What are the fundamental characteristics that data communications depend on?

- **Delivery:** deliver data to the correct destination
- **Accuracy:** deliver the data accurately
- **Timeliness:** deliver data in a timely manner
- **Jitter:** the variation in the packet arrival time

Explain the components of a data communication system.

Component	Description	Example
Message	The information (data) to be communicated	Text, numbers, pictures, audio, and video
Sender	The device that sends the data message	A computer, workstation, telephone handset, video camera, etc.
Receiver	The device that receives the message	Same as sender
Transmission medium	The physical path by which a message travels from sender to receiver	Twisted-pair wire, coaxial cable, fiber-optic cable, and radio waves
Protocol	A set of rules that govern data communications	

What are some advantages and disadvantages of Mesh Topology?

Advantages:

- Uses dedicated links, avoiding traffic problems.
- Robust – a damaged link does not affect others.
- Provides privacy and security as messages travel along dedicated links.
- Easy fault identification and isolation.

Disadvantages:

- Requires a large amount of cabling and I/O ports.
- Difficult installation and reconnection.
- Wiring bulk can exceed available space.
- Expensive hardware needed to connect each device.

Explain the direction of data flow in Simplex, Half-Duplex, and Full-Duplex systems.

---

1. **Simplex:** Unidirectional. One device transmits, the other receives. Example: keyboards, monitors.
2. **Half-Duplex:** Each station can transmit or receive, but not simultaneously. Example: Walkie-Talkies.
3. **Full-Duplex:** Both stations can transmit and receive simultaneously. Example: Telephone network.

List and explain the key network criteria.

Edit

Delete

- **Performance:** Measured by transit time and response time, affected by number of users, transmission medium, hardware capabilities, and software efficiency. Evaluated using throughput and delay.
- **Reliability:** Based on accuracy of delivery, frequency of failure, recovery time from failure, and network robustness in catastrophes.
- **Security:** Involves protecting data from unauthorized access, damage, and implementing recovery policies.

What are the main features of Star Topology?

- Each device connects to a central hub via a point-to-point link.
- No direct traffic between devices; all transmissions go through the hub.
- The hub acts as an exchange, relaying data to connected devices.

Discuss the advantages of Mesh Topology.

- Dedicated links avoid traffic problems.
- Robustness - a damaged link does not impact others.
- Provides high levels of privacy and security through dedicated links.
- Fault identification and isolation are straightforward.

What are the advantages of Star Topology?

- Less expensive than mesh since each device is connected only to the hub
- Each device needs only one link and one I/O port to connect it to any number of others
- Installation and configuration are easy
- Less cabling is needed than mesh
- Additions, moves, and deletions involve only one connection: between that device and the hub
- Robustness - if one link fails only that link is affected – all other links remain active
- Easy to fault identification and to remove parts
- No disruptions to the network when connecting or removing the devices

Explain the disadvantages of Star Topology.

- Dependency of the whole topology on one single point, the hub. If the hub goes down, the whole system is dead
- Often more cabling is required in a star than in some other topologies (such as ring or bus)

What advantages does Bus Topology offer?

- Ease of installation
- Less cabling than mesh or star topologies
- Only the backbone cable stretches through the entire facility
- Each drop line has to reach only as far as the nearest point on the backbone

Differentiate between Star Topology and Ring Topology.

Edit

Delete

Adding another point to the comparison table:

Feature	Star Topology	Ring Topology
Connection	Devices are connected to a central hub	Each device has a point-to-point connection with only two other devices
Fault Tolerance	Robust, as one link failure doesn't affect the entire network	Less fault tolerant; a break in the ring can disable the entire network unless a dual ring or switch is used
Cabling	Generally requires more cabling than ring topology	Requires less cabling as each device only connects to its two neighbors
Traffic Direction	Can handle bi-directional traffic	Typically handles unidirectional traffic
Uses	Local-area networks, High-speed LANs	Token Ring networks, some office buildings, and school premises
Privacy Concerns	Since hub is a physical layer device, it will broadcast anything it receives, compromising privacy	Ring topology provides more privacy as the data is only sent to the intended recipient, reducing the risk of eavesdropping

How does communication occur in a packet-switched network?

- Computer network communication between the two ends involves blocks of data called packets.
- Switches function for both storing and forwarding because a packet is an independent entity that can be stored and sent later.
- A router in a packet-switched network has a queue that can store and forward the packet.
- If packets arrive at one router when the thick line is already working at its full capacity, the packets should be stored and forwarded in the order they arrived.
- Packet-switched network is more efficient than a circuit-switched network, but the packets may encounter some delays.

Describe the two most common types of switched networks.

Edit

Delete

1. **Circuit-Switched Network:** In a circuit-switched network, a dedicated path or circuit is established between the communicating devices for the entire duration of the communication. Think of it like making a phone call where a direct line is reserved between you and the person you are calling until the call ends.
2. **Packet-Switched Network:** In a packet-switched network, data is divided into packets before being sent. Each packet contains information like the source and destination addresses. Routers then analyze this information to determine the best path to forward the packet, allowing for more efficient use of network resources and flexibility in routing. This can be likened to sending a letter where the message is broken down into smaller envelopes, each with the address of the sender and recipient, and then choosing different mail routes for each envelope based on traffic conditions.

What are the layers of the TCP/IP protocol suite? Explain each layer briefly with examples

---

The TCP/IP protocol suite is thought of as a five-layer model today. They are:

1. **Physical Layer:** This is the hardware layer responsible for the physical connection between devices. Example: Ethernet cable.
2. **Data Link Layer:** Responsible for node-to-node data transfer and error detection. Example: Ethernet frame.
3. **Network Layer:** Handles routing of data packets between devices. Example: IP address.
4. **Transport Layer:** Manages end-to-end communication and error recovery. Example: TCP and UDP.
5. **Application Layer:** Interfaces directly with the user and provides network services. Example: HTTP, FTP.

Explain the duties and services offered by the Transport Layer in detail.

The Transport Layer is responsible for the following duties and services:

1. **Port addressing:** It ensures that messages are directed to the correct application process on the receiving end by using port numbers.
2. **Segmentation and reassembly:** The transport layer divides larger messages into smaller segments for transmission and reassembles them at the destination.
3. **Connection control:** It manages the establishment, maintenance, and termination of connections between two processes.
4. **Flow control (end-to-end):** This ensures that the sender does not overwhelm the receiver with too much data at once.
5. **Error control (end-to-end):** It detects and corrects errors in the transmitted data.

Examples:

- UDP and TCP are common transport layer protocols.

Explain the OSI Model in detail.

Edit

Delete

1. OSI (Open Systems Interconnection) is developed by the International Standards Organization (ISO).
2. It contains seven layers: Application, Presentation, Session, Transport, Network, Data Link, and Physical.
3. The Presentation Layer handles data translation, encryption, decryption, and compression.
4. The Session Layer manages interaction types like simplex, half-duplex, and full-duplex, and is responsible for session recovery and connection management between applications.
5. The OSI model was not as successful as the TCP/IP protocol suite for several reasons, such as the timing of introduction, incomplete definitions of some layers, and performance issues in applications.
6. The Application Layer usually combines functionalities of the Application, Presentation, and Session layers of the OSI model.

1. **Application Layer:** This layer is the closest to the end user and provides services such as email, file transfer, and web browsing. It interacts directly with the user and is responsible for high-level protocols.
2. **Presentation Layer:** The Presentation Layer is responsible for data translation, encryption, decryption, and compression. It ensures that the data is in a readable format for the application layer to process.
3. **Session Layer:** The Session Layer manages connections and sessions between applications. It handles establishment, termination, synchronization, and maintenance of sessions for data exchange.
4. **Transport Layer:** This layer ensures reliable data transfer between end systems. It segments and reassembles data, provides error-checking mechanisms, and controls flow of data.
5. **Network Layer:** The Network Layer deals with routing packets from the source to the destination across multiple networks. It selects the best path for data transmission and handles logical addressing.

6. **Data Link Layer:** The Data Link Layer is responsible for node-to-node communication, framing, error detection, and flow control. It ensures reliable data transfer over the physical layer.
7. **Physical Layer:** This is the lowest layer of the OSI model and deals with physical connections and hardware devices. It defines the electrical, mechanical, and functional specifications for transmitting data over a physical medium.

What are some reasons for the lack of the OSI Model's success?

Edit

Delete

1. OSI was completed when TCP/IP was already fully in place, and changing the established system would be costly.
2. Some layers, such as the Presentation and Session layers, were never fully defined with actual protocols and corresponding software.
3. When OSI was implemented in various applications, it did not demonstrate a high level of performance to justify switching from the established TCP/IP protocol suite.

Explain HTTP and its types of connections.

Edit

Delete

HTTP (Hypertext Transfer Protocol) is the Web's application layer protocol and operates on a client-server model:

- The client initiates a TCP connection to the server on port 80.
- The server accepts the TCP connection and HTTP messages are exchanged.
- HTTP is 'stateless,' meaning the server maintains no information about previous client requests, making it simpler but less efficient for maintaining stateful connections.
- HTTP connections can be Persistent or Non-persistent:

1. **Persistent HTTP:**
   - TCP connection opened to a server.
   - Multiple objects can be sent over a single TCP connection.
   - TCP connection closed after communication.
2. **Non-persistent HTTP:**
   - TCP connection opened.
   - At most one object is sent over the connection.
   - TCP connection closed.
   - Downloading multiple objects requires multiple connections.

Explain the structure of an HTTP request message with an example.

An HTTP request message consists of:

- Request Line: includes the method, URL, and HTTP version.
- Header Lines: provide additional information about the request.
- Body: contains the data sent to the server.

For example:

```
GET /index.html HTTP/1.1
Host: www-net.cs.umass.edu
User-Agent: Firefox/3.6.10
Accept: text/html,application/xhtml+xml
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7
Keep-Alive: 115
Connection: keep-alive
```

List and explain the different HTTP methods.

The different HTTP methods are:

1. GET: Requests data from a specified resource.
2. POST: Submits data to be processed to a specified resource.
3. HEAD: Same as GET but it transfers the status line and header section only.
4. PUT: Uploads a representation of the specified URI.
5. DELETE: Deletes the specified resource.

For example:

```
GET /index.html HTTP/1.1
Host: www.example.com
```

```
POST /submit-form HTTP/1.1
Host: www.example.com
Content-Length: 27

name=John&age=30&submit=true
```

Differentiate between SMTP and HTTP.

SMTP	HTTP
Push protocol	Pull protocol
Uses persistent connections	Does not require persistent connections
ASCII command/response interaction with status codes	ASCII command/response interaction with status codes
Multiple objects sent in multipart messages	Each object encapsulated in its own response message

Describe the interaction between a client and server during an SMTP session.

- The client connects to the server on port 25.
- The server responds with a status code, such as 220 to indicate it is ready.
- The client sends commands like HELO, MAIL FROM, RCPT TO, and DATA.
- The server responds with corresponding status codes, such as 250 for successful commands.
- The actual message content is sent after the DATA command, ending with a line containing only a period.
- The client can quit the session using the QUIT command.

Sample SMTP Interaction:

```
S: 220 hamburger.edu
C: HELO crepes.fr
S: 250 Hello crepes.fr, pleased to meet you
C: MAIL FROM: <alice@crepes.fr>
S: 250 alice@crepes.fr... Sender ok
C: RCPT TO: <bob@hamburger.edu>
S: 250 bob@hamburger.edu ... Recipient ok
C: DATA
S: 354 Enter mail, end with '.' on a line by itself
C: Do you like ketchup?
C: How about pickles?
C: .
S: 250 Message accepted for delivery
C: QUIT
S: 221 hamburger.edu closing connection
```

Explain the steps involved when a client wants an IP address for a website.

Edit

Delete

1. **Query the local DNS:** The client queries the local DNS server. If the local DNS has the ip address available in it's cache it sends it. Otherwise, the further steps take place.
2. **Query the root DNS server:** The root DNS servers are strategically placed across the world (managed by 12 organizations). These root DNS servers do not have the IP address but know where to locate them.
3. **Query the TLD server:** The root server redirects the DNS query to the appropriate TLD (top level domain) server. Each TLD server manages a domain type. For example .com.
4. **Query the authoritative name server:** The TLD servers forward the DNS query to the authoritative name server which know everything about the domain (final authority).
5. **Return the result:** The obtained ip address is stored in the cache in the resolver (local DNS) for future use and sends it back to the client.

Explain the different types of DNS records (RR).

Edit

Delete

- **Type NS:** Name is a domain (e.g., foo.com) and value is the hostname of the authoritative name server for this domain.
- **Type A:** Name is a hostname and value is the IP address.
- **Type CNAME:** Name is an alias name for some 'canonical' (the real) name (e.g., www.ibm.com is really servereast.backup2.ibm.com) and value is the canonical name.
- **Type MX:** Value is the name of mail server associated with the name.

Explain the concept of multiplexing and demultiplexing in the transport layer.

Edit

Delete

Multiplexing and demultiplexing are crucial functions of the transport layer:

1. **Multiplexing:** This process occurs at the sender's end where data chunks from various sockets are gathered. Each data chunk is encapsulated with a transport header to create segments, which are then passed to the network layer.
2. **Demultiplexing:** At the receiver's end, the transport layer receives segments from the network layer. The job of providing these segments to the appropriate application process running in the host is called demultiplexing. The transport layer uses the headers containing source and destination IP addresses and port numbers to direct the segment to the correct socket.
3. These functions ensure that multiple applications can use the network simultaneously without interference.
4. For instance, a web browser and an email client can both operate on the same device and communicate over the network, with data being multiplexed at the sender and demultiplexed at the receiver.
5. Without these functions, the transport layer would not be able to deliver data efficiently and correctly to the intended process.