Module - 1
Data Communications
Computer Networks Theory
[CS-30003]
~Sankalp Nayak

Topics to be covered :

- Introduction to Computer Networks
- Analog signals and Digital Signals
- Data Transmission and Multiplexing
- Data Encoding Techniques
- Packet Switching and Circuit Switching
- Network Topologies
- Reference Models: ISO/OSI Model and TCP/IP Model.

1. DATA COMMUNICATIONS

- Data Communication is a process of exchanging data or information
- In case of computer networks this exchange is done between two devices over a transmission medium.
- This process involves a communication system which is made up of hardware and software. The hardware part involves the sender and receiver devices and the intermediate devices through which the data passes. The software part involves certain rules which specify what is to be communicated, how it is to be communicated and when. It is also called as a Protocol.
- The following sections describes the fundamental characteristics that are important for the effective working of data communication process and is followed by the components that make up a data communications system.

1.1 Characteristics of Data Communication

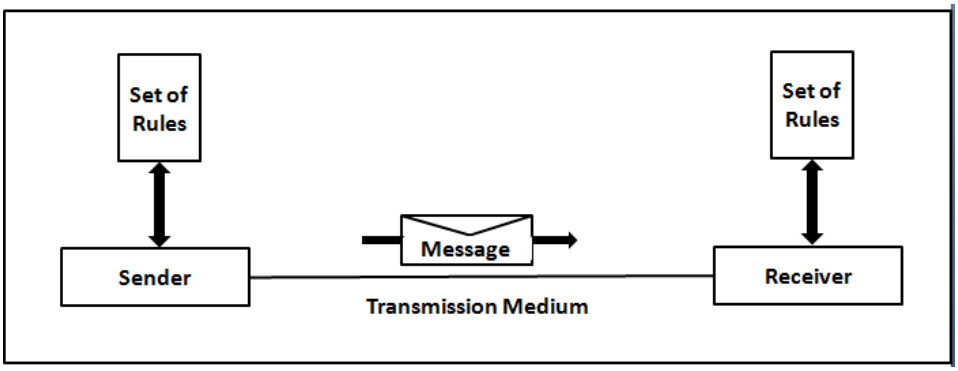
The effectiveness of any data communications system depends upon the following four fundamental characteristics:

1. **Delivery**: The data should be delivered to the correct destination and correct user.
2. **Accuracy**: The communication system should deliver the data accurately, without introducing any errors. The data may get corrupted during transmission affecting the accuracy of the delivered data.
3. **Timeliness**: Audio and Video data has to be delivered in a timely manner without any delay; such a data delivery is called real time transmission of data.
4. **Jitter**: It is the variation in the packet arrival time. Uneven Jitter may affect the timeliness of data being transmitted.

1.2 Components of Data Communication

A Data Communication system has five components as shown in the diagram below:

Fig. Components of a Data Communication System



1. **Message**

Message is the information to be communicated by the sender to the receiver.

2. **Sender**

The sender is any device that is capable of sending the data (message).

3. **Receiver**

The receiver is a device that the sender wants to communicate the data (message).

4. **Transmission Medium**

It is the path by which the message travels from sender to receiver. It can be wired or wireless and many subtypes in both.

5. **Protocol:** It is an agreed-upon set or rules used by the sender and receiver to communicate data.

- A protocol is a set of rules that governs data communication. A Protocol is a necessity in data communications without which the communicating entities are like two persons trying to talk to each other in a different language without know the other language.
- Without a protocol, 2 devices may be connected but not communicating.

1.3. COMPUTER NETWORK

Definition: A computer network can be defined as a collection of nodes, where a node can be any device capable of transmitting or receiving data. The communicating nodes are connected by communication links. A computer network should ensure the reliability of the data communication process, maintain security of the data, and optimize performance by achieving higher throughput and reducing delay times.

1) Performance

Performance can be measured using two main metrics:

- **Transit Time**: The time taken for a message to travel from one device to another.
- **Response Time**: The time elapsed between an inquiry and the response.

The performance of a network depends on several factors:

- **Number of Users**: More users can lead to network congestion, affecting performance.
- **Type of Transmission Medium**: Different mediums (e.g., wired, wireless) have varying speeds and reliabilities.
- **Efficiency of Software**: Efficient network software can improve data handling and speed.

Performance is often evaluated using two key networking metrics:

- **Throughput**: The amount of data successfully transmitted over a network in a given time period. Higher throughput indicates better performance.
- **Delay**: The time it takes for data to travel from the source to the destination. Lower delay times indicate better performance.

Good network performance is characterized by high throughput and low delay.

2) Reliability

Reliability in a network is measured by:

- **Frequency of Network Failures**: Fewer failures indicate higher reliability.
- **Time Taken to Recover from Failures**: Faster recovery times improve reliability.
- **Network Robustness in Disasters**: The network's ability to function during or after a disaster.

The fewer the failures and the quicker the recovery, the more reliable the network.

3) Security

Security refers to protecting data from unauthorized access or damage. It also involves:

- Implementing policies and measures to prevent data breaches.
- Ensuring recovery mechanisms are in place to restore data after a loss.

Security measures are essential to maintain data integrity and confidentiality.

1.4 Type of Connection

- A network is two or more devices interconnected through a communication medium. The medium provides the physical pathway between two devices. The connectivity between the devices is

classified into point-to-point and multipoint.

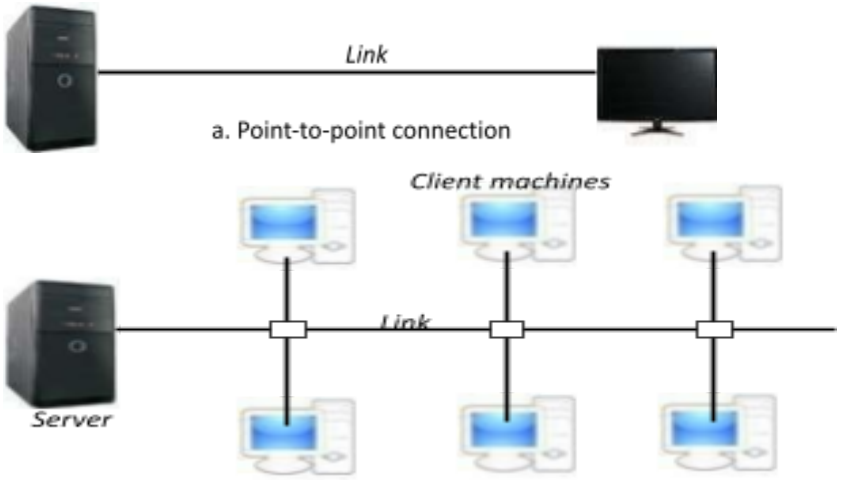
➔ **Point-to-point:** It provides a direct and dedicated link between two devices (normally source and destination). The entire transmission capacity of the link is shared for these two devices only (Fig a). For example: Point-to-Point connection b/w remote-control & TV for changing the channels. |

➔ **Multipoint:** A link is shared by many devices and the transmission capacity is shared by all devices connected (fig. b). The capacity of the channel is shared, either spatially or temporally (Figure 1.3b).

i) If the link is used simultaneously by many devices, then it is a spatially shared connection.

ii) If the user takes turns while using the link, then it is time shared (temporal) connection.

(spatially->space or temporally->time)



b. Multipoint

1.5 Network Types

Two popular types of networks:

1) LAN (Local Area Network)

2) WAN (Wide Area Network)

Network Type

Local Area Network (LAN)

- **Definition:** A LAN is a network confined to a relatively small area, such as a writing lab, school, or building.
- **Components:**
  - **Servers:** Generally not used directly by humans but run continuously to provide services like printing, software hosting, file storage, messaging, and security.
  - **Workstations:** Used by humans to interact with the network. These can include desktops, laptops, tablets, and other touch screen devices.

- **Configuration:** Servers tend to be more powerful than workstations. Workstations require appropriate storage, memory, and sometimes expensive displays based on user needs.

Wide Area Network (WAN)

- **Definition:** WANs connect networks in larger geographic areas, such as across states, countries, or continents.
- **Components:** Utilizes multiplexers, bridges, and routers to connect local and metropolitan networks to global communications networks like the Internet.
- **Applications:** Enables real-time communication between users half a world apart with workstations equipped with microphones and webcams.

Comparing Types of Network Coverage

Type	Coverage	Cost	Ownership
LAN	Single building or campus	Generally inexpensive	Typically privately owned
MAN	Single city or metropolitan area	Expensive to implement and maintain	Typically owned by private providers
WAN	Essentially unlimited geographic area	Cost varies widely	Typically owned by private providers

Network Types

Local Area Network (LAN)

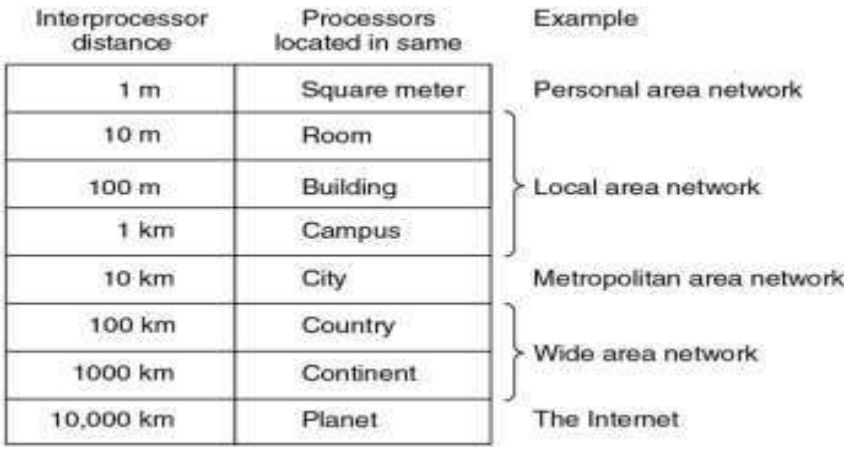
- **Usage:** Connects computers in a single office, building, or campus.
- **Ownership:** Usually privately owned.
- **Complexity:**
  - **Simple LAN:** May contain 2 PCs and a printer.
  - **Complex LAN:** Can extend throughout a company.
- **Addressing:** Each host in a LAN has a unique address.
- **Switch:** Recognizes destination addresses and guides packets to their destinations, reducing traffic and allowing multiple communications simultaneously.

Advantages of LAN:

1. **Resource Sharing:** Computer resources like printers and hard disks can be shared by all devices on the network.
2. **Expansion:** LANs can be connected to WANs to facilitate wider communication.

Wide Area Network (WAN)

- **Usage:** Connects networks over large geographic areas.
- **Applications:** Schools and businesses can communicate globally in seconds, enabling real-time collaboration and teleconferencing.



Interprocessor distance	Processors located in same	Example
1 m	Square meter	Personal area network
10 m	Room	Local area network
100 m	Building	
1 km	Campus	
10 km	City	Metropolitan area network
100 km	Country	Wide area network
1000 km	Continent	
10,000 km	Planet	The Internet

Figure: Classification of interconnected processors by scale.

Parameters	LAN	WAN
Expands to	Local Area Network	Wide Area Network
Meaning	LAN is used to connect computers in a single office, building or campus	WAN is used to connect computers in a large geographical area such as countries
Ownership of network	Private	Private or public
Range	Small: up to 10 km	Large: Beyond 100 km
Speed	High: Typically 10, 100 and 1000 Mbps	Low: Typically 1.5 Mbps
Propagation Delay	Short	Long
Cost	Low	High
Congestion	Less	More
Design & maintenance	Easy	Difficult
Fault Tolerance	More Tolerant	Less Tolerant
Media used	Twisted pair	Optical fiber or radio waves
Used for	College, Hospital	Internet
Interconnects	LAN interconnects hosts	WAN interconnects connecting devices such as switches, routers, or modems

2.1 Signals

- **Definition**: In data communication, what travels through the network at the physical layer are signals, which represent the data being transmitted.
- **Types of Signals**:
  - **Analog Signals**: Can be periodic or aperiodic. Periodic signals like sine waves are commonly used.
    - **Characteristics**:
    - **Peak Amplitude (A)**: The maximum value of the signal's intensity.
    - **Example**: If the voltage varies from 0 to 5V, the peak amplitude is 5V.

■ **Period (T) & Frequency (f)**: The time for one cycle and the number of cycles per second (measured in Hz). They are inversely related.

■ **Formula**: T = 1 / f

■ **Example**: For a signal with a frequency of 50 Hz, the period is T = 1 / 50 = 0.02 seconds.

■ **Phase (phi)**: Describes the signal's position relative to time 0, indicating a shift forward or backward.

■ **Example**: A sine wave with a phase shift of 90 degrees starts at its peak value instead of starting from zero.

■ **Wavelength (lambda)**: The distance a signal travels during one period, influenced by the signal's frequency and the propagation speed.

■ **Formula**: lambda = v / f, where v is the propagation speed of the signal.

■ **Example**: For a signal with a frequency of 3 MHz traveling at 3 x 10^8 meters per second, the wavelength is lambda = 3 x 10^8 / 3 x 10^6 = 100 meters.

○ **Digital Signals**: Represented by discrete levels of voltage (e.g., 0 and 1).

■ **Bit Rate (R)**: Measures the number of bits transmitted per second.

■ **Example**: A system transmitting 1000 bits per second has a bit rate of 1 kbps.

■ **Transmission Types**:

■ **Baseband**: Transmitting digital signals without converting them to analog.

■ **Example**: Ethernet networks typically use baseband transmission.

■ **Broadband**: Digital signals converted to analog for transmission.

■ **Example**: Cable television networks use broadband transmission.

2.2 Signal Impairment

● **Definition**: Signal quality can degrade as it travels through a medium, resulting in what is received differing from what was sent.

● **Causes**:

○ **Attenuation**: Loss of signal strength; requires amplification.

■ **Formula**: Attenuation (dB) = 10 x log10(P_in / P_out)

■ **Example**: If the input power is 100 mW and the output power is 50 mW, the attenuation is 10 x log10(100 / 50) = 3 dB.

○ **Distortion**: Changes in the signal's form, especially in composite signals with multiple frequencies.

■ **Example**: In a signal composed of frequencies 1 kHz and 2 kHz, if the 2 kHz component is delayed, distortion occurs.

○ **Noise**: Unwanted signals that interfere with the transmission. Measured using the Signal-to-Noise Ratio (SNR).

■ **Formula**: SNR (dB) = 10 x log10(Signal Power / Noise Power)

■ **Example**: If the signal power is 100 mW and noise power is 10 mW, SNR = 10 x log10(100 / 10) = 10 dB.

- **Data Rate Limits**: The speed at which data can be transmitted is limited by factors like bandwidth, signal levels, and channel quality.
  - **Nyquist Bit Rate (Noiseless Channel)**:
    - **Formula**: Bit Rate = 2 x B x log2(L), where B is the bandwidth and L is the number of signal levels.
    - **Example**: For a noiseless channel with a bandwidth of 3 kHz and 4 signal levels, the maximum bit rate is 2 x 3000 x log2(4) = 12 kbps.
  - **Shannon Capacity (Noisy Channel)**:
    - **Formula**: Capacity = B x log2(1 + SNR)
    - **Example**: For a channel with a bandwidth of 4 kHz and an SNR of 15, the capacity is 4000 x log2(1 + 15) = 16 kbps.

2.3 Digital Transmission

- **Conversion**: The process of converting data into signals for transmission.
  - **Digital-to-Digital Conversion**: Techniques include line coding, block coding, and scrambling.
    - **Line Coding Example**: Using NRZ (Non-Return-to-Zero) for binary data where 1 is represented by a positive voltage and 0 by zero voltage.
  - **Analog-to-Digital Conversion**: Techniques like Pulse-Code Modulation (PCM) and Delta Modulation (DM) are used to convert analog signals (e.g., human voice) into digital form.
    - **PCM Example**: A voice signal sampled at 8000 samples per second with 8 bits per sample has a bit rate of 64 kbps.

2.4 Analog Transmission

- **Digital-to-Analog Conversion**: Involves changing characteristics (amplitude, frequency, phase) of an analog signal based on digital data.
  - **Amplitude Shift Keying (ASK)**: Varying the amplitude.
    - **Example**: Binary data 0 and 1 are represented by 0V and 5V respectively.
  - **Frequency Shift Keying (FSK)**: Varying the frequency.
    - **Example**: Binary data 0 and 1 are represented by 1 kHz and 2 kHz signals respectively.
  - **Phase Shift Keying (PSK)**: Varying the phase.
    - **Example**: Binary data 0 and 1 are represented by 0° and 180° phase shifts.
- **Analog-to-Analog Conversion**: Used when the transmission medium is bandpass in nature.

- **Amplitude Modulation (AM)**: The carrier signal's amplitude follows the modulating signal.
  - **Example**: AM radio stations operate by varying the amplitude of a 1 MHz carrier signal to transmit audio signals.
- **Frequency Modulation (FM)**: The carrier signal's frequency follows the modulating signal.
  - **Example**: FM radio stations operate by varying the frequency of a carrier signal between 88 MHz and 108 MHz.
- **Phase Modulation (PM)**: The carrier signal's phase follows the modulating signal.
  - **Example**: In PM, the phase of a 100 MHz carrier signal is varied according to the input signal.

2.5 Multiplexing

- **Purpose**: To combine multiple signals for transmission over a single medium.
- **Frequency-Division Multiplexing (FDM)**: An analog method where signals are combined by allocating different frequency bands.
  - **Example**: In cable TV, different channels are transmitted using different frequency bands.
- **Time-Division Multiplexing (TDM)**: A digital method where signals are combined by allocating different time slots.
  - **Example**: In digital telephony, multiple voice signals are transmitted over a single channel using time slots.

2.6 Transmission Media

- **Guided Media**: Physical conduits that guide signals.
- **Twisted-Pair Cable**: Two insulated copper wires twisted together to reduce interference.
  - **Example**: Used in Ethernet networks, with categories like Cat 5, Cat 6.
- **Coaxial Cable**: A central core surrounded by insulation, used for higher frequency signals.
  - **Example**: Used in cable television and internet connections.
- **Fiber-Optic Cable**: Transmits signals as light through glass or plastic fibers, offering high-speed data transfer with minimal signal loss.
  - **Example**: Used for long-distance telecommunications and high-speed internet connections.

3. **Data Transmission and Multiplexing**

Data transmission and multiplexing are fundamental concepts in networking and communication systems. Data transmission refers to the process of sending data from one point to another, while multiplexing is a technique that allows multiple data streams to share a single communication channel.

3.1. Data Transmission

Data Transmission is the process of transferring data between two or more devices through a communication medium. This medium can be either wired (e.g., cables, fiber optics) or wireless (e.g., radio waves, microwaves).

Types of Data Transmission:

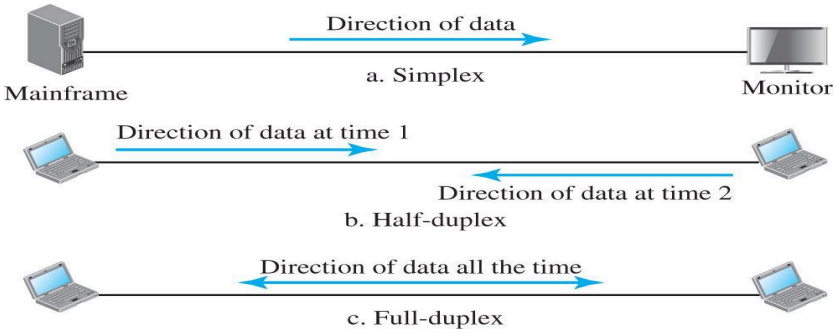
1. **Analog Transmission:**
   - **Description:** Analog transmission uses continuous signals that vary in amplitude, frequency, or phase to represent data. Common examples include traditional telephone systems and radio broadcasting.
   - **Characteristics:**
     - **Continuous Signals:** Represent data as a continuous waveform.
     - **Susceptible to Noise:** Analog signals are more prone to degradation and noise over long distances.
     - **Examples:** Voice communication over landline phones, AM/FM radio.
2. **Digital Transmission:**
   - **Description:** Digital transmission uses discrete signals, typically in binary form (0s and 1s), to represent data. It is more robust and less susceptible to noise compared to analog transmission.
   - **Characteristics:**
     - **Discrete Signals:** Data is represented in binary format.
     - **Error Detection and Correction:** Digital signals can include error detection and correction mechanisms to improve reliability.
     - **Examples:** Data transfer over the internet, digital telephony, and computer networks.

Modes of Data Transmission:

1. **Simplex:**
   - **Description:** Data transmission occurs in one direction only. One device sends data, and the other device only receives it.

- **Example:** Keyboard to computer communication.

2. **Half-Duplex:**
   - **Description:** Data transmission can occur in both directions, but not simultaneously. Devices take turns sending and receiving data.
   - **Example:** Walkie-talkies, where only one person can speak at a time.
3. **Full-Duplex:**
   - **Description:** Data transmission occurs in both directions simultaneously. Both devices can send and receive data at the same time.
   - **Example:** Telephone conversations, where both parties can speak and listen simultaneously.



Transmission Media:

1. **Wired Media:**
   - **Twisted Pair Cable:** Commonly used in telephony and Ethernet networks. It consists of pairs of wires twisted together to reduce electromagnetic interference.
   - **Coaxial Cable:** Used in cable television and broadband internet connections. It has a central conductor surrounded by insulation and shielding to prevent interference.
   - **Fiber Optic Cable:** Uses light signals to transmit data. It offers high bandwidth, long-distance transmission, and immunity to electromagnetic interference.
2. **Wireless Media:**
   - **Radio Waves:** Used in Wi-Fi, Bluetooth, and cellular networks. It provides mobility and ease of installation but is susceptible to interference.
   - **Microwaves:** Used in satellite communication and long-distance point-to-point communication. It requires line-of-sight transmission.
   - **Infrared:** Used in remote controls and short-range communication. It requires a clear line of sight and is limited to short distances.

3.2. Multiplexing

Multiplexing is a technique that combines multiple data streams into a single communication channel, maximizing the utilization of the available bandwidth. It allows

multiple signals to be transmitted over a single medium simultaneously, thereby improving efficiency.

Types of Multiplexing:

1. **Frequency-Division Multiplexing (FDM):**
   - **Description:** FDM divides the available bandwidth of a communication channel into multiple frequency bands, each carrying a separate signal.
   - **How it Works:** Each signal is modulated onto a different frequency carrier, allowing multiple signals to coexist without interference.
   - **Example:** Cable television, where different channels are transmitted on different frequencies over the same cable.
   - **Advantages:** Efficient use of bandwidth for analog signals, supports simultaneous transmission.
   - **Disadvantages:** Prone to interference and requires more bandwidth for each signal.
2. **Time-Division Multiplexing (TDM):**
   - **Description:** TDM divides the communication channel into time slots, with each signal assigned a specific time slot for transmission.
   - **How it Works:** Signals are transmitted in a round-robin fashion, with each signal getting a turn to use the channel during its assigned time slot.
   - **Example:** Digital telephony, where multiple phone calls are transmitted over the same line by interleaving their data in time slots.
   - **Advantages:** Efficient for digital signals, less interference compared to FDM.
   - **Disadvantages:** Requires synchronization between sender and receiver, delays may occur if time slots are not allocated properly.
3. **Types of TDM:**
   - **Synchronous TDM:** Fixed time slots are pre-allocated to each signal, regardless of whether data is being transmitted. This can lead to inefficient use of bandwidth if some time slots are unused.
   - **Asynchronous (or Statistical) TDM:** Time slots are dynamically allocated based on the demand of each signal. This improves bandwidth efficiency, as time slots are used only when data needs to be transmitted.
4. **Wavelength-Division Multiplexing (WDM):**
   - **Description:** WDM is a type of FDM used in fiber optic communication. It combines multiple optical signals, each with a different wavelength (color), onto a single optical fiber.
   - **How it Works:** Different data streams are transmitted simultaneously over different wavelengths of light, allowing for extremely high data rates.
   - **Example:** High-speed internet and long-distance communication over fiber optics.
   - **Advantages:** High bandwidth, supports multiple simultaneous data streams over a single fiber.

- **Disadvantages:** Expensive equipment required, complex to implement.

5. **Code-Division Multiplexing (CDM):**

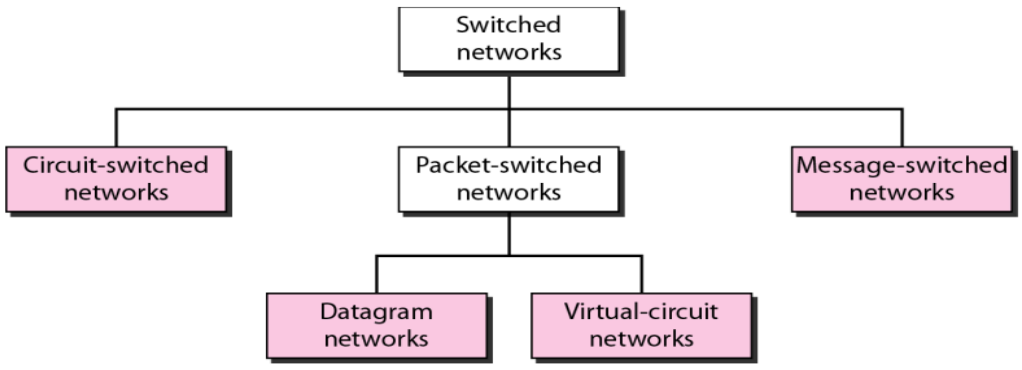
- **Description:** CDM assigns a unique code to each data signal and spreads it over the entire bandwidth available. Multiple signals can coexist on the same frequency band, distinguished by their unique codes.
- **How it Works:** Signals are combined and transmitted simultaneously. At the receiver's end, the signals are separated using their unique codes.
- **Example:** CDMA (Code Division Multiple Access) used in cellular networks.
- **Advantages:** Efficient use of bandwidth, resistant to interference, and allows multiple users to share the same channel.
- **Disadvantages:** Complex decoding process, limited by the number of unique codes available.

Conclusion

Data transmission and multiplexing are crucial for efficient communication in modern networks. Understanding different transmission methods (analog and digital) and transmission modes (simplex, half-duplex, full-duplex) lays the foundation for grasping more advanced concepts like multiplexing. Multiplexing techniques such as FDM, TDM, WDM, and CDM allow networks to handle multiple data streams simultaneously, maximizing the use of available resources and improving overall communication efficiency.

5. **Switching Techniques in Networking**

Switching is a fundamental concept in networking that involves establishing communication paths between nodes (devices) in a network. It directs data from one device to another through intermediary nodes until the data reaches its final destination. Switching techniques enable efficient data transfer and are categorized into three main types: **Circuit Switching**, **Packet Switching**, and **Message Switching**. Packet Switching can be further divided into two subcategories: **Virtual-Circuit Switching** and **Datagram Switching**.



Key Concepts

1. **Path Establishment:**

○ Nodes in the network establish a path for data by switching through directly connected neighboring nodes. Each node forwards the data to the next node until it reaches the destination.

2. **Node Switching:**

○ Each node in the network has the capability to "switch" or forward data to adjacent nodes, continuing the path to its final destination.

Types of Switching

1. Circuit Switching

- **Description:**

○ Circuit Switching establishes a dedicated communication path between two nodes for the duration of the connection. This path remains reserved exclusively for the communication session.

- **Usage:**

○ Commonly used in traditional telephone networks, where continuous communication is required.

- **Characteristics:**

○ **Reserved Bandwidth:** Bandwidth is reserved for the entire duration of the connection, ensuring a fixed data rate.

○ **Fixed Path:** A fixed path is established between the source and destination, ensuring continuous data flow.

○ **Inefficiency for Bursty Traffic:** Not suitable for data with varying traffic loads, as the dedicated path may remain underutilized.

Process of Circuit Switching:

1. **Circuit Establishment:** A connection is established before data transfer, involving setting up a path through intermediate switches.
2. **Data Transfer:** Data is transmitted continuously over the established path, which can be analog or digital.
3. **Circuit Disconnect:** After data transfer, the connection is terminated, and resources are released.

Switching Node Components:

- **Digital Switch:** Provides a full-duplex signal path between devices.
- **Network Interface:** Connects devices to the network.
- **Control Unit:** Manages connection setup, maintenance, and teardown.

Blocking vs. Non-blocking Networks:

- **Blocking Network:** May not connect all stations if all paths are in use, acceptable for voice traffic.
- **Non-blocking Network:** Allows simultaneous connections for all stations, ideal for data applications.

Technologies Used in Circuit Switching:

- **Space-Division Switching:** Uses separate physical paths for different connections. Examples include crossbar switches and multistage switches.
- **Time-Division Switching:** Uses time-division multiplexing (TDM) to share the same path among multiple connections at different time intervals.

Crossbar Switches:

- **Limitations:**
  - Number of crosspoints increases with the square of stations.
  - Costly for large switches.
  - Failure of a crosspoint disrupts connections.

Multistage Switches:

- **Advantages:** Fewer crosspoints, increased reliability, but potential for blocking.

2. Packet Switching

- **Description:**
  - Packet Switching divides data into smaller packets, each sent independently to the destination. These packets are reassembled at the destination. This method is more efficient than circuit switching and is commonly used in computer networks, including the Internet.
- **Characteristics:**
  - **Dynamic Path Selection:** Packets can take different paths based on network conditions, leading to efficient bandwidth use.
  - **Store-and-Forward Technique:** Each packet is temporarily stored at intermediate nodes before being forwarded, enabling better handling of network congestion.

Variations of Packet Switching:

1. **Virtual Circuit (Connection-Oriented):**
   - A complete route is established before sending data packets.
   - Intermediate nodes are informed of the route via a connection request packet.
   - Packets carry a virtual circuit identifier, ensuring they follow the same path.

○ **Advantages:** Packets arrive in order, making it suitable for long connections with large data transfers.

2. **Datagram (Connectionless):**

○ Each packet is treated independently without a pre-established route.

○ Packets may take different routes and can arrive out of order.

○ **Advantages:** Suitable for brief connections, quick establishment, and avoids network congestion by choosing different routes.

Switching Modes:

1. **Cut-through Mode:**

○ Fastest switching method with the lowest latency.

○ The switch forwards frames based on the destination MAC address without error checking.

2. **Store-and-Forward Mode:**

○ Entire frames are read and checked for errors using cyclic redundancy checks (CRC).

○ Ensures reliable delivery by dropping frames with errors.

3. **Fragment-Free Mode:**

○ Balances speed and reliability by checking frames minimally for validity, reducing latency.

Advantages of Packet Switching:

● **Virtual Circuit:** No separate routing is needed for each packet; packets arrive quickly and in order.

● **Datagram:** Avoids congestion and faulty nodes, and allows quick connection establishment.

3. Message Switching

● **Description:**

○ Message Switching sends the entire message from the source to the destination in one go, but it is stored and forwarded at intermediate nodes.

● **Characteristics:**

○ **Store-and-Forward Technique:** Similar to packet switching, but the entire message is stored at each node before being forwarded.

○ **No Need for a Dedicated Path:** Unlike circuit switching, message switching does not require a dedicated path.

○ **Increased Delay:** Each node must store the entire message, leading to higher delays compared to packet switching.

- **Usage:**
  - Used in early telecommunication networks and is less common in modern data networks due to high latency.

Conclusion: Switching techniques play a crucial role in efficiently managing data transmission in networks. Circuit switching is best suited for continuous communication, while packet switching is more flexible and efficient for data networks. Message switching, although less common, offers a unique approach to handling entire messages. Understanding these techniques helps in designing and managing networks effectively.

Feature	Datagram Network	Virtual Circuit Network
Connection Type	Connectionless	Connection-oriented
Path Determination	Each packet can take a different path	All packets follow the same path (pre-established)
Packet Routing	Independent routing for each packet	Fixed path defined during connection setup
Reliability	Generally less reliable, packets can be lost or out of order	More reliable, maintains order and integrity
Overhead	Lower overhead per packet (no need for connection setup)	Higher overhead due to connection setup and maintenance
Resource Reservation	No resource reservation	Resources are reserved for the duration of the connection
Use Case Examples	Internet (IP), DNS queries, streaming services	ATM networks, MPLS, traditional telephony
Congestion Handling	Dynamic, each packet can avoid congested paths	Fixed path can lead to congestion if the path becomes congested
Complexity	Simpler network management and routing	More complex due to the need for connection management
Setup Time	No setup time required	Setup time required before data transfer
Error Handling	Handled by higher layers	Built-in mechanisms for error detection and correction

6. Network Topologies

Network topology refers to the physical and logical arrangement of nodes, devices, and communication paths in a computer network. It influences how data flows and how efficiently a network operates.

Physical vs. Logical Topology

- **Physical Topology:** This represents the actual layout of the network, including the physical locations of devices and cables. It's typically depicted as a graph showing how various network components are connected.
- **Logical Topology: T**his describes the way data flows within the network, regardless of its physical design. It focuses on the path that data packets take from one device to another.

Types of Connections

- **Point-to-Point:** A direct connection between two devices. It's simple and ideal for small networks or direct communication between two nodes.
- **Point-to-Multipoint (Bus):** One device connects to multiple devices using a single communication line. This is commonly used in networks where broadcast communication is essential.

Basic Network Topologies

1. Point-to-Point Topology

- **Description:** A simple, direct connection between two nodes.
- **Advantages:** Easy to set up and maintain, ideal for small networks.
- **Disadvantages:** Limited scalability, as it's only practical for direct device-to-device communication.

2. Bus Topology

- **Description:** All devices share a single communication line (the bus). Data is sent in both directions, and all devices listen to the bus for their intended messages.
- **Characteristics:**
  - **Broadcast Communication:** Data sent from one device is broadcasted to all other devices on the bus.
  - **Connections: D**evices connect to the bus via drop lines and taps. A drop line connects the device to the bus, while a tap links to the main cable.
- **Advantages:**
  - **Cost-Effective:** Requires minimal cable, making it cheaper than other topologies like mesh or star.
  - **Easy Installation:** Straightforward to set up in small networks.
- **Disadvantages:**

- **Single Point of Failure:** A fault in the main cable can bring down the entire network.
- **Signal Degradation:** Reflections at the taps can degrade signal quality.
- **Low Security:** All devices receive the broadcasted data, which reduces security.
- **Not Suitable for Large Networks:** Limited by the number of devices and cable length.

3. Mesh Topology

- **Description:** Each device is connected to every other device, creating a network of point-to-point links.
- **Characteristics:**
  - **Dedicated Links:** Every device has a dedicated link to each other device, ensuring reliable communication.
  - **Multiple Channels:** The number of links grows rapidly with the number of devices, providing multiple paths for data.
- **Advantages:**
  - **Robustness:** If one link fails, data can still be transmitted through other links**.**
  - **Security:** Data travels on dedicated lines, making it difficult for unauthorized devices to intercept.
  - **Easy Fault Identification:** Faults are easily detected and isolated, making troubleshooting simpler.
- **Disadvantages:**
  - **High Cost:** Requires extensive cabling and hardware, leading to higher costs.
  - **Complex Installation:** Setting up and managing a mesh topology can be complicated and time-consuming.
  - **Limited Use in Traditional Networks:** More commonly used in wireless networks due to the high redundancy in wired environments.

4. Star Topology

- **Description:** All devices are connected to a central hub or switch, which manages the network.
- **Characteristics:**
  - **Central Hub:** Acts as a repeater or signal booster, enhancing communication efficiency.
  - **Point-to-Point Connections:** Each device has a direct link to the hub, facilitating easy data flow.
- **Advantages:**
  - **Easy to Install:** Simple to set up and reconfigure as devices can be added or removed without affecting the network.
  - **Robustness:** A failure in one link doesn’t affect the entire network, as the hub isolates the problem.

- **Centralized Management:** The hub manages data traffic, making it easier to control the network.

- **Disadvantages:**
- **Single Point of Failure:** If the hub fails, the entire network goes down.
- **Higher Cable Requirement:** Requires more cabling than bus or ring topologies, which can increase costs.

5. Ring Topology

- **Description: D**evices are connected in a circular arrangement, where each device is connected to two others.
- **Characteristics:**
- **Unidirectional/Bidirectional Traffic:** Data typically flows in one direction around the ring, though some configurations use dual rings for bidirectional communication.
- **Token Passing:** Data transmission is controlled by a token that circulates around the ring, ensuring orderly communication.
- **Advantages:**
- **Reduced Congestion:** Traffic is streamlined, reducing the chances of network congestion.
- **Simplified Fault Isolation:** Faults can be quickly identified, as devices raise alarms if they don't receive expected signals.
- **Disadvantages:**
- **Unidirectional Traffic:** If the ring is unidirectional, a failure in one part of the ring can disrupt the entire network.
- **Slower Performance:** Data must pass through multiple devices, which can slow down transmission rates.

6. Tree Topology

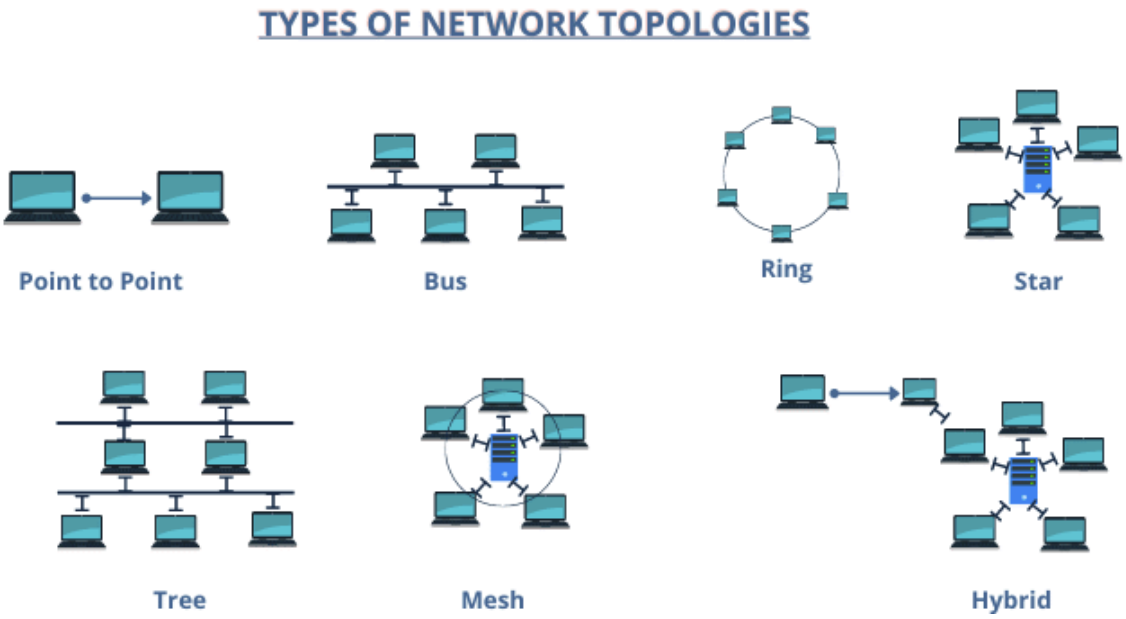
- **Description**: A hierarchical arrangement that combines star and bus topologies, forming a tree-like structure.
- **Characteristics**:
- Hierarchical Structure: The network is structured with a central root node branching out to various nodes.
- **Branching Factor**: Defines how many devices are connected at each level of the hierarchy.
- Advantages:
- Point-to-Point Wiring: Simplifies network management by allowing direct connections to the central node.
- **Scalability**: Easy to expand by adding more levels or branches.
- **Disadvantages**:

○ Complex Configuration: More difficult to set up and manage compared to simpler topologies.

○ Backbone Dependency: The network's backbone is critical; if it fails, large portions of the network can go down.

Considerations When Choosing a Topology

- **Cost:** Some topologies like bus are more cost-effective, while others like mesh may be more expensive due to cabling requirements.
- **Scalability:** Star topologies are easier to expand, making them suitable for growing networks.
- **Network Requirements:** The choice of topology should align with the specific needs of the network, including security, fault tolerance, and ease of management.



Summary: Each network topology has its own advantages and disadvantages. The choice of topology depends on factors such as cost, scalability, security, and network performance requirements. Understanding the characteristics of each topology helps in selecting the most suitable design for a given network environment.

7. NETWORK MODELS

7.1 Protocol Layering and Hierarchies

Protocol layering is a method used to manage complex communication tasks by dividing them into simpler, distinct layers. Each layer has its own protocol that manages specific aspects of the communication process, allowing for efficient and modular system design.

Key Concepts:

1. **Modularity:**

○ **Functionality:** Each layer functions independently as a "black box" with specific inputs and outputs. The internal details of how a layer processes data are hidden from other layers.

○ **Flexibility:** Layers can be updated or replaced without affecting other layers, as long as the inputs and outputs remain consistent.

2. **Advantages:**

○ **Separation of Concerns:** Different layers handle different tasks, making it easier to design, troubleshoot, and maintain the system.

○ **Selective Use:** Intermediate systems or devices may only require certain layers, rather than the entire stack, optimizing resource use.

3. **Disadvantages:**

○ **Complexity:** Layering introduces additional overhead as each layer must interact with the layers above and below it, potentially increasing system complexity.

Principles:

1. **Bidirectional Communication:**

○ **Capability:** Each layer must be able to handle communication in both directions. For example, if a layer is responsible for sending data in one direction, it should also be able to receive data in the opposite direction.

2. **Consistency:**

○ **Uniformity:** The protocol at each layer should be consistent across different devices. The data or objects processed by a layer at one end should be identical to those processed by the same layer at the other end to ensure proper communication.

Logical Connections:

● **Layer-to-Layer Communication:** Each layer establishes a logical connection through which it exchanges data with corresponding layers in other systems. This ensures that data moves efficiently through the communication stack.

Protocol Hierarchies

Networks use a hierarchical structure of layers, where each layer is responsible for specific tasks and services. This organization helps manage complex communication processes by breaking them down into manageable parts.

Key Concepts:

1. **Layers:**

○ **Function:** Networks are structured as a stack of layers, each with its own role. Each layer provides services to the layer above it and relies on the layer below it.

- **Abstraction:** Each layer abstracts the details of its operations, hiding the complexities of its implementation from other layers.

2. **Protocols:**

- **Definition:** At each layer, a protocol defines the rules for communication with the corresponding layer on another machine. These protocols specify how data is formatted, transmitted, and interpreted.
- **Peer Communication:** Layers on different machines communicate with each other using these protocols, ensuring consistency and proper data exchange.

3. **Interfaces:**

- **Interaction:** Layers communicate through well-defined interfaces. These interfaces outline how data is passed between layers, ensuring that each layer knows how to send and receive data to and from the layers directly above and below it.

4. **Communication Flow:**

- **Sending Data:** When data is transmitted, it moves down from the application layer through the intermediate layers to the physical layer, which handles the actual transmission over the network.
- **Receiving Data:** On the receiving end, data travels up from the physical layer through the intermediate layers back to the application layer, where it is processed and made usable.

Design Issues for Layers

1. **Identifying Senders and Receivers:**

- **Addressing Mechanisms:** Layers must include mechanisms to specify the source and destination of data. This ensures that data is delivered to the correct end system.

2. **Rules for Data Transfer:**

- **Data Flow Direction:** Defines how data should flow between layers, including whether it is sent or received.
- **Logical Channels:** Establishes channels for data transfer, allowing multiple data streams to be managed simultaneously.
- **Priorities:** Specifies how different data types or channels are prioritized during transmission.

3. **Error Control:**

- **Error Detection and Correction:** Layers must agree on methods for detecting and correcting errors that occur during data transmission, ensuring data integrity.

4. **Sequencing:**

- **Reassembly:** Ensures that data pieces are reassembled in the correct order at the receiving end, maintaining the proper sequence of the original message.

5. **Flow Control:**

- **Prevention of Overwhelm:** Manages the rate of data transmission to prevent a fast sender from overwhelming a slower receiver.

6. **Segmentation and Reassembly:**

○ **Handling Long Messages:** Breaks down long messages into smaller segments for transmission and reassembles them at the receiving end.

7. **Multiplexing and Demultiplexing:**

○ **Sharing Mediums:** Allows multiple data streams from different users to share the same communication medium efficiently.

8. **Routing:**

○ **Path Selection:** Determines the best path for data to travel from the source to the destination, optimizing network performance and efficiency.

Connection-Oriented and Connectionless Services

1. **Connection-Oriented:**

○ **Setup and Release:** Establishes a connection before data transfer and releases it afterward.

○ **Negotiation:** Parameters such as message size and quality of service are negotiated during setup.

○ **Analogy:** Similar to a telephone call where a connection is established and maintained during the conversation.

2. **Connectionless:**

○ **Direct Sending:** Sends data without setting up a connection first.

○ **Independent Routing:** Each message carries the destination address and is routed independently.

○ **Analogy:** Similar to mailing letters where each letter is sent without confirmation of receipt.

Relationship of Services to Protocols

1. **Services:**

○ **Layer Operations:** Define the operations provided by one layer to the layer above it. Services encapsulate the functionality that is exposed to higher layers.

2. **Protocols:**

○ **Communication Rules:** Govern the format and meaning of messages exchanged between peer entities within a layer. Protocols define how data is formatted, transmitted, and interpreted.

3. **Interaction:**

○ **Services vs. Protocols:** Services relate to the interfaces between layers, specifying what is offered to higher layers. Protocols relate to communication between peers, defining how data is exchanged within the same layer.

Network Architecture

Introduction:

- **Adaptability and Robustness:** Networks need to be flexible and reliable to meet varying demands and ensure connectivity.

Network Architectures:

- **Design Blueprints:** Provide general guidelines for designing and implementing networks, ensuring they are efficient and effective.

Layering and Protocols:

1. **Abstraction:**
   - **Encapsulation:** Each layer provides a specific interface that hides the complexity of its operations, simplifying interactions with other layers.
2. **Layering:**
   - **Building on Hardware:** Adds layers on top of basic hardware services to provide higher-level functionalities and services.
3. **Features of Layering:**
   - **Simplification and Modularity:** Enhances network design by breaking down complex tasks into manageable layers, improving modularity and maintainability.

Protocol Architecture

Major Protocol Architectures

1. **OSI Architecture**: Standard model for classifying communication functions.
2. **TCP/IP Protocol Suite**: The most widely used interoperable architecture.

While the OSI model is a theoretical framework used for understanding and designing a network architecture, the TCP/IP model is a more practical and widely used model based on the protocols around which the internet is built.

Introduction to OSI Model & Its Layers

The Open Systems Interconnection (OSI) Model was developed by the International Organization for Standardization (ISO). ISO is the organization, and OSI is the model. It was developed to allow systems with different platforms to communicate with each other. The platform could mean hardware, software, or the operating system. It is a network model that defines the protocols for network communications.

The purpose of the OSI model is to show how to facilitate communication between different systems without requiring changes to the logic of the underlying hardware and software. The OSI model is not a protocol; it is a model for understanding and designing a network architecture that is flexible, robust, and interoperable. It is a hierarchical model that groups its processes into layers. It has 7 layers as follows (from top to bottom):

1. Application Layer
2. Presentation Layer
3. Session Layer
4. Transport Layer
5. Network Layer
6. Data Link Layer
7. Physical Layer

Each layer has specific duties to perform and has to cooperate with the layers above and below it.

Layered Architecture of OSI Model

The OSI model has seven layers, each with its own dedicated task. A message sent from Device A to Device B has to pass through all layers at A from top to bottom, then all layers at B from bottom to top.

- At Device A, the message is sent from the top layer (Application Layer A), then it passes through all the layers until it reaches the Physical Layer. It is then transmitted through the transmission medium.
- At Device B, the message received by the Physical Layer passes through all its layers and moves upwards until it reaches its Application Layer.

Data Transfer Through Intermediate Nodes

As the message travels from Device A to Device B, it may pass through many intermediate nodes. These intermediate nodes usually involve only the first three layers of the OSI model.

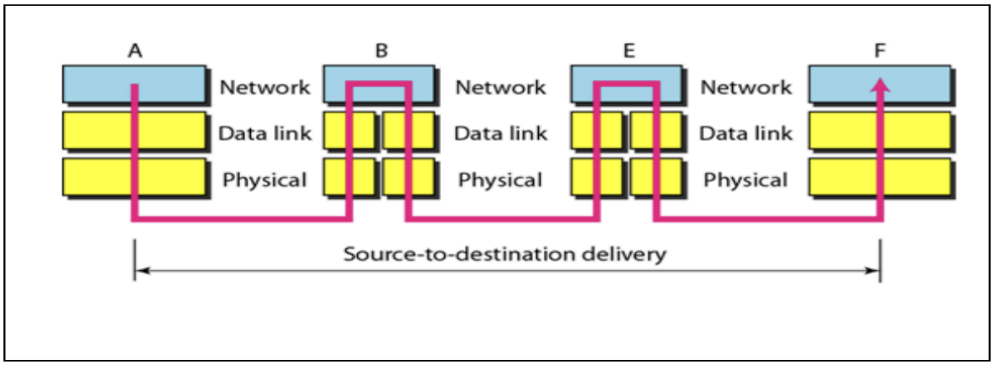


Fig: Data Transfer through Intermediate nodes

- The Data Link layer determines the next node where the message is supposed to be forwarded.
- The Network layer determines the final recipient.

Communication & Interfaces in the OSI Model

1. **Layer Communication**:
   - **Sending Device**:
     - Each layer in the sending device receives data from the layer directly above it.
     - The layer adds its own specific information (header) to the message it receives.
     - The entire package, including the added header, is passed down to the layer below.
   - **Receiving Device**:
     - Each layer in the receiving device removes the header added by its corresponding layer in the sending device.
     - The remaining data is then passed up to the layer above it.
2. **Dedicated Functions**:
   - Each layer has a unique function or service that is distinct from the other layers.
   - These functions ensure that each layer effectively contributes to the overall communication process.
3. **Service Invocation**:
   - **Sending Device**: Each layer calls upon the services offered by the layer below it to complete its tasks.
   - **Receiving Device**: Each layer relies on the services provided by the layer above it to interpret and process the received data.
4. **Peer-to-Peer Communication**:
   - Layers at corresponding levels on both the sending and receiving devices communicate directly with each other.
   - For example, Layer 2 on the sending device communicates and understands data from Layer 2 on the receiving device.
   - This type of communication is known as **peer-to-peer communication**.
5. **Encapsulation & Decapsulation Process**:
   - **Encapsulation**:
     - As data travels through the layers on the sending device:

- It starts at Layer 7 (Application Layer).
- Each layer adds its own header to the data it receives from the layer above.
- The complete package, including the added headers, is passed down to the next layer.
- This process of adding headers at each layer ensures that each layer's information is preserved and correctly transmitted.

- **Decapsulation**:
- On the receiving device:
- The process begins at the lowest layer (Physical Layer) and moves upwards.
- The corresponding layers remove the headers added by their peers in the sending device.
- The data, now stripped of its headers, is passed up to the next layer until it reaches the Application Layer.
- Decapsulation is essential for the correct interpretation and processing of the data by the receiving device.

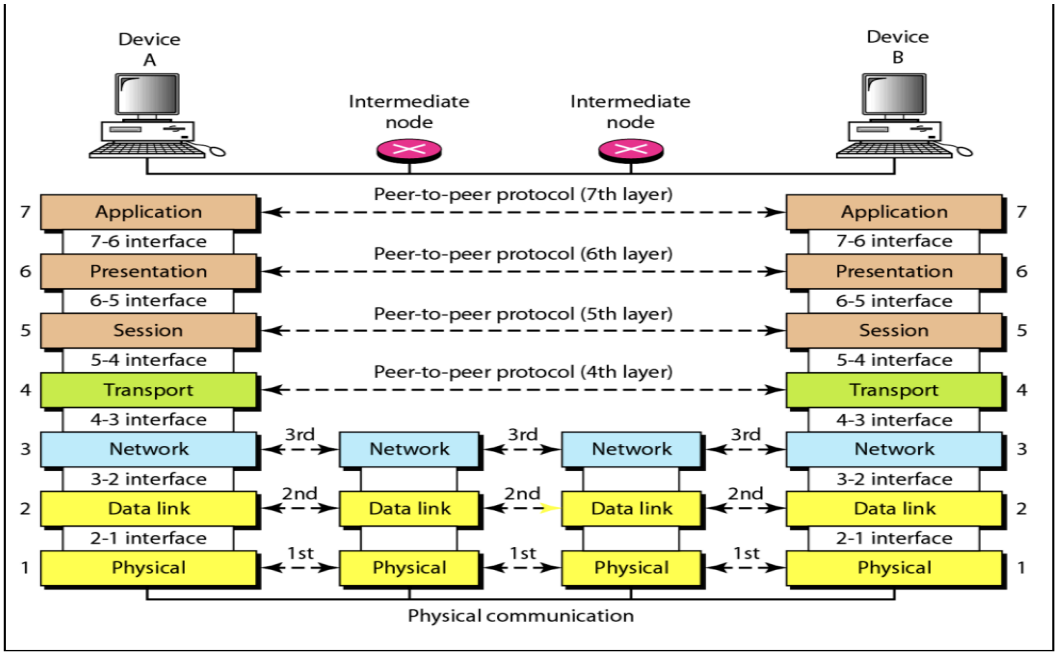


Fig: Communication & Interfaces in the OSI model

Description of Layers in the OSI Model

1. Physical Layer

- **Main Responsibility:**

The Physical Layer is responsible for the transmission of individual bits from one node to another over a physical medium.

- **Other Responsibilities:**

1. **Physical Characteristics of Interfaces and Medium:**

- Defines mechanical (e.g., cables, plugs) and electrical (e.g., modulation, signal strength, voltage levels) characteristics of the transmission medium.
- Specifies whether the transmission medium is wired or wireless.

2. **Representation of Bits:**

■ Determines the encoding method used to convert data (0s and 1s) into signals for transmission.

3. **Data Rate:**

■ Defines the transmission rate, or the number of bits sent per second.

4. **Synchronization of Bits:**

■ Ensures synchronization between the transmitter and receiver at the bit level.

5. **Line Configuration:**

■ Defines the nature of the connection:

■ **Point-to-Point:** A dedicated link between two devices.

■ **Multipoint:** A shared link among multiple devices.

6. **Physical Topology:**

■ Specifies the layout or topology (e.g., mesh, star, ring, bus) used to connect devices in the network.

7. **Transmission Mode:**

■ Defines the direction of data transfer between devices:

■ **Simplex:** One-way communication.

■ **Half-Duplex:** Two-way communication but not simultaneous.

■ **Full-Duplex:** Two-way communication simultaneously.

2. Data Link Layer

- **Main Responsibility:**

The Data Link Layer is responsible for moving frames from one node to another within the same network.

- **Other Responsibilities:**

1. **Framing:**

■ Divides the stream of bits received from the Network Layer into manageable data units called frames.

2. **Physical Addressing:**

■ Adds a header containing the physical addresses (MAC addresses) of the sender and receiver to each frame.

3. **Flow Control:**

■ Manages the data flow between sender and receiver, ensuring that the receiver is not overwhelmed by the sender.

4. **Error Control:**

■ Detects and corrects errors in the transmitted data.

■ Handles mechanisms to detect damaged, lost, or duplicate frames.

5. **Access Control:**

■ Determines which device has the right to use the communication channel in a multipoint connection.

3. Network Layer

- **Main Responsibility:**

The Network Layer is responsible for the delivery of packets from the source to the destination across multiple networks.

- **Other Responsibilities:**

1. **Logical Addressing:**

■ Adds a header containing logical addresses (IP addresses) of the sender and receiver, used to identify devices on the network.

2. **Routing:**

■ Determines the best path for packet transmission from source to destination using routers and gateways.

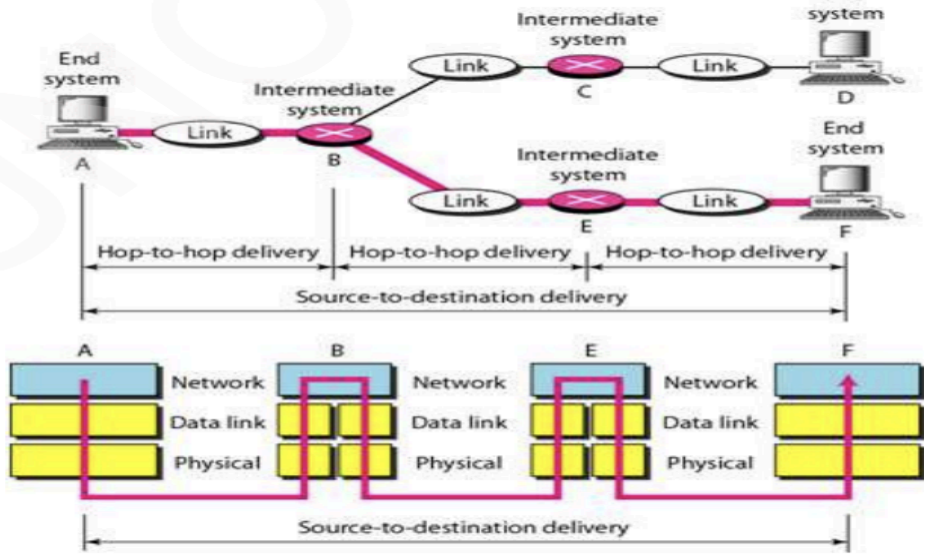


Figure 2.9 *Source-to-destination delivery*

4. Transport Layer

- **Main Responsibility:**

The Transport Layer ensures the reliable process-to-process delivery of the entire message.

- **Other Responsibilities:**

1. **Service Point Addressing:**

■ Adds a header with port numbers to ensure that the message reaches the correct process on the destination computer.

2. **Segmentation and Reassembly:**

■ Divides the message into segments, each with a sequence number for reassembly and error detection at the destination.

3. **Connection Control:**

■ Can be either:

■ **Connectionless:** Treats each segment independently.

■ **Connection-Oriented:** Establishes a connection before transmitting data.

4. **Flow Control and Error Control:**

■ Manages end-to-end flow control and error control, ensuring that data is transmitted accurately and in order.

5. Session Layer

- **Main Responsibility:**

The Session Layer manages and controls the dialog between two systems.

- **Other Responsibilities:**

1. **Dialog Control:**
   - Manages communication sessions in either half-duplex or full-duplex mode.
2. **Synchronization:**
   - Adds synchronization points, called checkpoints, to manage data transfer and ensure that only unsent data is retransmitted in case of failure.

6. Presentation Layer

- **Main Responsibility:**

The Presentation Layer handles the syntax and semantics of the information exchanged between two systems.

- **Other Responsibilities:**

1. **Translation:**
   - Converts data between the format required by the network and the format understood by the computer.
2. **Encryption and Decryption:**
   - Encrypts data at the sender's side and decrypts it at the receiver's side to ensure data security.
3. **Compression:**
   - Reduces the size of data for efficient transmission, particularly important for multimedia data.

7. Application Layer

- **Main Responsibility:**

The Application Layer provides services directly to the user and enables user interaction with the network.

Summary

The OSI model is a comprehensive framework for understanding and designing network architecture. Each of the seven layers has specific functions and responsibilities, ensuring robust, flexible, and interoperable network communication. Understanding each layer's role is crucial for developing and managing effective network systems.

TCP/IP Model

The TCP/IP (Transmission Control Protocol/Internet Protocol) model is a set of protocols that dictate how data should be transmitted over the internet. It was developed by the U.S. Department of Defense in the 1970s and serves as the foundation for all modern

internet communications. Unlike the OSI model, which has seven layers, the TCP/IP model has four layers. Each layer is responsible for specific tasks and communicates with the layers above and below it.

1. Application Layer

Main Responsibility:

- The Application Layer is responsible for providing end-user services and enabling applications to access network services. This layer is where interaction with the network begins for most users.

Functions and Protocols:

- **Functions:**
  - It provides various services like file transfer, email, remote login, and network management.
  - It facilitates data exchange between software applications and the underlying network.
  - The Application Layer also ensures that the applications work with the correct data format and provide user-friendly interfaces.
- **Protocols:**
  - **HTTP (Hypertext Transfer Protocol):** Used for transmitting web pages.
  - **FTP (File Transfer Protocol):** Used for file transfers between computers.
  - **SMTP (Simple Mail Transfer Protocol):** Used for sending emails.
  - **DNS (Domain Name System):** Translates domain names to IP addresses.

2. Transport Layer

Main Responsibility:

- The Transport Layer is responsible for end-to-end communication between devices. It ensures that data is delivered error-free, in sequence, and with no losses or duplications.

Functions and Protocols:

- **Functions:**
  - Provides reliability by establishing, maintaining, and terminating connections between devices.
  - It performs error detection and recovery, flow control, and segmentation of data into smaller units.
  - The Transport Layer can work in a connection-oriented or connectionless manner.

- **Protocols:**
  - **TCP (Transmission Control Protocol):** A connection-oriented protocol that ensures reliable data transfer. It establishes a connection before data is sent and guarantees that all data arrives at its destination.
  - **UDP (User Datagram Protocol):** A connectionless protocol that sends data without establishing a connection. It is faster but less reliable than TCP, often used in applications like streaming or gaming where speed is prioritized over reliability.

3. Internet Layer

Main Responsibility:

- The Internet Layer, also known as the Network Layer, is responsible for routing data packets across the network. It determines the best path for data to travel from the source to the destination.

Functions and Protocols:

- **Functions:**
  - The primary function is to handle the movement of packets across networks.
  - It manages logical addressing (IP addresses) and ensures that each packet is directed to its correct destination.
  - The Internet Layer handles packet fragmentation and reassembly.
- **Protocols:**
  - **IP (Internet Protocol):** The core protocol of the Internet Layer, responsible for logical addressing and routing of packets. There are two versions: IPv4 and IPv6.
  - **ICMP (Internet Control Message Protocol):** Used for sending error messages and operational information. For example, the "ping" command uses ICMP to test network connectivity.
  - **ARP (Address Resolution Protocol):** Resolves IP addresses to MAC addresses, enabling communication within a local network.

4. Network Access Layer

Main Responsibility:

- The Network Access Layer, also known as the Link Layer, is responsible for the physical transmission of data over a network medium. It manages the hardware aspects of networking, including the hardware addresses and the physical medium of transmission.

Functions and Protocols:

- **Functions:**
  - This layer is responsible for defining how data is physically transmitted across the network.
  - It involves framing, physical addressing (MAC addresses), and media access control.
  - The Network Access Layer ensures that data is placed onto the network medium and successfully received by the next device in the path.
- **Protocols:**
  - **Ethernet:** The most widely used protocol for local area networks (LANs), defining wiring and signaling for the physical layer.
  - **PPP (Point-to-Point Protocol):** Used for direct communication between two network nodes, often used in serial connections.
  - **Wi-Fi:** A wireless protocol that allows devices to connect to a network without physical cables.

Key Differences between OSI and TCP/IP Models:

- **Layers:** OSI has 7 layers, while TCP/IP has 4.
- **Flexibility:** TCP/IP is more flexible and scalable, allowing for easier integration of new protocols.
- **Development:** TCP/IP was developed first, specifically for practical use in the internet, whereas OSI was a theoretical model designed later.

Conclusion:

The TCP/IP model is the backbone of internet communication. Understanding each layer's role and the protocols it employs is crucial for grasping how data is transmitted across networks. This model not only facilitates communication between devices but also ensures the reliability and efficiency of data transmission.

The table comparing the OSI and TCP/IP models:

Aspect	OSI Model	TCP/IP Model
Full Form	Open System Interconnection	Transmission Control Protocol
Development	Developed by ISO (International Standard Organization)	Developed by ARPANET (Advanced Research Project Agency Network)
Purpose	Independent standard, generic protocol for communication	Standard protocols for internet development and host connections
Transport Layer	Guarantees the delivery of packets	Does not guarantee packet delivery but is still reliable
Approach	Vertical approach	Horizontal approach
Session & Presentation Layers	Separate layers	Combined into the Application layer
Model Type	Reference model for building various networks, including TCP/IP	Implemented model derived from the OSI model
Network Layer	Provides both connection-oriented and connectionless services	Provides only connectionless service
Protocol Replacement	Protocols are hidden and easily replaceable with technology changes	Protocols cannot be easily replaced
Number of Layers	7 Layers	4 Layers
Service, Protocol, Interface Separation	Clearly defines and separates services, protocols, and interfaces; protocol-independent	No clear separation; protocol-dependent
Usage	Low usage	Highly used
Standardization	Provides standardization for devices like routers, motherboards, switches	Does not standardize devices but connects various computers