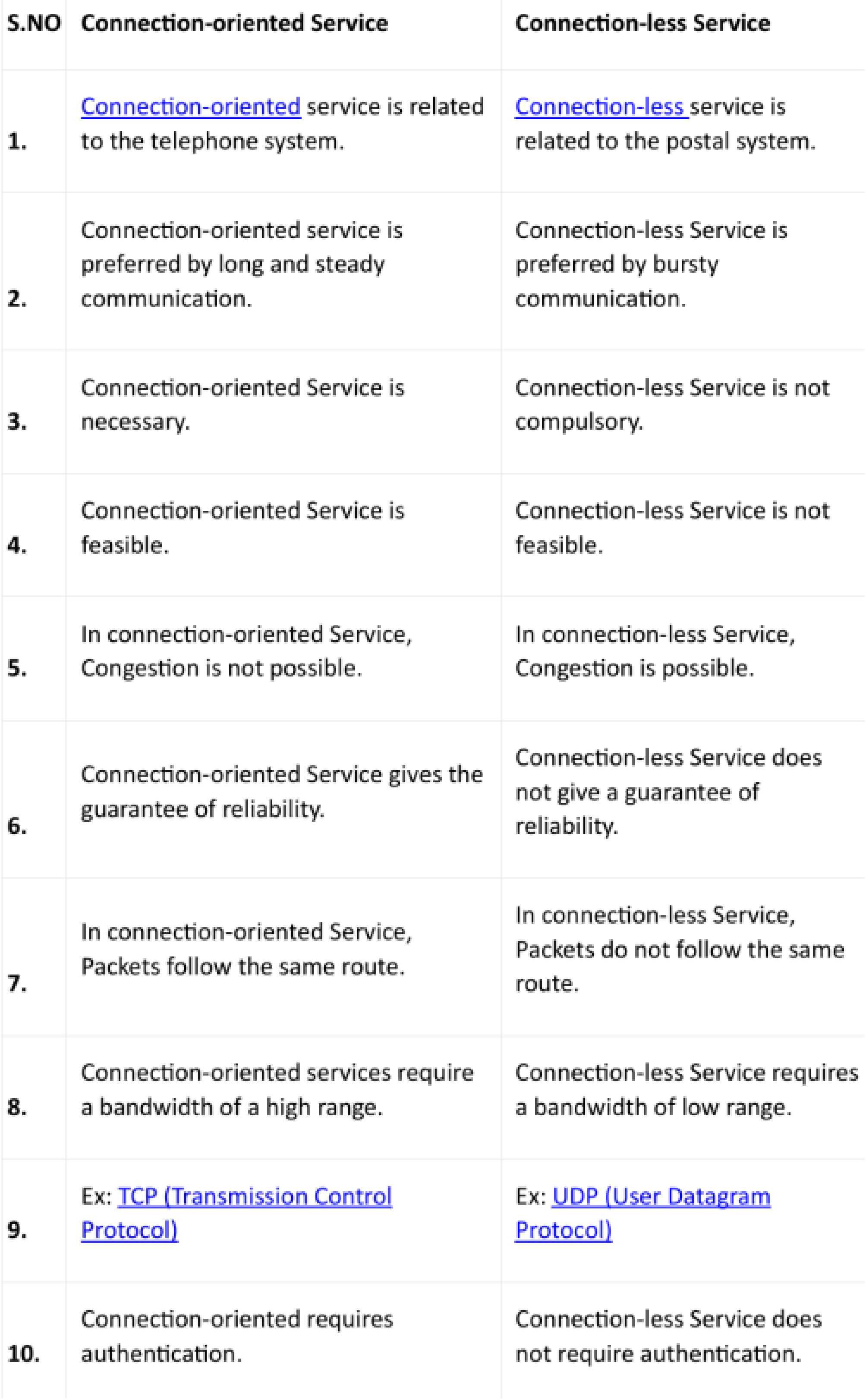
CN CAE1 & CAE 3

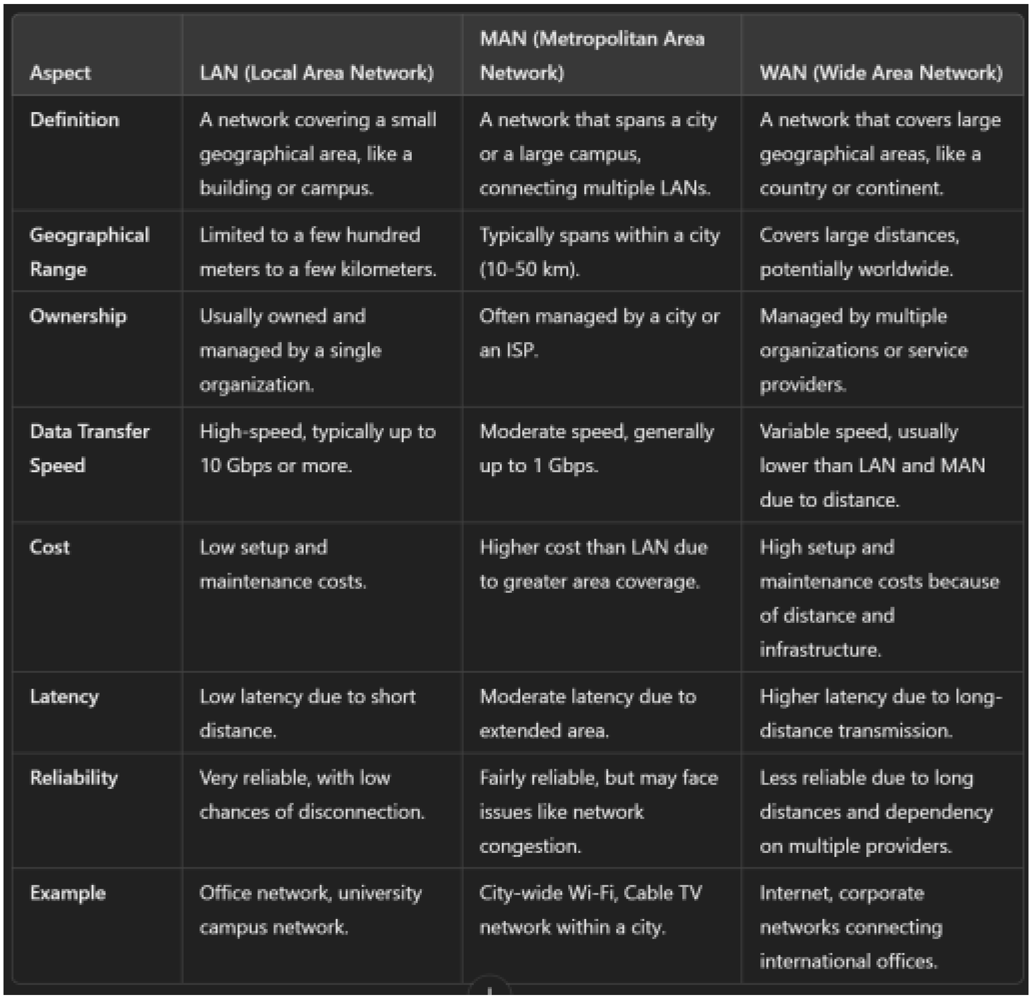
1. CO VS CS



S.NO	Connection-oriented Service	Connection-less Service
1.	Connection-oriented service is related to the telephone system.	Connection-less service is related to the postal system.
2.	Connection-oriented service is preferred by long and steady communication.	Connection-less Service is preferred by bursty communication.
3.	Connection-oriented Service is necessary.	Connection-less Service is not compulsory.
4.	Connection-oriented Service is feasible.	Connection-less Service is not feasible.
5.	In connection-oriented Service, Congestion is not possible.	In connection-less Service, Congestion is possible.
6.	Connection-oriented Service gives the guarantee of reliability.	Connection-less Service does not give a guarantee of reliability.
7.	In connection-oriented Service, Packets follow the same route.	In connection-less Service, Packets do not follow the same route.
8.	Connection-oriented services require a bandwidth of a high range.	Connection-less Service requires a bandwidth of low range.
9.	Ex: TCP (Transmission Control Protocol)	Ex: UDP (User Datagram Protocol)
10.	Connection-oriented requires authentication.	Connection-less Service does not require authentication.

2. CO AXIAL CABEL -- PDF

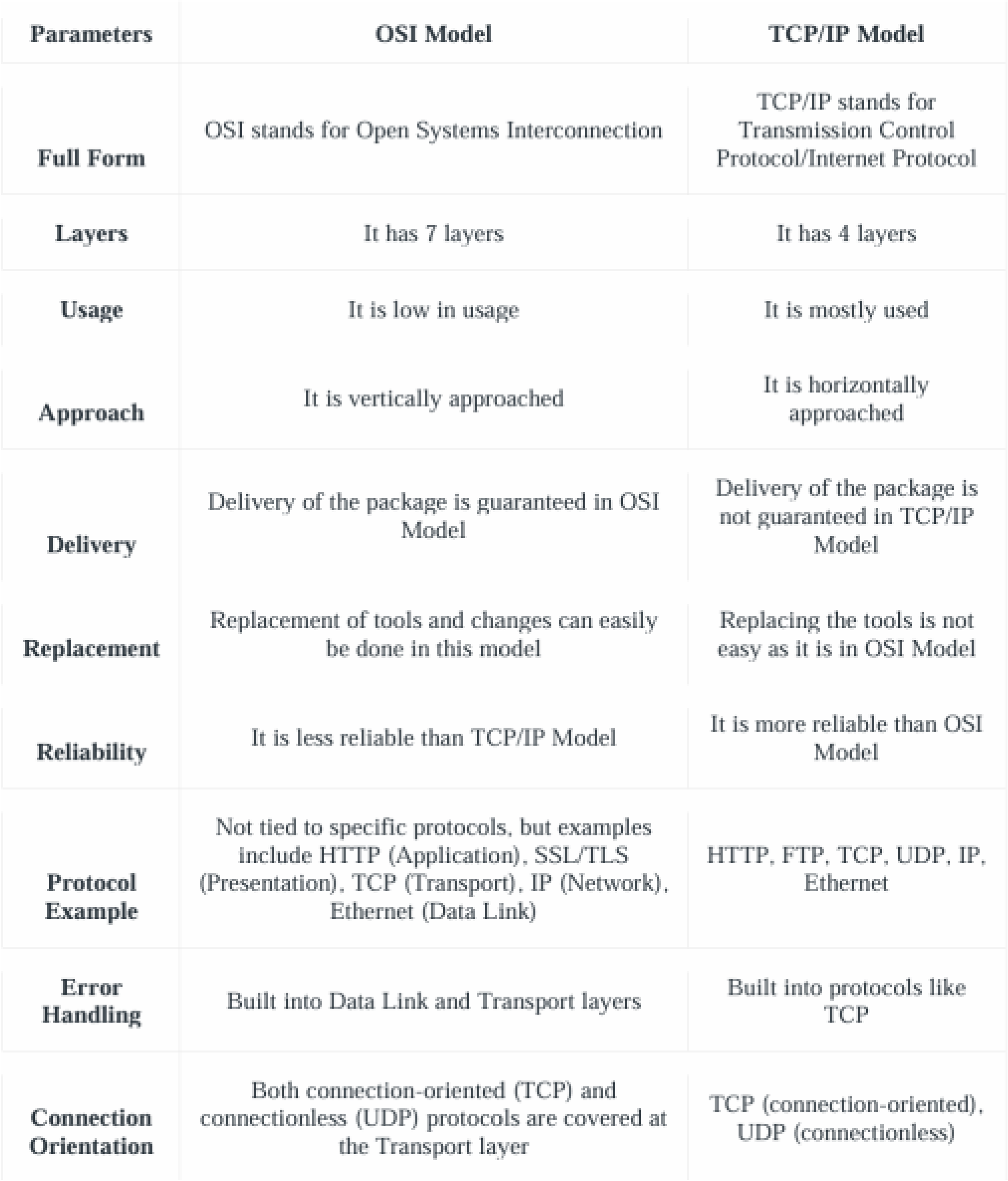
3. difff LAN MAN WAN



Aspect	LAN (Local Area Network)	MAN (Metropolitan Area Network)	WAN (Wide Area Network)
Definition	A network covering a small geographical area, like a building or campus.	A network that spans a city or a large campus, connecting multiple LANs.	A network that covers large geographical areas, like a country or continent.
Geographical Range	Limited to a few hundred meters to a few kilometers.	Typically spans within a city (10-50 km).	Covers large distances, potentially worldwide.
Ownership	Usually owned and managed by a single organization.	Often managed by a city or an ISP.	Managed by multiple organizations or service providers.
Data Transfer Speed	High-speed, typically up to 10 Gbps or more.	Moderate speed, generally up to 1 Gbps.	Variable speed, usually lower than LAN and MAN due to distance.
Cost	Low setup and maintenance costs.	Higher cost than LAN due to greater area coverage.	High setup and maintenance costs because of distance and infrastructure.
Latency	Low latency due to short distance.	Moderate latency due to extended area.	Higher latency due to long-distance transmission.
Reliability	Very reliable, with low chances of disconnection.	Fairly reliable, but may face issues like network congestion.	Less reliable due to long distances and dependency on multiple providers.
Example	Office network, university campus network.	City-wide Wi-Fi, Cable TV network within a city.	Internet, corporate networks connecting international offices.

4. OSI --> PDF

5. TCP/IP VS OSI



Parameters	OSI Model	TCP/IP Model
Full Form	OSI stands for Open Systems Interconnection	TCP/IP stands for Transmission Control Protocol/Internet Protocol
Layers	It has 7 layers	It has 4 layers
Usage	It is low in usage	It is mostly used
Approach	It is vertically approached	It is horizontally approached
Delivery	Delivery of the package is guaranteed in OSI Model	Delivery of the package is not guaranteed in TCP/IP Model
Replacement	Replacement of tools and changes can easily be done in this model	Replacing the tools is not easy as it is in OSI Model
Reliability	It is less reliable than TCP/IP Model	It is more reliable than OSI Model
Protocol Example	Not tied to specific protocols, but examples include HTTP (Application), SSL/TLS (Presentation), TCP (Transport), IP (Network), Ethernet (Data Link)	HTTP, FTP, TCP, UDP, IP, Ethernet
Error Handling	Built into Data Link and Transport layers	Built into protocols like TCP
Connection Orientation	Both connection-oriented (TCP) and connectionless (UDP) protocols are covered at the Transport layer	TCP (connection-oriented), UDP (connectionless)

6. ISSUES

1. Physical Layer

Design Issues:

- **Transmission Medium:** The physical path for data transfer, such as copper wires, fiber optics, or wireless channels. The choice affects data rate, distance, and noise immunity.
- **Bit Rate Control:** Determining the number of bits transmitted per second. This involves managing the speed of data transmission to match the capacity of the medium.
- **Synchronization:** Ensuring the sender and receiver are aligned in time, so data bits are correctly interpreted.

2. Data Link Layer

- **Framing:** Dividing data into frames for easier error detection and retransmission. Proper framing ensures data integrity and efficient handling.
- **Error Detection and Correction:** Identifying and correcting errors that occur during transmission. Techniques include parity bits, checksums, and cyclic redundancy checks (CRC).
- **Flow Control:** Managing the pace of data transmission to prevent buffer overflow at the receiver. Methods like sliding window protocols are used to regulate the flow.
- **Medium Access Control (MAC):** Determining how multiple devices share the same transmission medium. This involves collision detection (CSMA/CD) and collision avoidance (CSMA/CA) mechanisms.

3. Network Layer

Design Issues:

- **Routing:** Determining the best path for data packets to travel across networks. Routing algorithms like OSPF, BGP, and RIP are used to find optimal paths.
- **Logical Addressing:** Assigning unique addresses to devices to ensure proper delivery of packets. IP addresses (IPv4/IPv6) are used for this purpose.
- **Packet Forwarding:** Moving packets from the source to the destination across various intermediate nodes. This includes handling congestion and ensuring data integrity.

4. Transport Layer

- **Reliable Data Transfer:** Ensuring all data reaches its destination accurately and in order. Protocols like TCP handle retransmissions and sequencing.
- **Connection Management:** Establishing, maintaining, and terminating connections between devices. This includes handling connection establishment (handshake process), data transfer, and teardown.
- **Flow Control:** Regulating data flow between sender and receiver to prevent buffer overflow. Techniques like sliding window protocols are used.
- **Error Detection and Correction:** Detecting errors and recovering lost or corrupted data. This involves checksums, acknowledgments, and retransmissions.

5. Session Layer

- **Session Management:** Establishing, maintaining, and terminating sessions between applications. This includes handling session initiation, data exchange, and termination.
- **Synchronization:** Managing the flow of data and keeping track of checkpoints. This ensures that communication can resume from the last checkpoint in case of interruptions.
- **Dialog Control:** Managing the exchange of data between devices, including half-duplex and full-duplex communication. This ensures proper coordination between communicating parties.

6. Presentation Layer

- **Data Translation:** Converting data between the application layer format and the network format. This includes handling different character encoding schemes (e.g., ASCII, EBCDIC).
- **Encryption and Decryption:** Ensuring data security by encrypting data before transmission and decrypting it upon receipt. Protocols like SSL/TLS are used for secure communication.
- **Data Compression:** Reducing the size of data to improve transmission efficiency. This includes lossy (e.g., JPEG) and lossless (e.g., PNG) compression techniques.

7. Application Layer

- **Network Services to Applications:** Providing services such as file transfer, email, and remote login. This includes handling protocols like HTTP, FTP, SMTP, and DNS.
- **User Authentication:** Verifying the identity of users to ensure secure access to network resources. This includes implementing authentication mechanisms like passwords, biometrics, and tokens.
- **Data Integrity:** Ensuring the accuracy and consistency of data during transmission. This involves implementing integrity checks and error detection mechanisms.

7. Describe the need of networking and how OSI Model of networking useful for seamless communication between devices.

Need for Networking:

- **Resource Sharing:** Enables sharing of resources like printers, files, and internet connections.
- **Communication:** Facilitates communication through emails, messaging, and video conferencing.
- **Data Sharing:** Allows data exchange between different devices and users.
- **Reliability and Redundancy:** Provides backup and failover capabilities.

Usefulness of OSI Model:

- **Standardization:** Ensures interoperability between different vendors and technologies.
- **Layered Approach:** Simplifies troubleshooting and maintenance.
- **Modularity:** Allows independent development and updates of each layer.
- **Scalability:** Facilitates network expansion without major changes to the existing architecture.

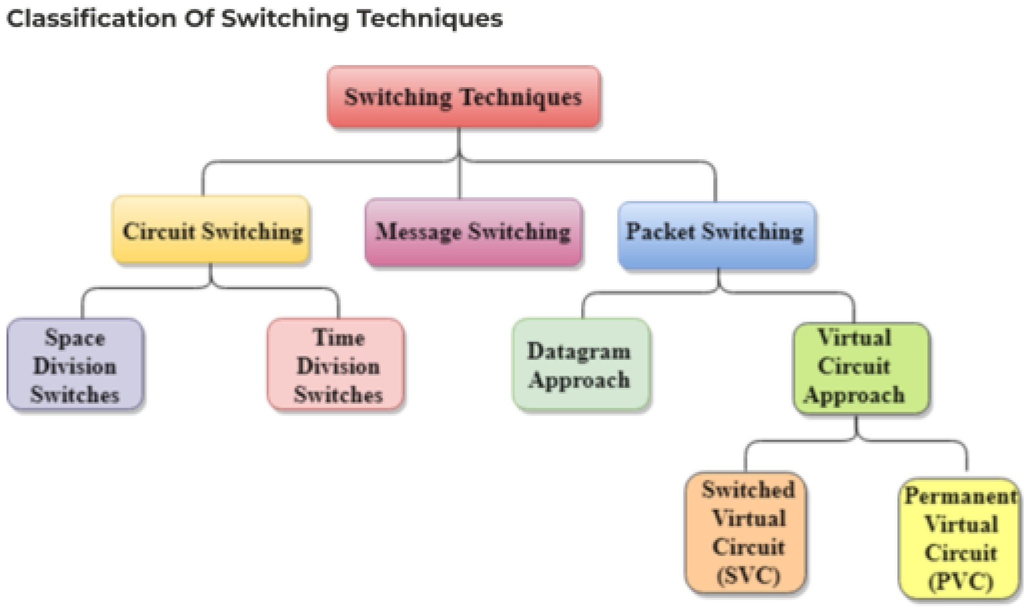
8.13. Explain switching techniques.

In large networks, there can be multiple paths from sender to receiver.

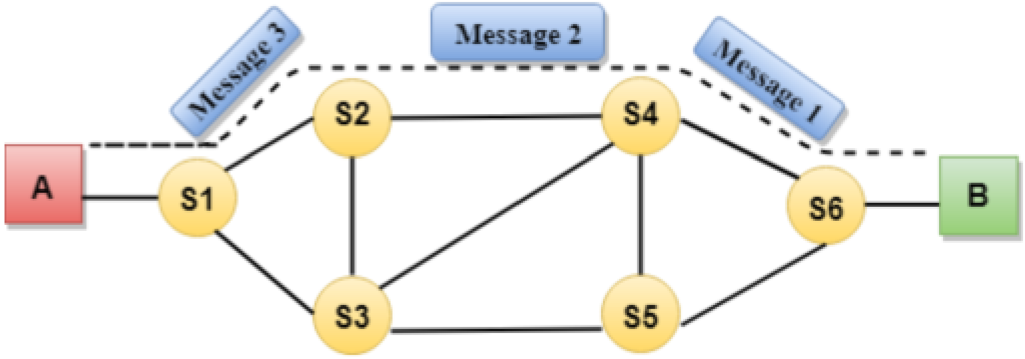
The switching technique will decide the best route for data transmission.

Switching technique is used to connect the systems for making one-to-one communication

Switching Techniques:

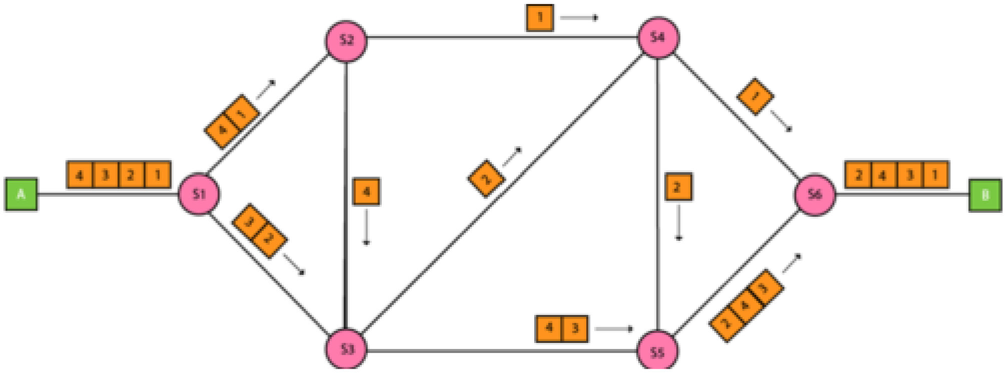


1. **Circuit Switching:**



- **Definition:** Establishes a dedicated communication path between two devices before data transfer begins. once the connection is established then the dedicated path will remain to exist until the connection is terminated
- **Phases:** Involves three phases: connection establishment, data transfer, and connection release.
- **Example:** Traditional telephone networks.
- **Advantages:** Provides a continuous and reliable connection.
- **Disadvantages:** Inefficient for data traffic with varying loads, as the dedicated path remains idle when no data is being transferred.

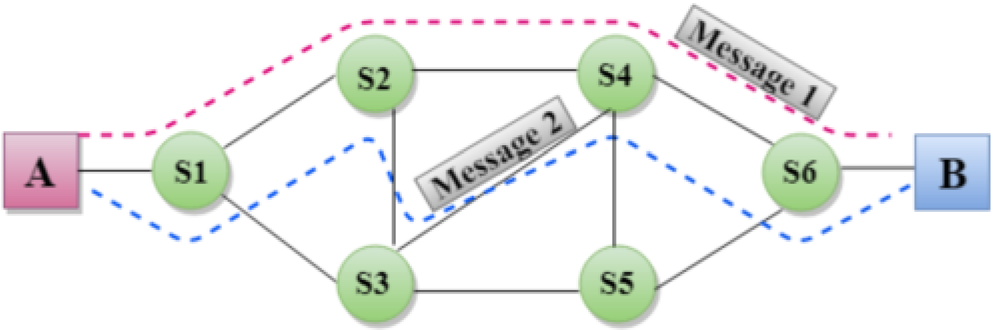
2. **Packet Switching:**



- **Definition:** Data is divided into packets and each packet is routed independently based on the destination address.
- Every packet contains some information in its headers such as source address, destination address and sequence number.
- Packets will travel across the network, taking the shortest path as possible.

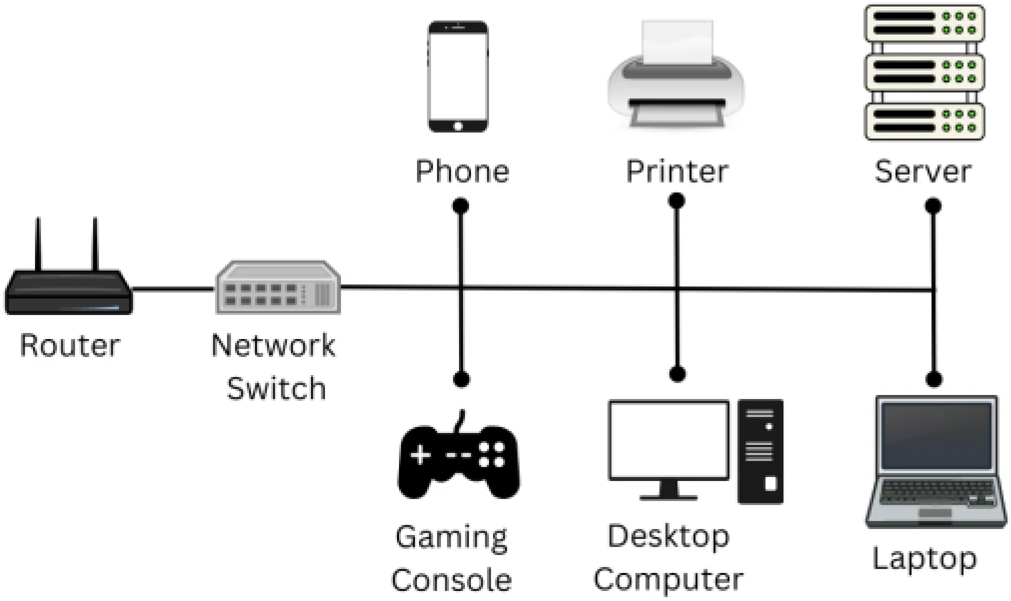
- **Types:**
    - **Datagram Packet Switching:** Each packet is treated independently and can take different routes.
    - **Virtual Circuit Packet Switching:** A pre-defined path is established, but packets are still switched.
- **Example:** Internet.
- **Advantages:** Efficient use of network resources and better handling of bursty data traffic.
- **Disadvantages:** Packets can arrive out of order and require reassembly.

3. **Message Switching:**



- **Definition:** Entire messages are routed from the source to the destination, one hop at a time.
- they can provide the most efficient routes.
- there is no establishment of a dedicated path between the sender and receiver.
- **Store-and-Forward:** Each intermediate node stores the entire message before forwarding it.
- **Example:** Email systems.
- **Advantages:** No need for a dedicated path, and it handles large messages efficiently.
- **Disadvantages:** Can introduce significant delays due to the store-and-forward mechanism.

9. Describe working of switch with the help of diagram.

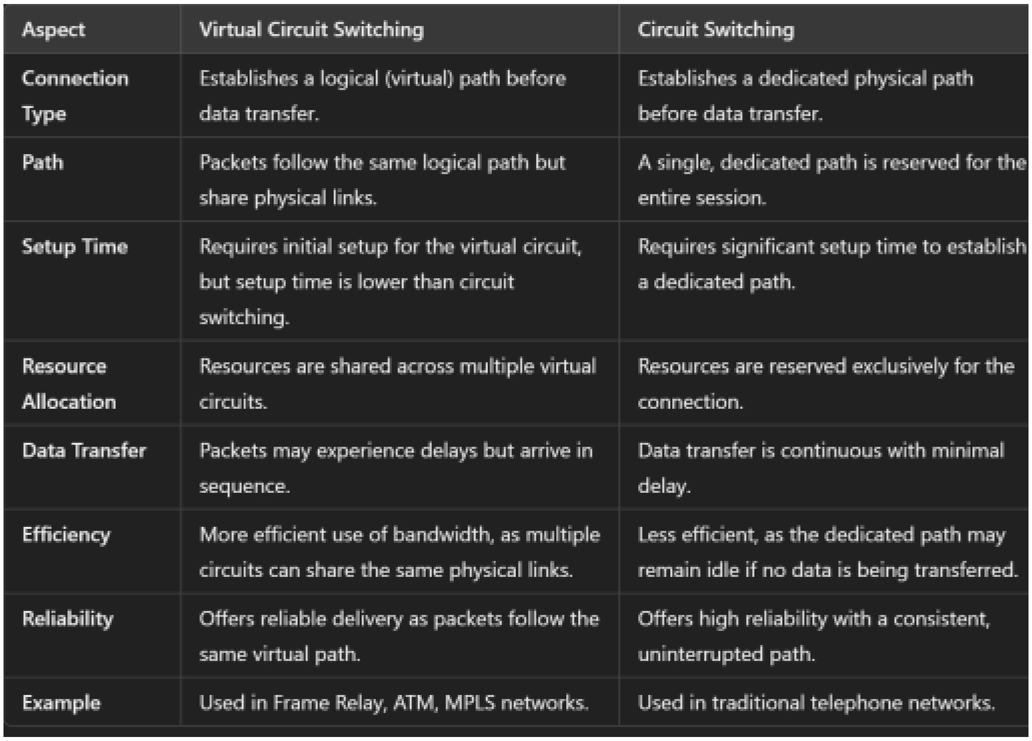


Explanation:

- **Switch Basics:** A switch is a network device that connects multiple devices and uses MAC addresses to forward data only to the intended recipient.
- **MAC Address Table:** The switch maintains a table mapping each port to the MAC addresses of the devices connected to it.
- **Packet Handling:**
    1. **Receiving a Frame:** The switch receives a data frame on one of its ports.
    2. **Learning MAC Addresses:** The switch reads the source MAC address from the frame and updates its MAC address table with the port number.
    3. **Forwarding Decision:** The switch looks up the destination MAC address in its table.
    4. **Forwarding the Frame:** If the destination MAC address is found, the frame is forwarded to the corresponding port. If not, it is broadcasted to all ports except the one it was received on.
- **Advantages:** Reduces network congestion, improves bandwidth utilization, and isolates collision domains.

10. FIBRE OPTICAL CABEL ->> PDF

11. DIFF



Aspect	Virtual Circuit Switching	Circuit Switching
Connection Type	Establishes a logical (virtual) path before data transfer.	Establishes a dedicated physical path before data transfer.
Path	Packets follow the same logical path but share physical links.	A single, dedicated path is reserved for the entire session.
Setup Time	Requires initial setup for the virtual circuit, but setup time is lower than circuit switching.	Requires significant setup time to establish a dedicated path.
Resource Allocation	Resources are shared across multiple virtual circuits.	Resources are reserved exclusively for the connection.
Data Transfer	Packets may experience delays but arrive in sequence.	Data transfer is continuous with minimal delay.
Efficiency	More efficient use of bandwidth, as multiple circuits can share the same physical links.	Less efficient, as the dedicated path may remain idle if no data is being transferred.
Reliability	Offers reliable delivery as packets follow the same virtual path.	Offers high reliability with a consistent, uninterrupted path.
Example	Used in Frame Relay, ATM, MPLS networks.	Used in traditional telephone networks.

flex	Medium flex	low flex
res util	eff	inefff
cost	low	high
quatilty	varies	constant

12. transmission media --> pdf

13. refer 8. and book

14 . pdf

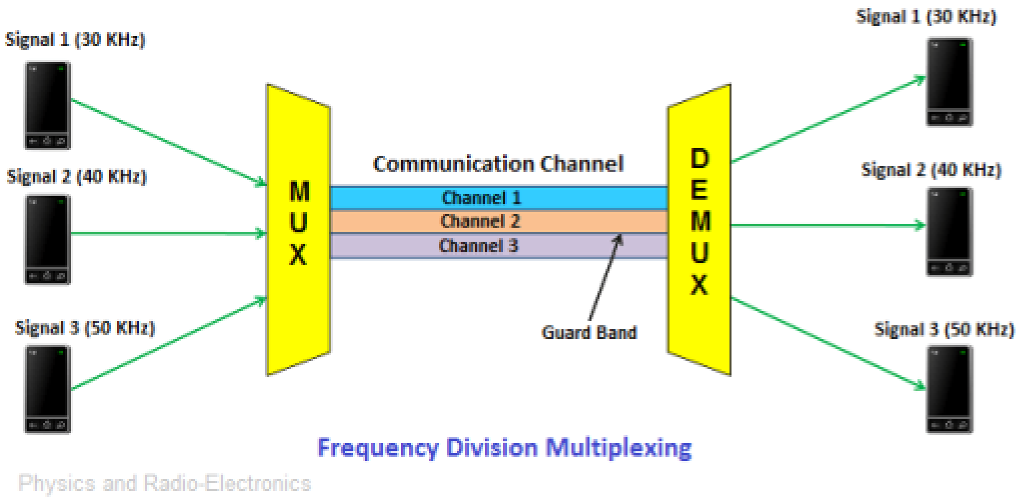
15. multiplexing

multiplexing is a technique that allows multiple signals or data streams to share the same communication channel or transmission medium.

It enables efficient utilization of bandwidth, allowing several devices or data sources to communicate over a single channel simultaneously.

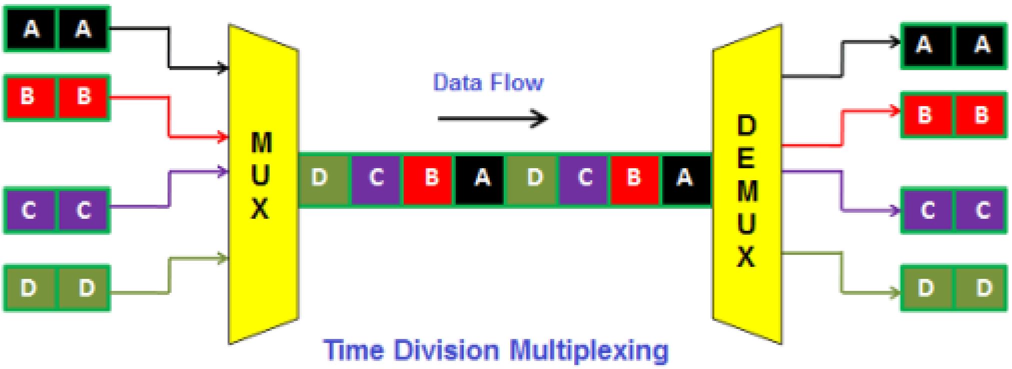
There are several types of multiplexing techniques, each suited to different kinds of data transmission requirements.

1. Frequency Division Multiplexing (FDM)



- **Concept**: In FDM, the available bandwidth of a channel is divided into multiple frequency bands, and each band is allocated to a separate data stream. Multiple signals are transmitted simultaneously but on different frequencies.
- **Application**: FDM is commonly used in radio and television broadcasting, as well as in cable TV. Each channel operates on a unique frequency.
- **Advantages**:
  - Allows simultaneous transmission of multiple signals.
  - Minimizes interference between signals due to separate frequency bands.
- **Disadvantages**:
  - Requires large bandwidth.
  - Bandwidth is fixed, so each user is allocated a specific frequency, which may be underutilized if the user is idle.

2. Time Division Multiplexing (TDM)



- **Concept**: In TDM, time is divided into slots, and each user is allocated a specific time slot in a repeating cycle. During each time slot, only one signal or data stream is transmitted, and the cycle repeats for continuous communication.
- **Types**:
    - **Synchronous TDM**: Each user is assigned a time slot whether they have data to send or not. If a user has no data, the time slot remains empty.
    - **Asynchronous TDM** (or Statistical TDM): Only active users are assigned time slots, allowing better bandwidth utilization.
- **Application**: TDM is commonly used in digital telephony and cellular networks.
- **Advantages**:
    - More efficient than FDM in terms of bandwidth usage, especially in statistical TDM.
    - Suitable for digital data transmission.
- **Disadvantages**:
    - Requires precise timing to allocate time slots accurately.
    - If many users are idle, synchronous TDM can lead to inefficient bandwidth usage.

3. Wavelength Division Multiplexing (WDM)

- **Concept**: WDM is a multiplexing technique used specifically in optical fiber communication. It combines multiple data streams, each on a separate light wavelength (or color), within a single optical fiber. Each wavelength carries an independent signal, allowing multiple signals to be transmitted simultaneously.
- **Types**:

    - **Dense Wavelength Division Multiplexing (DWDM)**: Used for high-capacity networks, with closely spaced wavelengths, allowing for more channels.
    - **Coarse Wavelength Division Multiplexing (CWDM)**: Uses wider wavelength spacing, typically for shorter-distance communication.
- **Application**: WDM is widely used in fiber-optic communication systems for long-distance data transmission.
- **Advantages**:
    - Allows very high data rates over a single optical fiber.
    - Efficient for long-distance communication.
- **Disadvantages**:
    - Requires expensive optical equipment, such as wavelength-specific lasers.
    - Requires precise control of wavelengths to avoid interference.

CAE 3

1. Transport Layer Services Provided to the Upper Layer

The **Transport Layer** is the fourth layer in the OSI model, positioned between the Network Layer and the Application Layer.
It provides essential services that facilitate reliable **communication** between application processes running on different hosts, enabling end-to-end **data transmission.**
The transport layer protocols, such as **TCP (Transmission Control Protocol)** and **UDP (User Datagram Protocol)**, are implemented in end systems but not in network routers.
These protocols provide distinct sets of services to cater to different types of network requirements.
The transport layer offers the following key services:

1. **End-to-End Delivery**:
    - Ensures the reliable delivery of entire messages from the source to the destination across the network. This service allows application processes on one host to communicate seamlessly with application processes on another host, handling the complexities of underlying network layers.
2. **Reliable Delivery**:
    - The transport layer provides mechanisms for **reliable delivery**, ensuring that data is transmitted accurately and completely. This includes:

        - **Error Control**: Ensures that packets are checked for errors and retransmitted if necessary. While the Data Link Layer checks for node-to-node errors, the Transport Layer performs end-to-end error checking.
        - **Sequence Control**: Maintains the correct order of packets during transmission, enabling accurate reassembly at the destination.
        - **Loss Control**: Uses sequence numbers to identify and retransmit lost segments.
        - **Duplication Control**: Prevents duplicate packets from reaching the destination by using sequence numbers to detect duplicates.

3. **Flow Control**:
    - Flow control prevents the sender from overwhelming the receiver by adjusting the data transmission rate according to the receiver's capacity. This helps reduce packet loss and network congestion. The **Sliding Window Protocol** is commonly used, allowing for efficient data transmission and preventing receiver buffer overflow.
4. **Multiplexing and Demultiplexing**:
    - **Multiplexing** allows multiple application processes to share a single network connection, improving transmission efficiency. It has two types:
        - **Upward Multiplexing**: Multiple transport connections use the same network path, optimizing network usage by combining transmissions bound for the same destination.
        - **Downward Multiplexing**: One transport connection uses multiple network paths to enhance throughput, useful in networks with low bandwidth.
    - **Demultiplexing**: The transport layer uses port numbers to identify different application processes and ensure each application receives its intended data.
5. **Addressing**:
    - The transport layer assigns a unique address to each application process, referred to as a **Transport Service Access Point (TSAP)**, allowing it to deliver data to the correct application on the destination host. This addressing mechanism ensures that data generated by an application on one machine is directed to the correct application on another machine.

17. Explain Flow Control and Buffering in Transport Layer

1. **Flow Control**: This mechanism ensures a balance between the data transmission speed of the sender and the receiver’s ability to process data. Key methods include:

- **Sliding Window Protocol**: Limits the amount of data a sender can send before needing an acknowledgment, dynamically adjusting the window size based on network conditions.
- **Stop-and-Wait Protocol**: The sender transmits one packet and waits for an acknowledgment before sending the next.

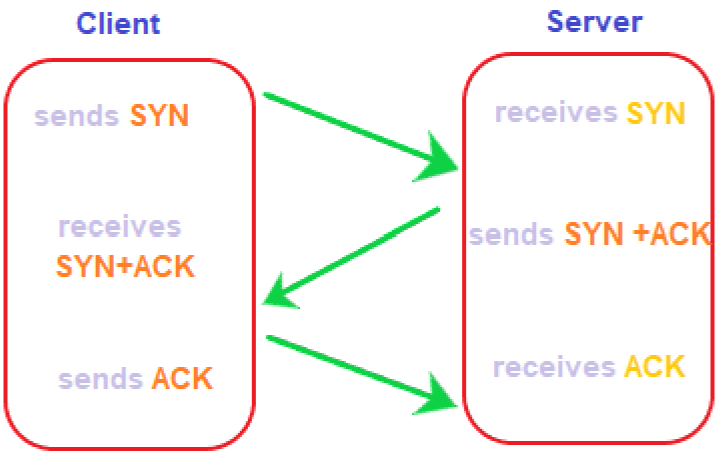
2. **Buffering**: The transport layer uses buffers (temporary storage) to handle data during transmission.
    - **Sender Buffering**: Holds data that is waiting to be sent.
    - **Receiver Buffering**: Stores incoming data until it is processed, allowing for smooth data flow even when transmission rates vary.

18. Explain with Diagram about Establishing and Releasing a Connection in Transport Layer

The **Transport Layer** is responsible for delivering data between application processes across a network.
The **transport layer** provides a reliable and error-free data connection.
establishing and releasing connection is done using TCP
TCP is a connection-oriented protocol, which means that it first establishes the connection between the sender and receiver in the form of a **handshake**
Handshake refers to the process to establish connection between the client and server.
Handshake is simply defined as the process to establish a communication link

The **Transport Layer** typically uses a **three-way handshake** for establishing a reliable connection and a **four-way handshake** for connection termination. This process is common in TCP.

1. **Connection Establishment**:



- **Step 1**: SYN
- Client sends a SYN (synchronize) packet to the server to initiate a connection. It acts as a **connection request** between the client and server. It informs the server that the client wants to establish a connection
- **Step 2**: SYN-ACK
- Server responds with a SYN-ACK packet, acknowledging the client's request. The ACK segment informs the client that the server has received the connection request and it is ready to build the connection
- **Step 3**: ACK
- ACK (Acknowledgment) is the last step before establishing a successful TCP connection between the client and server
- Client sends an ACK packet, completing the three-way handshake and establishing the connection.
- After these three steps, the client and server are ready for the data communication process.
- TCP connection and termination are full-duplex, which means that the data can travel in both the directions simultaneously.

2. **Connection Termination**: Any device establishes a connection before proceeding with the termination. TCP requires 3-way handshake to establish a connection between the client and server before sending the data. Similarly, to terminate or stop the data transmission, it requires a 4-way handshake

step 1 : fIN
FIN refers to the **termination request** sent by the client to the server. The first FIN termination request is sent by the client to the server. It depicts the start of the termination process between the client and server.

Step 2: FIN_ACK_WAIT

The client waits for the ACK of the FIN termination request from the server. It is a **waiting state** for the client.

Step 3: ACK

The server sends the ACK (Acknowledgement) segment when it receives the FIN termination request. It depicts that the server is ready to close and terminate the connection.

Step 4: FIN _WAIT_2

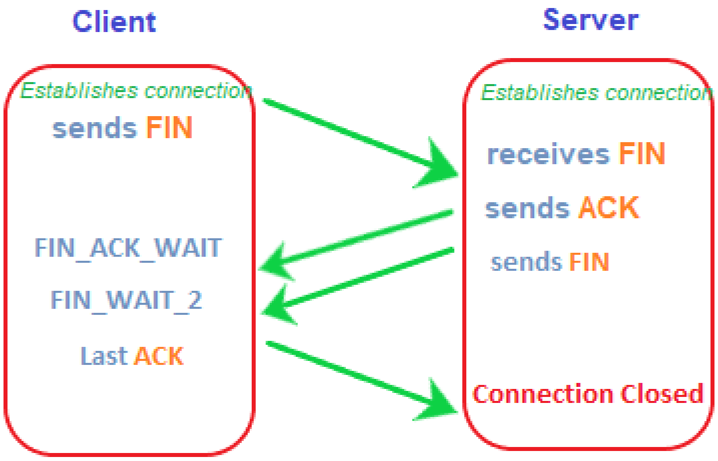
The client waits for the FIN segment from the server. It is a type of approved signal sent by the server that shows that the server is ready to terminate the connection.

Step 5: FIN

The FIN segment is now sent by the server to the client. It is a confirmation signal that the server sends to the client. It depicts the successful approval for the termination.

Step 6: ACK

The client now sends the ACK (Acknowledgement) segment to the server that it has received the FIN signal, which is a signal from the server to terminate the connection. As soon as the server receives the ACK segment, it terminates the connection.



1.

   - **Step 1**: Client sends a FIN (finish) packet to signal termination.
   - **Step 2**: Server sends an ACK packet in response.
   - **Step 3**: Server sends its own FIN packet to the client.
   - **Step 4**: Client responds with an ACK, completing the termination process.

19. Explain Crash Recovery in Transport Layer

Crash Recovery in the transport layer is essential for ensuring **data continuity and reliability** in case of connection failures.

In network communication crash recovery plays an important role in the transport layer in ensuring the **reliable exchange of data between endpoints.**

The need for crash recovery arises when unforeseen disruptions, like h**ardware failures, network congestion, transmission errors, or packet loss**, occur during *network communication*.

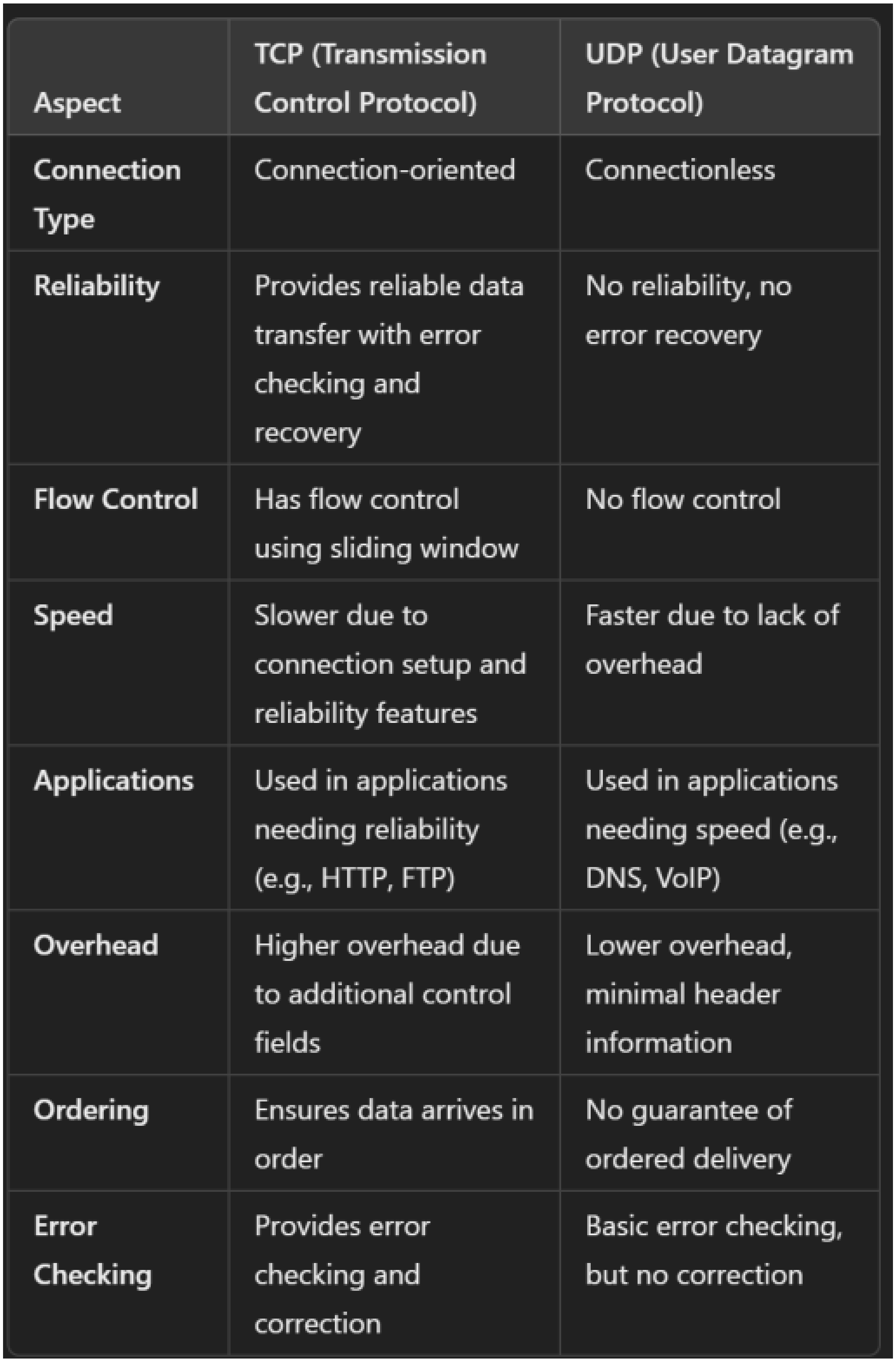
These disruptions can lead to **data loss or corruption,** affecting the reliability and timeliness of communication services.

Techniques include:

1. **Timeout Mechanism**: If no acknowledgment is received within a specific timeframe, the transport layer retransmits the lost data.
2. **Retransmission Queue**: Maintains a queue of unacknowledged packets for possible retransmission.
3. **Checkpointing and Rollback**: Saves the state at checkpoints, allowing recovery to a known point in case of failure.
4. **Sequence Numbers**: Tracks packets, helping in reordering and identifying missing packets after recovery.

TCP uses these mechanisms to ensure data reliability and integrity.These mechanisms guarantee the efficient transmission of data, even in network failures

20, tcp vs upd -- refer pdf for more detail



Aspect	TCP (Transmission Control Protocol)	UDP (User Datagram Protocol)
Connection Type	Connection-oriented	Connectionless
Reliability	Provides reliable data transfer with error checking and recovery	No reliability, no error recovery
Flow Control	Has flow control using sliding window	No flow control
Speed	Slower due to connection setup and reliability features	Faster due to lack of overhead
Applications	Used in applications needing reliability (e.g., HTTP, FTP)	Used in applications needing speed (e.g., DNS, VoIP)
Overhead	Higher overhead due to additional control fields	Lower overhead, minimal header information
Ordering	Ensures data arrives in order	No guarantee of ordered delivery
Error Checking	Provides error checking and correction	Basic error checking, but no correction

21. Explain Transport Service Primitives in Detail

Transport Service Primitives are basic operations provided by the transport layer for managing connections and data transfer.

They define how applications interact with the transport layer.

A primitive means operation.

To access the service a user process can access these primitives. These primitives are different for connection oriented service and connectionless service.

1. **LISTEN**: A server uses this primitive to indicate its readiness to accept incoming connections. It blocks waiting for an incoming connection.
2. **CONNECT**: Initiates a connection to a remote server. Used by a client to start communication.Response is awaited.
3. **SEND**: Used to transmit data across the connection.
4. **RECEIVE**: Waits for incoming data and retrieves it when available.
5. **DISCONNECT**: Terminates the connection gracefully, ensuring all data is sent and acknowledged before closing.

These primitives enable applications to establish connections, exchange data, and close connections in a controlled manner.

Connection Oriented Service Primitives

- There are 4 types of primitives for Connection Oriented Service :

CONNECT	This primitive makes a connection
DATA, DATA-ACKNOWLEDGE,	Data and information is sent using thus primitive
CONNECT	Primitive for closing the connection
RESET	Primitive for reseting the connection

Connectionless Oriented Service Primitives

- There are 2 types of primitives for Connectionless Oriented Service:

Primitive	Meaning
Unitdata	**Unitdata primitive is simply required to send packet of data or information.**
Facility, Report	**This primitive is required for getting details about the performance and working of the network such as delivery statistics or report.**