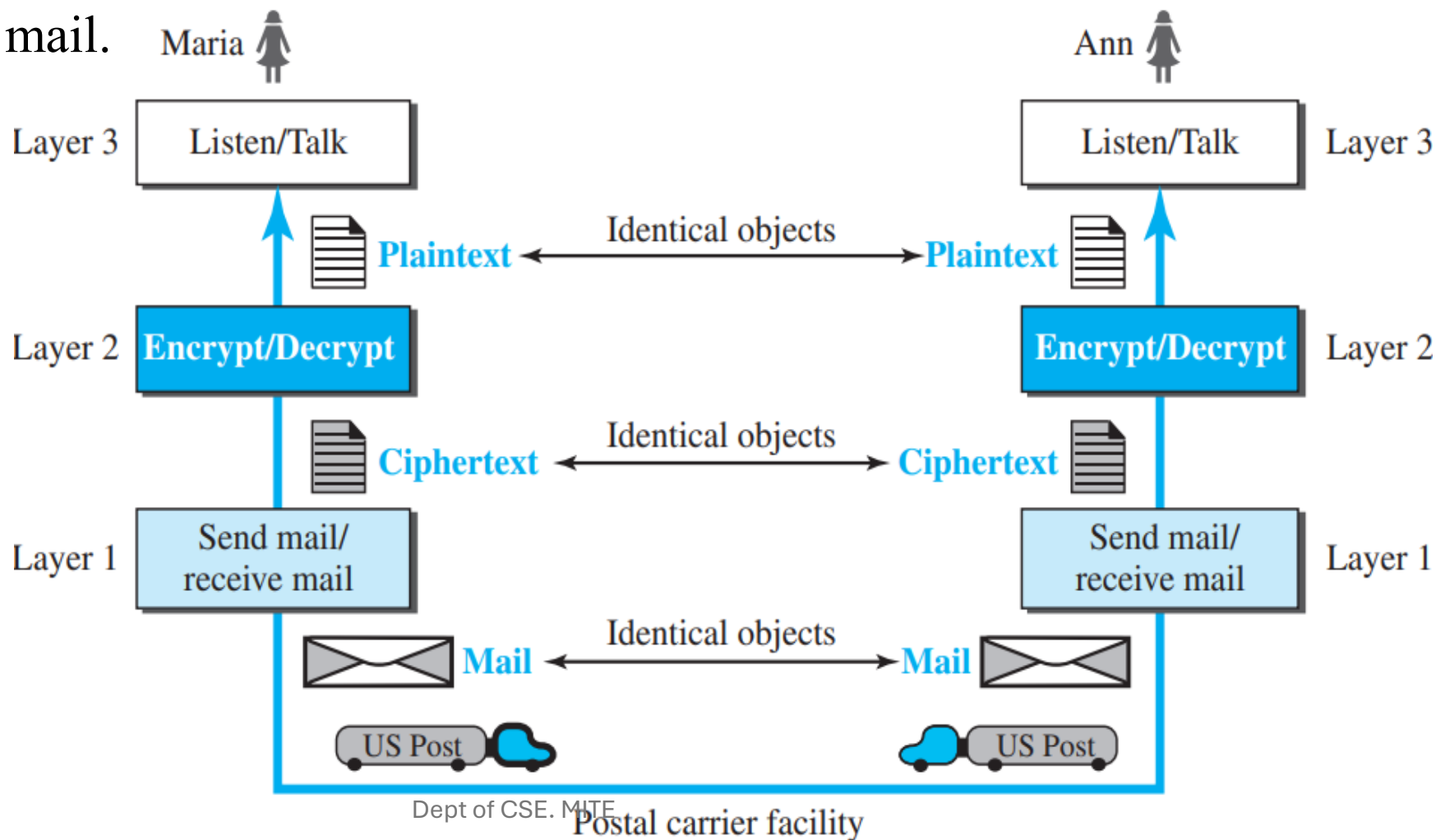
- Communication rules:
 - Greet each other.
 - Use appropriate language and timing.
 - Engage in a two-way dialog.
- Even simple communication follows specific rules (protocols).



Network Models

2.1 Protocol Layering:

Second Scenario (Multi-layer Communication): Long-distance communication between Maria and Ann using mail.



Network Models

2.1 Protocol Layering:

Second Scenario (Multi-layer Communication): Long-distance communication between Maria and Ann using mail.

Message Creation (Layer 3 - Communication Layer):

- Maria creates a **plaintext** message by speaking to a machine that simulates Ann's presence.
- The message is passed to Layer 2 for encryption.

Encryption/Decryption (Layer 2 - Security Layer):

- **Maria's Side:** The Layer 2 machine encrypts the plaintext, turning it into **ciphertext** for security.
- **Ann's Side:** The ciphertext is decrypted back into plaintext by Ann's Layer 2 machine.

Sending/Receiving Mail (Layer 1 - Physical Layer):

- **Maria's Side:** The Layer 1 machine puts the ciphertext in an envelope and mails it.
- **Ann's Side:** The Layer 1 machine collects the letter, removes the ciphertext, & passes it to Layer 2.

Network Models

2.1 Protocol Layering:

Two Principles of Protocol Layering:

1. First Principle (Bidirectional Communication):

Each layer must handle two tasks: **one in each direction.**

- Layer 3: Listen and talk.
- Layer 2: Encrypt and decrypt.
- Layer 1: Send and receive mail.

2. Second Principle (Identical Objects):

Objects at the same layer on both sides must be identical.

- Layer 3: Plaintext on both sides.
- Layer 2: Ciphertext on both sides.
- Layer 1: Mail on both sides.

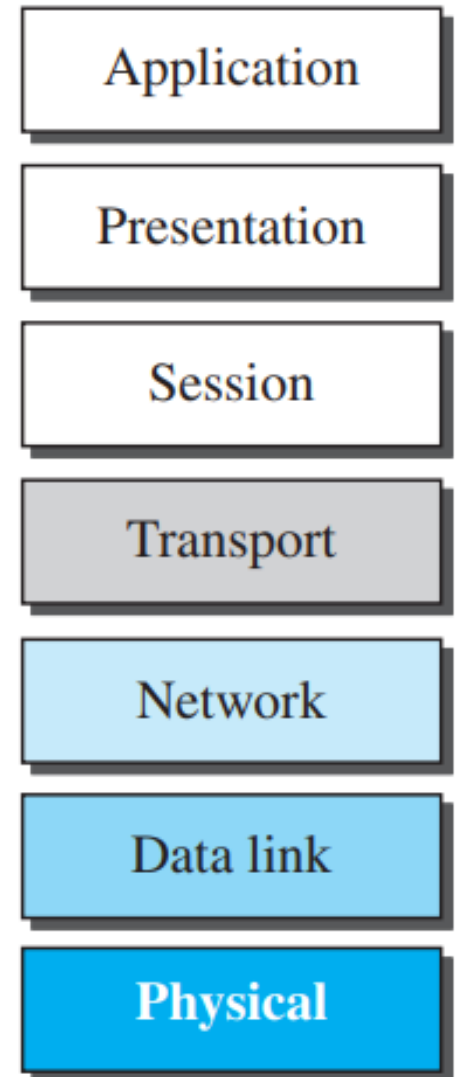
Network Models

2.3 The OSI Model

OSI Model (Open Systems Interconnection Model)

The **OSI model** is a conceptual framework that standardizes the functions of a communication system into **seven distinct layers**.

Each layer is responsible for specific tasks in data communication. This model was developed by the International Organization for Standardization (ISO).



OSI Model

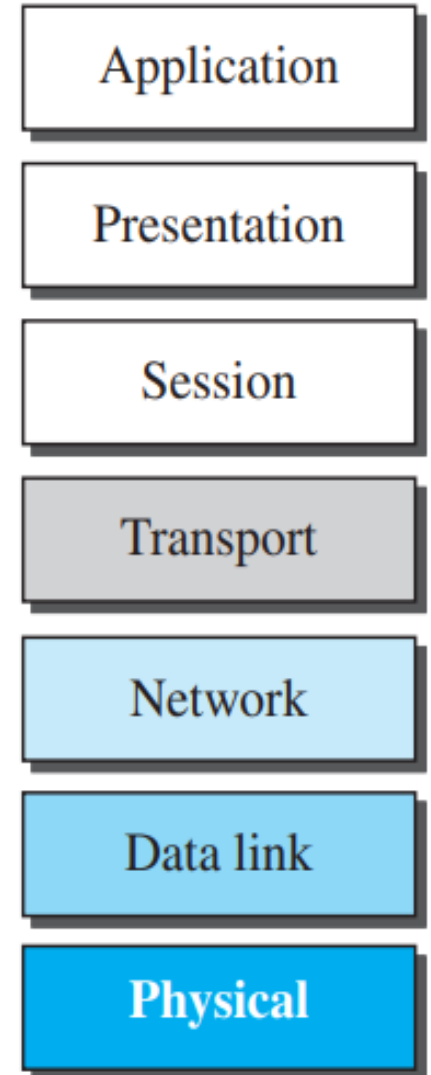
Network Models

OSI Model (Open Systems Interconnection Model)

1. Physical Layer (Layer 1): Deals with the physical connection of devices, including cables, signals, and data rates.

2. Data Link Layer (Layer 2): Ensures reliable data transfer across the physical network by detecting and possibly correcting errors (**Frames**-adds MAC header & trailer).

3. Network Layer (Layer 3): Responsible for logical addressing, routing, and forwarding of packets (**Packets**- IP address + protocol info)



OSI Model

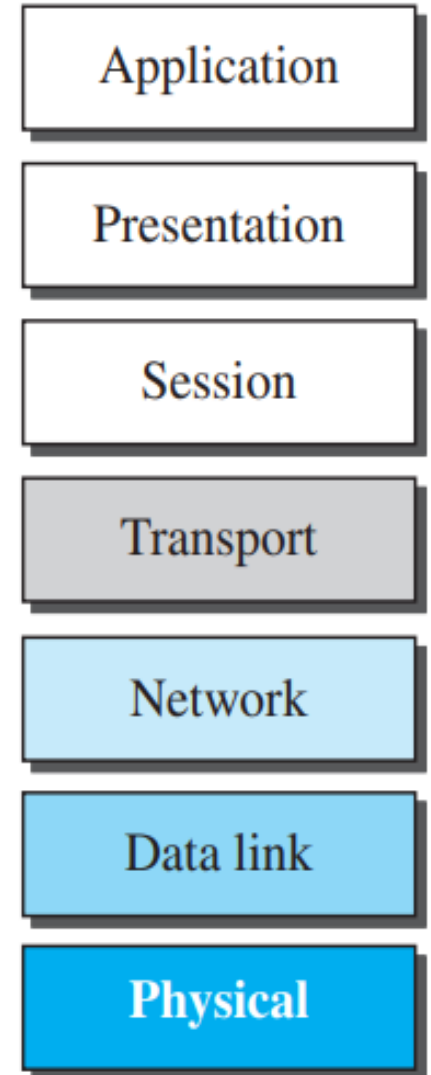
Network Models

OSI Model (Open Systems Interconnection Model)

4. Transport Layer (Layer 4): Ensures end-to-end communication, error recovery, and data flow control(**Segments**- Port num, Seq num, Ack num).

Examples: TCP, UDP.

5. Session Layer (Layer 5): Manages and controls the connections (sessions) between applications on different devices.



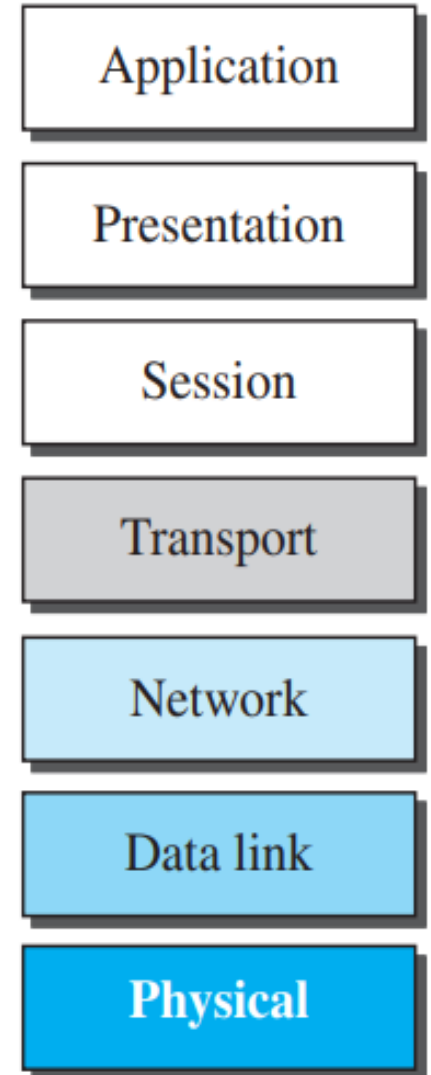
OSI Model

Network Models

OSI Model (Open Systems Interconnection Model)

6. Presentation Layer (Layer 6): Transforms data into a format understandable by the application layer. Handles encryption, decryption, and data compression.

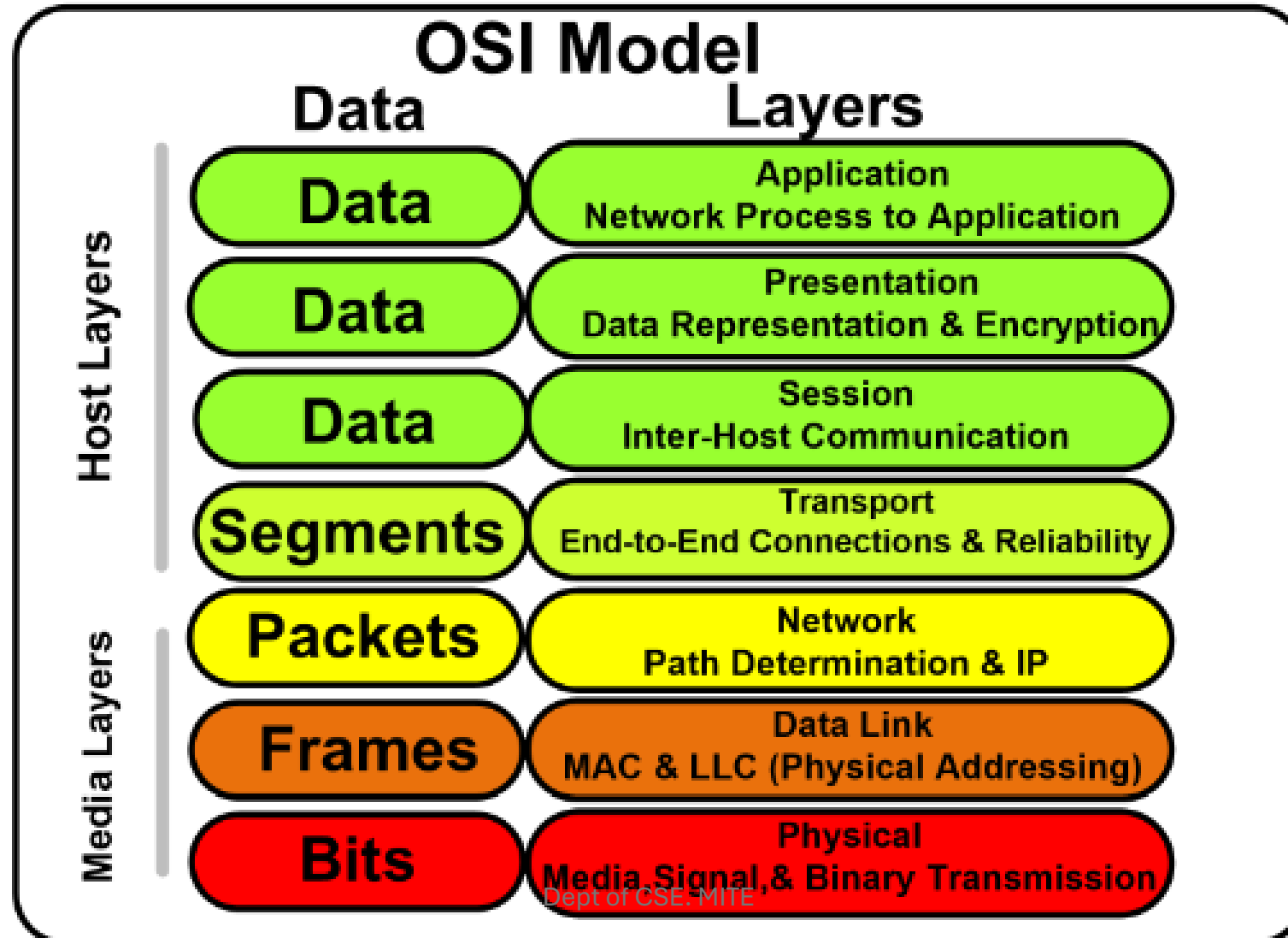
7. Application Layer (Layer 7): Interfaces directly with end-user applications, facilitating things like file transfer, email, and web browsing. Examples: HTTP, FTP, SMTP



OSI Model

Network Models

OSI Model (Open Systems Interconnection Model)



Network Models

OSI Model (Open Systems Interconnection Model)

Bit: A single binary digit, 0 or 1. Example: 1

Byte: 8 consecutive bits. Example: 10101010 (represents the character "A" in ASCII)

Frame: A logical grouping of bytes with header and trailer.

Example: Ethernet frame - Destination MAC: 00:11:22:33:44:55, Source MAC: 66:77:88:99:00:11, Data: 10101010

Packet: A logical grouping of frames with header and data.

Example: IP packet - Source IP: 192.168.1.100, Destination IP: 8.8.8.8, Data: 10101010

Segment: A logical grouping of packets with header and data.

Example: TCP segment - Source Port: 8080, Destination Port: 80, Sequence Number: 12345, Data: 10101010

Network Models

OSI Model (Open Systems Interconnection Model)- Example

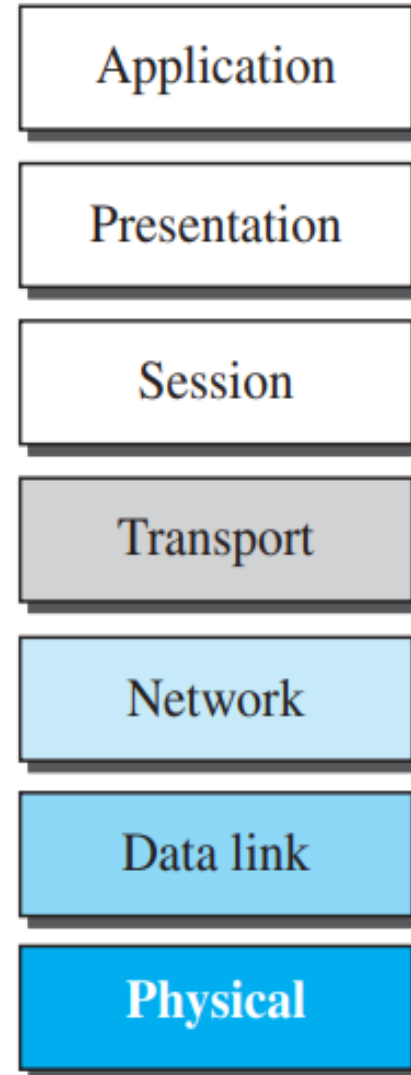
Maria sends an email to Ann using the OSI model:

Layer 7: Application Layer Maria composes an email using her email client (e.g., Gmail).

Email content: "Hello Ann, hope you're doing well."

Layer 6: Presentation Layer Email client encrypts and compresses the email. Format: MIME (Multipurpose Internet Mail Extensions)

Layer 5: Session Layer Establishes a connection between Maria's email client and Ann's email server. Session ID: 123456



OSI Model

Network Models

OSI Model (Open Systems Interconnection Model)- Example

Maria sends an email to Ann using the OSI model:

Layer 4: Transport Layer Breaks down data into segments and assigns sequence numbers.

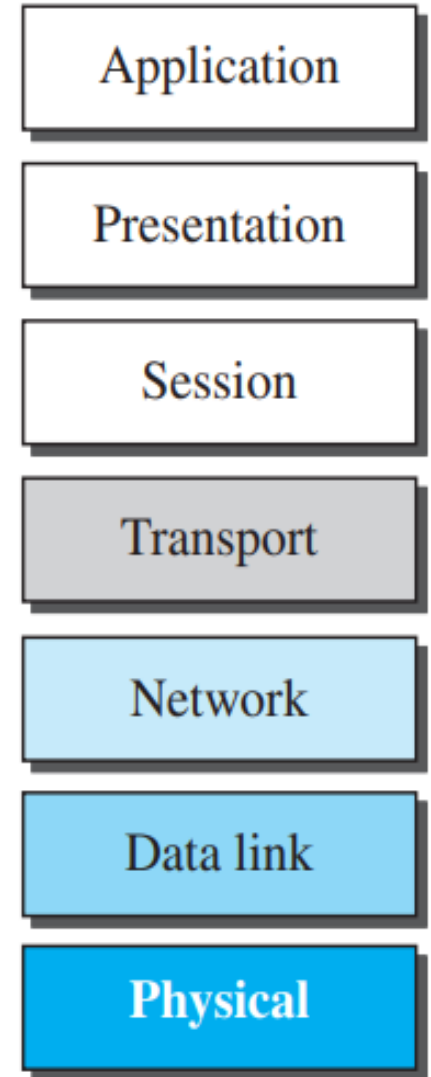
Protocol: TCP (Transmission Control Protocol) Source Port: 1024

Destination Port: 25 (SMTP)- Assigned to each process.

Layer 3: Network Layer Routes data packets between networks using IP addresses.

Source IP: 192.168.1.100 (Maria's network)

Destination IP: 203.0.113.50 (Ann's email server)



OSI Model

Network Models

OSI Model (Open Systems Interconnection Model)- Example

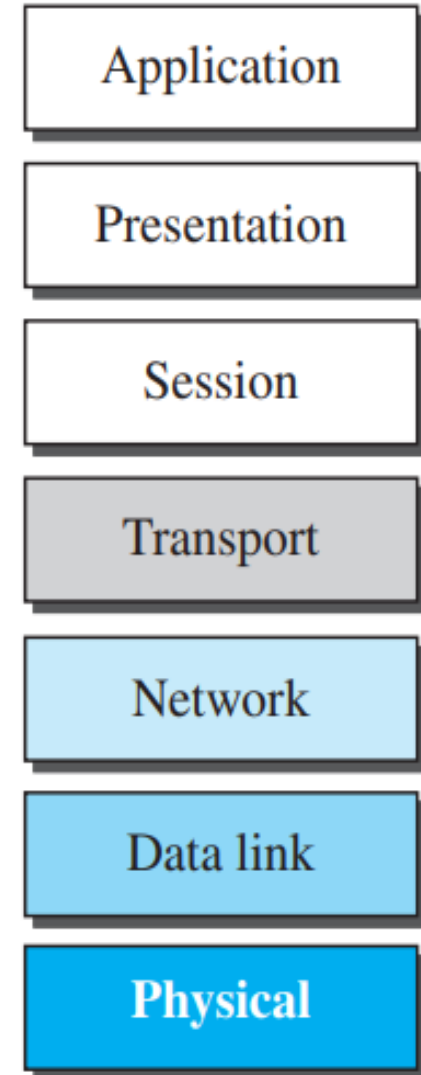
Maria sends an email to Ann using the OSI model:

Layer 2: Data Link Layer: Frames data for transmission over the local network.

Protocol: Ethernet

MAC Address: 00:11:22:33:44:55 (Maria's router)

Layer 1: Physical Layer: Transmits raw bits over the network medium (e.g., Wi-Fi, Ethernet cable).



OSI Model

Network Models

OSI Model (Open Systems Interconnection Model)- Example

Maria sends an email to Ann using the OSI model:

Receiving End

Layer 1: Physical Layer Ann's email server receives raw bits from the network medium.

Bits → H - 01001000
e - 01100101
l - 01101110
l - 01101110
o - 01101111

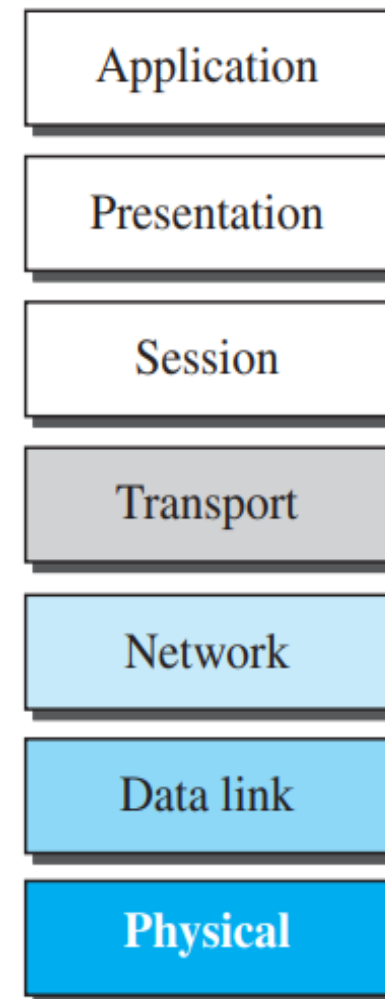
Hello → 48 65 6C 6C 6F

Bytes

Layer 2: Data Link Layer Decrypts and reassembles frames.

Frames-> MAC address+ length of bits+ Sequence

Layer 3: Network Layer Routes data to Ann's email client.



OSI Model

Network Models

OSI Model (Open Systems Interconnection Model)- Example

Maria sends an email to Ann using the OSI model:

Receiving End

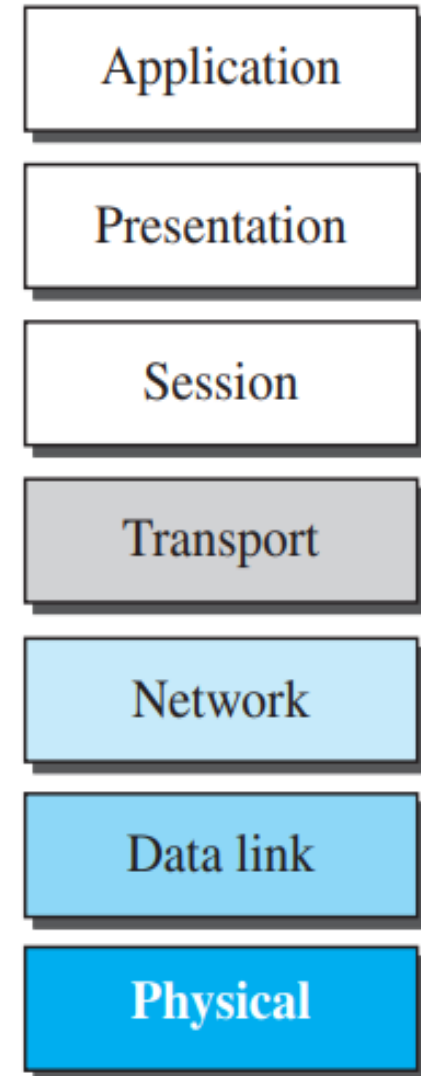
Layer 4: Transport Layer Reassembles segments.

Layer 5: Session Layer Establishes a connection.

Layer 6: Presentation Layer Decrypts and decompresses email.

Layer 7: Application Layer Ann's email client displays the email:

"Hello Ann, hope you're doing well."



OSI Model

Network Models

2.2 TCP/IP PROTOCOL SUITE

- A set of protocols organized into layers, widely used for internet communication.
- It follows a **hierarchical model** where higher layers depend on the services of lower layers.
- Originally a four-layer model, but now considered a five-layer model.
- **TCP (Transmission Control Protocol)** ensures reliable data transmission by managing packet delivery, error checking, and data reassembly.
- **IP (Internet Protocol)** handles addressing and routing, ensuring that data packets are sent to the correct destination.

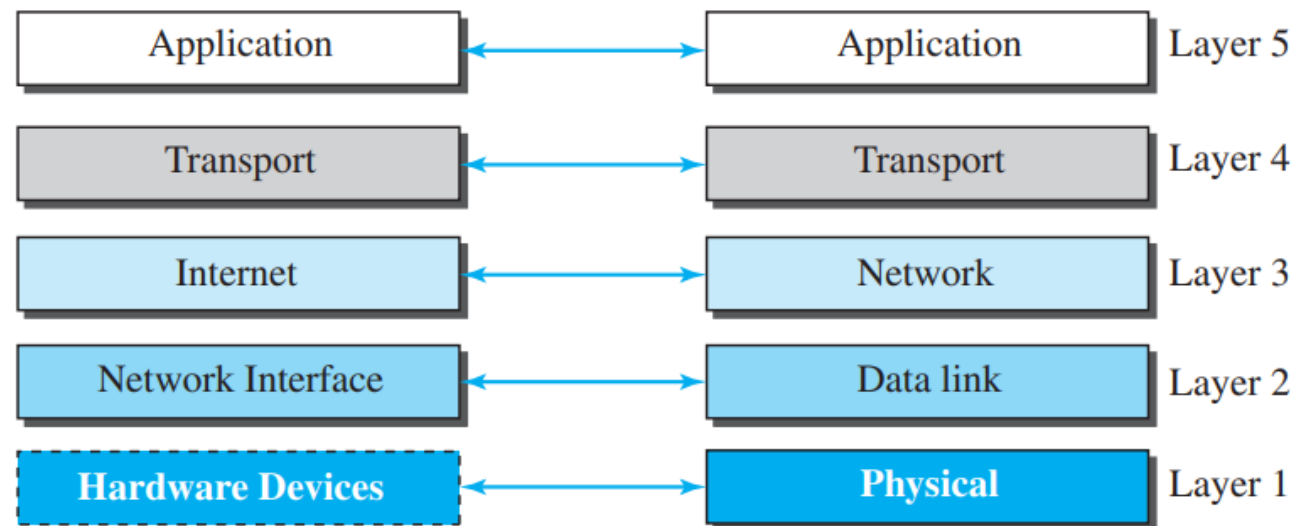
Network Models

TCP/IP PROTOCOL SUITE

Layered Architecture of TCP/IP

The **layered architecture** of TCP/IP is a structured framework that divides communication tasks into specific layers, each with its own responsibilities.

This architecture ensures data is transmitted efficiently and reliably between devices across a network. TCP/IP is composed of five layers, with each layer interacting with the one above and below it.



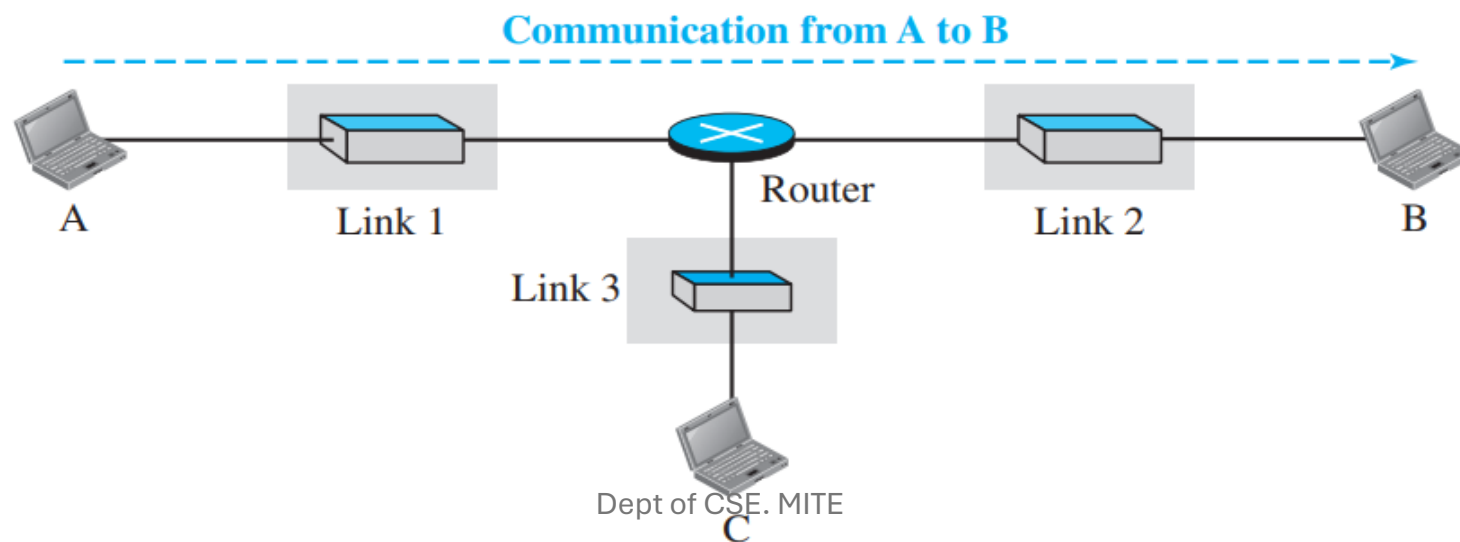
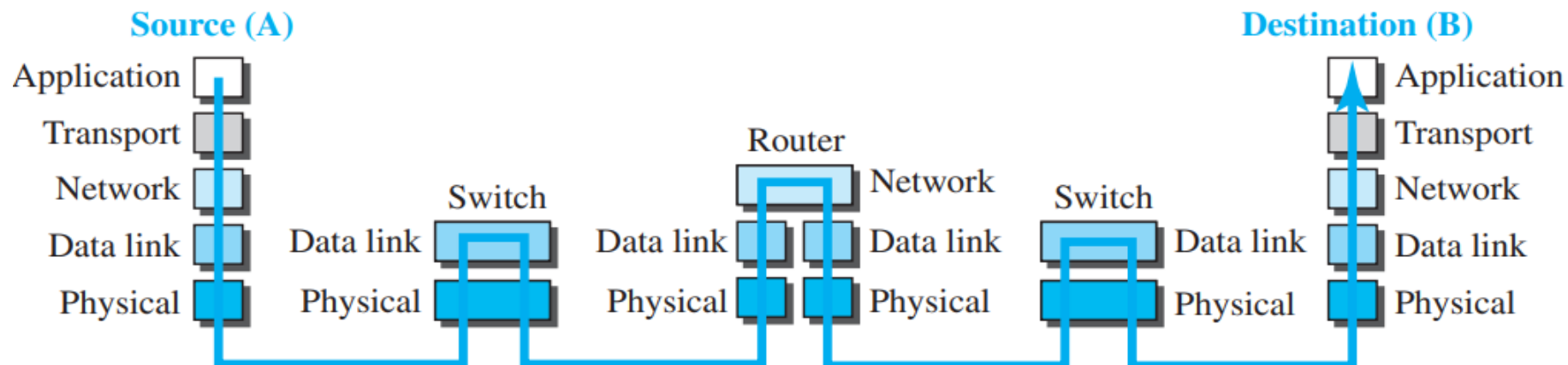
Dept of CSE, MITE
a. Original layers

b. Layers used in this book

Network Models

TCP/IP PROTOCOL SUITE

Layered Architecture of TCP/IP



Network Models

TCP/IP PROTOCOL SUITE

TCP/IP layered communication between two hosts, **Source (A)** and **Destination (B)**, connected through a network with **switches** and a **router**.

1. Source Host (A):

1. The source computer (A) sends data starting from the **Application Layer** and passes it down through the **Transport, Network, Data Link, and Physical Layers**.
2. The message is converted into physical signals and transmitted through **Link 1** to the first switch.

2. Switch (Link 1):

1. The switch in **Link 1** operates at the **Data Link and Physical Layers**.
2. It forwards the data based on physical and MAC addressing (Data Link Layer) and passes it onto the next device.

Network Models

TCP/IP PROTOCOL SUITE

Router:

1. The router connects **Link 1** and **Link 2**, operating at three layers: **Physical, Data Link, and Network**.
2. It routes packets based on IP addresses (Network Layer) and transfers data between networks (from Link 1 to Link 2).

•**Switch (Link 2):** Similar to the first switch, it operates at the **Data Link and Physical Layers** to forward the data within Link 2

•**Destination Host (B):**

•At the destination **computer (B)**, the message is received at the **Physical Layer** and travels up through the layers (Data Link, Network, Transport, and Application Layers).

•Finally, the **message reaches the Application Layer**, where it can be used by the receiving application.

Network Models

TCP/IP PROTOCOL SUITE

Layer Mapping:

OSI Layer	TCP/IP Equivalent	Example from Email Communication
Application (7)	Application (4)	Maria's email application (Gmail, Outlook) sending the email to Ann.
Presentation (6)	Application (4)	Encoding email content (e.g., encryption or format conversion).
Session (5)	Application (4)	Establishing and managing the session for the email exchange.
Transport (4)	Transport (3)	TCP segments the email, ensuring reliable transmission.
Network (3)	Internet (2)	IP routing of the email packets from Maria's network to Ann's network.
Data Link (2)	Network Access (1)	Framing and addressing email data for physical transmission.
Physical (1)	Network Access (1)	The actual transmission of the email data over physical media (e.g., Wi-Fi, Ethernet).

Network Models

Layers in the TCP/IP Protocol Suite

Logical connections simplify the understanding of each layer's role in TCP/IP communication.

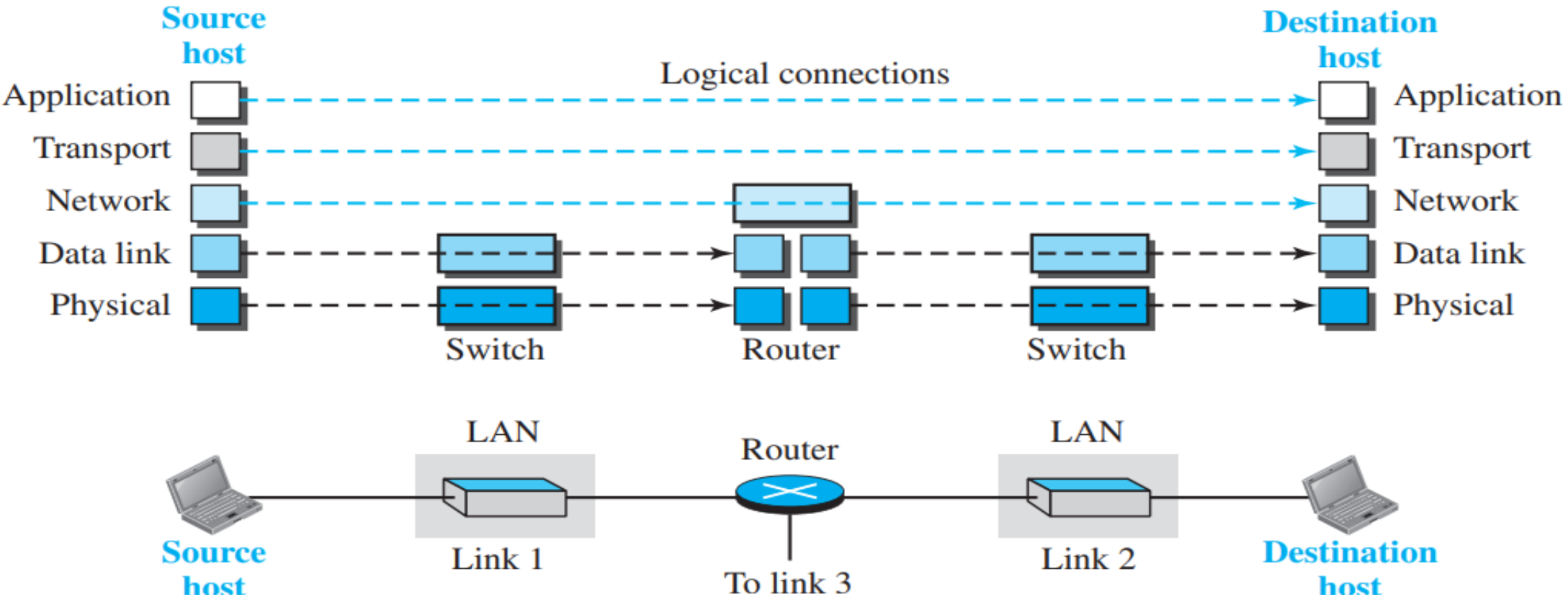
- **Application, Transport, and Network Layers:** Their duty is **end-to-end** communication between the source and destination hosts.

- **Data Link and Physical Layers:** Their duty is **hop-to-hop** communication, between devices like hosts, switches, and routers.

In the **top three layers**, **data units (packets) are unchanged by routers or switches**. In the **bottom two layers**, packets may be modified by routers but not by link-layer switches.

Network Models

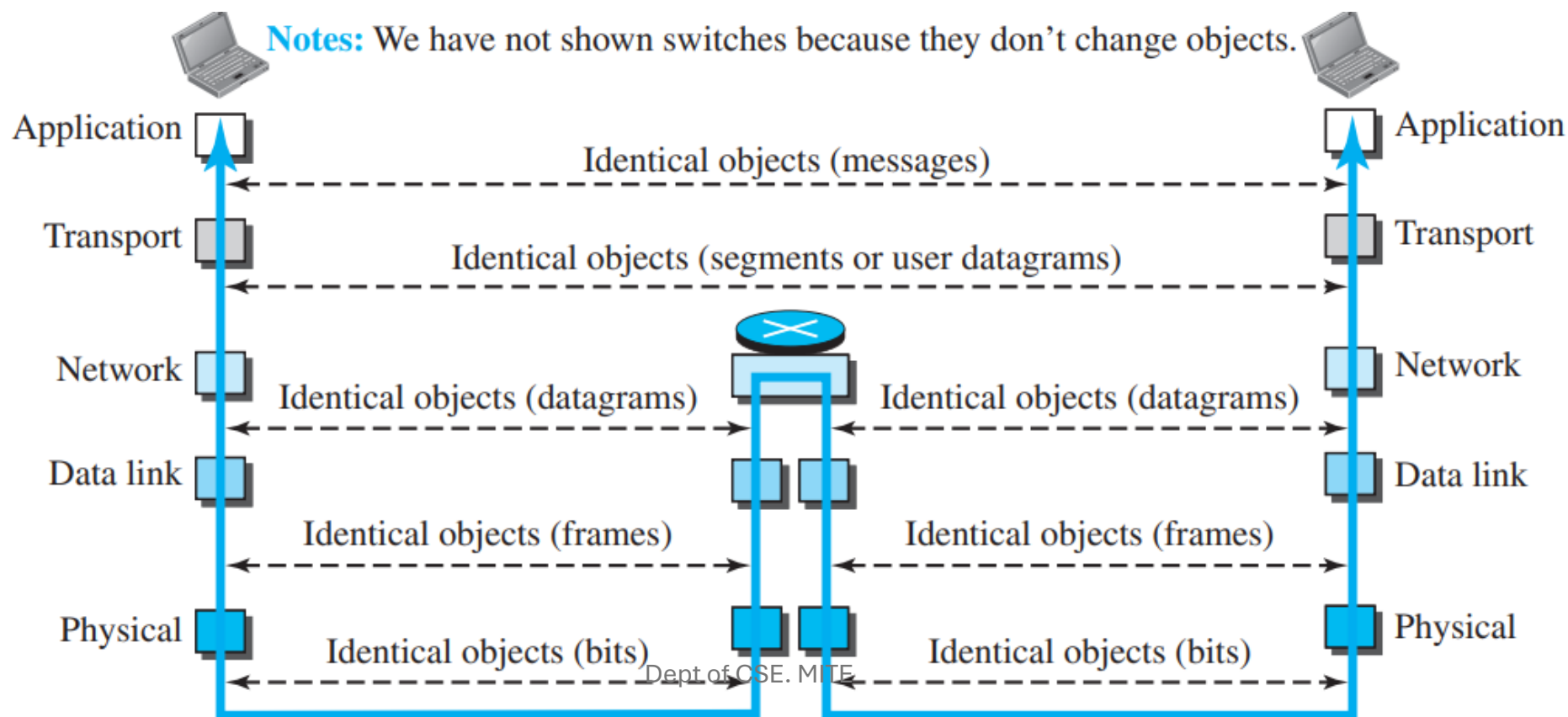
Layers in the TCP/IP Protocol Suite



Network Models

Layers in the TCP/IP Protocol Suite

Although the logical connection at the network layer is between two hosts, identical objects exist only between two hops because a router may fragment packets and send more packets than it receives. The link between two hops does not alter the object.



Network Models

Layers in the TCP/IP Protocol Suite

1. Application Layer:

- Function:** Provides **network services directly to user applications** (e.g., web browsers, email clients). It allows software to send and receive data over the network.

- Protocols:** HTTP, FTP, SMTP, DNS.

- Example:** When you browse the web, HTTP is used to request and receive web pages.

2. Transport Layer:

- Function:** Ensures **reliable data transmission between devices**. It manages error detection, recovery, and flow control, ensuring complete data delivery.

- Protocols:** TCP (Transmission Control Protocol), UDP (User Datagram Protocol).

- Example:** TCP ensures that packets are delivered in order and without errors, reassembling them on the receiving side.

Network Models

Layers in the TCP/IP Protocol Suite

3. Network Layer:

- **Function:** Handles **logical addressing** (IP addresses) and **routing of data packets** from the source to the destination across different networks.
- **Protocols:** IP (Internet Protocol), ICMP (Internet Control Message Protocol).
- **Example:** The IP protocol routes packets from your device to a remote server by determining the best path across networks.

4. Data Link Layer:

- **Function:** Provides **physical addressing** (MAC addresses) and **ensures error-free delivery** of frames (data packets) between devices on the same local network (e.g., between a computer and a router).
- **Protocols:** Ethernet, ARP (Address Resolution Protocol), Wi-Fi.
- **Example:** On a local network, Ethernet ensures that data is delivered between devices using MAC addresses.

Network Models

Layers in the TCP/IP Protocol Suite

5. Physical Layer:

- **Function:** Defines the **hardware aspects of data transmission**, including **cables, radio frequencies, and electrical signals**. It converts the data into physical signals that can be transmitted over the network medium.
- **Protocols:** Ethernet (physical cable standards), Wi-Fi (wireless communication standards).
- **Example:** Data is transmitted as electrical signals over an Ethernet cable or as radio signals in Wi-Fi.

Network Models

Layers in the TCP/IP Protocol Suite

The example of **sending an email** from **Maria** to **Ann** to explain how the TCP/IP layers work together:

1. Application Layer:

1. **Function:** Maria writes an email using an application like Gmail or Outlook.
2. **What happens:** The email application uses the **SMTP protocol** (Simple Mail Transfer Protocol) to create and format the message for transmission.

2. Transport Layer:

1. **Function:** Ensures the email message is reliably delivered to Ann.
2. **What happens:** The **TCP protocol** breaks Maria's email into smaller segments called **packets**.
TCP ensures these packets are sent in the correct order and checks for any errors in transmission.

Network Models

Layers in the TCP/IP Protocol Suite

3. Network Layer: Function: The frames are converted into **packets** with IP addresses for Maria and Ann.

Example: The packets travel through different routers across the internet, from Maria's IP address to Ann's IP address.

4. Data Link Layer:

The data is packaged into **frames** and sent to the next device on the network (e.g., from Maria's computer to the local router). MAC addresses are used to identify the source and destination within the local network.

5. Physical Layer: The email message is broken into binary data (1s and 0s) and sent over the network.

Network Models

Encapsulation and Decapsulation Process:

1. Encapsulation at the Source Host:

- **Application Layer:** The data is called a **message** and is passed to the transport layer.
- **Transport Layer:** Adds a header (with source and destination application details) to the message, creating a **segment** (TCP) or **user datagram** (UDP).
- **Network Layer:** Adds a header (with source and destination IP addresses) to the segment, creating a **datagram**.
- **Data Link Layer:** Adds a header (with link-layer addresses) to the datagram, creating a **frame**, which is passed to the physical layer for transmission.

Network Models

Encapsulation and Decapsulation Process:

2. Encapsulation and Decapsulation at the Router:

- The data-link layer **decapsulates the datagram from the frame** and sends it to the network layer.
- The network layer **checks the source and destination addresses**, consults its forwarding table, and forwards the datagram without changes, unless fragmentation is needed.
- The **next link's data-link layer encapsulates the datagram in a frame** and sends it to the physical layer for transmission.

Network Models

Encapsulation and Decapsulation Process:

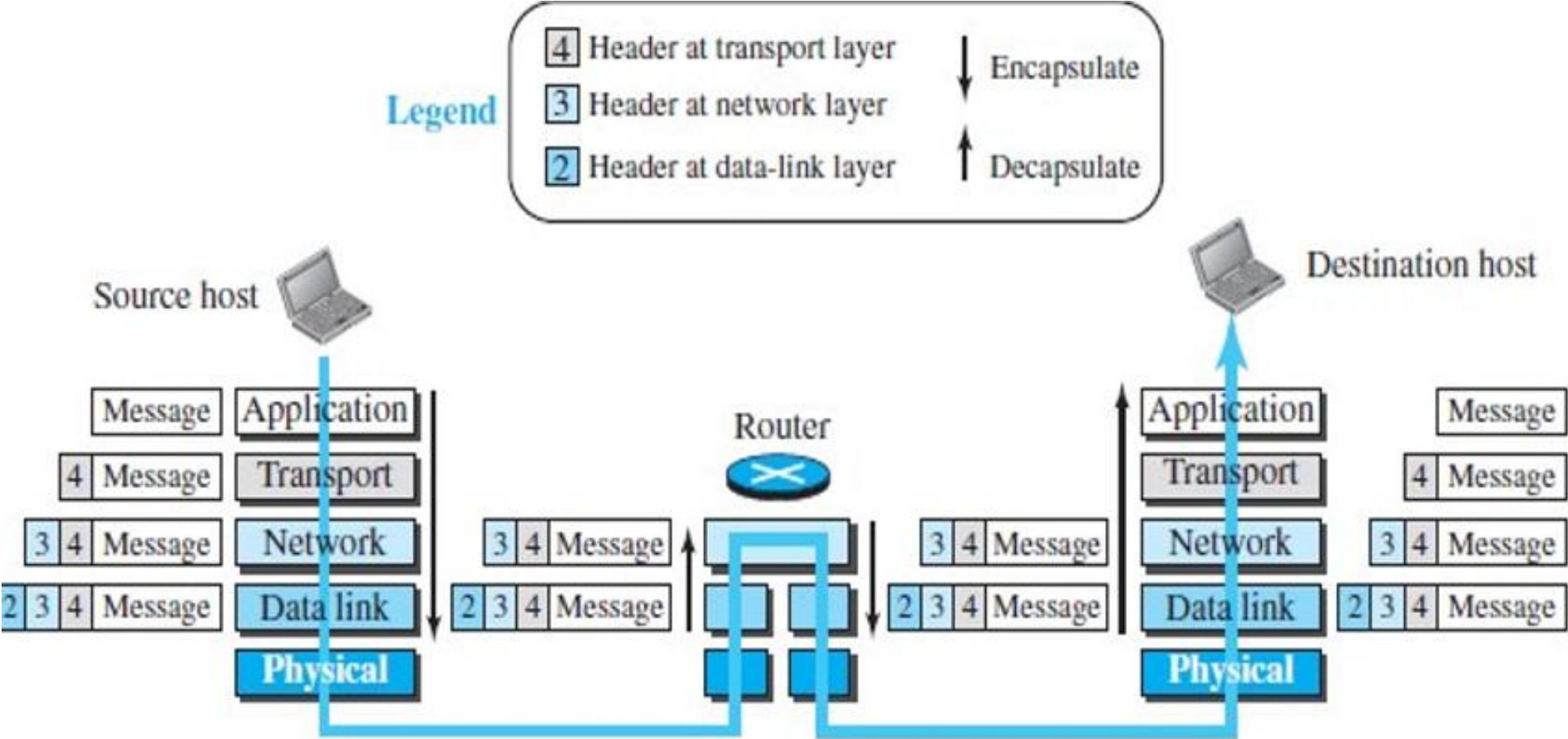
3. Decapsulation at the Destination Host:

- Each layer removes its **header** and **passes the payload to the next layer**.
- The message reaches the **Application Layer**, where it is finally delivered to the user application.

Each step involves **encapsulation** (adding headers) or **decapsulation** (removing headers) to ensure proper data delivery.

Network Models

Encapsulation and Decapsulation Process:



Network Models

Addressing in Protocol Layering:

In the Internet's protocol layering, **addressing** is crucial for communication between two parties, requiring both a **source** and **destination** address. There are typically **four types of addresses** (since the physical layer doesn't use addresses):

1.Application Layer: Uses **names** like domain names (e.g., someorg.com) or email addresses (e.g., somebody@coldmail.com).

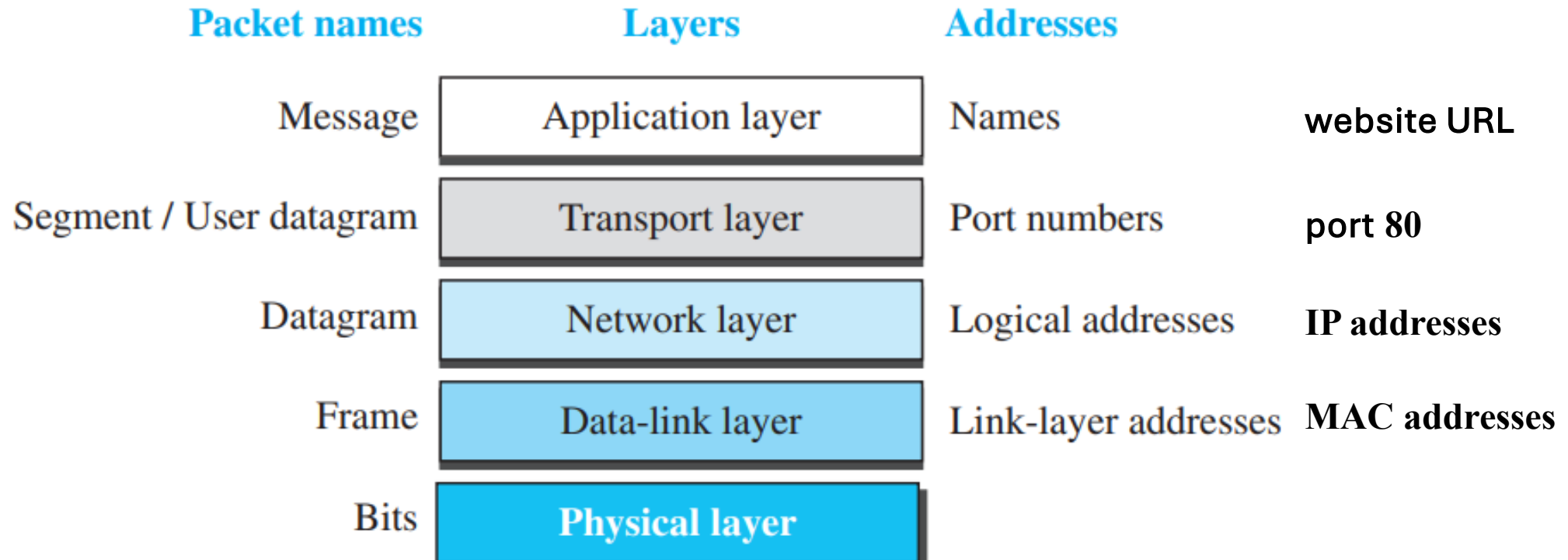
2.Transport Layer: Uses **port numbers** to identify specific programs or services on the source and destination devices.

3.Network Layer: Uses **IP addresses**, which are global and identify devices across the Internet.

4.Data Link Layer: Uses **MAC addresses** to uniquely identify devices within a local network (LAN or WAN).

Network Models

Addressing in Protocol Layering:



Network Models

Multiplexing and Demultiplexing:

In the **TCP/IP protocol suite**, **multiplexing** occurs at the source. **Demultiplexing** occurs at the destination.

- Multiplexing:** A protocol at a layer encapsulates data from several upper-layer protocols (e.g., TCP can handle data from HTTP, FTP).

- Demultiplexing:** A protocol decapsulates and directs data to the correct upper-layer protocol (e.g., IP directs packets to TCP or UDP).

Each protocol header includes a field to identify which protocol the data belongs to.

Network Models

Multiplexing and Demultiplexing:

At the Source (Multiplexing):

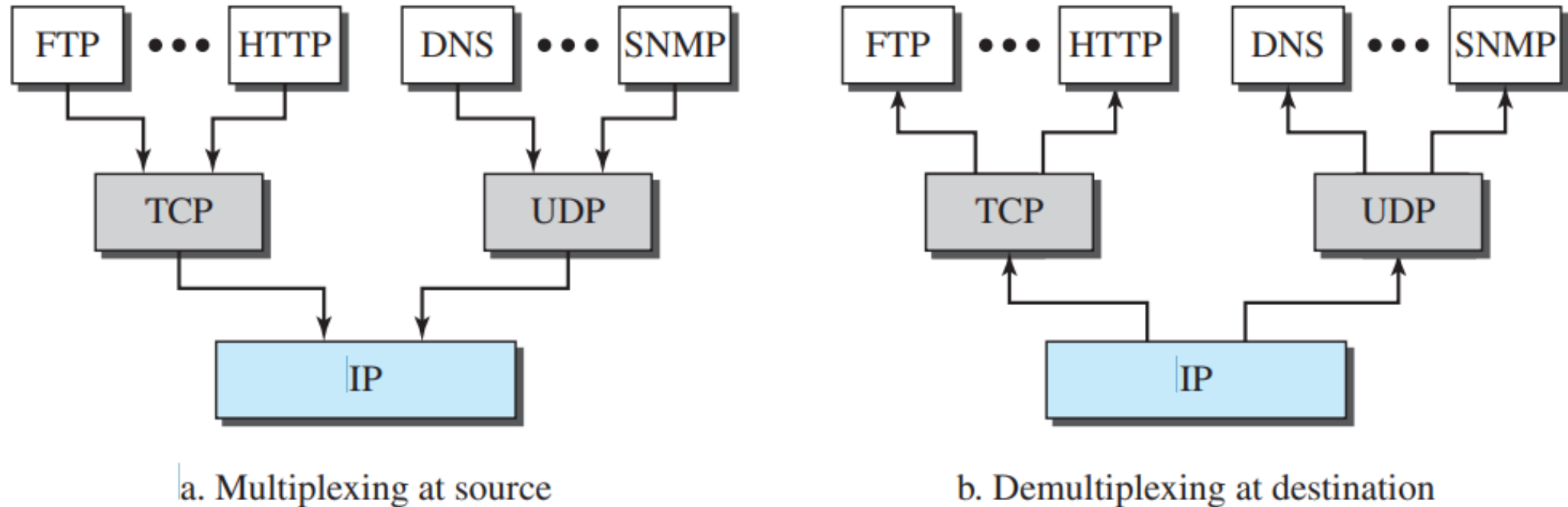
1. You open your **browser and request a web page (HTTP)**, and at the same time, you start an **FTP session to upload a file**.
2. Your computer's **transport layer uses TCP** for both **HTTP and FTP**.
3. **TCP** adds a **port number** (e.g., port 80 for HTTP, port 21 for FTP) to each of these data streams and forwards them to the **IP** layer.
4. The **IP** layer **encapsulates these TCP segments** into **datagrams** and sends them over **the network**.

At the Destination (Demultiplexing):

1. The **destination server** receives the **data from the network layer (IP)** and **passes it to the transport layer**.
2. The **TCP** at the **server looks at the port numbers**:
 1. If it's port 80, it forwards the data to the **web server (HTTP)**.
 2. If it's port 21, it forwards the data to the **FTP server**.
3. This way, the **server knows which data belongs to the website and which data belongs to the file transfer**.

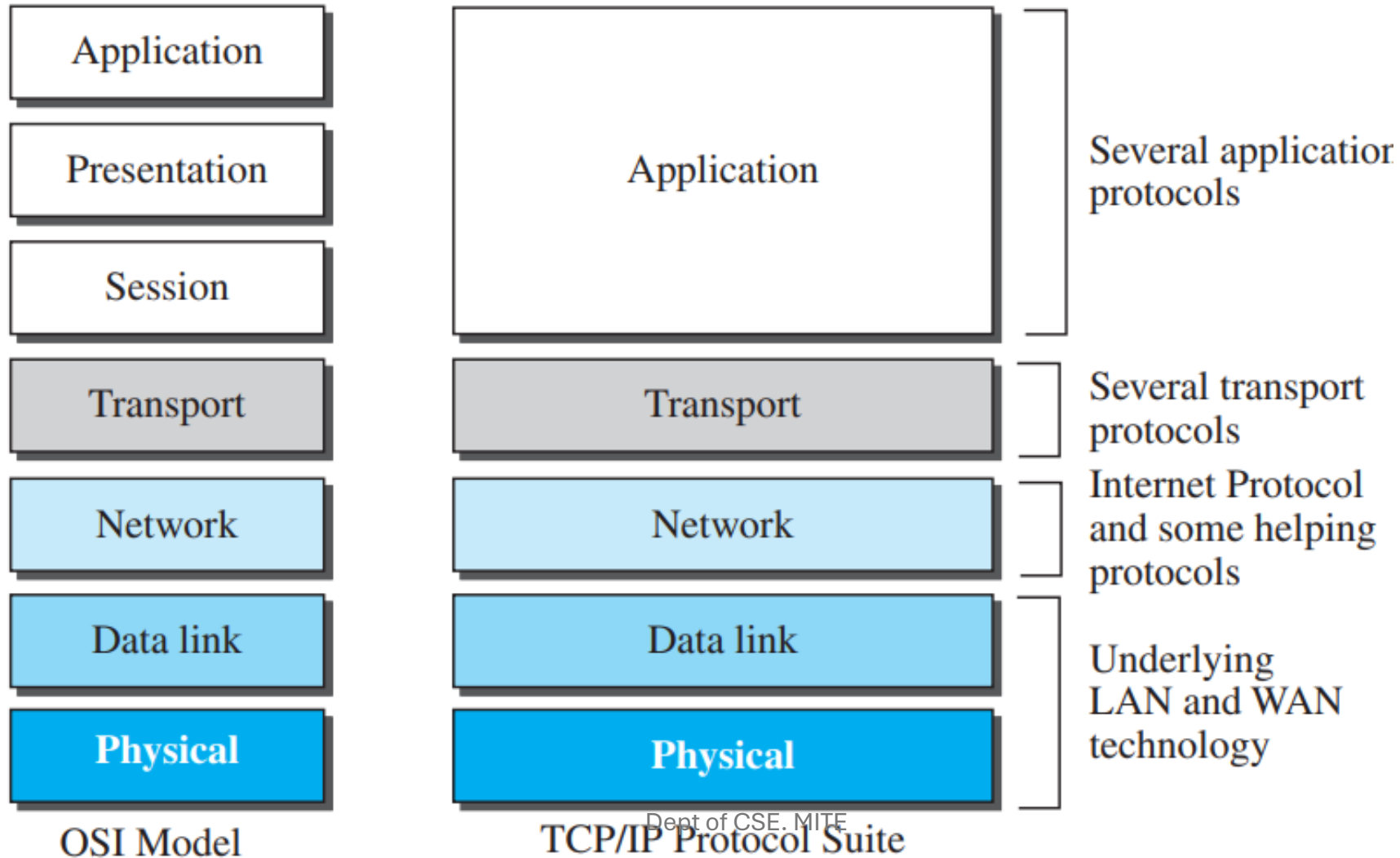
Network Models

Multiplexing and Demultiplexing:



Network Models

Comparison of OSI and TCP/IP Models:



Network Models

Lack of OSI Model's Success:

1. Timing: OSI was developed after TCP/IP was already well-established, with significant investment in TCP/IP. Changing to OSI would have been costly.

2. Incomplete Definition: Some OSI layers, like the **presentation** and **session layers**, were not fully defined, and corresponding protocols and software were not fully developed.

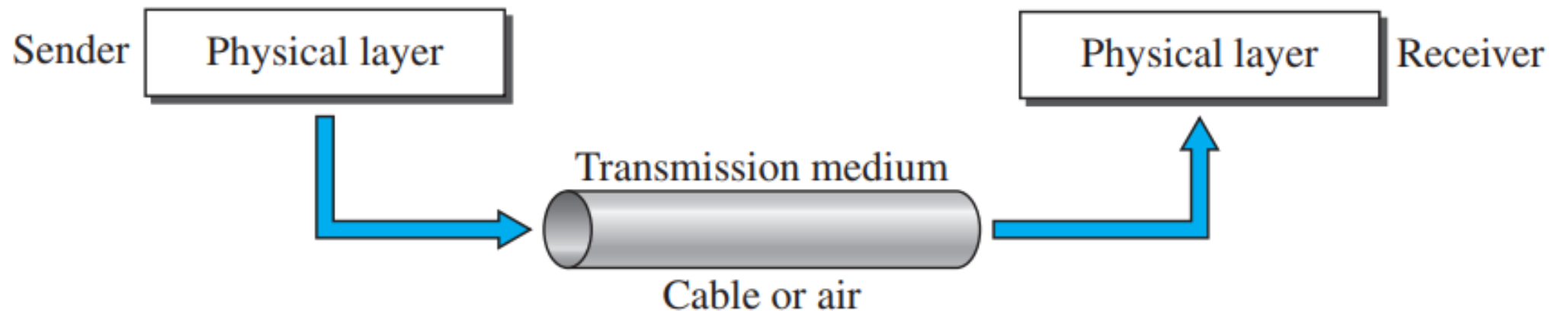
3. Performance: When OSI was implemented, it did not perform well enough to convince the Internet community to switch from TCP/IP(these layers remained theoretical or underutilized in practice).

Chapter 3 Introduction to Physical Layer

Physical Layer: Transmission Medium

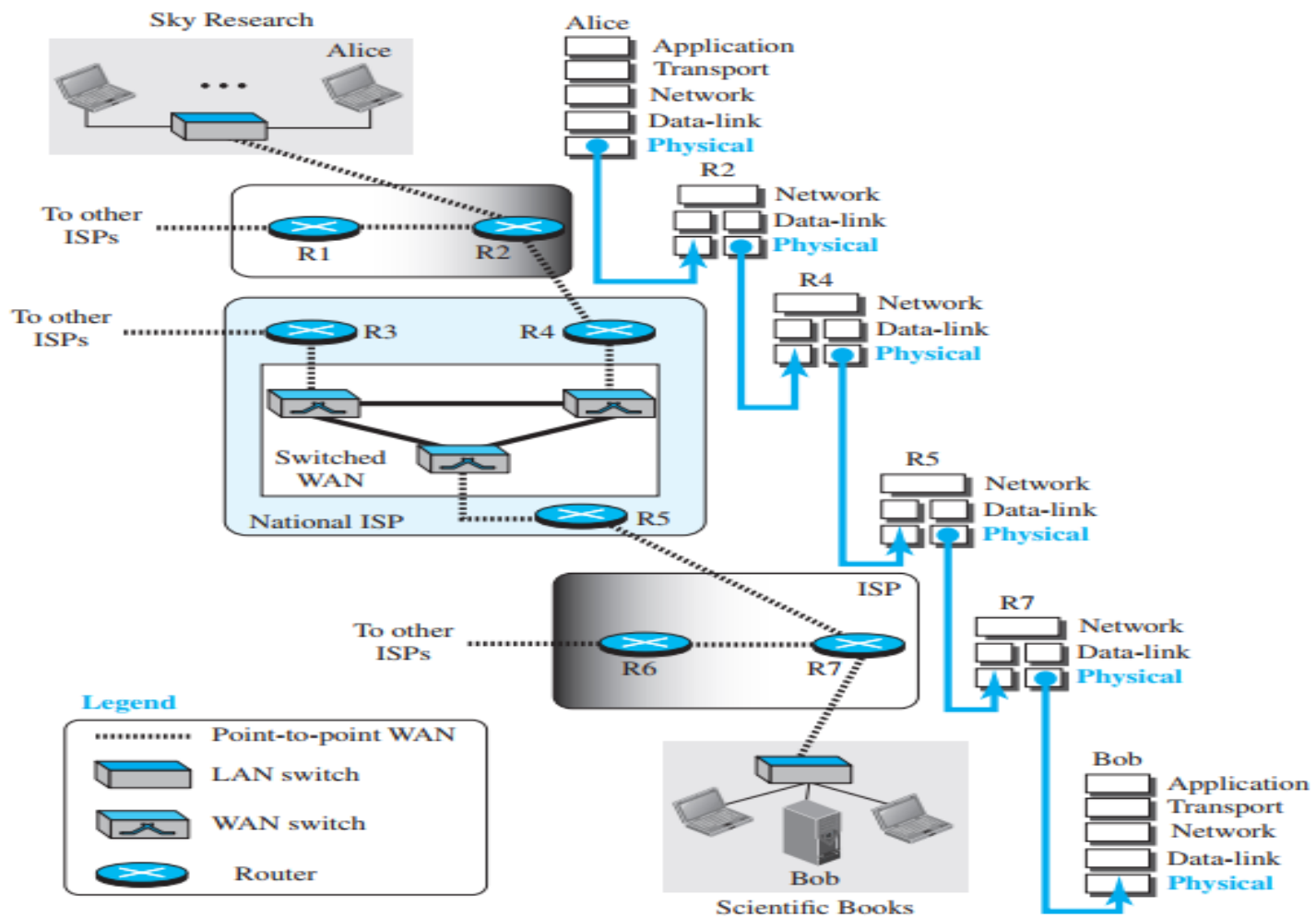
This layer deals with the **hardware aspects of communication, such as cables, connectors, electrical signals, and the transmission medium itself.**

The transmission medium is critical in ensuring the **data is properly transmitted across networks** and is **classified into two categories: guided media and unguided media.**



Physical Layer: Transmission Medium

Figure 3.1 Communication at the physical layer



Physical Layer: Transmission Medium

Five Levels of Communication

Communication between Alice's computer and Bob's server occurs at five levels — **Application, Transport, Network, Data-Link, and Physical.**

•Logical vs Physical Communication:

The first four layers (Application to Data-Link) handle **logical communication**, meaning how data is formatted, sent, and interpreted. The **Physical layer** handles **actual transmission** of data as signals.

•Role of the Physical Layer:

At the Physical layer, data is **converted into signals** (like electrical pulses, light, or radio waves) to travel across the network media (wires, fiber, air).

Physical Layer: Transmission Medium

Five Levels of Communication

Transmission Path:

The data travels from **host-to-router**, **router-to-router**, and finally **router-to-host** and **switches** are also involved in this physical communication.

•Data Becomes Signals:

To **transmit data**, the medium must **change it into signals**. These signals represent the data and can be **analog** (continuous) or **digital** (discrete 0s and 1s).

•Final Communication Process:

Although Alice and Bob are **exchanging data**, at the lowest level, this involves the **exchange of signals** between devices using different transmission media and network components.

Physical Layer: Transmission Medium

3.1. DATA AND SIGNALS

Data can exist in two basic forms: **analog** and **digital**.

- **Analog data** refers to information that is **continuous**, meaning it can take on an **infinite number of values within a range**.
- An example is an **analog clock**, where the hands move smoothly and continuously to show time.
- In contrast, **digital data** is **discrete**, meaning it has clearly defined, separate values.
- A **digital clock** changes time suddenly, such as jumping from 8:05 to 8:06, without showing the in-between values.

Physical Layer: Signals

DATA AND SIGNALS

- **Analog** data is common in the real world. For example, the **human voice creates sound waves that change smoothly and continuously**. When we speak, these analog sound waves travel through the air.
- On the other hand, **digital data represents** information using a series of **0s and 1s** - the binary system used by computers. This kind of **data has a fixed number of possible values and is stored in a digital format** in computer memory.
- **Digital data** can be sent **directly as a digital signal**, or it **can be converted** (modulated) into an **analog signal if the transmission medium** (like radio or telephone lines) requires analog format.

Physical Layer: Signals

Analog and Digital Signals

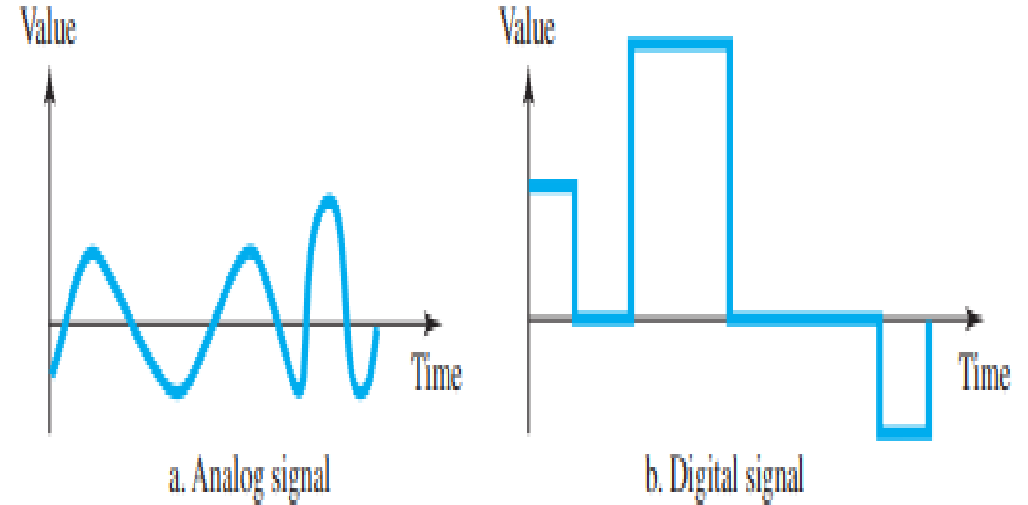
- A **digital signal** has a **limited number of distinct values**.
- The most common digital signal uses **just two levels**, typically represented as **0 and 1**.
- These signals don't transition gradually; instead, they change instantly from one value to another creating a **step-like** or **square wave** pattern.
- Each step in a digital signal marks a **specific time** where the **value suddenly changes**.

Physical Layer: Signals

Analog and Digital Signals

- To understand signals better, a graph with two perpendicular axes. The horizontal axis represents time, and the vertical axis shows the signal's strength or value.
- When we draw an analog signal, we see a smooth, continuous curve. On the other hand, a digital signal appears as a series of sudden jumps, like vertical lines that go up or down at fixed points in time.

Figure 3.2 Comparison of analog and digital signals



Physical Layer: Signals

Periodic and Nonperiodic Signals

- Both analog and digital signals can be **further classified based on their behavior over time** they can be **either periodic or nonperiodic**.
- The term **periodic** means that the signal follows a **repeating pattern**. That is, **the signal completes a specific shape or wave form within a fixed amount of time** (called a period) and **then repeats the same pattern over and over**.
- The **completion of one full pattern is called a cycle**. For example, a sine wave that repeats itself at regular intervals is a classic example of a periodic signal.

Physical Layer: Signals

Periodic and Nonperiodic Signals

- On the other hand, a **nonperiodic (also called aperiodic) signal does not repeat. It changes randomly or unpredictably over time without forming a consistent pattern or cycle.**
- In real-world communication systems, **signals that carry actual data are often nonperiodic** because **the data itself changes over time and rarely forms a repeating structure.**

3.4 Transmission Impairment

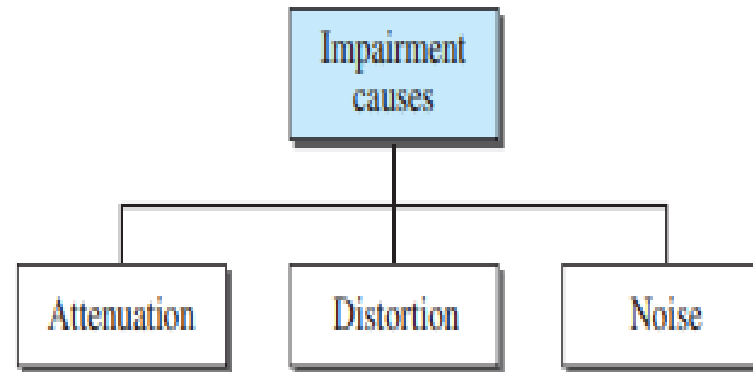
- When a signal travels through a **transmission medium** (like copper wire, fiber optics, or air), the medium is **not perfect**. This means it **cannot preserve the original signal exactly** as it is.
- Due to **imperfections in the medium**, the **signal gets altered** during its **journey from the sender to the receiver**. This alteration is known as **signal impairment**.
- Transmission Impairment is **the loss or degradation of signal quality as it travels through a communication medium**(What is sent is not exactly what is received).

Transmission Impairment

Major Causes of Transmission Impairment:

- There are **three main causes** of signal impairment:
 1. Attenuation,
 2. Distortion, and
 3. Noise

Figure 3.26 *Causes of impairment*

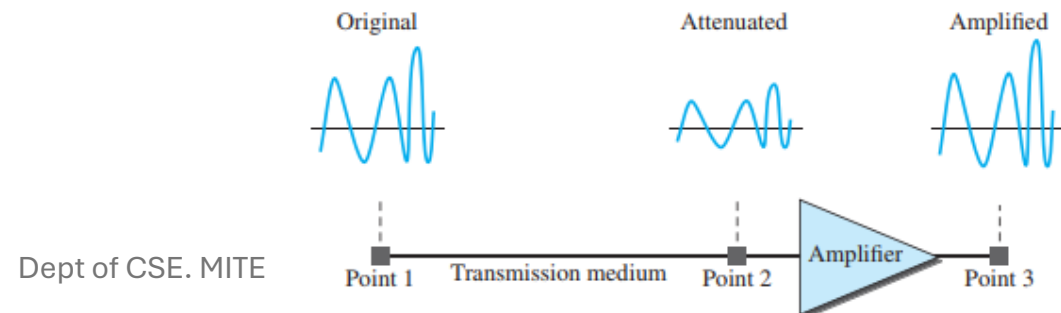


Transmission Impairment

1. Attenuation

- Attenuation means **loss of signal energy** (or strength) during transmission.
- When a signal travels through a medium, part of its energy is lost in overcoming the resistance of the medium. This lost energy often turns into heat.
- **Example:** A long electrical wire gets warm when carrying signals—this is due to energy loss.
- **Amplifier:** To **compensate** for attenuation, we use an **amplifier** to **boost** the signal's power before it becomes too weak.

Figure 3.27 Attenuation



Transmission Impairment

1. Attenuation

- **Measurement Unit: Decibel (dB)**
- A **decibel (dB)** measures how much signal power is **gained** or **lost** between two points:

- Point 1: Just after the signal is sent (transmitted).

$$\text{dB} = 10 \cdot \log_{10} \left(\frac{P_2}{P_1} \right)$$

- Point 2: Just before the signal is received.

- **If dB is negative** → signal **lost** (attenuated).
- **If dB is positive** → signal **gained** (amplified).

Transmission Impairment

Attenuation

Ex1: Suppose a signal travels through a transmission medium and its power is reduced to one-half. This means that $P_2 = \frac{1}{2} P_1$. In this case, the attenuation (loss of power) can be calculated as

$P_1=2; P_2=1$ (which is half of 2) $\frac{P_2}{P_1} = 0.5 \Rightarrow \text{dB} = 10 \log_{10}(0.5) = -3 \text{ dB}$

Ex2: A signal travels through an amplifier, and its power is increased 10 times. This means that $P_2 = 10P_1$. In this case, the amplification (gain of power) can be calculated as

Power becomes $10\times \rightarrow 10 \log_{10} \frac{P_2}{P_1} = 10 \log_{10} \frac{10P_1}{P_1} = 10 \log_{10} 10 = 10(1) = 10 \text{ dB}$

Transmission Impairment

Attenuation

Ex 3: Sometimes the decibel is used to measure signal power in milliwatts. In this case, it is referred to as dBm and is calculated as $\text{dBm} = 10 \log_{10} P_m$, where P_m is the power in milliwatts. Calculate the power of a signal if its $\text{dBm} = -30$.

Solution

We can calculate the power in the signal as $\text{dBm} = 10 \log_{10} P_m$

$$\text{dBm} = -30$$

$$-30 = 10 \log_{10} P_m$$

$$-3 = \log_{10} P_m$$

Convert from log form to normal form: $P_m = 10^{-3}$ $\log_{10} x = y \Rightarrow x = 10^y$

The signal power is $10^{-3} = 0.001 \text{ mW}$

Transmission Impairment

2. Distortion

Distortion refers to any **change in the shape or form of a signal** as it travels through a transmission medium.

Why it happens:

A **composite signal is made of multiple frequency components**. Each of **these components can travel at different speeds in the medium** (due to different propagation delays).

Because of this:

- The frequencies arrive at different times
- The phases of these components change
- The original shape of the signal is distorted

At the sender, all components are in phase

At the receiver, components become out of phase → Distortion

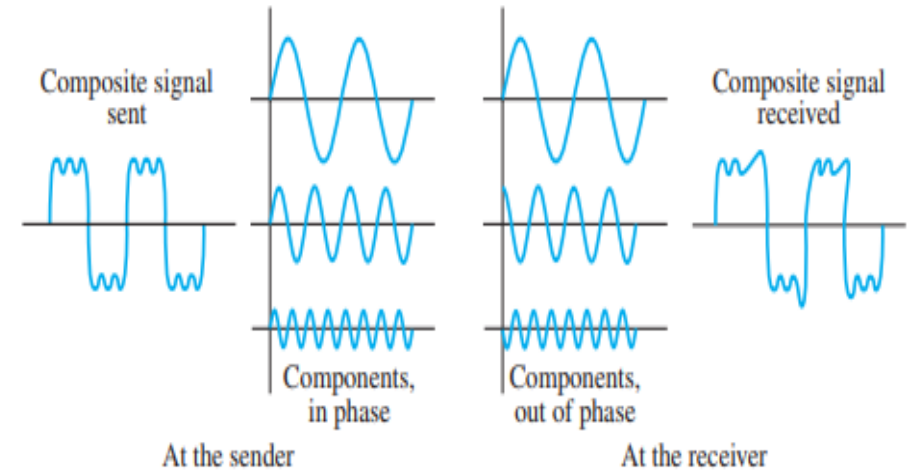
Transmission Impairment

Distortion

Example:

- Imagine sending a square wave signal (which is composed of many frequencies). If higher frequencies are delayed more than lower ones, the wave **loses its shape** and may become unrecognizable.
- Shows how a signal that is originally sharp and regular becomes **distorted** at the receiver because its components are **out of sync**.

Figure 3.29 Distortion



Transmission Impairment

Distortion

Attenuation

Definition

Loss of signal **power** over distance

Effect

Signal becomes **weaker**

Cause

Medium resistance, absorption, spreading

Solution

Amplifiers, repeaters

Distortion

Change in **signal shape or waveform**

Signal becomes **altered or deformed**

Different frequencies traveling at different speeds, non-linearities

Equalizers, filters, waveform correction

Transmission Impairment

3. Noise

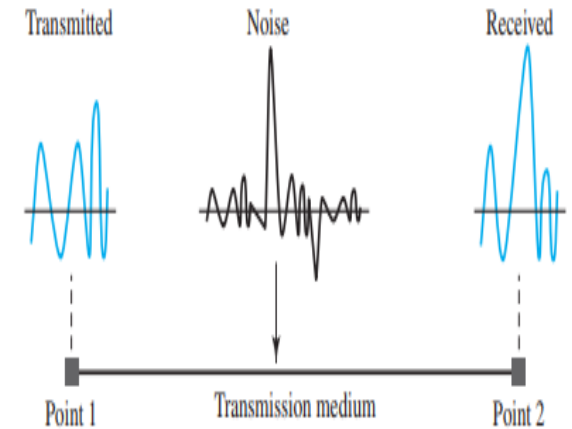
Noise is any unwanted electrical signal that interferes with the original transmitted signal during communication. It can distort or corrupt the signal, leading to errors at the receiver.

Types of Noise:

1. Thermal Noise

- Thermal noise is generated by the **random motion of electrons** in a conductor due to heat.
- It exists in all electrical devices and is also known as **white noise**.
- It is **independent of the transmitted signal** and present even when no signal is being sent
- FM radio or TV volume up with no signal.

Figure 3.30 Noise



Transmission Impairment

Noise

2. Induced Noise

- Induced noise is caused by **external electromagnetic sources**, such as motors, transformers, and appliances. These devices act as **sending antennas**, while the transmission medium acts as a **receiving antenna**, picking up unwanted signals.

3. Crosstalk

- Crosstalk occurs when a **signal from one wire unintentionally affects another wire nearby**. It happens due to **electromagnetic interference**, where one wire acts as a transmitter and the other as a receiver. Common in **twisted-pair cables**.

4. Impulse Noise

- Impulse noise consists of **sudden, short-duration, high-energy pulses** that can severely corrupt data. It is caused by **power lines, lightning strikes, or switching systems**, and is especially **damaging to digital signals**.

Transmission Impairment

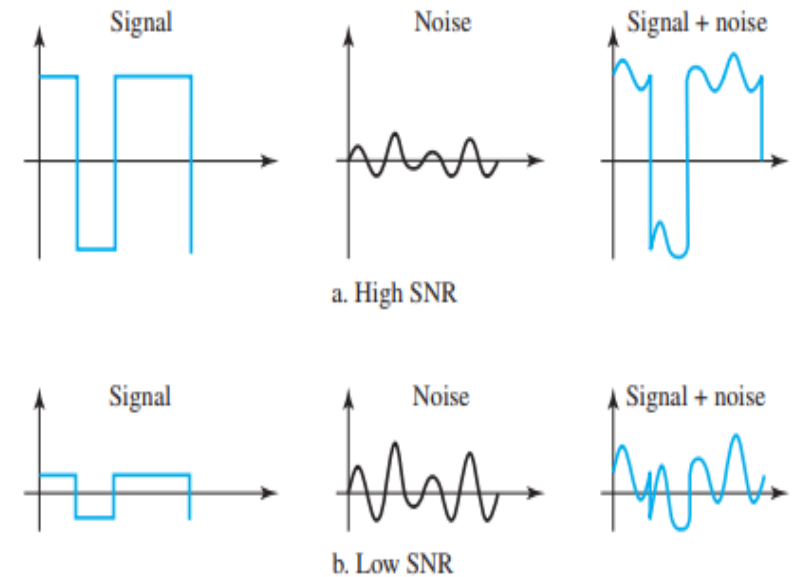
Signal-to-Noise Ratio (SNR)

- **SNR** is the **ratio** of the **power of the signal** to the **power of the noise** in the channel. It helps determine how much a signal is **affected by noise**.

$$\text{SNR} = \frac{\text{Average Signal Power}}{\text{Average Noise Power}}$$

- A **higher SNR** → clear signal, less noise
- A **lower SNR** → noisy signal, possible errors

Figure 3.31 Two cases of SNR: a high SNR and a low SNR



SNR in Decibels (SNRdB)

Since powers can vary over a wide range, we express SNR in **decibels** using: $\text{SNR}_{dB} = 10 \cdot \log_{10} \left(\frac{P_{\text{signal}}}{P_{\text{noise}}} \right)$

Signal-to-Noise Ratio (SNR)

Signal-to-Noise Ratio (SNR)

A signal has a power of 50 mW and the noise power is 0.5 mW.

1. Calculate the **Signal-to-Noise Ratio (SNR)** in linear form.
2. Convert the SNR to **decibels (dB)**.

$$SNR = \frac{P_s}{P_n} = \frac{50}{0.5} = 100$$

$$SNR_{dB} = 10 \log_{10} \left(\frac{P_s}{P_n} \right)$$

$$SNR_{dB} = 10 \log_{10}(100) = 10 \times 2 = 20 \text{ dB}$$

1. The **Signal-to-Noise Ratio (SNR)** in linear form

2. SNR to **decibels**

Ratio of signal power to noise power - expressed **in two different ways**.

Transmission Impairment

Noise

Signal-to-Noise Ratio (SNR)

Example 3.31

The power of a signal is 10 mW and the power of the noise is 1 μ W; what are the values of SNR and SNR_{dB}?

Solution

The values of SNR and SNR_{dB} can be calculated as follows:

$$1 \text{ mW} = 1000 \mu\text{W}$$

$$\text{SNR} = (10,000 \mu\text{W}) / (1 \mu\text{W}) = 10,000 \quad \text{SNR}_{\text{dB}} = 10 \log_{10} 10,000 = 10 \log_{10} 10^4 = 40$$

Example 3.32

The values of SNR and SNR_{dB} for a noiseless channel are

$$\text{SNR} = (\text{signal power}) / 0 = \infty \quad \longrightarrow \quad \text{SNR}_{\text{dB}} = 10 \log_{10} \infty = \infty$$

We can never achieve this ratio in real life; it is an ideal.

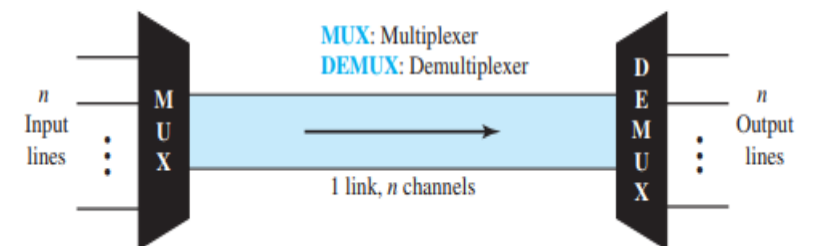
Multiplexing

Multiplexing is a technique that allows **multiple signals or data streams to be transmitted simultaneously** over a **single communication link**. This is done by efficiently **sharing the bandwidth** of the link among the multiple data sources.

How Multiplexing Works:

- On the **sender side**, a **Multiplexer (MUX)** combines the data streams from **multiple sources** into **one composite signal** (many-to-one).
- On the **receiver side**, a **Demultiplexer (DEMUX)** separates the composite signal into the **original data streams** and delivers each one to its intended destination (one-to-many).
- **One physical link** is shared by **n logical channels**.
- **Link** = physical medium (e.g., cable, fiber)
- **Channel** = logical path that carries one signal

Figure 6.1 Dividing a link into channels

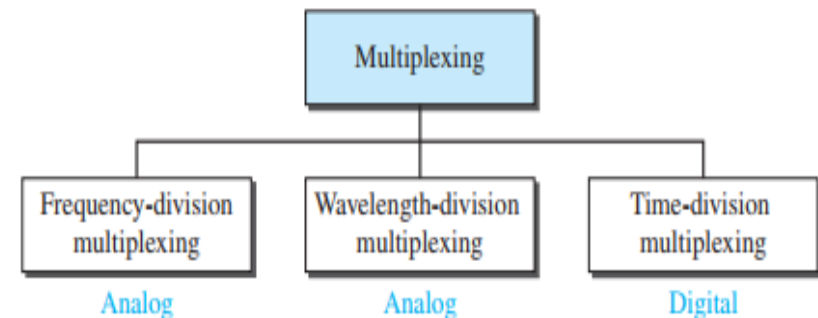


Multiplexing

- There are three basic multiplexing techniques:
 1. Frequency-division multiplexing,
 2. Wavelength-division multiplexing, and
 3. Time-division multiplexing.

The first two are techniques designed for analog signals, the third, for digital signals (see Figure 6.2).

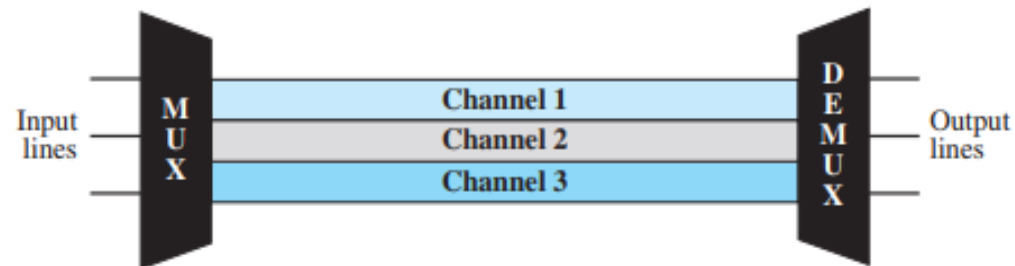
Figure 6.2 Categories of multiplexing



Frequency-Division Multiplexing

- **Frequency-Division Multiplexing (FDM)** is an **analog multiplexing technique** in which **multiple signals** are transmitted **simultaneously over a single communication link**, with each signal assigned a **unique frequency range (channel)** within the total available bandwidth.

Figure 6.3 *Frequency-division multiplexing*



Frequency-Division Multiplexing

How It Works

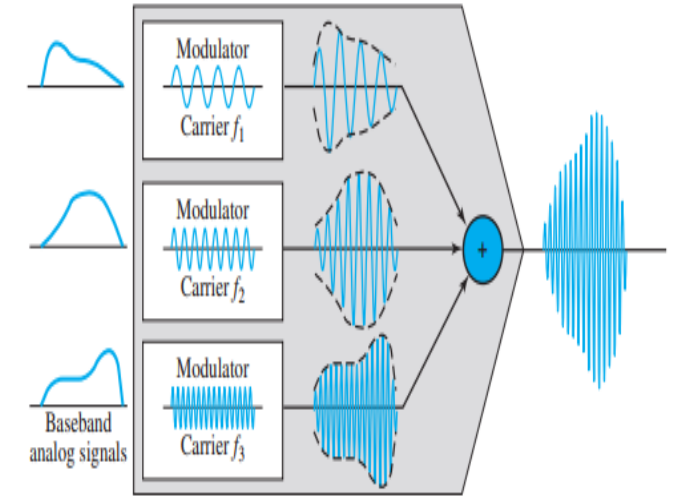
- Imagine a road divided into lanes.
- Each car (signal) stays in its own lane (frequency) so they don't crash into each other.
- In FDM, each signal is **placed on a different frequency**, just like each car stays in its own lane.
- These separate frequencies (called **channels**) are **combined** into one signal and sent together.
- At the receiver, the signals are **separated again** using filters.

Frequency-Division Multiplexing

Figure 6.4 FDM process

Multiplexing Process

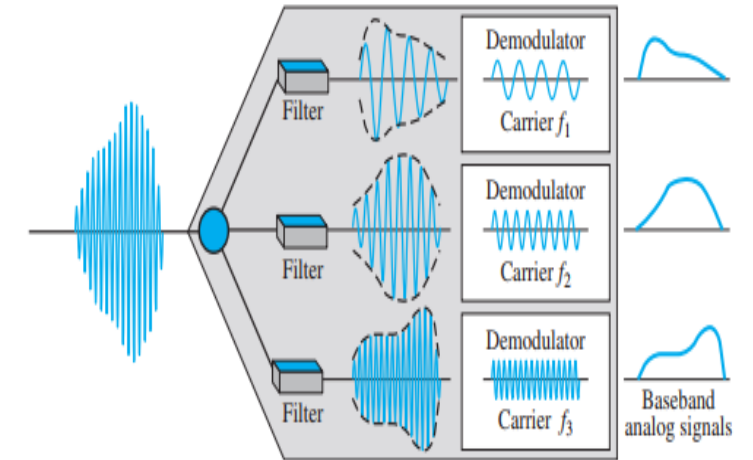
- **Each device** (or source) wants to send its own signal.
- All signals are in **similar frequency ranges** (e.g., audio signals).
- The **Multiplexer** takes each signal and:
 - Assigns it a **unique carrier frequency** (like f_1 , f_2 , f_3).
 - **Modulates** the signal onto that carrier frequency (like tuning it onto a separate radio channel).
- These modulated signals are then **combined** into one **composite signal**.
- The combined signal is sent over a **single communication channel** (like a wire or fiber).
- **Example:** Think of multiple radio stations being broadcast at the same time on different frequencies.



Frequency-Division Multiplexing

Demultiplexing Process

- The **Demultiplexer** receives the combined signal.
- It uses **filters** to **separate each frequency range** (just like tuning into one radio station at a time).
- Each separated signal is then **demodulated** to get back the original message.
- The clean individual signals are sent to their respective **output devices**.
- **Example:** Your car radio can pick one station (filter and demodulate it) from many that are being broadcast together.



Wavelength-Division Multiplexing (WDM)

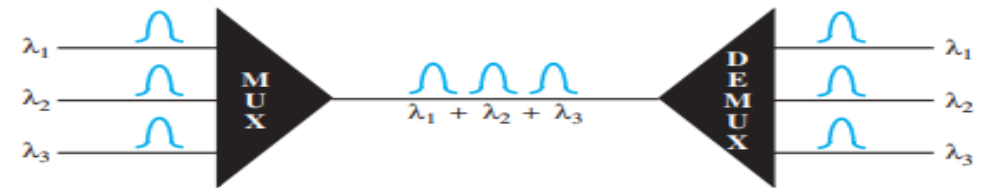
WDM is a technique used in fiber-optic communication that combines multiple optical signals (light beams), each with a different wavelength, into a single fiber to transmit data simultaneously.

How It Works:

- Each data source sends light of a unique wavelength (like different colors).
- A multiplexer (MUX) merges all these light signals into one beam.
- The fiber-optic cable carries the combined beam.

At the receiver side, a demultiplexer (DEMUX) separates the combined light into original individual wavelengths.

Figure 6.10 Wavelength-division multiplexing



Wavelength-Division Multiplexing (WDM)

WDM is similar to FDM, but it works with **light instead of electrical signals**.

Each wavelength = one channel. Signals don't interfere because they use different wavelengths ($\lambda_1, \lambda_2, \lambda_3 \dots$).

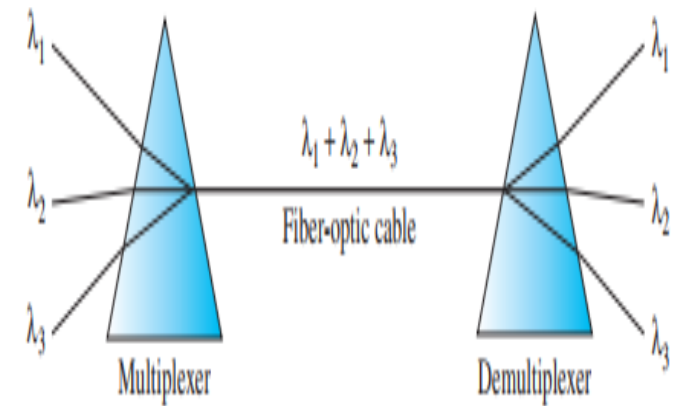
Think of WDM like a rainbow:

- Each color (wavelength) is different.
- All colors travel together in a single beam of light.
- Later, they can be separated using a prism, just like a DEMUX.

Types: WDM (basic): Fewer channels, spaced far apart.

DWDM (Dense WDM): More channels, very close spacing, allows higher data capacity.

Figure 6.11 Prisms in wavelength-division multiplexing and demultiplexing



Time Division Multiplexing (TDM)

How It Works:

- Each source (1, 2, 3...) gets a **time slot** in which it can send data.
- A **multiplexer (MUX)** collects one unit of data from each input in order and puts them into a **frame**.
- The combined data (frame) is sent over a single fast link.
- A **demultiplexer (DEMUX)** at the other end separates the frame back into individual data units and sends them to the right destination.

Time Division Multiplexing (WDM)

Synchronous Time-Division Multiplexing

- Synchronous TDM is a digital multiplexing technique.
- **Time Slot** A fixed time period assigned to each device. Each device gets one time slot per frame, whether it's sending data or not.
- **Frame** A complete cycle of time slots, one slot for each device. If there are n devices, then each frame has n slots.

Slot Duration and Speed

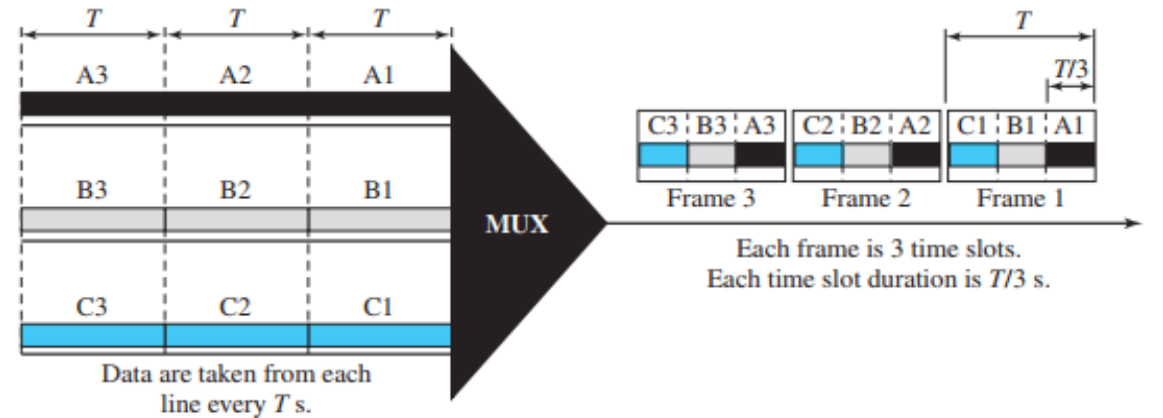
- Suppose each device sends 1 unit of data in time **T seconds**.
- When multiplexed, all data units are sent in just **T seconds**.
- So each output time slot becomes **T/n seconds**, where **n** is the number of devices.

This means the **data travels faster** after multiplexing.

Time Division Multiplexing (WDM)

Synchronous Time-Division Multiplexing

Figure 6.13 Synchronous time-division multiplexing



Let's say we have 3 devices: A, B, and C.

- **Frame 1:** Sends A1, B1, C1
- **Frame 2:** Sends A2, B2, C2
- **Frame 3:** Sends A3, B3, C3
- Each device sends **1 unit per frame**, and this cycle keeps repeating.

Time Division Multiplexing (TDM)

Interleaving is the process of taking turns quickly when sending and receiving data from multiple sources.

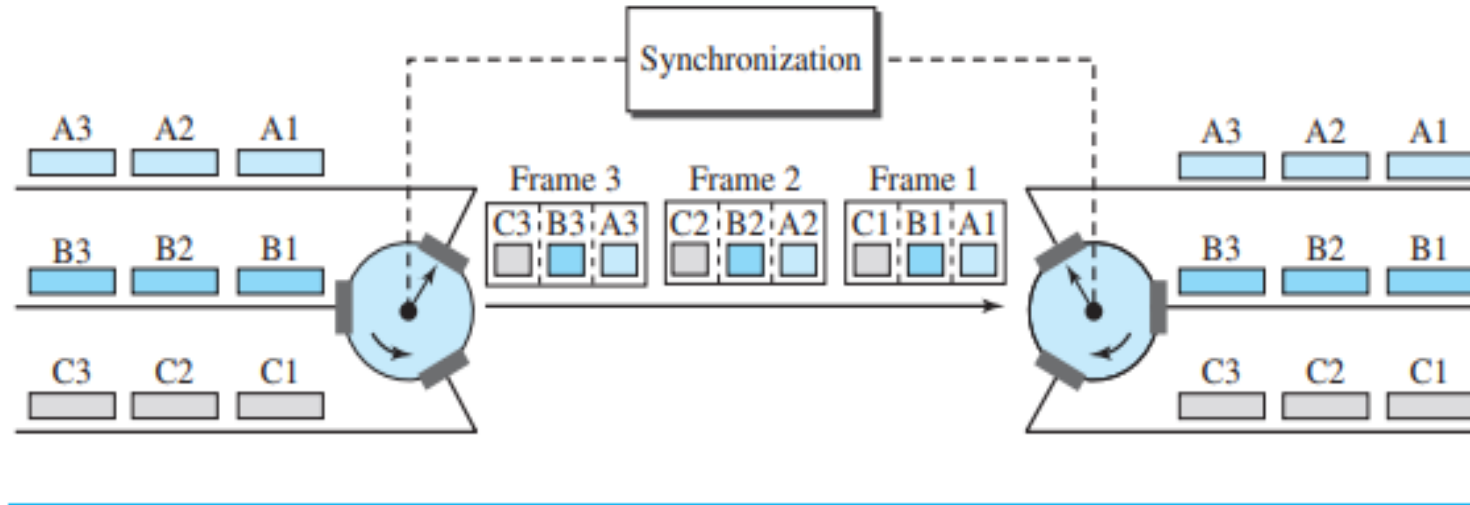
Why is it called “Interleaving”?

- Because the data from multiple sources is **mixed together like threads in a fabric**, but in a **fixed order**, so it can be separated correctly at the destination.
- Think of two spinning wheels (like fan blades) – one for sending and one for receiving.
- The sending wheel (multiplexer) quickly gives each device a chance to send its data one by one.
- The receiving wheel (demultiplexer) picks up the data and delivers it to the correct receiver in the same order.
- Both wheels spin at the same speed and are perfectly timed.

Time Division Multiplexing (WDM)

Interleaving

Figure 6.15 Interleaving

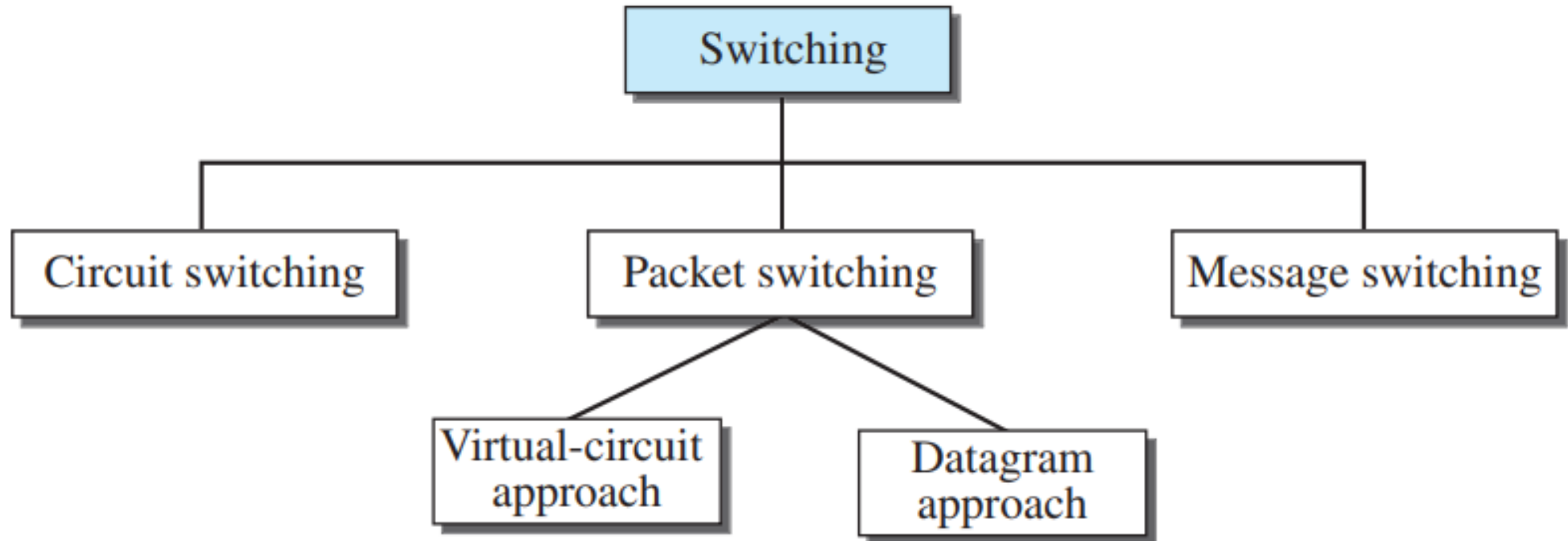


Chapter 8 Switching

8.1 Switching

- **Switching** is a technique used in communication **networks to direct data from the source to the destination** through various intermediate devices like switches or routers.
- It involves **choosing a path for data transmission across a network.**
- Switching ensures that **data can travel efficiently across shared network resources, allowing multiple devices to communicate simultaneously.**

Switching



Switching

Circuit Switching:

- **Process:** In circuit switching, a **dedicated communication path is established between the sender and receiver for the entire duration of the communication session.** This path is exclusive, and no other data can use the route until the session ends.

- **Example:** Traditional **telephone systems are a prime example.** When you make a call, a dedicated circuit is set up between your phone and the receiver's phone, and this circuit remains open until the call ends.

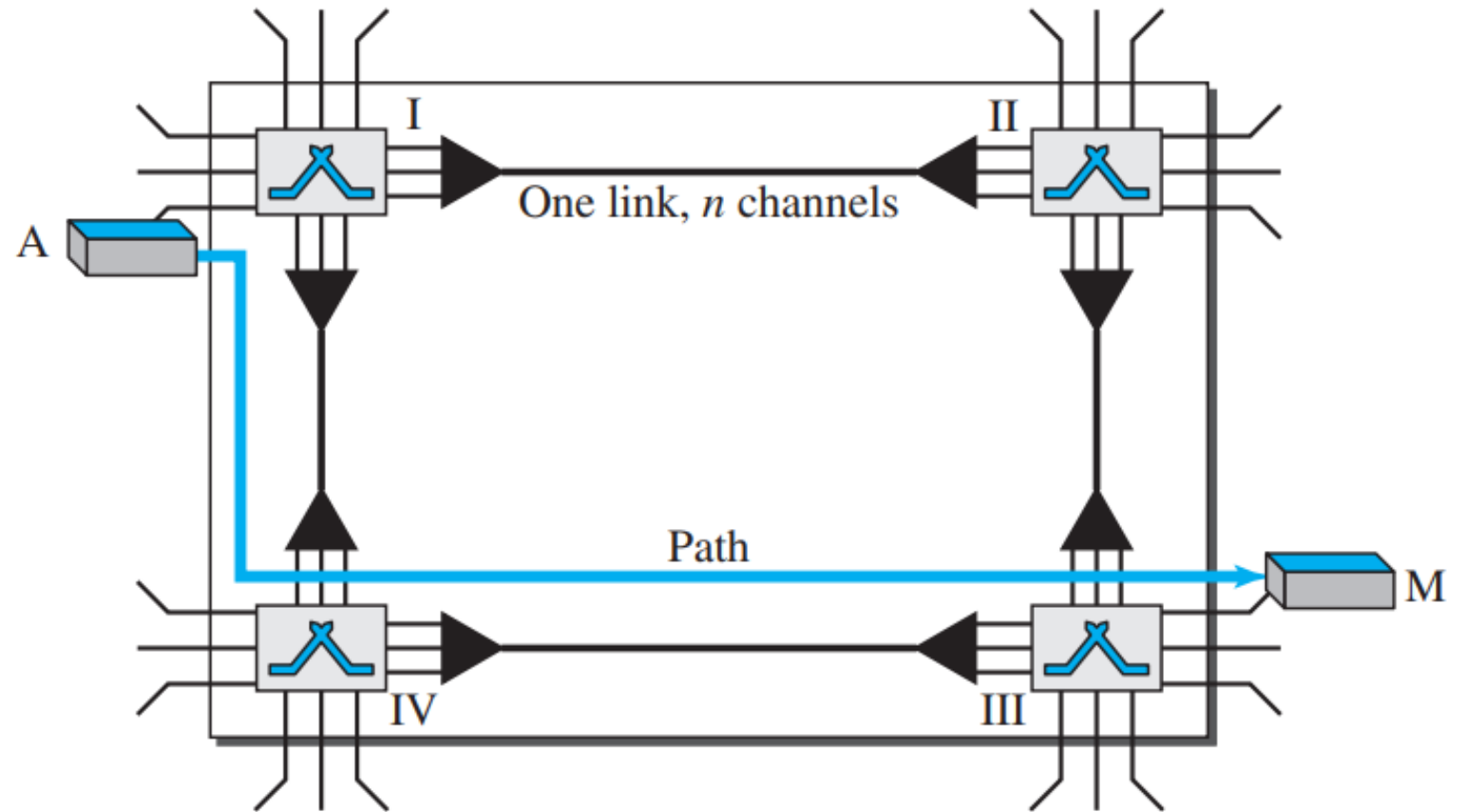
Switching

Circuit Switching:

- **Connection Setup:** A dedicated path is established between the sender and receiver before communication can begin.
- **Data Transmission:** Once the circuit is set up, data is transmitted in a continuous stream without interruption.
- **Dedicated Path:** The communication channel remains reserved for the call, ensuring a constant connection, but it's inefficient if there are periods of silence or inactivity.
- **Tear Down:** After the communication is complete, the dedicated circuit is released for other users.

Switching

Circuit Switching:



Efficiency: Circuit switching is **less efficient** due to the **dedicated path** being idle during inactivity.

Delay: It has low transmission delay but may **experience setup delay before communication starts.**

Switching

Circuit Switching:

Advantages:

- Guaranteed bandwidth: Since the path is dedicated, the full capacity of the circuit is available.
- Reliable transmission: Once the circuit is established, communication is consistent.

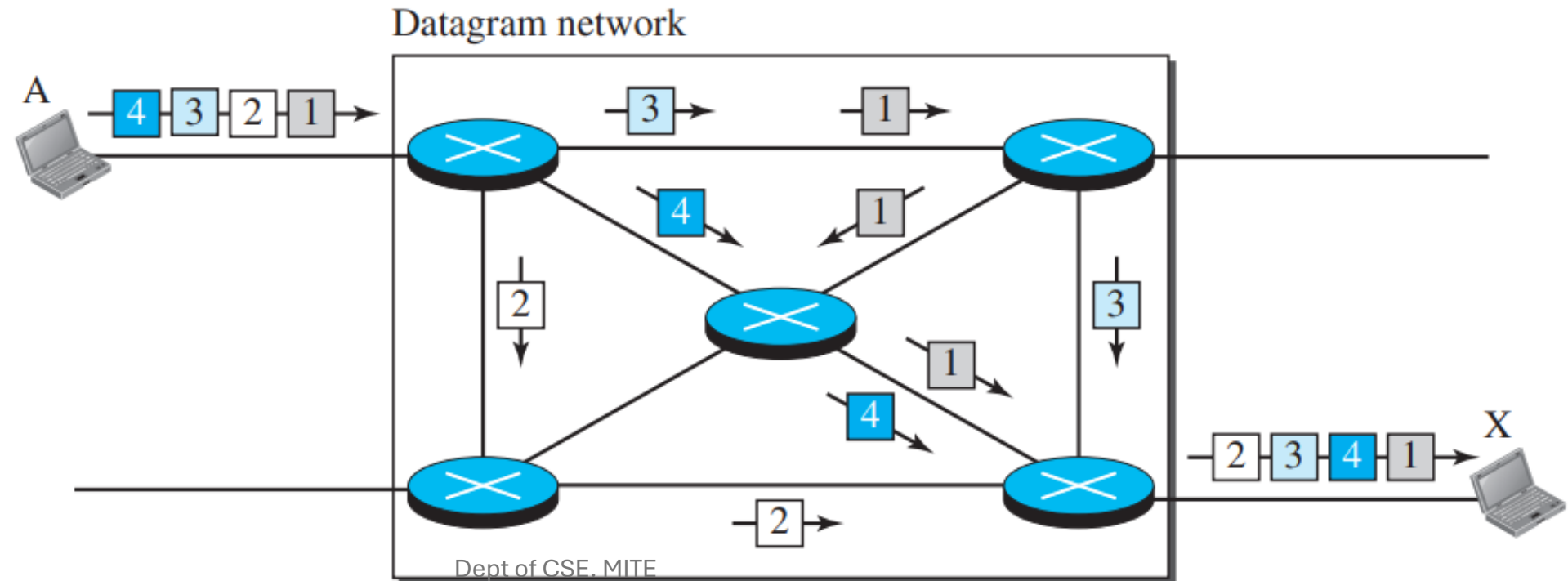
Disadvantages:

- Inefficient: The circuit remains open even during periods of silence, wasting resources.
- Delays: Establishing the circuit can take time, especially over long distances.

Switching

Packet Switching:

- **Process:** In packet switching, **data is divided into small units called packets**, which are **sent independently** over the network. **Each packet can take a different route**, and they are reassembled in the correct order at the destination.



Switching

Packet Switching:

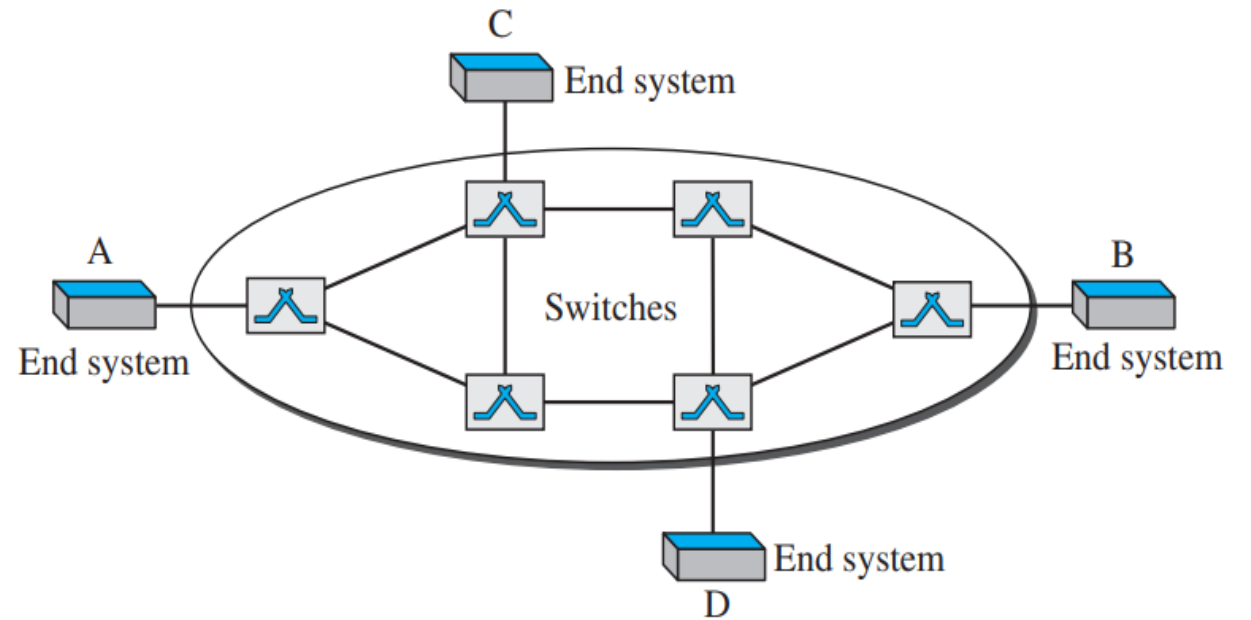
- **Two Approaches:**

- **Datagram Approach**: Each packet is treated independently, and packets can take different paths to the destination. Each packet contains the destination address, and the network routes each packet individually.
- **Virtual Circuit Approach**: Before transmitting packets, a logical path (virtual circuit) is established, and all packets follow this path. This combines aspects of both packet and circuit switching.

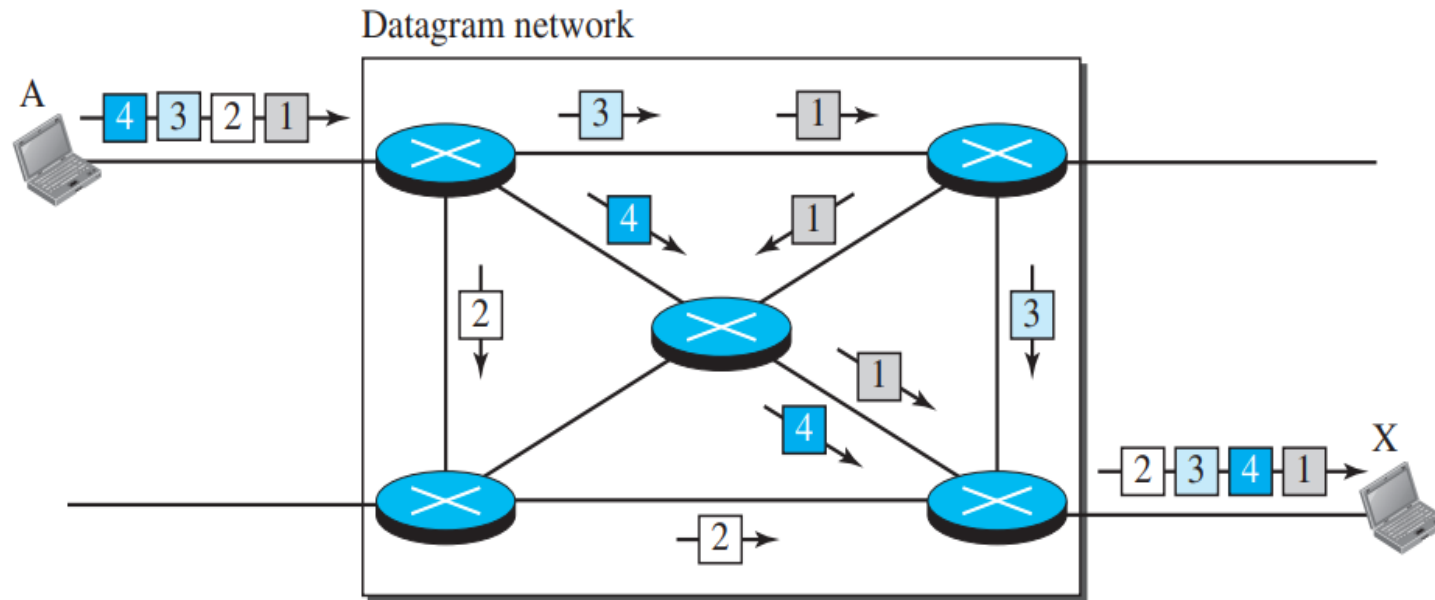
Switching

Packet Switching:

- Two Approaches:



Datagram Approach



Virtual Circuit Approach

Switching

Packet Switching:

- Two Approaches:

Aspect	Datagram Switching	Virtual Circuit Switching
Path Establishment	No dedicated path; packets routed independently	Logical path established for all packets
Order of Packets	Packets may arrive out of order	Packets arrive in correct order
Reliability	Less reliable due to variable routes	More reliable due to consistent route
Resource Allocation	Resources not reserved; packets share network resources dynamically	Logical resources reserved for the session

Switching

Packet Switching:

Example: The **Internet uses packet switching**. When you send data over the internet (like an email), it is broken into packets, sent across various routes, and reassembled at the destination.

Advantages:

- Efficient:** Resources are used only when data is transmitted, and bandwidth is shared among multiple users.
- Resilient:** If one route fails, packets can be redirected through other available routes.

Disadvantages:

- Possible delays:** Packets may arrive out of order or be delayed if network congestion occurs.
- Complexity:** Packet reassembly and routing require more complex protocols.

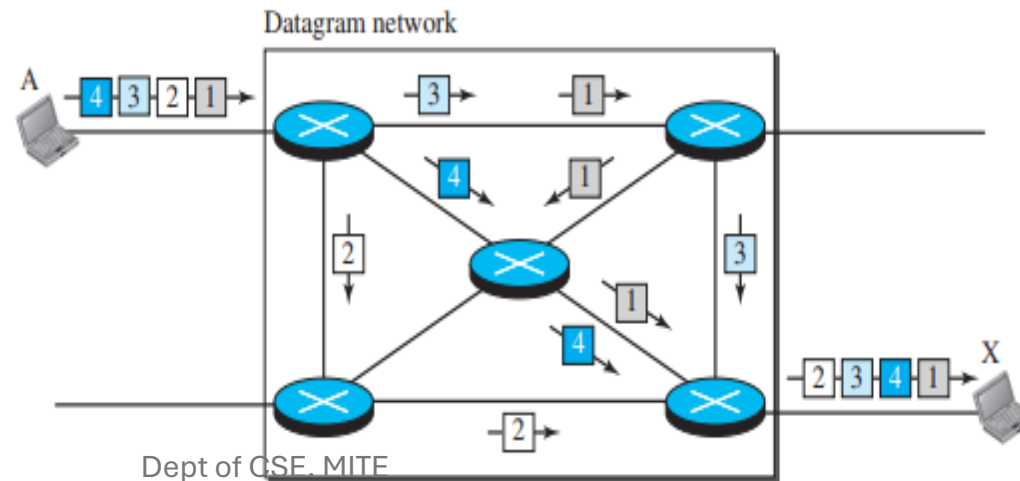
Switching

Packet Switching

Datagram Network (Connectionless Switching)

- In a **datagram network**, each packet is handled separately. Even if multiple packets are part of the same message, the network treats each one as **independent**. These packets are called **datagrams**.

Figure 8.7 A datagram network with four switches (routers)



Switching

Datagram Networks

Packets in a datagram network may **take different paths** to reach the same destination. This happens because some paths might be busy or full.

As a result:

- Packets can arrive **out of order**
- Packets may experience **different delays**
- Some packets may even get **lost or dropped**

Datagram networks are also called **connectionless networks**. This means:

- Routers don't store any information about connections.
- There is **no need to establish a connection before sending data**.
- Packets are just sent whenever needed.

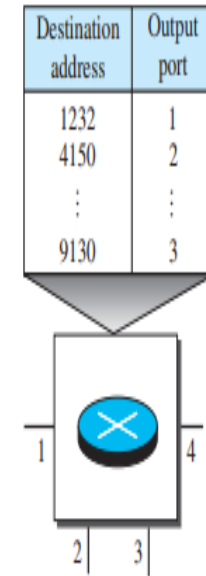
Switching

Datagram Networks

Figure 8.8 Routing table in a datagram network

Switching in a datagram network happens at the **network layer**, and the switches are called **routers**.

- Each router has a **routing table** that tells it which port to use to send a packet based on its **destination address**.
- These tables are **dynamic** and are updated often to reflect changes in the network.
- The **destination address stays the same** in the packet header throughout the journey.



Switching

Datagram Networks

Efficiency of Datagram Networks

- Datagram networks are more efficient than circuit-switched networks.
- They only use resources when data is being sent.
- If there is a delay between packets, the network can serve other users in the meantime, This saves bandwidth and makes better use of the network.

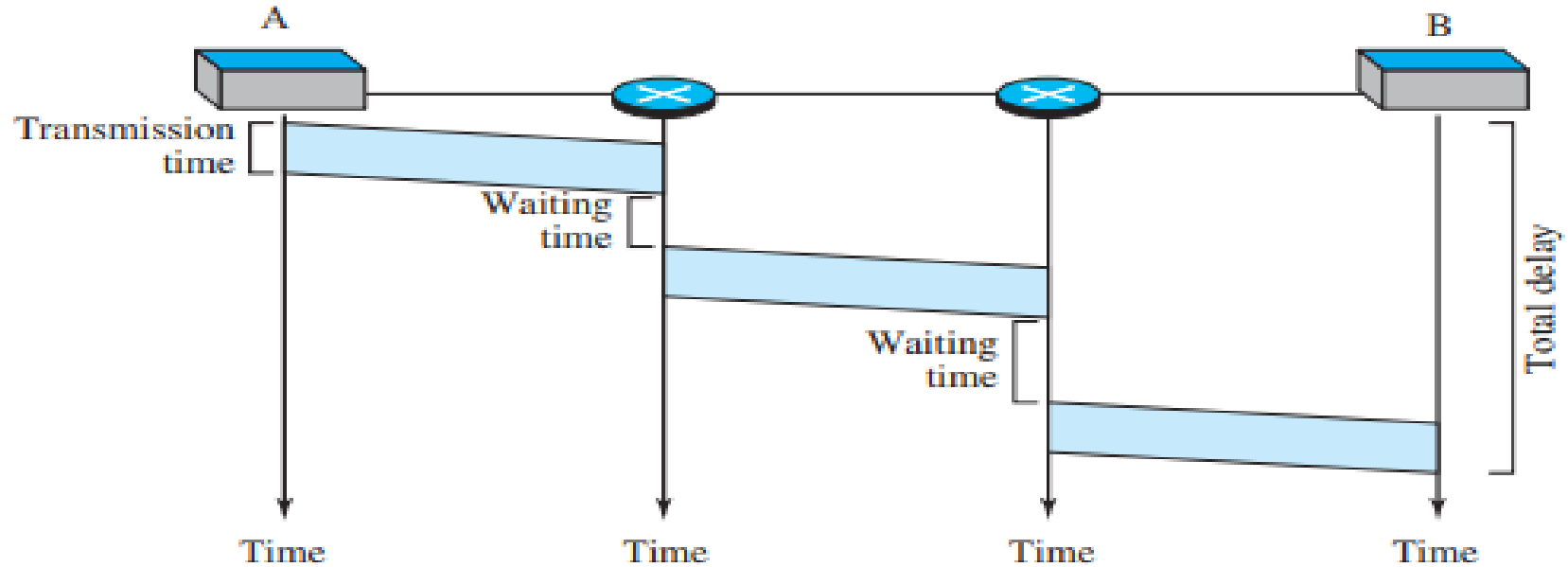
Delay in Datagram Networks

- There can be more delay in datagram networks because: Each packet might wait at a switch before being forwarded.
- Packets from the same message may travel on different paths So, the delay is not uniform.

Switching

Datagram Networks

Figure 8.9 Delay in a datagram network



The packet travels through two switches. There are three transmission times ($3T$), three propagation delays (slopes 3τ of the lines), and two waiting times ($w_1 + w_2$). We ignore the processing time in each switch. The total delay is

$$\text{Total delay} = 3T + 3\tau + w_1 + w_2$$

Switching

Virtual-Circuit Networks

A **virtual-circuit network** is a cross between a circuit-switched network and a datagram network. It has some characteristics of both.

1. As in a circuit-switched network, there are setup and teardown phases in addition to the data transfer phase.
2. Resources can be allocated during the setup phase, as in a circuit-switched network, or on demand, as in a datagram network.
3. As in a datagram network, data are packetized and each packet carries an address in the header.
4. As in a circuit-switched network, all packets follow the same path established during the connection.
5. A virtual-circuit network is normally implemented in the data-link layer, while a circuit-switched network is implemented in the physical layer and a datagram network in the network layer.

Switching

Virtual-Circuit Networks

Addressing in Virtual-Circuit Networks

Global Addressing

- Every source and destination has a global address, like a unique name in the network.
- This global address is mainly used during the setup phase to create the connection.

Virtual-Circuit Identifier (VCI)

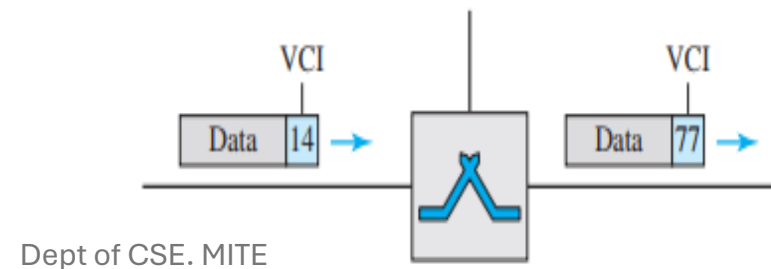
- After the connection is made, data is sent using a short address called the VCI (Virtual-Circuit Identifier).
- This VCI is local — it is used only between two switches. When a packet arrives at a switch, it has one VCI.

Switching

Virtual-Circuit Networks

- When it leaves that switch, the VCI is changed for the next link. Each switch keeps its own list of VCIs.
- So, the packet changes VCI values as it travels from one switch to another. This makes VCIs short and easy to manage because they only need to be unique at each switch.

Figure 8.11 *Virtual-circuit identifier*



Switching

Virtual-Circuit Networks

Three Phases in a Virtual-Circuit Network

- Just like in circuit-switching, a **virtual-circuit network** works in **three phases**:
- **Setup Phase**
- **Data Transfer Phase**
- **Teardown Phase**

Data Transfer Phase

- After setup is complete, data frames are sent from the source to the destination.
- Each **switch** has a **switching table** that tells it:
 - On which port the frame came in
 - What the incoming VCI is

Switching

Virtual-Circuit Networks

Data Transfer Phase

- Which port to send the frame out from.
- What new VCI to use on the outgoing port
- For example, if a frame arrives at port 1 with VCI 14, the switch looks in its table, changes the VCI to 22, and sends it out from port 3.
- Each switch **changes the VCI** as the frame moves along the path.
- This continues until **all data frames are sent**. All frames follow the same **virtual path**, though no real circuit is built.

Switching

Virtual-Circuit Networks

Figure 8.12 Switch and tables in a virtual-circuit network

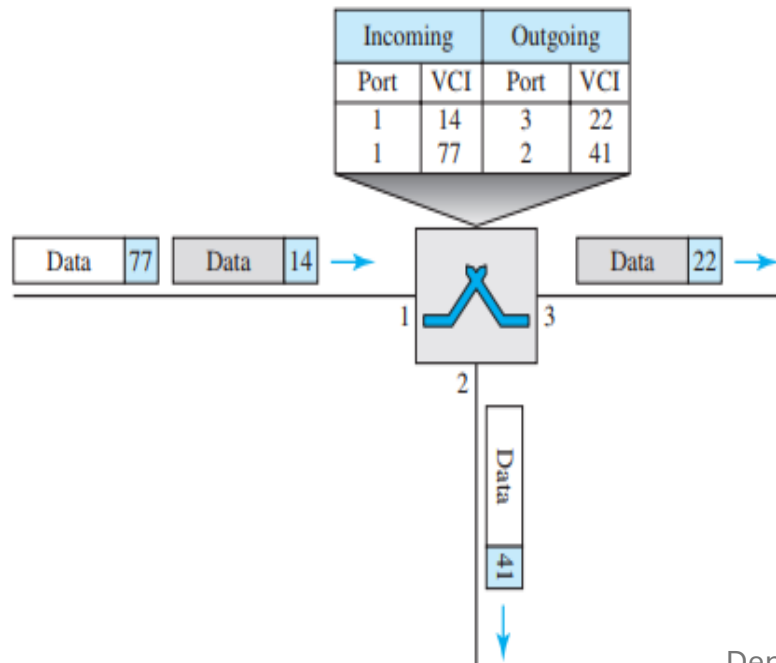
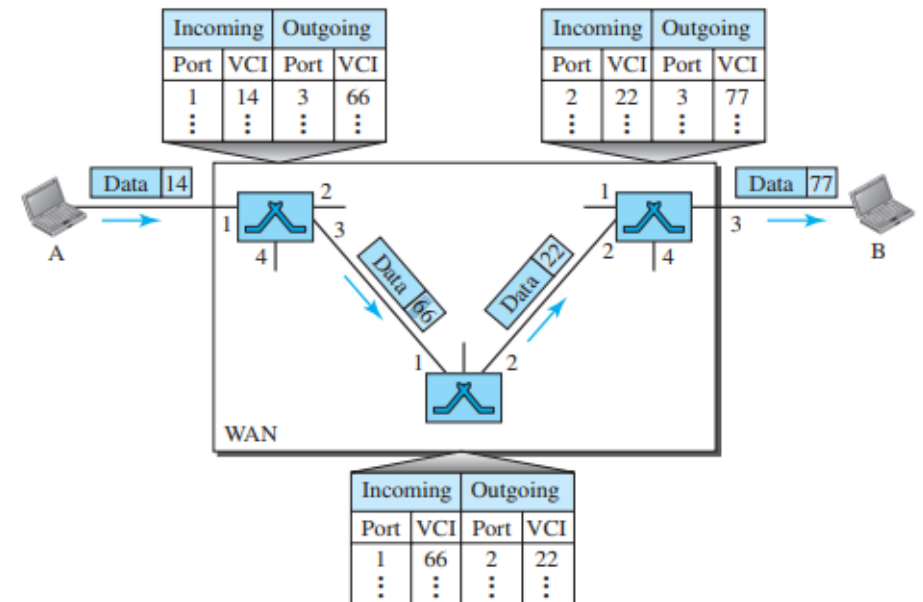


Figure 8.13 Source-to-destination data transfer in a virtual-circuit network



Switching

Virtual-Circuit Networks

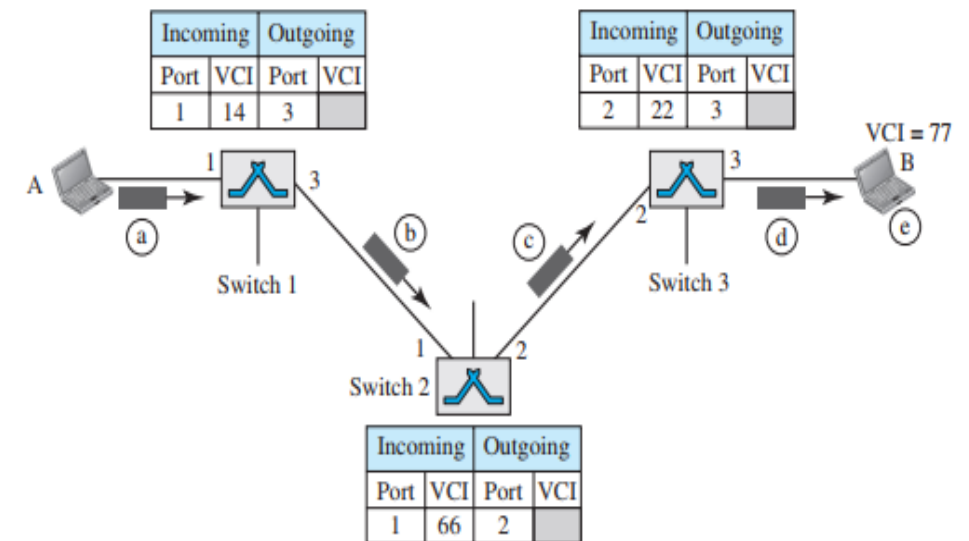
Setup Phase

- Before sending data, the source must **set up the path** to the destination.
- This happens in **two steps**:

a. Setup Request

b. Acknowledgment

Figure 8.14 Setup request in a virtual-circuit network



Switching

Virtual-Circuit Networks

- a. Source A sends a setup frame to switch 1.
- b. Switch 1 receives the setup request frame. It knows that a frame going from A to B goes out through port 3. The switch, in the setup phase, acts as a packet switch; it has a routing table which is different from the switching table.
- c. Switch 2 receives the setup request frame. The same events happen here as at switch 1; three columns of the table are completed: in this case, incoming port (1), incoming VCI (66), and outgoing port (2).
- d. Switch 3 receives the setup request frame. Again, three columns are completed: incoming port (2), incoming VCI (22), and outgoing port (3).
- e. Destination B receives the setup frame, and if it is ready to receive frames from A, it assigns a VCI to the incoming frames that come from A, in this case 77. This VCI lets the destination know that the frames come from A, and not other sources

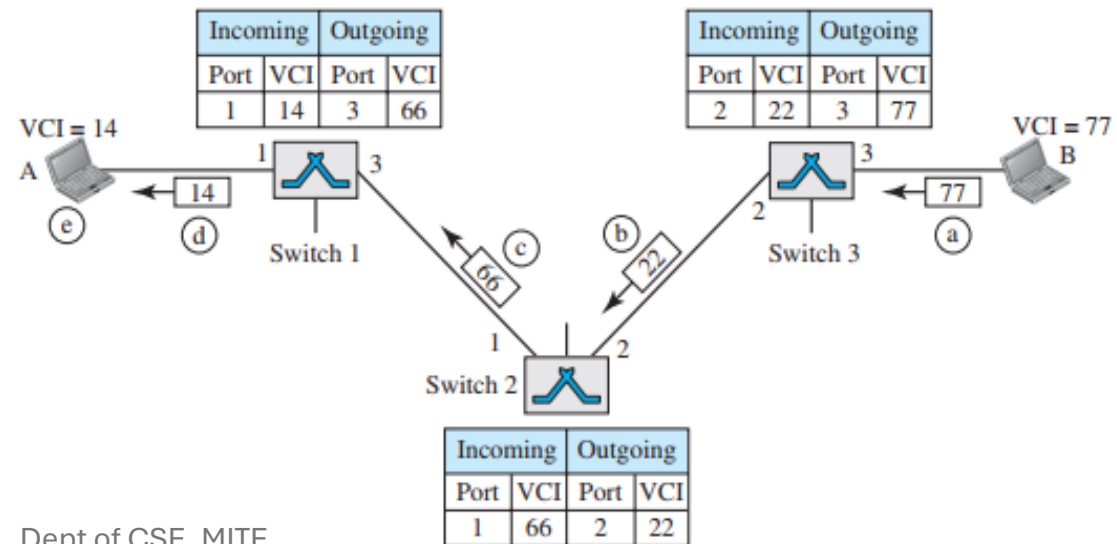
Switching

Virtual-Circuit Networks

Acknowledgment

- A special frame, called the *acknowledgment frame*, completes the entries in the switching tables

Figure 8.15 Setup acknowledgment in a virtual-circuit network



Switching

Virtual-Circuit Networks

Acknowledgment

- a) The destination sends an acknowledgment to switch 3. The acknowledgment carries the global source and destination addresses so the switch knows which entry in the table is to be completed.
- b) Switch 3 sends an acknowledgment to switch 2 that contains its incoming VCI in the table, chosen in the previous step. Switch 2 uses this as the outgoing VCI in the table.
- c) Switch 2 sends an acknowledgment to switch 1 that contains its incoming VCI in the table, chosen in the previous step. Switch 1 uses this as the outgoing VCI in the table.
- d) Finally switch 1 sends an acknowledgment to source A that contains its incoming VCI in the table, chosen in the previous step.
- e) The source uses this as the outgoing VCI for the data frames to be sent to destination B.

Switching

Virtual-Circuit Networks

Teardown Phase (Virtual-Circuit Network)

- After all data is sent, Source A sends a teardown request to end the connection.
- Destination B replies with a teardown confirmation.
- Every switch along the path deletes the virtual circuit entry from its table.
- The connection between A and B is now officially closed.

Efficiency in Virtual-Circuit Networks

- Resource reservation can be done either:
 - During setup → same delay for all packets.
 - On demand → delay may differ for each packet.
 - Even without booking (reserving), the source can still check availability, like calling a restaurant to ask about wait time.

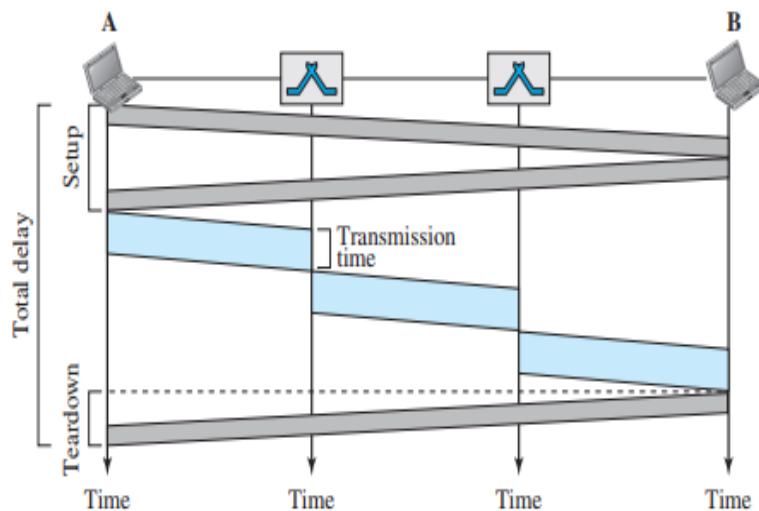
Switching

Virtual-Circuit Networks

Delay in Virtual-Circuit Networks

- There is a **one-time delay** during **setup** and **one-time delay** during **teardown**.
- If resources were already reserved during setup, then **individual packets face no wait**.
- But if resources are **allocated on demand**, packets may face **different delays**.

Figure 8.16 Delay in a virtual-circuit network



Dept of CSE, MITE

- 3 transmission times ($3T$)
- 3 propagation delays (3τ)
- Setup and teardown delays included
- **Total Delay = $3T + 3\tau + \text{setup delay} + \text{teardown delay}$**

Switching

Message Switching:

- **Process:** In message switching, the **entire message is treated as a single data unit and transmitted from the source to the destination.**

The message is stored temporarily at each intermediate node until a suitable path is available to forward it. This is often referred to as a "**store-and-forward**" system.

- **Example:** Early email systems used message switching. The entire message was stored at a server and forwarded only when network conditions allowed.

Switching

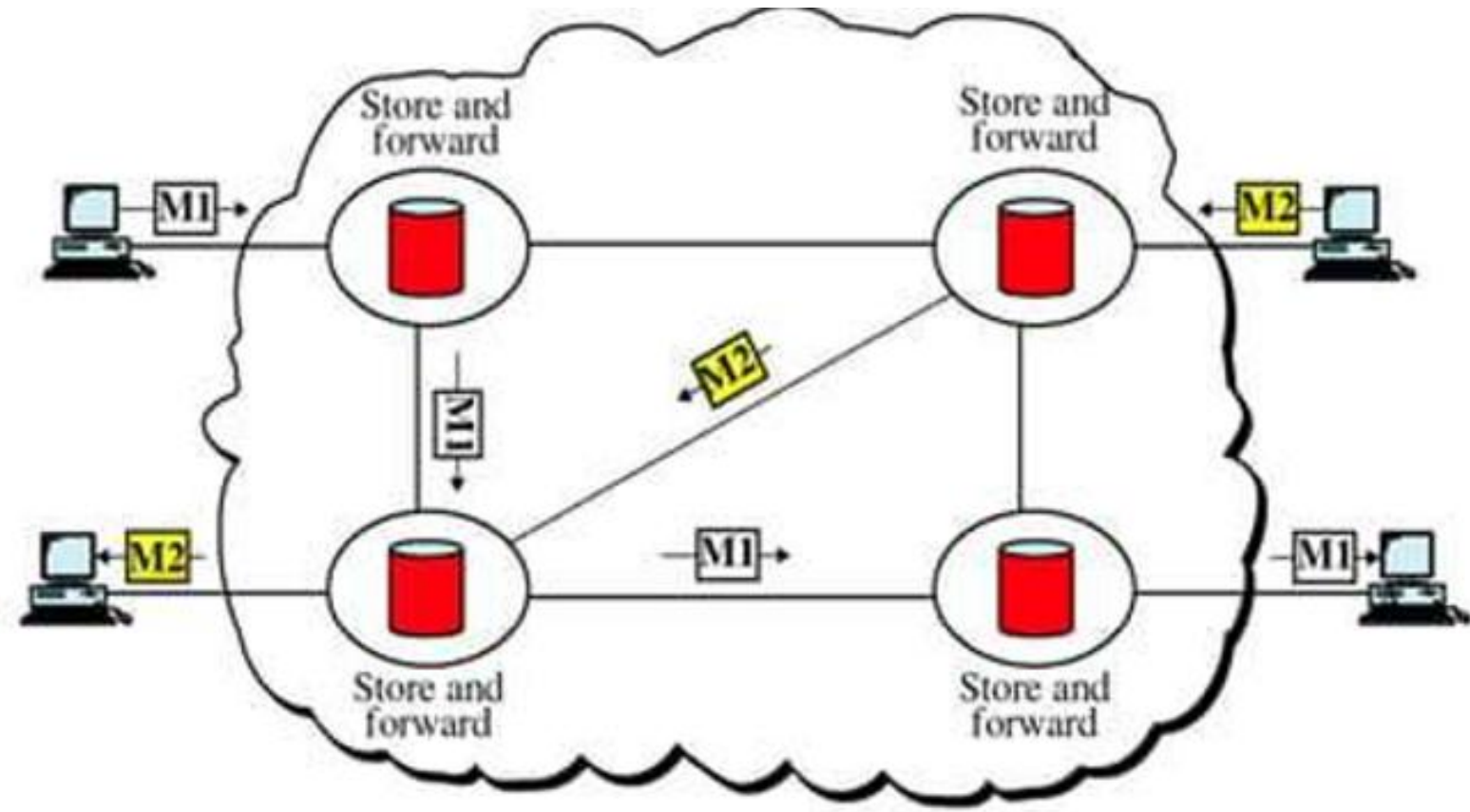
Message Switching:

Features:

- **No Dedicated Path:** Unlike circuit switching, message switching does not require a dedicated communication channel between the source and destination.
- **Variable Delays:** Since messages are stored and forwarded through multiple nodes, delays are variable and can depend on the load of intermediate nodes and the overall network traffic.
- **Efficient Use of Resources:** It can efficiently use network resources since the system doesn't require an end-to-end connection during the entire message transmission.

Switching

Message Switching:



Switching

Method	Description	Example	Pros	Cons
Circuit Switching	Dedicated path established for the entire communication session	Telephone networks	Reliable, consistent connection	Inefficient, delays in setup
Packet Switching	Data is broken into packets, sent independently through various routes	Internet	Efficient use of resources, fault-tolerant	Packets may arrive out of order, delays
Message Switching	Complete messages are stored at intermediate nodes before forwarding	Early email systems	No need for dedicated path, flexible routing	High delay, requires large storage