

Data Communications and Computer Networks

Unit-I

Introduction to Data Communications

Communication: Communication is defined as a process in which more than one computer transfers information, instructions to each other and for sharing resources.

(or)
Communication is defined as a process or act in which we can send or receive data.

Data Communication: Data Communication is the process of transferring data from one place to another or between two locations.

It allows electronic and digital data to move between two networks.

Computer Network: A computer network is a collection of computing devices that are connected with each other for the purpose of information and resource sharing among a wide variety of users.

Components of Data Communication:

A communication system is made up of the following components

1) Message: A message is a piece of information that is to be transmitted from one person to another.
It could be a text file, an audio file, a video file, etc.

2) Sender: It is simply a device that sends data messages.

It can be a computer, mobile, telephone, laptop, video camera, or workstation, etc.

3) Receiver: It is a device that receives messages. It can be a computer, telephone mobile, workstation, etc.

4) Transmission Medium/ Communication Channels:

Communication Channels are the medium that connect two or more workstations.

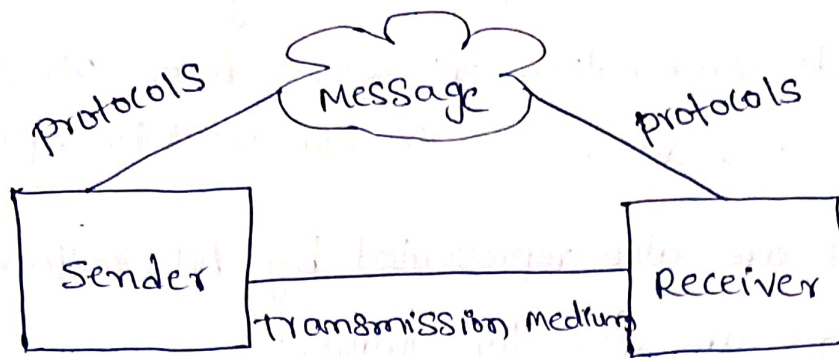
Workstations can be connected by either wired media or wireless media.

5) Set of rules (Protocol): When someone sends the data (the sender)

it should be understandable to the receiver also otherwise it is meaningless.

Therefore, there are some set of rules (protocols) that is followed by every computer connected to the internet and they are: TCP (Transmission Control Protocol)

IP (Internet Protocol)



TCP (Transmission control protocol): It is responsible for dividing messages into packets on the source computer and reassembling the received packet at the destination or recipient computer. It also makes sure that the packets have the information about the source of the message data, the destination of the message data, the sequence in which the message data should be re-assembled, and checks if the message has been sent correctly to the specific destination.

IP (Internet protocol): IP is responsible for handling the address of the destination computer so that each packet is sent to its proper destination.

Data Representation: Different forms of data representations such as text, numbers, images, audio & video.

Text: Texts include combinations of alphabets in small case (a, b, c, d) as well as upper case (A, B, C, D--).

It is stored as a pattern of bits (0 or 1).

→ Text can be represented by sequence of bits (Ex: ASCII)

American Standard code for Information Interchange.

Numbers: Numbers are also represented by bit patterns.

Numbers include combination of digits from 0 to 9.

However a code such as ASCII is not used to represent numbers.

Image: Images are also represented by bit patterns.

In computer images are digitally stored.

Picture or image is a matrix of pixel elements (ie) pixels.
→ A pixel which is the smallest element of an image.

Audio: It represents sound by continuous (analog) signal.

Video: It can be analog or digital signal.

Images: RGB (Red, Green, Blue)

YCM (Yellow, cyan & Magenta)

Data Flow: Communication between two devices can be simplex, half-duplex, full-duplex.

Simplex:

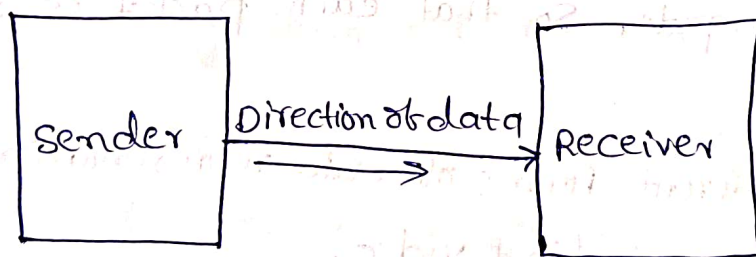


Fig: Simplex

→ In simplex mode, the communication is unidirectional as on a one way street. Only one of the two devices on a link can transmit the other can only receive.

→ Keyboard and traditional monitors are examples of simplex devices.

→ The keyboard can only introduce input the monitor can only accept output.

→ The Simplex mode can use the entire capacity of the channel to send data in one direction.

2) Half-Duplex :

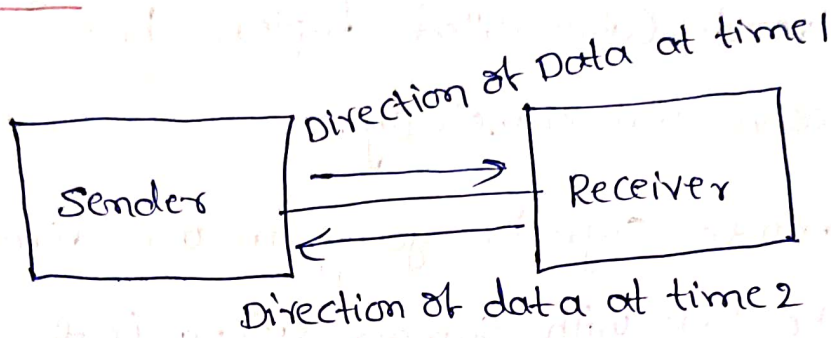


Fig : Half-Duplex

→ In half duplex mode each station can both transmit and receive but not at the same time.

→ When one device is sending the other can only receive and vice versa.

→ In a half duplex transmission the entire capacity of a channel is taken over by whichever of the two devices is transmitting at the time.

→ Examples of Halfduplex walkie-talkies, CB (Citizens band) radios.

→ The half duplex mode is used in cases where there is no need for communication in both directions at the same time. The entire capacity of channel can be utilized for one direction.

3) Full Duplex: / duplex

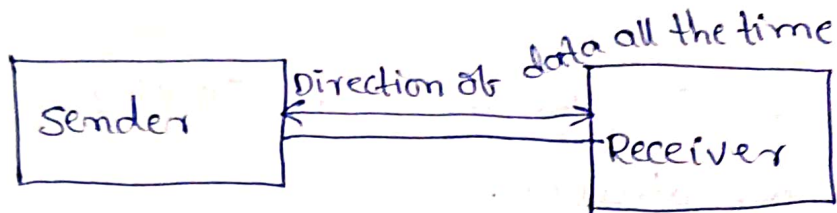


Fig: Full Duplex

→ In full-duplex mode (also called duplex), both stations can transmit and receive simultaneously.

→ The full-duplex mode signals going in one direction share the capacity of the link with signals going in the other direction.

→ This sharing can occur in two ways either the link must contain two physically separate transmission paths, one for sending and the other for receiving or the capacity of the channel is divided signals traveling in both directions.

→ Example of full-duplex communication is telephone network

Networks Distributed processing:

Networks: A Network is a set of devices connected by communication links.

→ A node can be a computer, printer, or any other device capable of sending and/or receiving data generated by other nodes on the network.

Distributed processing: Most networks use distributed processing, in which a task is divided among multiple computers.

→ Instead of a single large machine being responsible for all aspects of a process, each separate computer handles a subset of the task.

Advantages of distributed processing:

- 1) security
- 2) Distributed data bases
- 3) faster problem solving
- 4) security through redundancy
- 5) collaborative processing

Network criteria:

Data Communication
Network Criteria

performance

Reliability

Security

→ A Network must be able to meet a certain number of criteria.

→ The most important of these are performance, reliability and security.

Performance: It can be measured in many ways, including transit time and response time.

→ transit time is the amount of time required for a message to travel from one device to another.

Reliability: In addition to accuracy of delivery network reliability is measured by the frequency of failure, the time it takes a link to recover from a failure.

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

Network Applications:

- 1) Marketing & sales
- 2) Financial services
- 3) Manufacturing
- 4) Electronic messaging
- 5) Teleconferencing
- 6) Cable Television.

Physical structures :

Types of connection :

- A network is two or more devices connected through links.
- A link is a communications pathway that transfers data from one device to another.
- For communication to occur, two devices must be connected in same way to the same link at the same time.
- There are two possible types of connections

1) Point-Point 2) Multipoint

1) Point-Point :

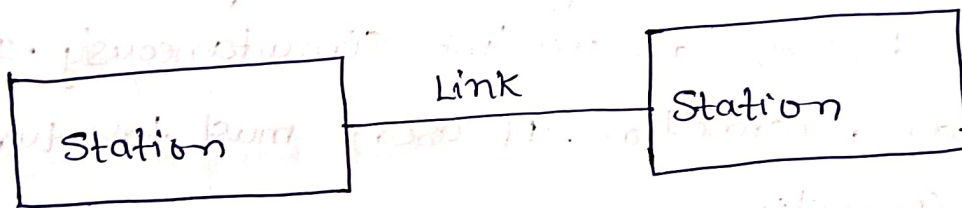


Fig: Point-Point

- Point-Point connection provides a dedicated link between two devices.
- The entire capacity of the links is reserved for transmission between those two devices.
- Most point-Point connections use an actual length of wire or cable to connect the two ends. but other options, such as Microwave or Satellite links are also possible.
- When you change television channels by infrared remote control you are establishing a point-point connection between the remote control & the television's control system.

2) Multipoint : / Multidrop :

→ A multipoint (also called Multidrop) connection is one in which more than two-specific devices share a single link.

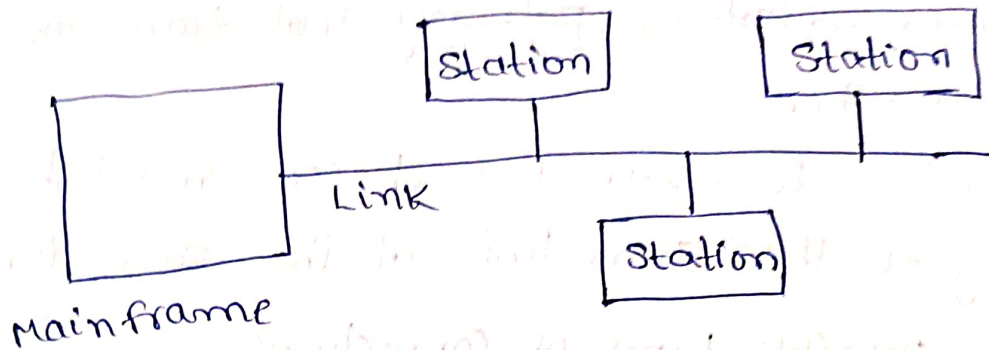


Fig: Multipoint

→ In a multipoint environment the capacity of the channel is shared either spatially or temporally.

→ If several devices can use link simultaneously. It is a spatially shared connection. If users must take turns. It is a timeshared connection.

Network Models:

→ Networking is divided into multiple layers. Each layer is involved in some particular task and is independent of all other layers.

→ Layers share data between them and they depend on each other only to take input and send output.

Layered tasks :

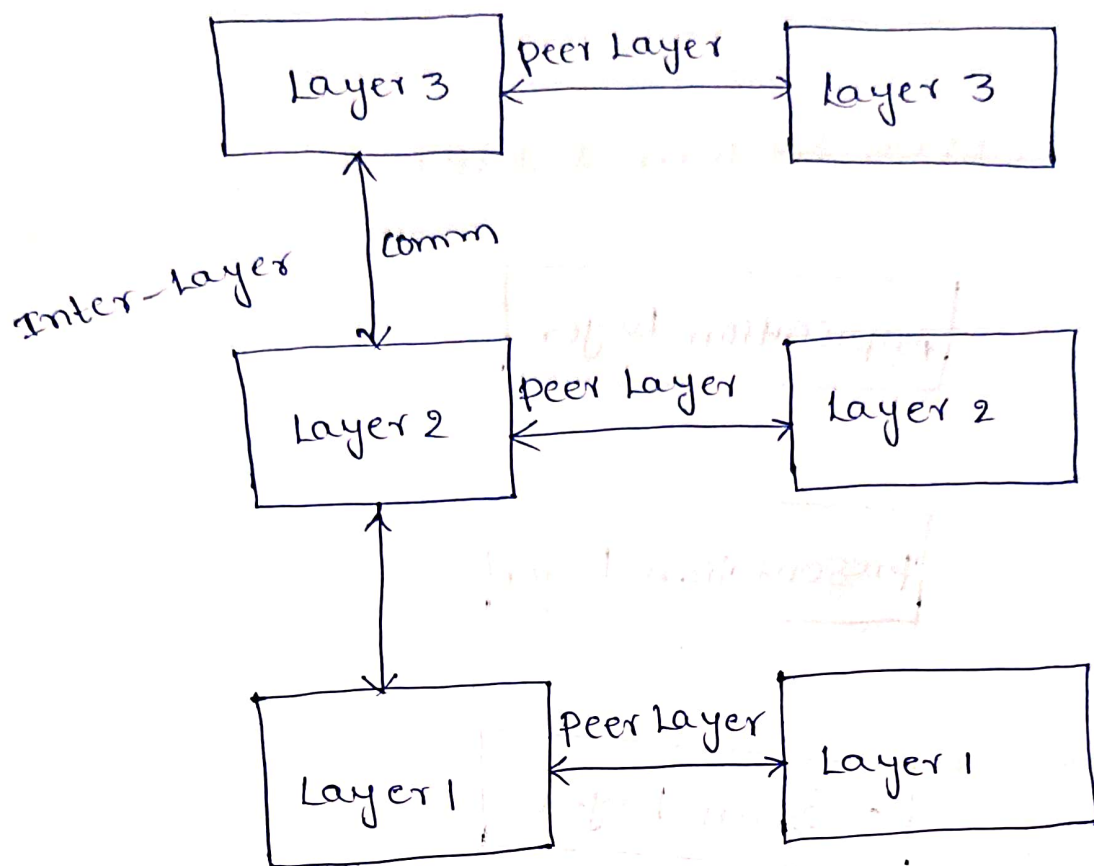


Fig: Layered Architecture of Network model

- In Layered Architecture of network model, one whole network process is divided into small tasks.
- Each small task is then assigned to particular layer which works dedicatedly to process the task only. Every layer does only specific work.
- In Layered communication system one layer of a host deals with the task done by or to be done by its peer layer at the same level on the remote host. Task is either initiated by layer at the lowest level or at the top most level.
- If the task is initiated by the top most layer, it passed on to the layer below it for further processing.
- The lower layer does the something.

→ It process the task and passes on to lower layer. If the task is initiated by lower most layer, then the reverse path is taken.

OSI Model : (Open System Interconnect) :

Application Layer

Presentation Layer

Session Layer

Transport Layer

Network Layer

Data Link Layer

Physical Layer

→ Open System Interconnect is an open standard for all communication system. OSI model is established by International Standard Organization (ISO). This model has seven layers. They are

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

1) Physical Layer: This layer defines the hardware, cabling wiring, power, output, pulse rate etc.

2) Data Link Layer: This layer is responsible for reading and writing data from and onto the line. Link errors are detected at this layer.

3) Network Layer: This layer is responsible for address assignment and uniquely addressing hosts in a network.

4) Transport Layer: This layer is responsible for end-to-end delivery between hosts.

5) Session Layer: This layer maintains sessions between remote hosts. For example once user/password authentication is done, the remote host maintains this session for a while and does not ask for authentication again in that time span.

6) Presentation Layer: This Layer defines how data in the native format of remote host should be presented in the native format of host.

7) Application Layer: This layer is responsible for providing interface to the application user. This layer encompasses protocols which directly interact with the user.

→ Dataflow through the OSI Model in a Step-by-Step process.

1) Physical: Frames are converted into bits & transmitted physically.

2) DataLink Layer: packets are framed and send to the next device.

3) Network Layer: Segments are packaged into packets and routed.

4) transport Layer: data is broken into segments for reliable delivery.

5) session Layer: connections are established and managed.

6) Presentation Layer: data is formatted and encrypted.

7) Application Layer: applications create the data.

3) Internet Model:

Application layer

transport Layer

Internet Layer

Link Layer

→ Internet model uses tcp/ip protocol suit.. Also known as Internet suit.

→ Internet model which contains four layered architecture.

→ OSI model is general communication Model but Internet model is what the Internet uses for all its communication.

→ The Internet is independent of its underlying network architecture so is its model.

Link Layer: This layer provides mechanism of sending and receiving actual data. This layer is independent of underlying network architecture and hardware.

2) Internet Layer: Internet protocol (IP) works on this layer.

This layer facilitates host addressing and recognition. This layer defines routing.

3) Transport Layer: This layer defines how data should flow between hosts. Major protocol at this layer is transmission control protocol (TCP).

→ This layer ensures data delivered between hosts is in order and is responsible for end-to-end delivery.

4) Application Layer: This layer defines the protocol which enables user to interact with the network.

Ex: FTP, HTTP, etc.

FTP: File Transfer Protocol

HTTP: Hypertext Transfer Protocol

→ Network Topologies: Network topology is the arrangement with which computer systems or network devices are connected to each other.

→ topologies may define both physical and logical aspect of the network.

→ Both logical and physical topologies could be same or different in the same network.

1) Bus topology

2) Ring topology

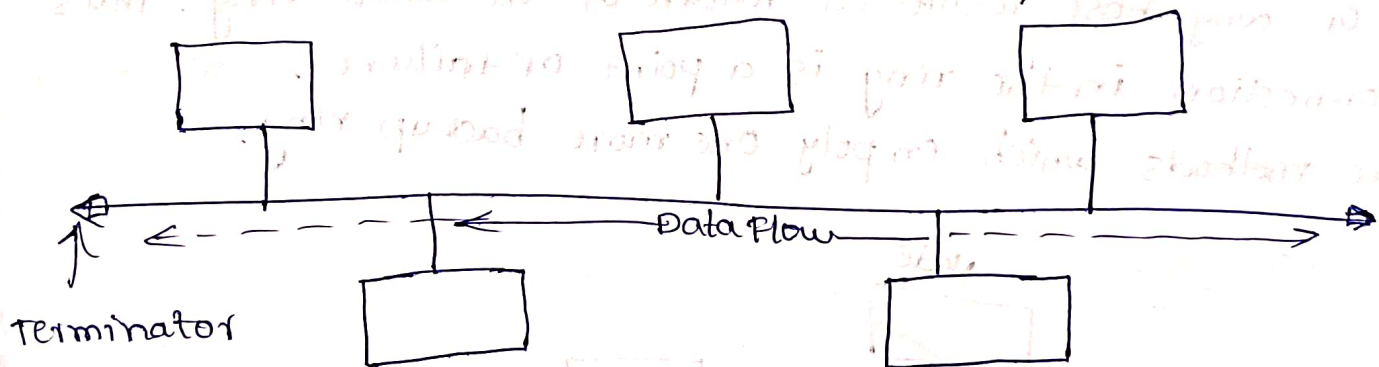
3) Star topology

4) Mesh topology

5) Hybrid topology

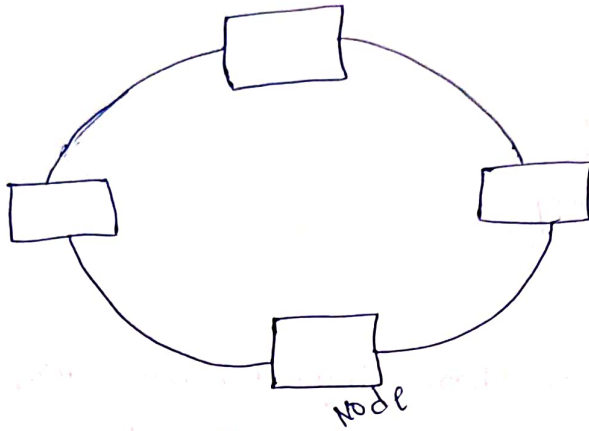
1) Bus Topology:

- In Bus topology all devices share single communication line or cable.
- Bus topology may have problem while multiple hosts sending data at the same time.
- It is one of the simple forms of networking where a failure of a device does not affect the other devices. But failure of the shared communication line can make all other devices stop functioning.



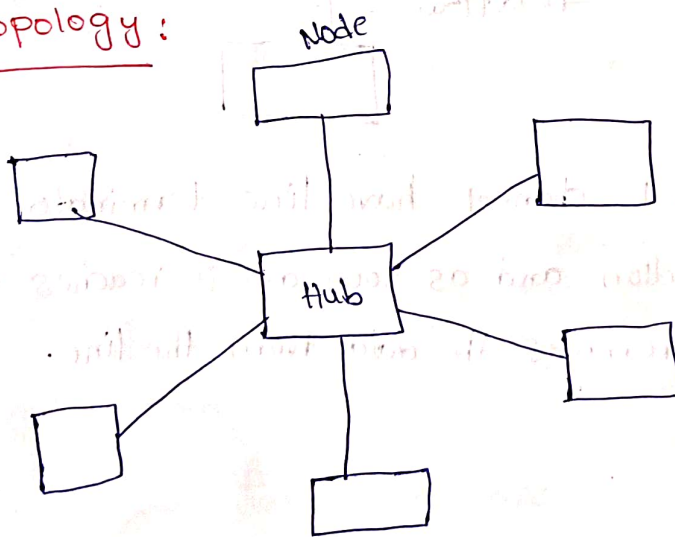
Both ends of the shared channel have line terminator. The data sent in only one direction and as soon as it reaches the extreme end, the terminator removes the data from the line.

2) Ring Topology :



- In Ring topology each host machine connects to exactly two other machines, creating a circular network structure.
- When one host tries to communicate or send message to a host which is not adjacent to it, the data travels through all immediate hosts.
- Failure of any host results in failure of the whole ring, thus every connection in the ring is a point of failure.
- There are methods which employ one more backup ring.

3) Star topology :

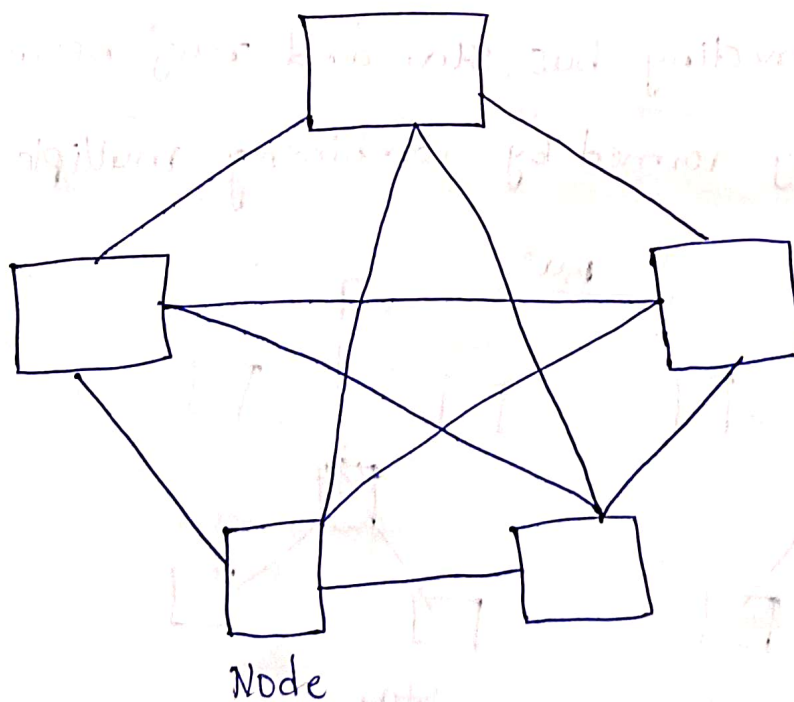


→ All ho
as hu
→ That is
hub.
→ AS i
Connect
→ Every
hub.
→ Sta
only
Mesh
→ In
→ This
host
to fe

- All hosts in star topology are connected to a central device, known as hub device, using a point-to-point connection.
- That is there exists a point-to-point connection between hosts and hub.
- As in Bus topology, hub acts as single point of failure. If hub fails, connectivity of all hosts to all other hosts fails.
- Every communication between hosts, takes place through only the hub.
- Star topology is not expensive as to connect one more host, only one cable is required and configuration is simple.

Mesh topology:

- In this type of topology, a host is connected to one or multiple hosts.
- This topology has hosts in point-to-point connection with every other host or may also have hosts which are in point-to-point connection to few hosts only.



→ Hosts in mesh topology also work as relay for other hosts which do not have direct point-to-point links.

→ Mesh topology comes into two types:

1) Full Mesh: All hosts have a point-point connection to every other host in the network, thus for every new host $n(n-1)/2$ connections are required. It provides the most reliable network structure among all network topologies.

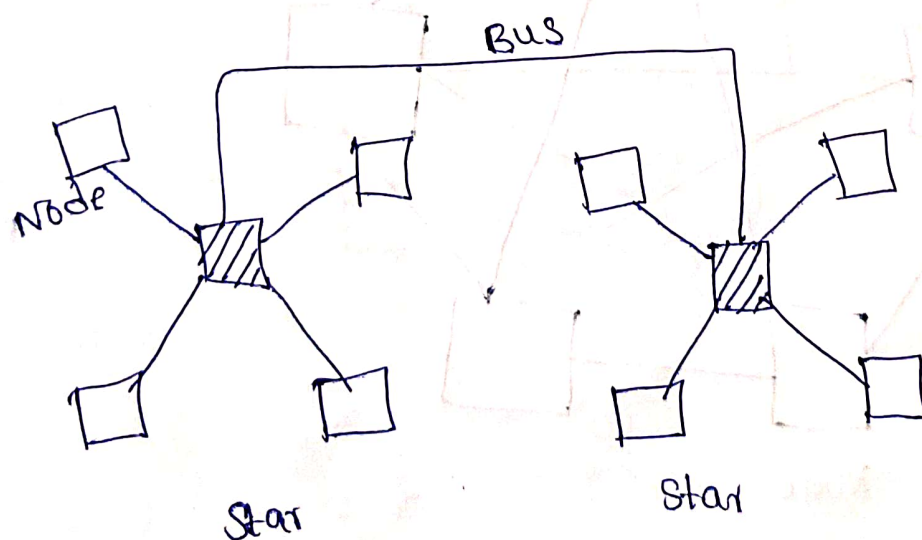
2) partially mesh: Not all hosts have point-point connection to every other host. Hosts connect to each other in some arbitrary fashion. This topology exists where we need to provide reliability to some hosts out of all.

5) Hybrid topology:

→ A network structure whose design contains more than one topology is said to be hybrid topology.

→ It is formed by connecting bus, star and ring networks together.

→ It is a large topology formed by combining multiple topologies.



Categories of Networks Interconnection of N/w :

→ Based on the size & networks are categorized into the following types:

- 1) Local Area Network (LAN)
- 2) Wide Area Network (WAN)
- 3) Metropolitan Area Network (MAN)

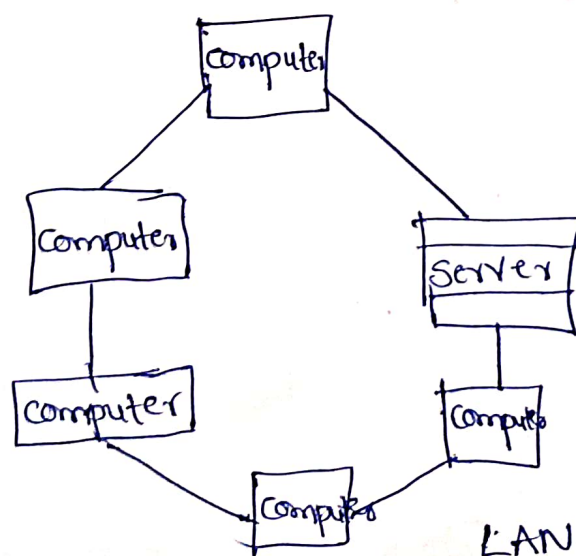
1) Local Area Network (LAN) :

→ LAN is the most frequently used Network. A LAN is a computer network that connects computers through a common communication path contained within a limited area that is locally:

→ usually LAN covers an organization offices, schools, colleges or universities

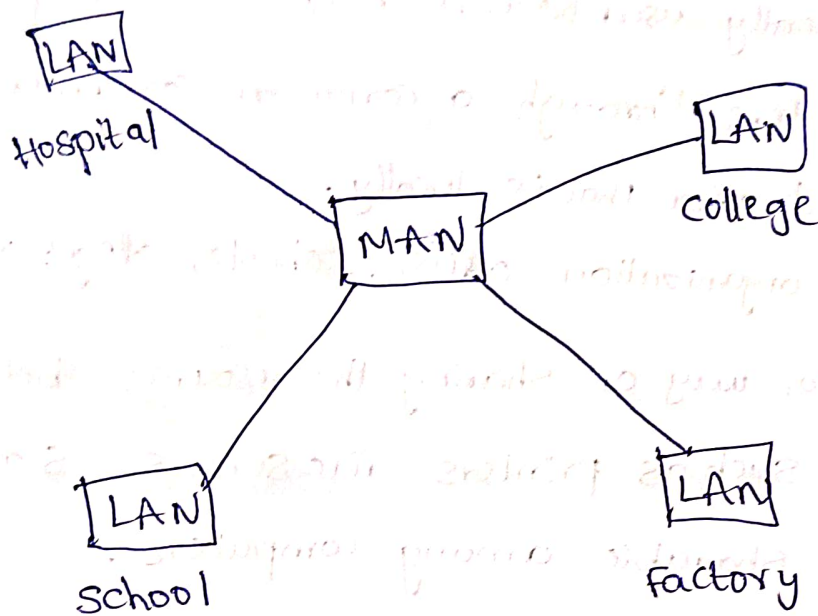
→ LAN provides a useful way of sharing the resources between end users. The resources such as printers, file servers, scanners, and Internet are easily sharable among computers.

→ The two important technologies involved in this network are Ethernet and wifi. It ranges up to 2km & transmission speed is very high with easy maintenance and low cost.



2) Metropolitan Area Network (MAN):

- The Metropolitan Area Network (MAN) generally expands throughout a city such as cable TV network. (or)
- MAN is a computer network that connects multiple local area networks (LAN) within a metropolitan area, such as a city or a large campus.
- It acts as a bridge between LAN and WAN, covering a larger geographic area than a LAN but a smaller area than a WAN.

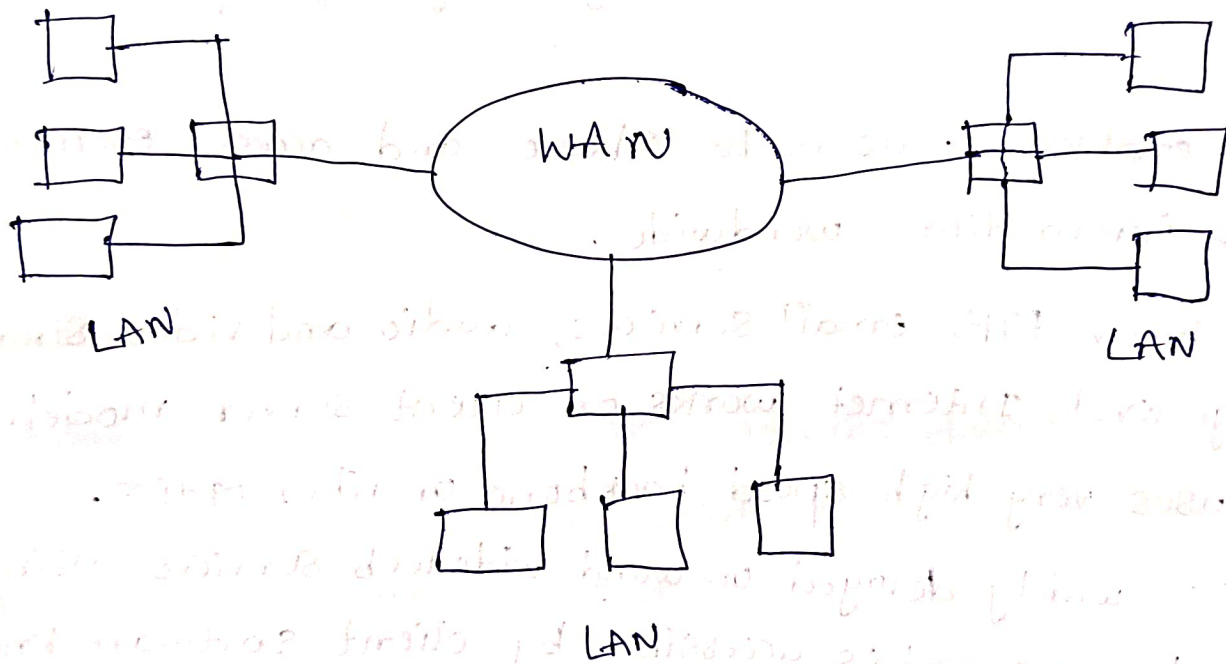


- Backbone of MAN is high-capacity and high-speed fiber optics.
- MAN works in between Local Area Network and wide Area Network. MAN provides uplink for LANs to WANs or Internet.

Wide Area Network (WAN):

→ A wide Area Network is a Computer Network that spans a large geographical area, typically connecting multiple ^{local} area Networks (LANs) or other networks.

→ WAN enable communication and data sharing between devices at different locations, including branch offices, data centers, and cloud services.



WAN may use advanced technologies such as Asynchronous Transfer Mode (ATM), Frame Relay, and Asynchronous Optical Network (SONET). WAN may be managed by multiple administration.

→ The Internet :

- A Network of networks is called an Internetwork or simply the Internet. It is the largest network in existence on this planet.
- The Internet hugely connects all WANs and it can have connection to LANs and Home networks.
- Internet uses TCP/IP Protocol Suite and uses IP as its addressing protocol.
- Present day, Internet is widely implemented using IPv4. Because of shortage of address spaces, it is gradually migrating from IPv4 to IPv6.
- Internet enables its users to share and access enormous amount of information worldwide.
- It uses WWW, FTP, email services, audio and video streaming etc. At huge level, Internet works on client-server model.
- Internet uses very high speed backbone of fiber optics.
- Internet is widely deployed on world wide web services using HTML linked pages and is accessible by client software known as web Browsers.
- When a user requests a page using some web browser located on some web server anywhere in the world, the web server responds with the proper HTML page. The communication delay is very low.

→ Internet is serving many purposes and is involved in many aspects of life. Some of them are:

→ web sites

→ Email

→ Instant messaging

→ Blogging

→ Social Media

→ Marketing

→ Networking

→ Resource Sharing

→ Audio and Video Streaming

→ protocols and standards

protocols:

- A protocol is a set of rules that governs data communications.
- It defines what is communicated, how it is communicated and when it is communicated.

Key elements of Protocol:

Syntax:

- Structure or format of data, meaning the order in which they are presented.

Semantics:

- Refer to the meaning of each section of bits, how a particular pattern is interpreted and what action to be taken.

Timing:

→ Refers to when data should be sent and how fast can they be sent.

Standards

→ Standards are essential in creating and maintaining an open and competitive market for equipment manufacturers.

→ Required to guarantee national and international interoperability of data and telecommunications technology and process.

→ Categories of data communications standards

→ De facto :

→ Standards that have not been approved by an organizational body but have been adopted through widespread use (eg: Model TCP/IP).

→ De jure :

→ Those that have been legislated by an official recognized body, eg: OSI model.

Standards organizations:

Standards Creation Committees:

- ISO (International organization for standardization)
(International Standard organization)
- ITU - (International telecommunications union - Telecommunications Standards)
 - Initially known as CCITT (Consultative committee for international telegraphy and telephony)
- ANSI (American National Standards Institute)
- IEEE (Institute of Electrical and Electronics Engineers)
- EIA (Electronic Industries Association)

Forums

- Made up of representatives from interested corporations to speed acceptance and use of new technologies in the telecom industry.

Regulatory Agencies:

- Government agencies: to protect public interest by regulating radio, tv and wire/cable communications.

Internet standards :

- Internet standards, also known as Internet Engineering Task Force (IETF) standards.
- An Internet standard is a thoroughly tested specification that is useful to and adhered to by those who work with the Internet.
- It is a formalized regulation that must be followed. There is a strict procedure by which a specification attains Internet standard status.
- A specification begins as an Internet draft.
- An Internet draft is a working document (a work in progress) with no official status and a 6-month lifetime.
- Upon recommendation from the Internet authorities, a draft may be published as a Request for comment (RFC).
- Each RFC is edited, assigned a number and made available to all interested parties.
- RFCs go through maturity levels and are categorized according to their requirement level.

→ Internet standards are established guidelines and protocols that ensure interoperability and reliable communication across the Internet.

→ They define how data is transmitted, formatted, and managed, enabling different devices and systems to communicate effectively. Some examples include TCP/IP, HTTP, HTML, etc.

→ TCP/IP (Transmission Control Protocol/Internet Protocol)

→ The foundation of the Internet, defining how data is transmitted and routed across networks.

→ HTTP (Hypertext Transfer Protocol): The protocol used for web browsing, enabling communication between web servers & clients (e.g.: web browsers).

HTML: (Hypertext markup language): The language used to create web pages, defining the structure and content of web content.

DNS (Domain Name System): Translates human-readable domain names (like google.com) into IP addresses that computers use to communicate.

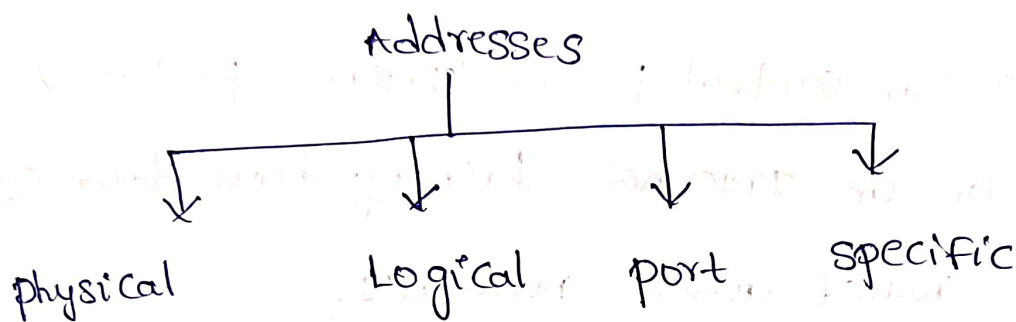
FTP: (File Transfer Protocol): A standard protocol for transferring files between computers.

Email protocols: Standards for sending and receiving emails.

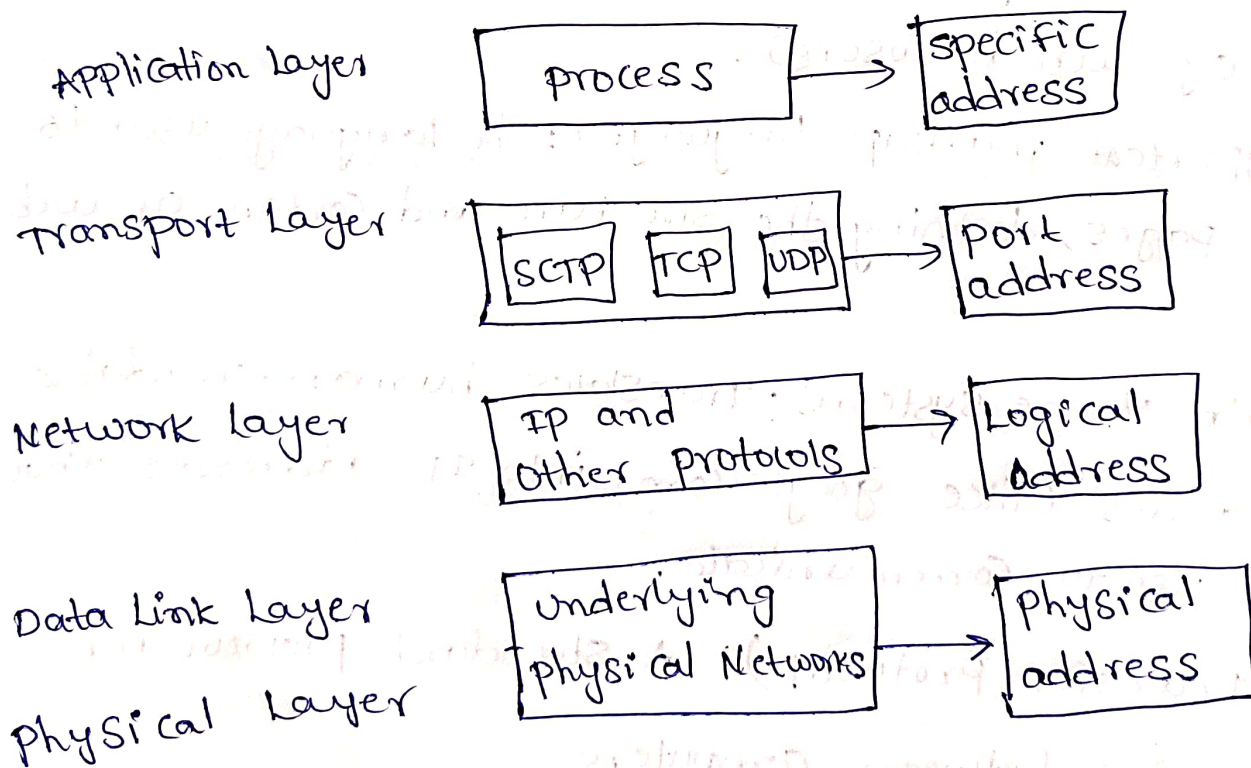
→ Addressing:

→ Four levels of addresses are used in an Internet employing TCP/IP protocols: Physical (Link) addresses, Logical (IP) addresses, Port addresses, & Specific addresses.

Address in TCP/IP



Relationship of layers and addresses in TCP/IP



Physical Addresses:

- The physical address, also known as the Link address. It is the address of a node as defined by its WAN & LAN.
- It is included in the frame used by the data link layer.
- It is the lowest-level address.
- Physical addresses have authority over the network (LAN & WAN). The size and format of these addresses vary depending on the network.
- For example: Ethernet uses a 6-byte (48-bit) physical address is imprinted on the network interface card (NIC).
- Most Local-Area networks use a 48-bit (6 byte) physical address written as 12 hexadecimal digits.
- Every byte (2 hexadecimal digits) is separated by a colon.

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

A 6 byte (12 hexadecimal digits) physical Address.

Logical Address:

- Logic addresses are necessary for universal communications that are independent of underlying physical networks.
- Physical addresses are not adequate in an internetwork environment where different networks can have different address formats.
- A universal addressing system is needed in which each host can be identified uniquely.

→ The logical address are designed for this purpose. A logical address in the Internet is currently a 32-bit address that can uniquely define a host connected to the Internet.

→ No two publicly addressed and visible hosts on the Internet can have the same IP address.

SCP : Stream Control Transmission Protocol

UDP : User datagram protocol.

Port Addresses :

→ The IP address and physical address are necessary for a quality of data to travel from a source to the destination host.

→ However, arrival at the destination host is not the final objective of data communications on the Internet.

→ A system that sends nothing but data from one computer to another is not complete. today computers are devices that can run multiple processes at the same time.

Specific Address :

→ Some addresses have user friendly addresses that are designed for that specific address.

→ For example include the e-mail address and the universal Resource Locator (URL) (www.gmail.com). The first defines the recipient of an e-mail.

→ The second is used to find a document on the world wide web.

→ Ethernet is a family of wired computer networking technologies commonly used in LAN.

Difference between OSI and TCP/IP

OSI

- 1) OSI stands for open system interconnect
- 2) It has 7 layers
- 3) Delivery of the packet is guaranteed in OSI Model
- 4) Developed by ISO
- 5) Vertical Approach
- 6) protocols in the OSI model are hidden and can be easily replaced when the technology changes.
- 7) Network layer provides connection oriented & connectionless service.
- 8) It is low in usage
- 9) Less reliable

TCP/IP

- 1) TCP/IP stands for Transmission Control protocol / Internet protocol
- 2) It has 4 layers
- 3) Delivery of the packet is not guaranteed in TCP/IP Model.
- 4) Developed by ARPANET (Advanced Research project Agency Network)
- 5) Horizontal Approach
- 6) protocols can not be easily replaced.
- 7) Network layer provides only connectionless service
- 8) It is mostly used.
- 9) It is more reliable

10) protocol independent Standard

11) The smallest size of the OSI header is 5 bytes.

12) In this model, Session & presentation layers are separated
(ie) both layers are different.

10) protocol dependent Standard.

11) The smallest size of TCP/IP header is 20 bytes.

12) In this, the Session and presentation layers are not different layers. Both layers are included in application layer.

Wireless Links and Network Characteristics:

Various characteristics of wireless links and network are:

1) Attenuation: It is a situation where the strength of electromagnetic signals reduces gradually. This is because the signals which are transmitted over long distance get dispersed in all the direction. Due to this only a small part received by receiver.

2) Interference: The Receiver can receive the signal from any other sender other than the intended sender.

3) Multipath propagation: It is a situation where a receiver can receive multiple signals from the same sender. This is because these signals can be reflected back when gets hit by various obstacles such as walls, ground or objects.

4) Error: Characteristics of wireless network makes the error and error detection issue become more difficult compared to wired network.

→ If error is considered as the signal to noise ratio (SNR)

→ SNR is high → Strong signal

→ SNR is low → Corrupted signal

WIFI: 802.11 Wireless LANs: The 802.11 Architecture:

The main objective of IEEE 802.11 Standard are,

1) A simple and robust WLAN that can provide Asynchronous & time bounded services.

2) MAC Layer should be capable of managing multiple physical layers each of them showing different transmission characteristics and medium sense.

IEEE 802.11 Architecture and Services:

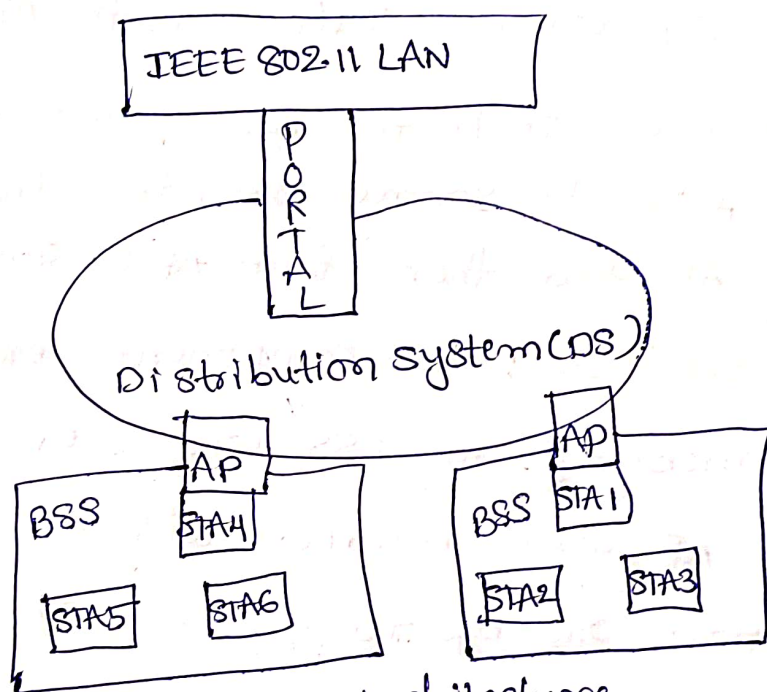


Fig: IEEE 802.11 Architecture

The IEEE 802.11 architecture consists of various components. They are STA (station), Basic Service Set (BSS), Access point (AP), Extended service set (ESS) and distribution service (DS).

1) Station (STA): This component is used to connect to the wireless medium. Station (STA) can be any device, it can be a mobile device, or a Network Interface card (NIC) etc. It provides the services of authentication, deauthentication, privacy and delivery of the data.

2) Basic Service Set (BSS):

→ A BSS consists of a set of STA that communicate with each other. Due to uncertainty of wireless propagation, the BSS cannot locate itself at a particular location.

→ When BSS include an AP (Access point), then the BSS is called as infrastructure BSS, but often referred to as BSS.

3) Access point (AP):

An AP is just like a station (STA). It also provides the same services as STA. In an infrastructure BSS, every mobile STA communicate with the AP.

→ If suppose one mobile STA in the BSS want to communicate with other mobile STA in the same BSS, then the communication is first sent to AP and then from AP to other mobile STA.

4) Extended Service Set (ESS):

This component extends the range (area) of mobility. An ESS consists of infrastructure BSS's, where one AP communicates with the other AP to move the traffic from one BSS to another and it also facilitates to move mobile STA's from one BSS to another BSS. This is the responsibility of an AP to perform this communication through a medium known as Distribution Service (DS).

5) Distribution Service (DS):

This component is the backbone of the WLAN and can be constructed of either wired or wireless networks. The DS allows the AP to communicate and to exchange frames for STA's in their BSS's.

Unit-II

Data Link Layer

→ Data Link Layer is second layer of OSI layered model. This layer is one of the most complicated layers and has complex functionalities and liabilities.

→ Data Link Layer hides the details of underlying hardware and represents itself to upper layer as the medium to communicate.

→ Data link layer works between two hosts which are directly connected in some sense. This direct connection could be point to point or broadcast.

→ Data link layer is responsible for converting data stream to signals bit by bit and to send that over the underlying hardware. At the receiving end, data link layer picks up data from hardware which are in the form of electrical signals, assembles them in a recognizable frame format and hands over to upper layer.

Data Link Layer has two sub-layers:

1) Logical Link Control: It deals with protocols, flow-control, and error control.

2) Media Access Control: It deals with actual control of media

Functionality of Data-Link Layer:

2

Data Link layer does many tasks on behalf of upper layer. These are

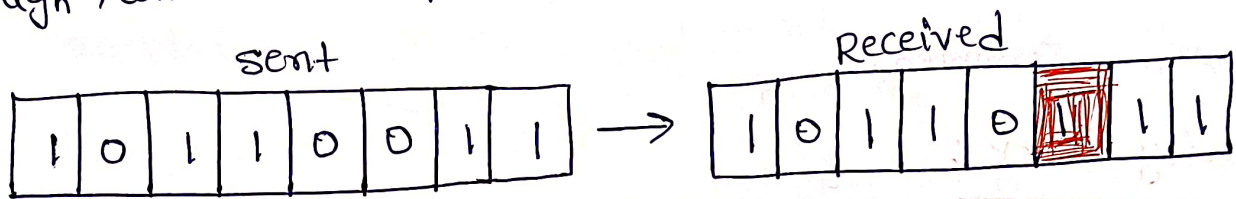
- 1) Framing: Data Link Layer takes packets from Network Layer and encapsulates them into frames. Then, it sends each frame bit by bit, on the hardware. At receiver end data link layer picks up signals from hardware and assembles them into frames.
- 2) Addressing: Data Link Layer provides layer-2 hardware addressing mechanism. Hardware address is assumed to be unique on the link. It is encoded into hardware at the time of manufacturing.
- 3) Synchronization: when data frames are sent on the link, both machines must be synchronized in order to transfer to take place.
- 4) Error Control: Sometimes signals may have encountered problem in transition and the bits are flipped. These errors are detected and attempted to recover actual data bits. It also provides error reporting mechanism to the sender.
- 5) Flow Control: Stations on same link may have different speed or capacity. Data link layer ensures flow control that enables both machine to exchange data on same speed.
- 6) Multi Access: when host on the shared link tries to transfer the data, it has a high probability of collision.

3

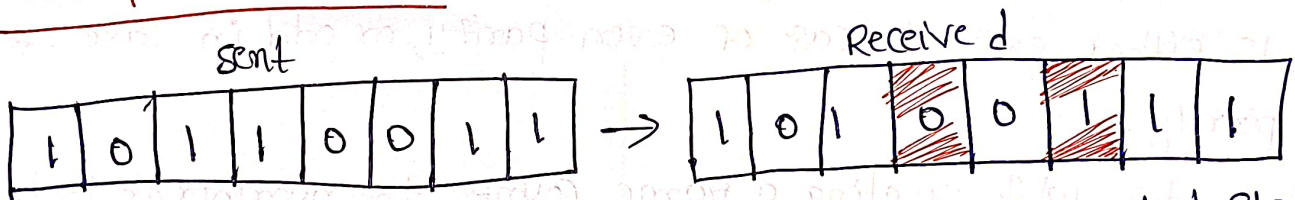
→ Data Link Layer uses some error control mechanism to ensure that frames (data bit streams) are transmitted with certain level of accuracy.

Types of Errors : There may be three types of errors.

1) Single bit error : In a frame, there is only one bit, anywhere though, which is corrupt.

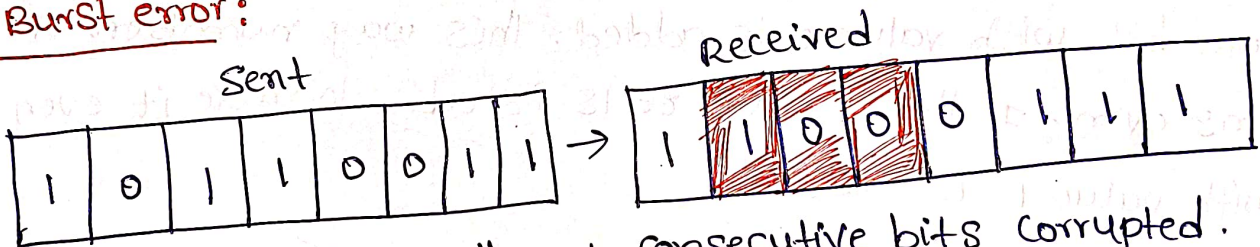


2) Multiple bits error :



Frame is received with more than one bits in corrupted state.

3) Burst error :



Frame contains more than 1 consecutive bits corrupted.

→ Error control mechanism may involve two possible ways.

→ Error detection

→ Error correction

Error Detection:

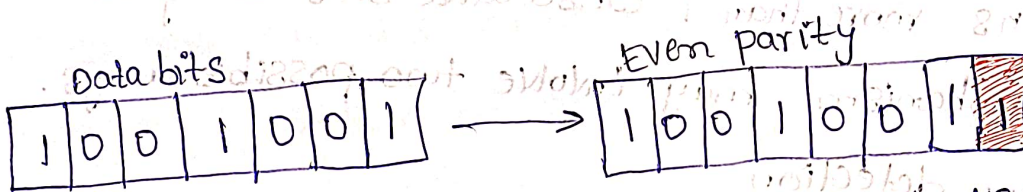
→ Errors in the received frames are detected by means of parity check and Cyclic Redundancy check & checksum.

→ In both cases, few extra bits are sent along with actual data to confirm that bits received at other end are same as they were sent. If the counter-check at receiver end fails, the bits are considered corrupted.

1) Parity Check:

one extra bit is sent along with the original bits to make number of 1s either even in case of even parity, or odd in case of odd parity.

→ The sender, while creating a frame counts the number of 1s in it. For example, if even parity is used and number of 1s is even then one bit with value 0 is added. This way number of 1s remains even. If the number of 1s is odd, to make it even a bit with value 1 is added.



→ The receiver simply counts the number of 1s in a frame. If the count of 1s is even and even parity is used, the frame is considered to be not-corrupted and is accepted. If the count of 1s is odd and odd parity is used, the frame is still not corrupted.

CRC Example:- Estimate the remainder obtained by dividing x^7+x^5+1 by the generator polynomial x^3+1 .

Sol:- Codeword $\rightarrow x^7+x^5+1$

Generator $\rightarrow x^3+1$

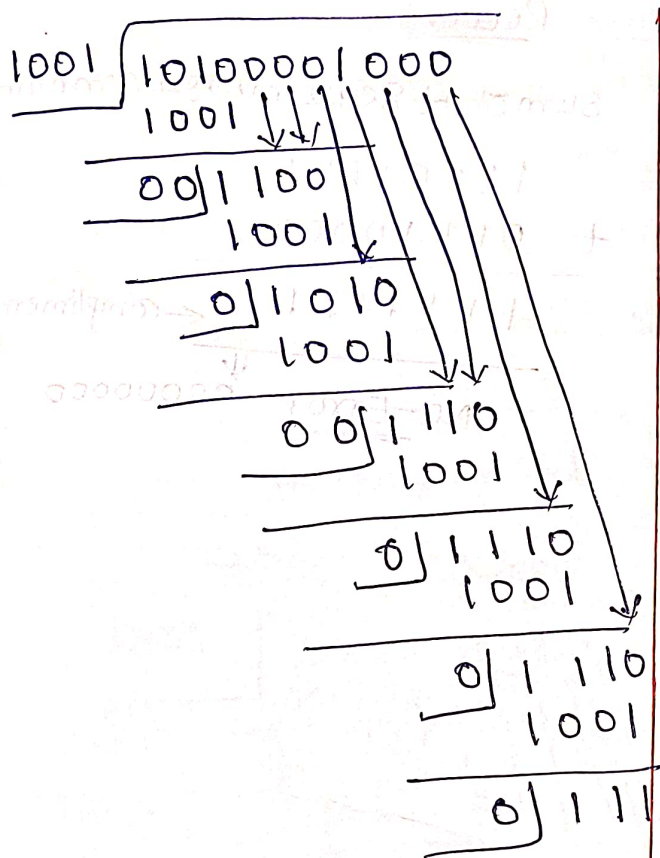
$$x^7+x^5+1 \Rightarrow 1 \cdot x^7 + 0 \cdot x^6 + 1 \cdot x^5 + 0 \cdot x^4 + 0 \cdot x^3 + 0 \cdot x^2 + 0 \cdot x^1 + 1 \cdot x^0$$

$$M(x) = 10100001, \quad G(x) = x^3+1 \Rightarrow 1x^3 + 0x^2 + 0x^1 + 1x^0$$

$$G(x) \Rightarrow 1001$$

$n-1 = 4-1 = 3$ we add 000 at end of code

Sender



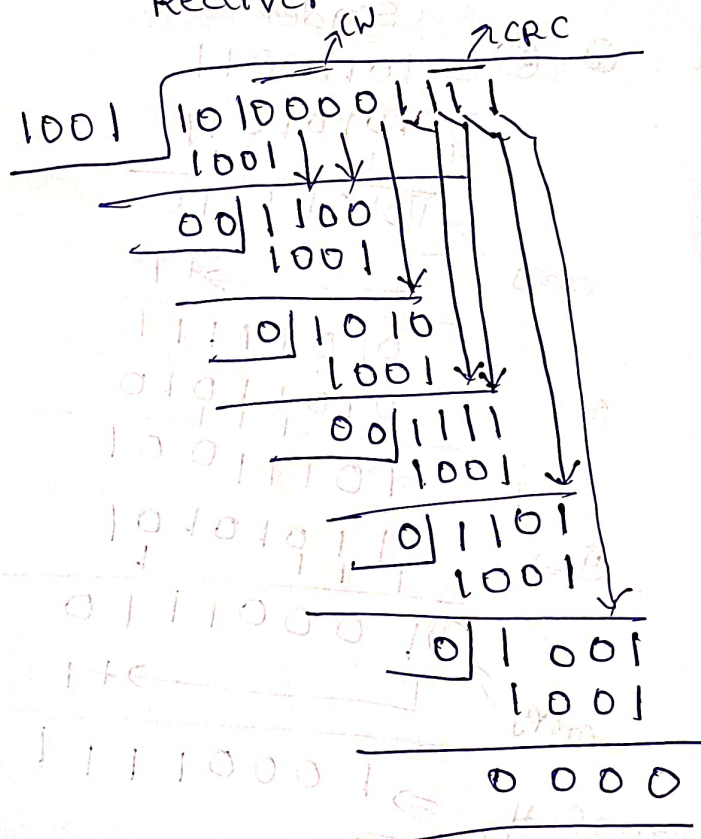
~~CRC~~ $\Rightarrow 111$

CRC

Along with Codeword we have to send CRC.

Codeword + CRC at Receiver Side.

Receiver



No-Error.

2) checksum: It is used for error detection during data transmission. Its primary purpose is to ensure the integrity of data by verifying whether any errors or corruption occurred during transit from sender to receiver.

Example :-

10110011 | 10101011 | 01011010 | 11010101
 ① ② ③ ④

Sender

① + ② ⇒

$$\begin{array}{r} 10110011 \\ + 10101011 \\ \hline 01011110 \end{array}$$

carry → +1

③ ⇒

$$\begin{array}{r} 01011111 \\ + 01011010 \\ \hline 10111001 \end{array}$$

④ ⇒

$$\begin{array}{r} 11010101 \\ + 10001110 \\ \hline 10001111 \end{array}$$

carry → +1

Sum of 4 segments ⇒ 10001111

Compliment of Sum is ⇒ 01110000

Receiver

Sum of 4 segments + Compliment

$$\begin{array}{r} 10001111 \\ + 01110000 \\ \hline 11111111 \end{array}$$

← Compliment

00000000

No Error

3) cyclic Redundancy Check (CRC) : CRC is a different approach to detect if the received frame contains valid data. This technique involves binary division of the data bits being sent. The divisor is generated using polynomials. The sender performs a division operation on the bits being sent and calculates the remainder. Before sending the actual bits, the sender adds the remainder at the end of the actual bits. Actual data bits plus the remainder is called a codeword. The sender transmits data bits as codewords.

→ Determine if there are errors in received codeword

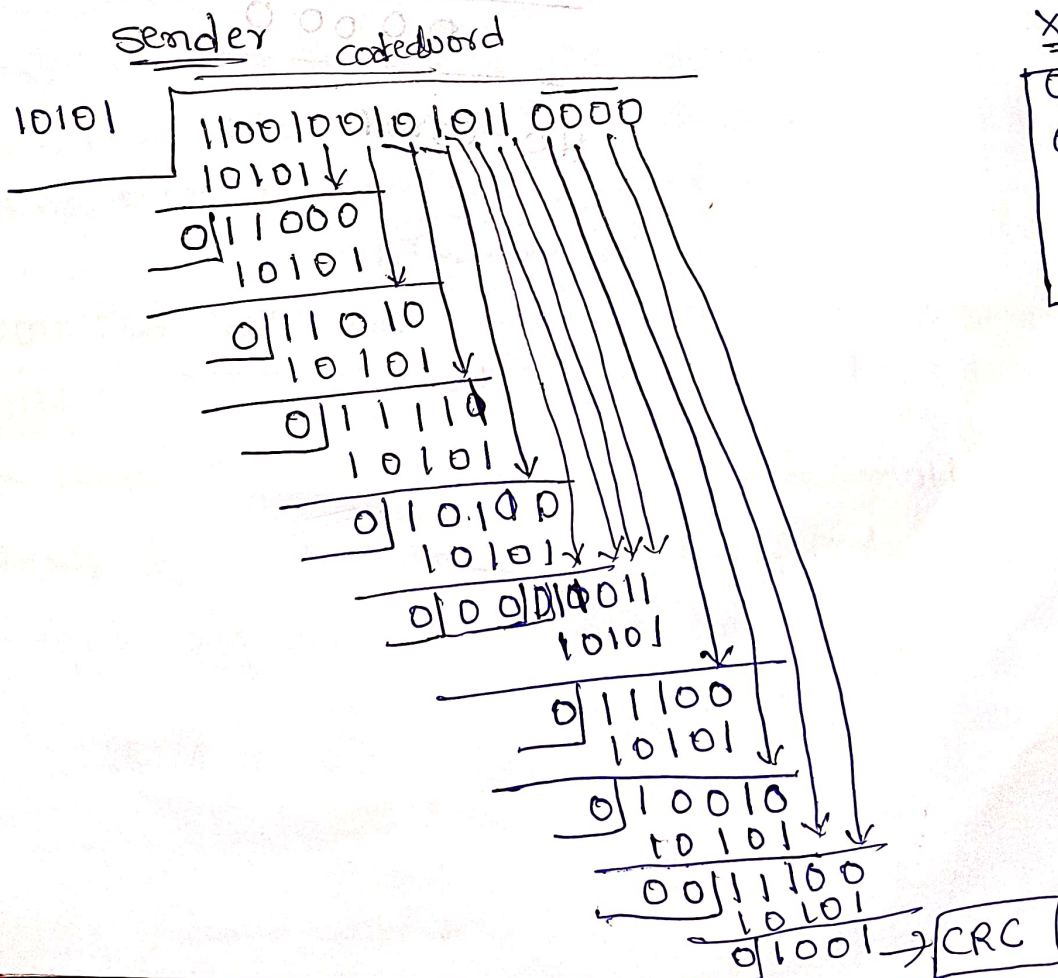
1100100101011 when the polynomial is 10101.

sol:-

Given data $M(x) = 1100100101011$

polynomial $G(x) = 10101$

$\Rightarrow 5 - 1 = 4$
we have to add 4 zero's at the end of the codeword.

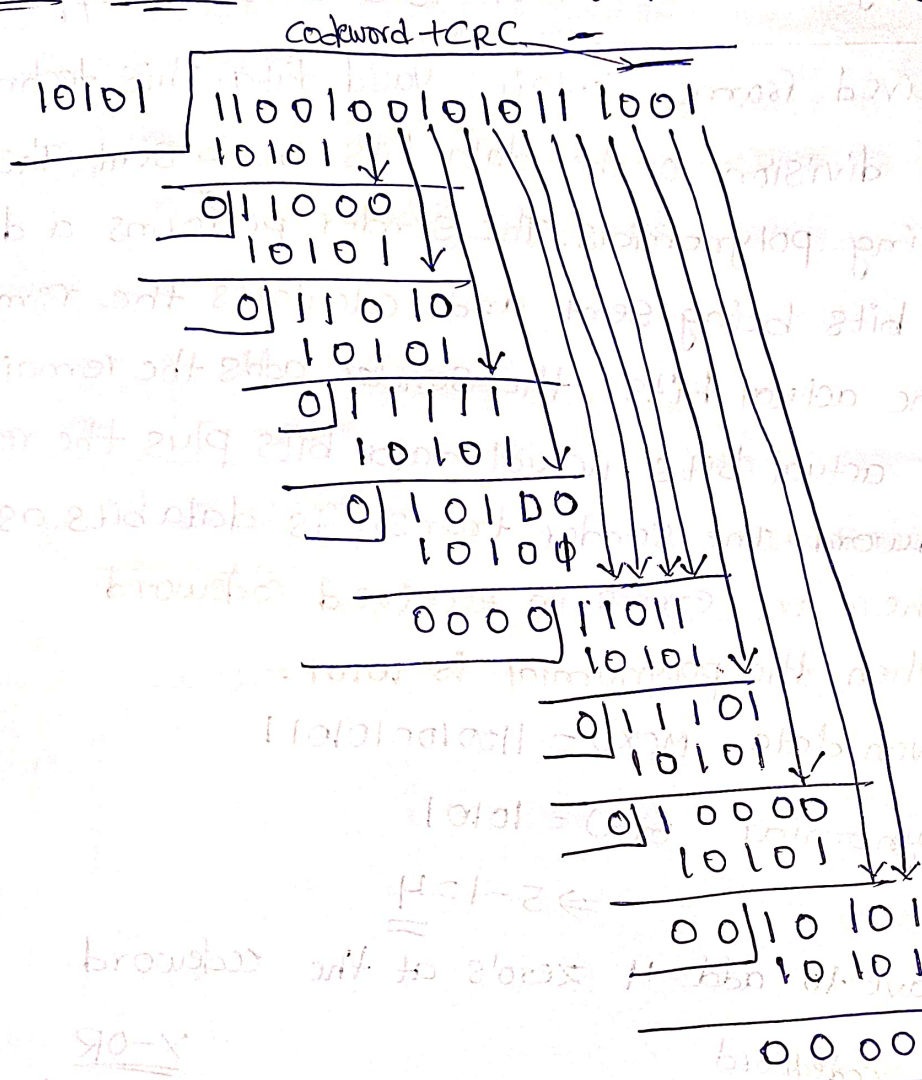


X-OR

0	0	0
0	1	1
1	0	1
1	1	0

At Receiver Side : Codeword + CRC we have to send

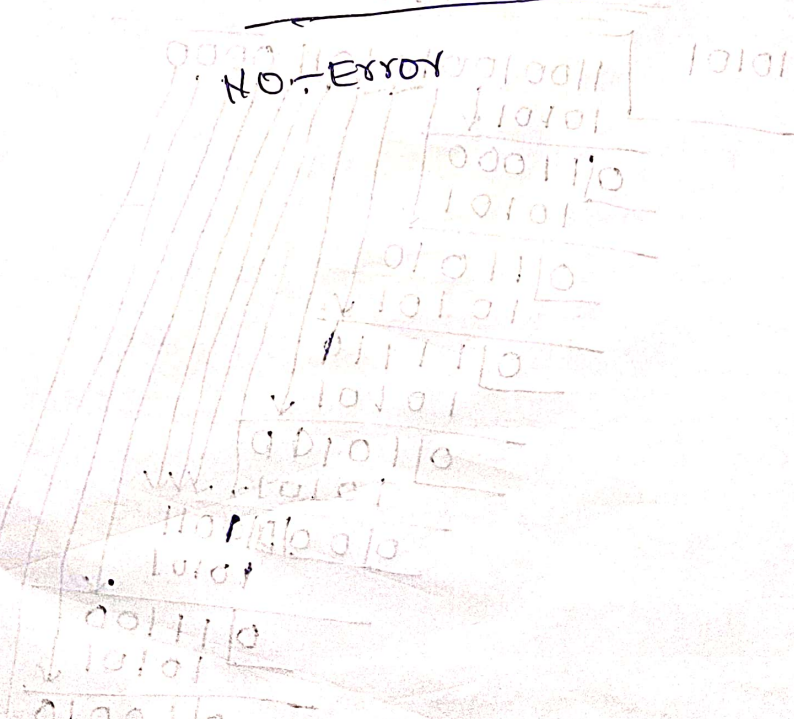
8



2-0K

0	0	0
1	1	0
1	0	1
0	1	1

No Error



→ Error correction: In the digital world, error correction can be done in two ways.

1) Backward Error correction: When the receiver detects an error in the data received, it requests back the sender to retransmit the data unit.

2) Forward Error correction: When the receiver detects some error in the data received, it executes error-correcting code, which helps it to auto-recover and to correct some kind of errors.

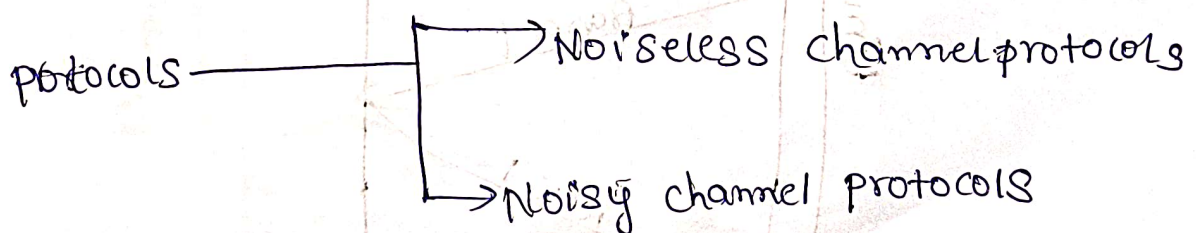
→ The first one Backward Error correction, is simple and can only be efficiently used where retransmitting is not expensive. Example: fiber optics. But in case of wireless transmission retransmitting may cost too much.

→ For m data bits, r redundant bits are used. r bits can provide 2^r combinations of information. In $m+r$ bit codeword, there is possibility that the r bits themselves may get corrupted. So the number of r bits used must inform about $m+r$ bit locations plus no-error information (i.e) $m+r+1$.

$$2^r \geq m+r+1$$

→ PROTOCOLS: Noiseless channels & Noisy channels protocols in DLL:

→ To transmit the data from one node to another, data link layer combines framing, flow control & error control schemes. protocols are divided into two categories.



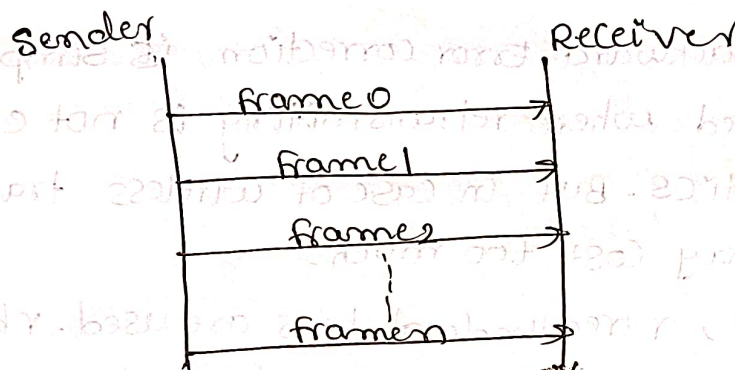
16
1) Noisless Channels: These channels are error free channels.

The following two protocols are introduced for noiseless channel.

1) Simplest protocol

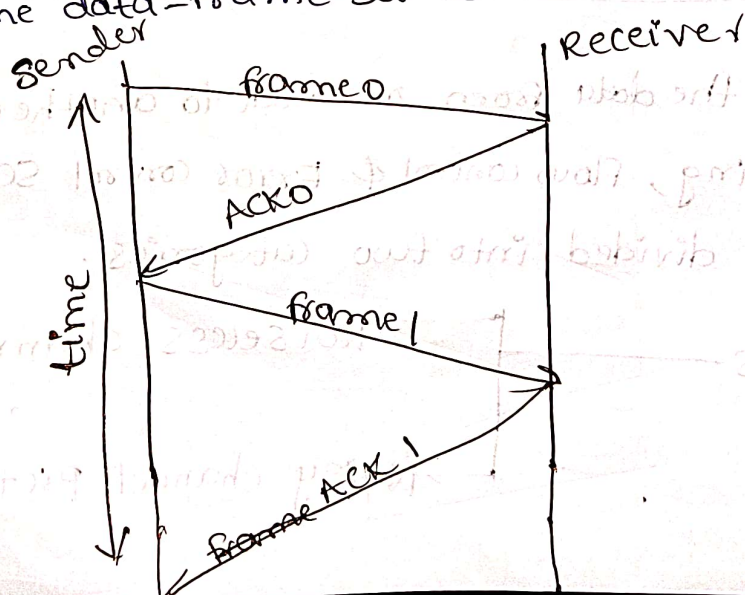
2) Stop & wait protocol

1) Simplest protocol:- According to this protocol the target has to process the received frame that is Short Intervals. It is a unidirectional protocol that has no errors or flow control.



Transmission of data is only in single direction i.e. simplex transmission methodology is adopted. In this there are no error that takes place in physical layer.

2) Stop and wait protocol: In this the sender after transmitting a data frame to stop & wait until the acknowledgement of the data-frame set is received.



→ In this it uses a timer, once the sender forwards the frame it should receive the ACK from the receiver before the timer expires. If the timer expires the previous frame is forwarded again to the receiver. This indicates the sender needs to keep the copy of frame until it receives ACK.

2) Noisy channels :-

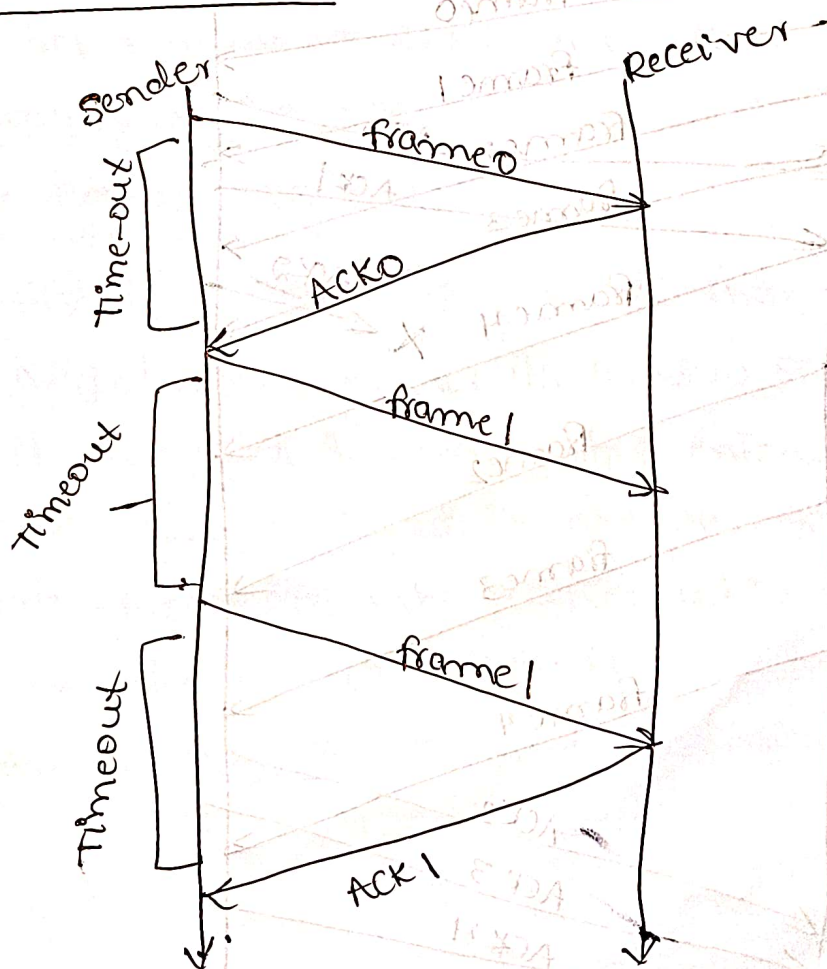
→ These channels are called error creating channels. They ignore the error or generate the error control mechanisms. The following three protocols are introduced for Noisy channels.

1) Stop-and-wait ARQ (Automatic Repeat Request)

2) Go-back-N - ARQ

3) Selective Repeat ARQ

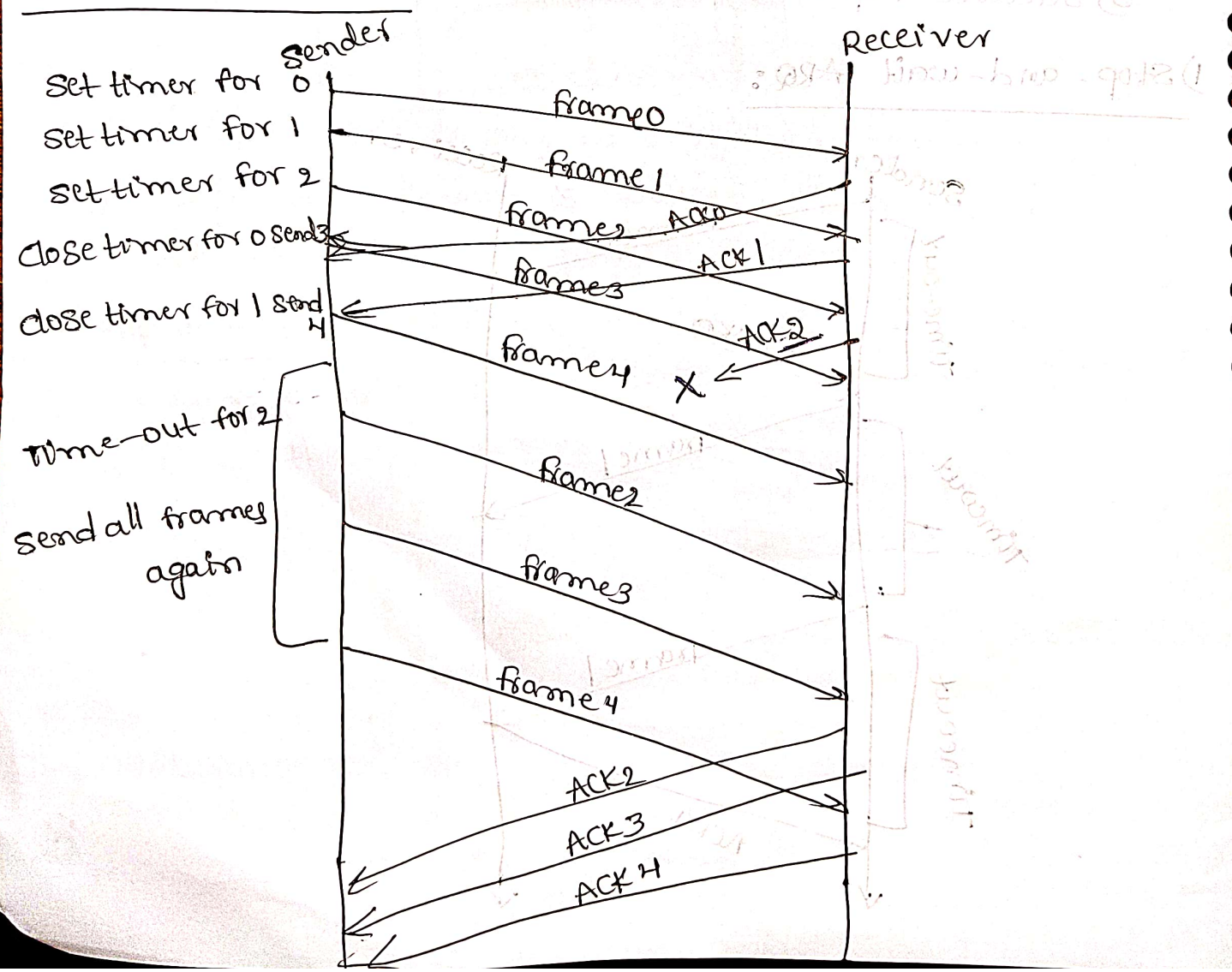
1) Stop-and-wait ARQ:



→ In Stop-and-wait ARQ the sender transmits the frame and then waits for the response. The receiver after receiving the frame sends positive Acknowledgement (ACK) frame if the frame received correctly and negative Acknowledgement (NAK) frame if the data frame receiver contains errors.

→ In some situations it is also possible the ACK frame transmitted by Receiver may be lost. If sender does not receive any ACK before timer expires. It will retransmit the frame. This duplicates frame at the receiver. In order to avoid this each frame is numbered by sequence number. In this ARQ, only one frame is transmitted at a time.

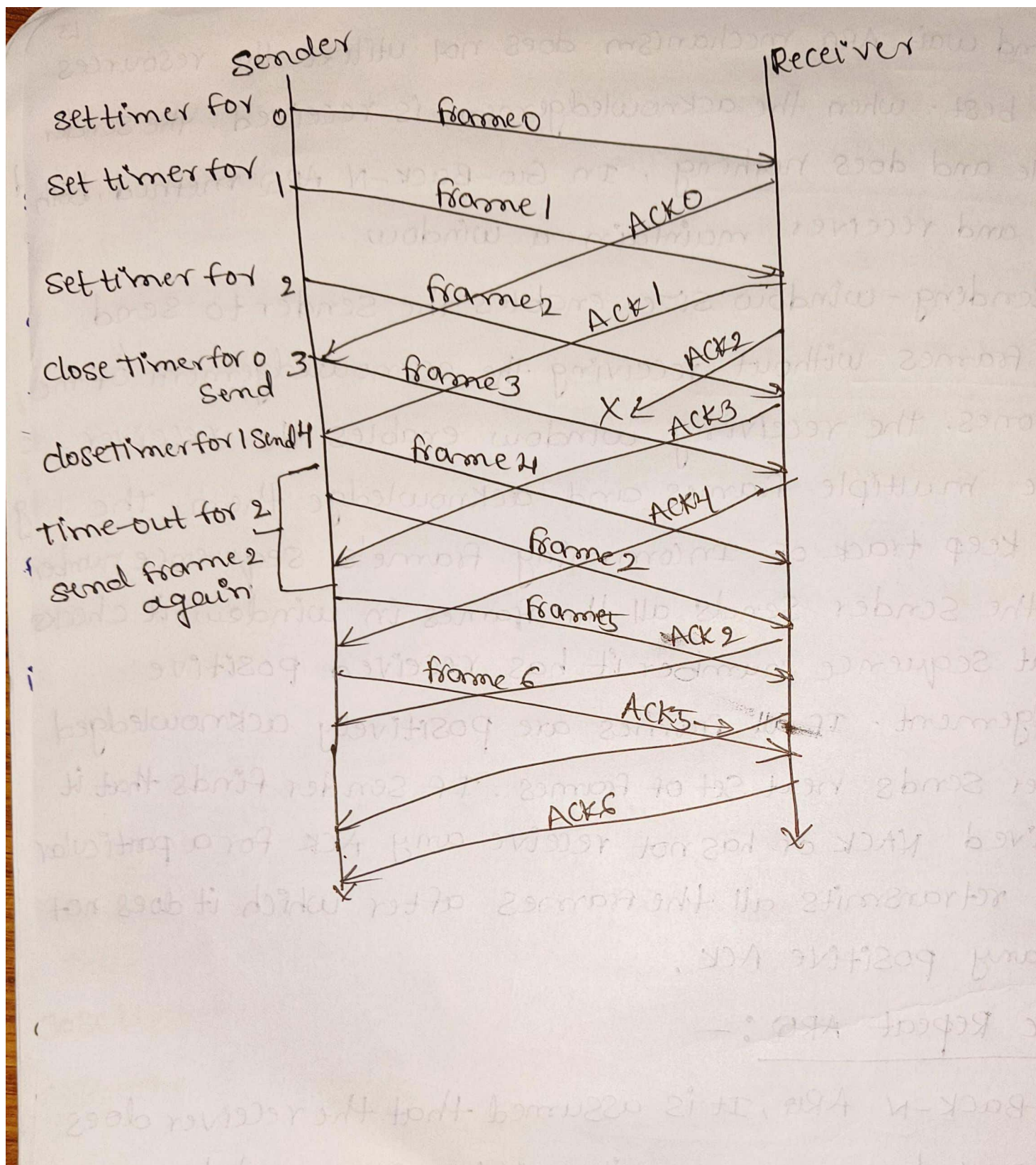
2) Go-Back-N-ARQ :

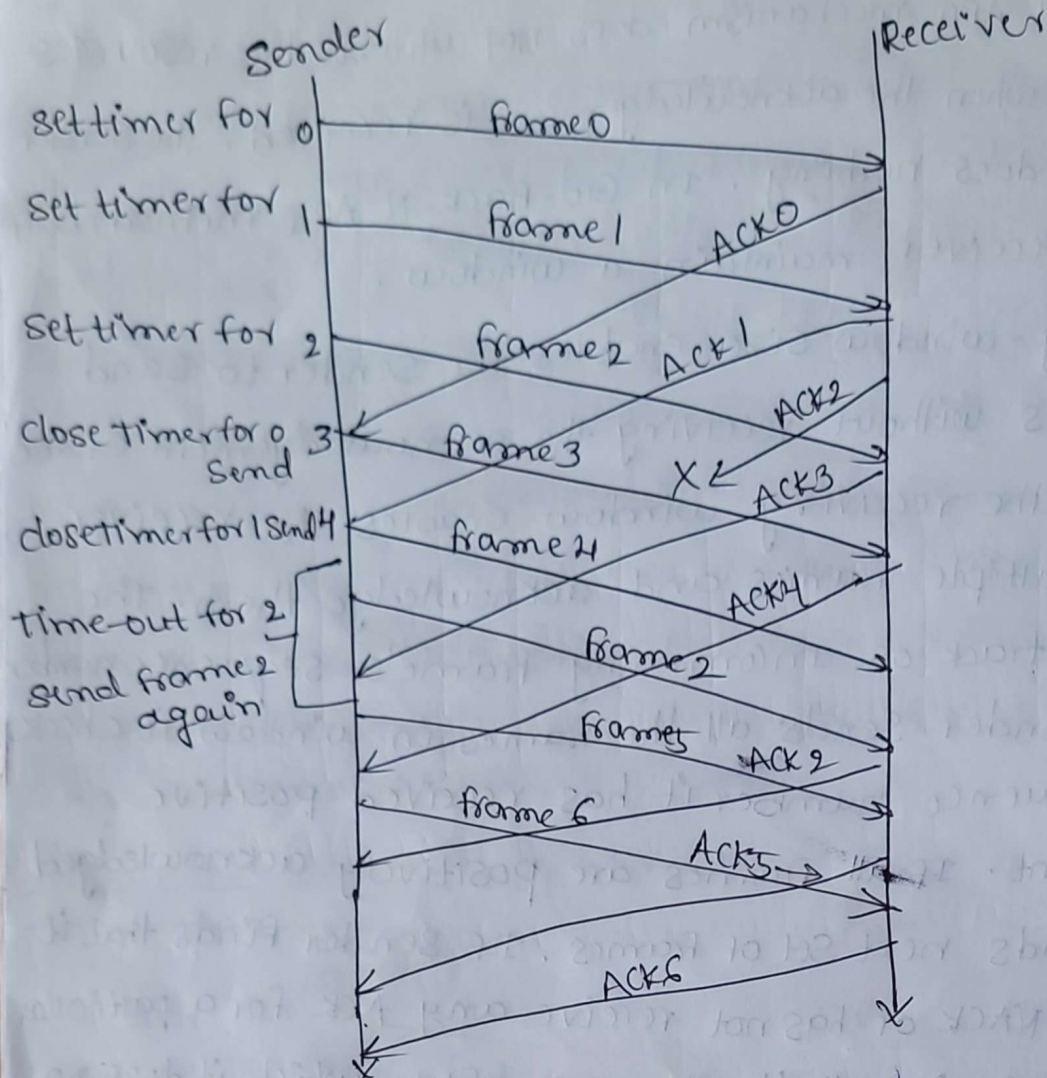


- Stop and wait ARQ mechanism does not utilize the resources at their best. When the acknowledgement is received, the sender sits idle and does nothing. In Go-Back-N ARQ method, both sender and receiver maintain a window.
- The sending-window size enables the sender to send multiple frames without receiving the acknowledgement of the previous ones. The receiving-window enables the receiver to receive multiple frames and acknowledge them. The receiver keeps track of incoming frame's sequence number.
- When the sender sends all the frames in window, it checks up to what sequence number it has received positive acknowledgement. If all frames are positively acknowledged, the sender sends next set of frames. If sender finds that it has received NACK or has not received any ACK for a particular frame, it retransmits all the frames after which it does not receive any positive ACK.

3) Selective Repeat ARQ: -

- In Go-Back-N ARQ, it is assumed that the receiver does not have any buffer space for its window size and has to process each frame as it comes. This enforces the sender to retransmit all the frames which are not acknowledged.
- In Selective Repeat ARQ, the receiver while keeping track of sequence numbers, buffers the frames in memory and sends NACK for only frame which is missing or damaged.





→ The 802.11 MAC Sublayer protocol or wifi or (WLAN) :

Wifi or WLAN uses high frequency radio waves instead of cables for connecting the devices in LAN. Users connected by WLANs can move around within the area of Network coverage.

Avoidance of collision by 802.11 MAC Sublayer :

→ In wireless systems, the method of collision detection does not work. It uses a protocol called carrier sense multiple Access with Collision avoidance (CSMA/CA).
 → When a frame is ready, the transmission station checks whether the channel is idle or busy.

- 15
- If the channel is idle, the station waits for an inter-frame gap (IFG) amount of time and then sends the frame.
 - After sending the frame, it sets a timer.
 - The station then waits for acknowledgment from the receiver. If it receives the acknowledgement before expiry of timer, it makes a successful transmission.
 - otherwise it waits for a back-off time period & restarts the algorithm.

Co-ordination Functions in 802.11 MAC Sublayer:

IEEE 802.11 MAC sublayer uses two co-ordination functions for collision avoidance before transmission.

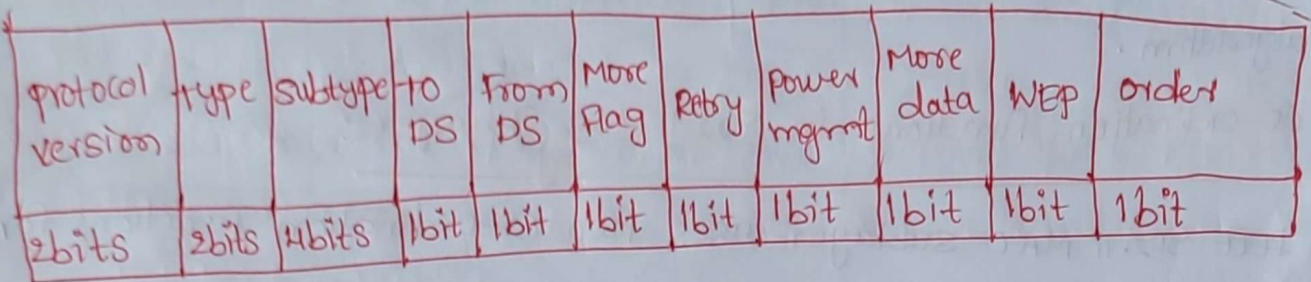
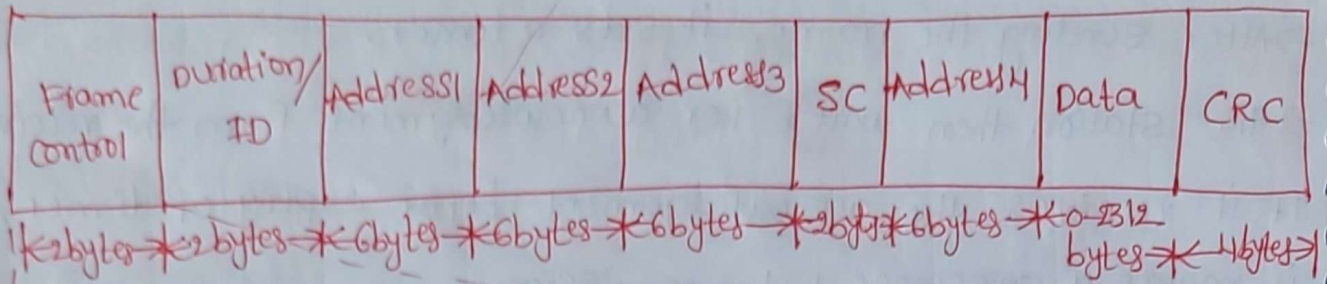
- 1) Distributed co-ordination function (DCF)
- 2) point co-ordination function (PCF).

Distributed Co-ordination Function: It uses CSMA/CA as access method as wireless LAN can't implement CSMA/CD. It only offers asynchronous service.

point Co-ordination Function: It is implemented on top of DCF and mostly used for time sensitive transmission. It uses centralized connection free polling access method. It offers both asynchronous and time-bounded service.

→ IEEE 802.11 Frame Format :

16



Frame Control (FC): It is 2 bytes long field which defines type of frame and some control information. Various field present in FC are,

- i) version: It is a 2 bit long field which indicates the current protocol version which is fixed to be 0 for now.
- ii) Type: It is a 2 bit long field which determines the function of frame (ie) Management (00), control (01) or data (10). The value 11 is reserved.
- iii) Subtype: It is a 4 bit long field which indicates subtype of the frame like 0000 for association request, 1000 for beacon.
- iv) To DS: It is a 1 bit long field which when set indicates that destination frame is for DS (distribution system).
- v) From DS: It is a 1 bit long field which when set indicates frame coming from DS.

(vi) More Flag: (More Fragments): It is 1 bit long field which when set to 1 means frame is followed by other fragments.

(vii) Retry: It is 1-bit long field. If the current frame is a retransmission of an earlier frame this bit is set to 1.

(viii) power Mgmt: (power Management): It is 1-bit long field that indicates the mode of a station after successful transmission of a frame. Set to 1 the field indicates that the station goes into power save mode. If the field is set to 0, the station stay active.

(ix) More data: It is 1-bit long field that is used to indicate receiver that a sender has more data to send than the current frame. This can be used by an access point to indicate to a station in power-save mode that more packets are buffered or it can be used by a station to indicate to an access point after being polled that more polling is necessary as the station has more data ready to transmit.

(x) WEP: It is a 1-bit long field which indicates that the (wired equivalent privacy) standard security mechanism of 802.11 is applied.

(xi) Order: It is 1-bit long field. If this bit is set to 1 the received frames must be processed in strict order.

2) Duration/ID: It is 2 bytes long field which contains the value indicating the period of time in which the medium is occupied (in μs).

3) Address (1 to 4): These are 6 bytes long fields which contain in standard IEEE 802 MAC address (48 bit each). The meaning of each address depends on the DA bits in the frame control field.

4) SC (Sequence Control): It is 16 bits long field which consists of 2 sub-fields (ie) sequence number (12 bits) & Fragment number (4 bits). Since acknowledgement mechanism frames may be duplicated hence, a sequence number is used to filter duplicate frames.

5) Data: It is a variable length field which contain information specific to individual frames which is transferred transparently from sender to the receivers.

6) CRC (Cyclic Redundancy Check): It is 4 bytes long field which contains a 32 bit CRC error detection sequence to ensure error free frame.

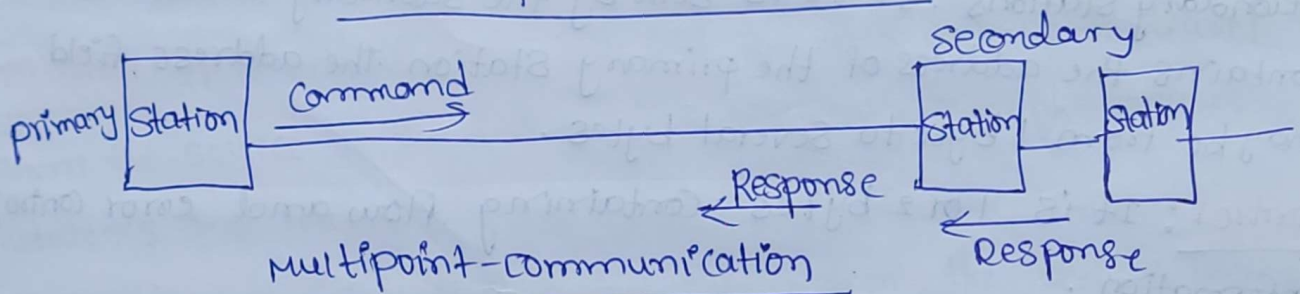
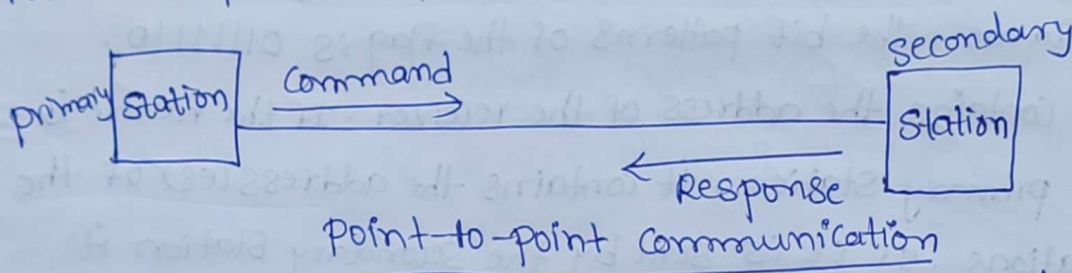
→ HDLC: (High-Level Data Link Control):

→ High-level data link control (HDLC) is a group of communication protocols of the data link layer for transmitting data between network points or nodes. Since it is a data link protocol, data is organized into frames.

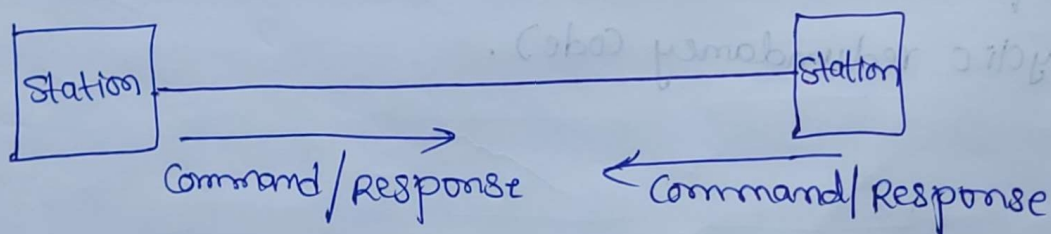
→ A frame is transmitted via the network to destination that verifies its successful arrival. It is a bit-oriented protocol that is applicable for both point-to-point & multipoint communications.

Transfer Modes: HDLC supports two types of transfer modes, normal response mode and asynchronous balanced mode.

i) Normal Response Mode (NRM): Here two types of stations are there, a primary station that send commands and secondary station that can respond to received commands. It is used for both point-to-point and multipoint communications.

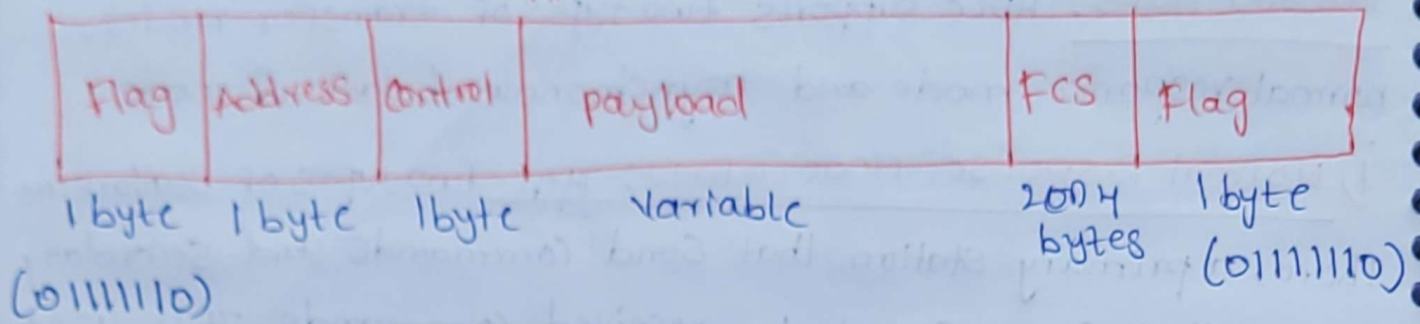


ii) Asynchronous Balanced Mode (ABM): Here, the configuration is balanced (i.e.) each station can both send commands and respond to commands. It is used for only point-to-point communications.



HDLC Frame: HDLC is a bit-oriented protocol where each frame contains up to six fields. The structure varies according to the type of frame. The fields of a HDLC frame are:

HDLC Frame



Flag: It is an 8-bit sequence that marks the beginning and the end of the frame. The bit patterns of the flag is 01111110.

Address: It contains the address of the receiver. If the frame is sent by the primary station, it contains the address(es) of the secondary stations. If it is sent by the secondary station it contains the address of the primary station. The address field may be from 1 byte to several bytes.

Control: It is 1 or 2 bytes containing flow and error control information.

payload: This carries the data from the network layer. Its length may vary from one network to another.

FCS (Frame Check Sequence): It is a 2 byte or 4 byte frame check sequence for error detection. The standard code used is CRC (Cyclic redundancy code).

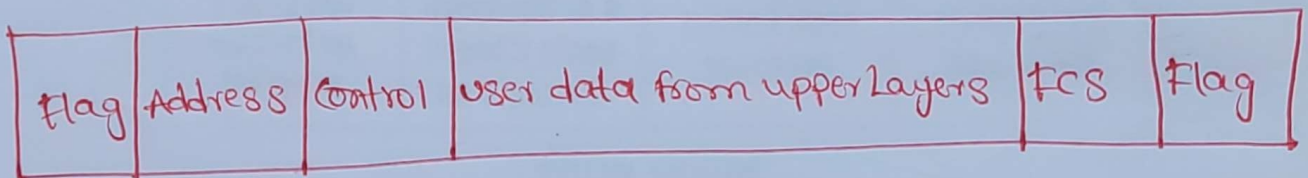
Types of HDLC Frames: There are three types of HDLC frames, the types of frame is determined by the control field of the frame.

i) I-frame: I-frame or Information frames carry user data from the network layer. They also include flow and error control information that is piggybacked on user data. The first bit of control field of I-frame is 0.

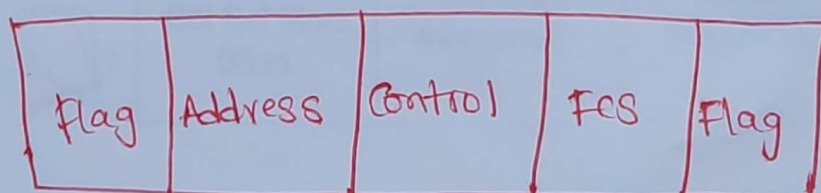
ii) S-frame: S-frame or Supervisory frames does not contain information field. They are used for flow and error control when piggybacking is not required. The first two bits of control field of S-frame is 10.

iii) U-frame: U-frames or un-numbered frames are used for myriad miscellaneous functions, link link management. It may contain an information field, if required. The first two bits of control field of U-frame is 11.

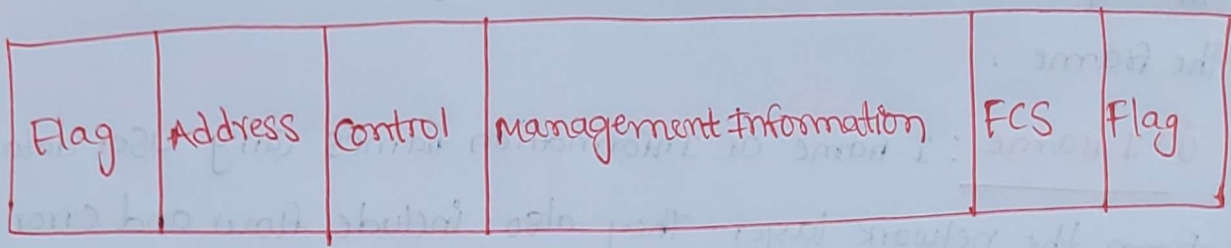
I-frame:



S-frame:



U-Frame:



Multiple Access protocols:

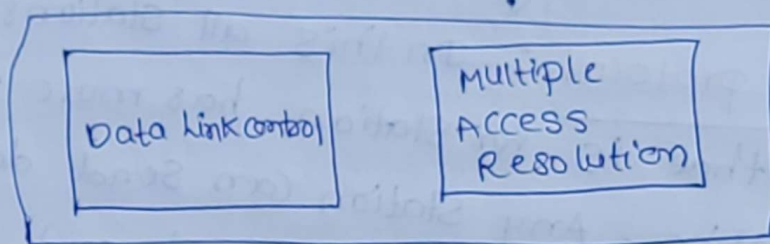
→ Multiple Access protocols are methods used in computer networks to control how data is transmitted when multiple devices are trying to communicate over the same network.

→ These protocols ensure that data packets are not sent and received efficiently, without collisions or interference. They help manage the network traffic so that all devices can share the communication channel smoothly and effectively.

→ The Data Link Layer is responsible for the transmission of data between two nodes. Its main functions are:

→ Data Link Control

→ Multiple Access Control

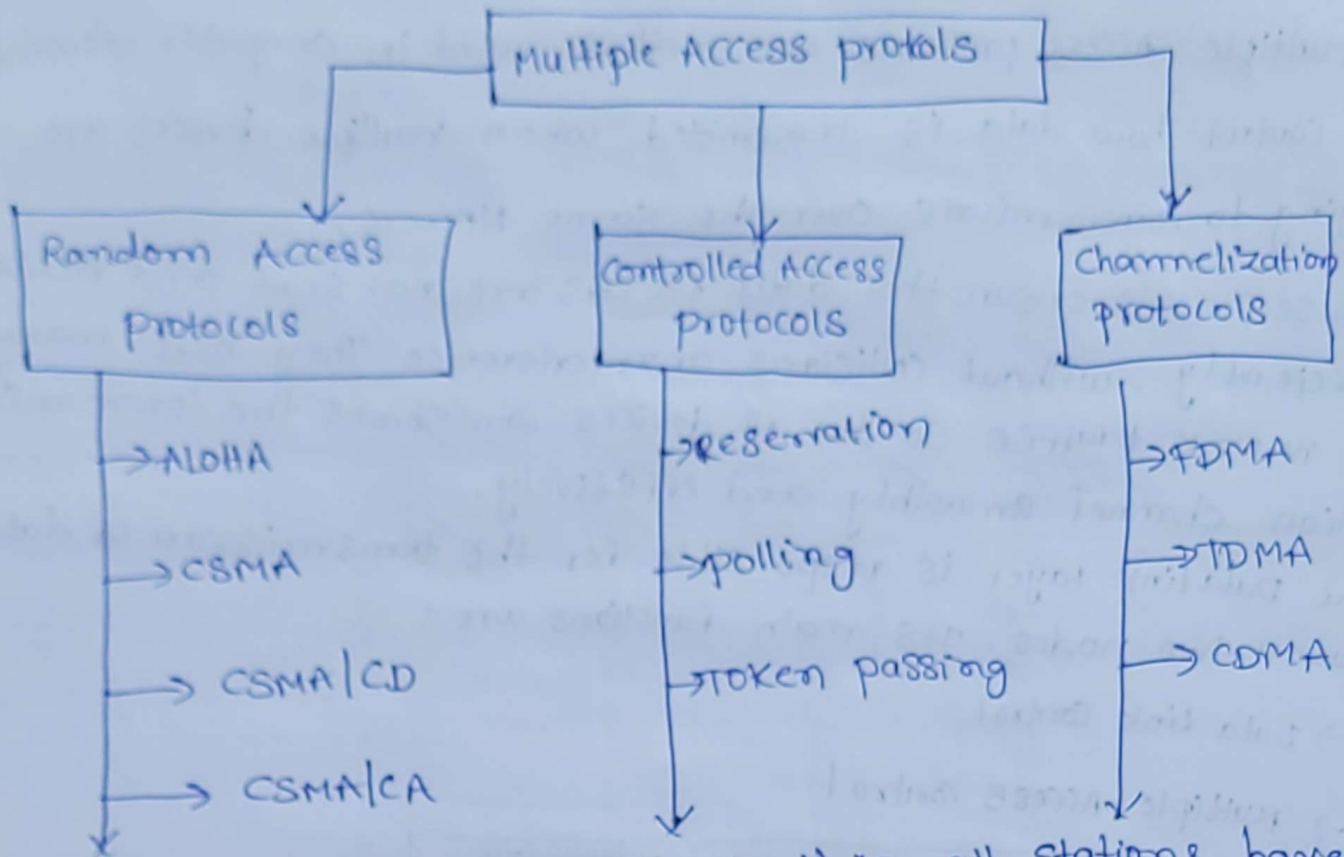


Data Link Layer Functions

Data Link Control: The Data Link Control is responsible for the reliable transmission of messages over transmission channels, by using techniques like framing, error control and flow control. For Datalink refer to stop & wait ARQ.

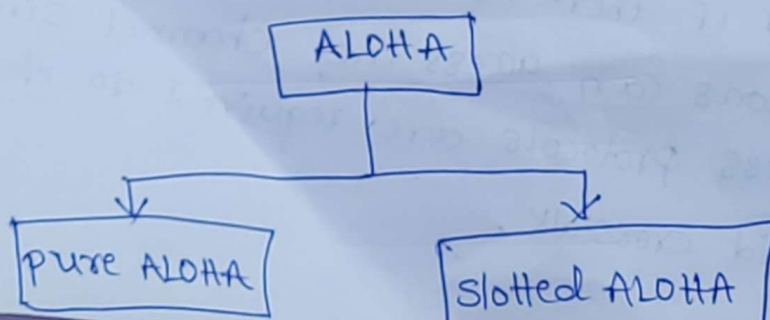
Multiple Access Control: If there is a dedicated link between the sender and the receiver then data link control layer is sufficient, however if there is no dedicated link present then multiple stations can access the channel simultaneously. Hence Multiple Access protocols are required to decrease collision and avoid crosstalk.

→ Thus, protocols are required for sharing data on non dedicated channels. Multiple Access protocols can be subdivided further as



1) Random Access protocol: In this all stations have same superiority that is no station has more priority than another station. Any station can send data depending on medium's state (idle or busy). It has two features:

- There is no fixed time for sending data
- There is no fixed sequence of stations sending data.
- The random access protocols are further subdivided as:



ALOHA: It was designed for wireless LAN but is also applicable for shared medium. In this multiple stations can transmit data at the same time and can hence lead to collision and data being garbled.

Pure ALOHA: pure ALOHA is an algorithm states that every station is allowed to send the data whenever it wants to send. But the problem is that there may be chance of collision and thus, the collided frames will be destroyed.

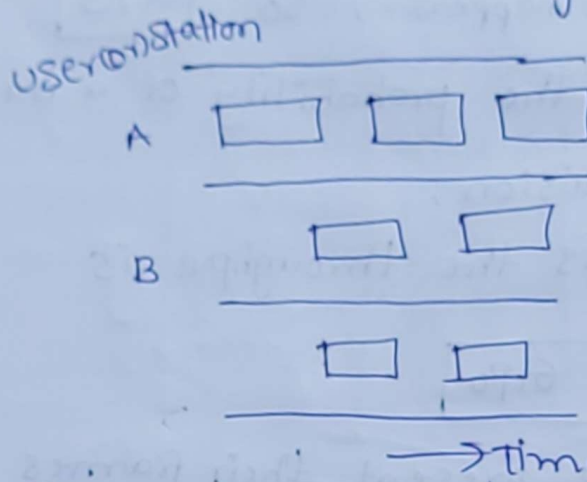


Fig: Frame Generation in an pure ALOHA System

→ The station A, B, C have their own frames. It suppose two frames wants to use a common channel at the same time, the collision will occur. With the help of this ALOHA channel, the station will be in two modes, typing or waiting.

→ Initially, all station prepare a frame and when the frame is ready, the station transmits it and checks the channel to see if it is successful.

→ If it is successful then the station goes for preparing the frames. If it is not successful then the station has to wait and the frame will be retransmitted again & again until the frame is successfully sent.

let S stands for station

K stands for Retransmission frames in a frame time.

G is mean no. of frames.

→ If pure ALOHA is at low load ($S \approx 0$) there might be very few collision, so $G \approx S$

→ If pure ALOHA is at high load ($S \approx 1$) there might be no. of collisions can happen, so $G \approx S$

→ let p_0 be denoted for the probability of a frame which does not suffer a collision.

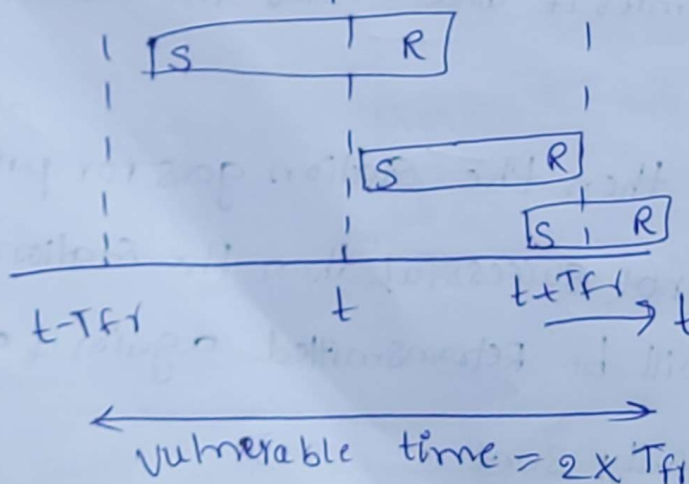
→ For low and high loads the throughput is

$$S = G p_0$$

→ If the station wants to present their frames with collision, then the probability that K frame, S, are generated during a given frame time is given by the poisson distribution.

$$P[K] = \frac{G^K e^{-G}}{K!}$$

Vulnerable time of pure ALOHA:



vulnerable time is defined as the length of the time at which collision may take place.

$$\text{vulnerable time of pure ALOHA} = 2 \times T_{fr}$$

Throughput of pure ALOHA: The Average no. of Successful transmissions is

$$S = G \times e^{-2G}$$

→ Slotted ALOHA: In Slotted ALOHA, time is splitted into slots of T_{fr} and in this a station is allowed to send frames only at the beginning of the time slot.

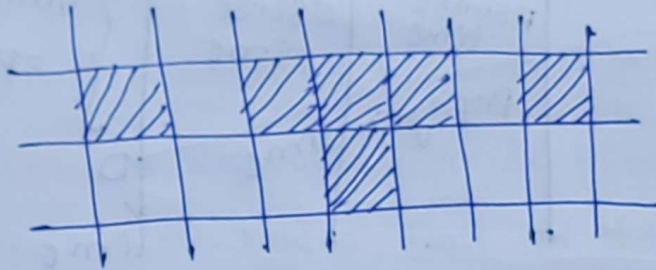


Fig: Frames in Slotted ALOHA N/W

If any station cannot send frames at the beginning of time slot then it must wait until the beginning of the next time slot. This indicates that the station which started at the beginning of this slot has already finished sending its frame. Of course if two stations send frames at the beginning of the same time slot then collision may take place, but vulnerable time is decreased to T_{fr} , which is one-half of pure ALOHA.

$$\text{Vulnerable time of Slotted ALOHA} = T_{fr}$$

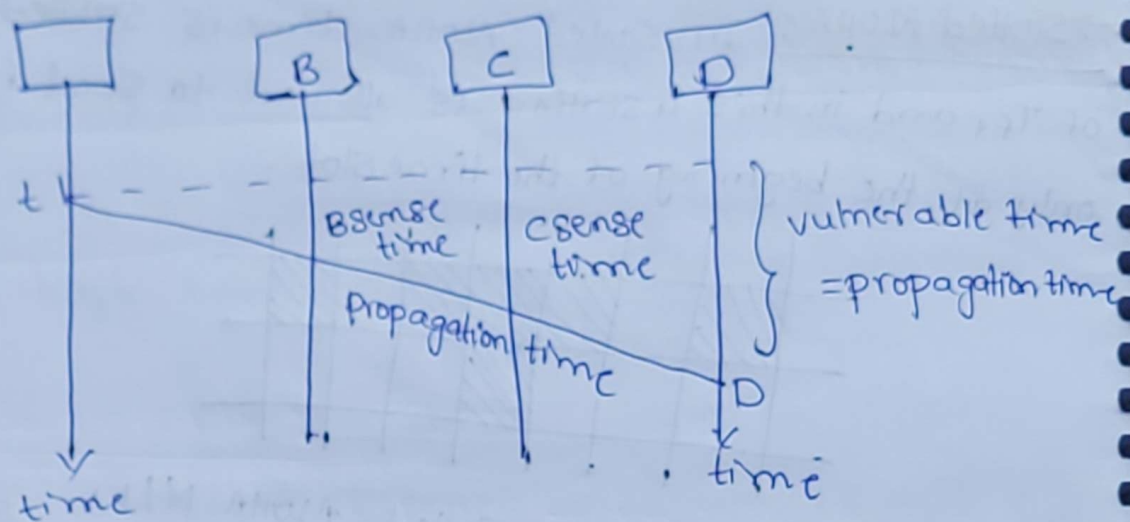
$$\text{Throughput for Slotted ALOHA} \Rightarrow S = G \times e^{-G}$$

26
Carrier Sense Multiple Access (CSMA): This method was

developed to decrease the chance of collisions when two or more stations start sending their signals over the data link layer.

→ Carrier sense multiple access requires that each station first check the state of the medium before sending.

vulnerable time = propagation time (t_p)



→ The persistence methods can be applied to help the station take action when the channel is busy/idle.

Types of modes in CSMA: There are 4 types of access modes available in CSMA. It is also referred as 4 different types of CSMA protocols which decide the time to start sending data across shared media.

1-persistent: If senses the shared channel first and delivers the data right away if the channel is idle. If not, it must wait and continuously track for the channel to become idle and then broadcast the frame without condition as soon as it does.

2) Non-persistent: It first assesses the channel before transmitting data. If the channel is Idle the node transmits data right away. If not, the station must wait for an arbitrary amount of time (not continuously), and when it discovers the channel is empty, it sends the frames.

3) p-persistent: It consists of the 1-persistent and non-persistent modes combined. Each node observes the channel in the 1-persistent mode, and if the channel is Idle, it sends a frame with a probability p . If the data is not transferred, the frame restarts with the following time slot after waiting for a $(q = 1 - p)$ probability random period.

4) 0-persistent: A supervisory node gives each node a transmission order. Nodes wait for their time slot according to their allocated transmission sequence when the transmission medium is Idle.

Advantages of CSMA:

- Increased Efficiency: CSMA ensures that only one device communicates on the network at a time, reducing collisions and improving network efficiency.
- Simplicity: It is a simple protocol that is easy to implement and does not require complex hardware or software.
- Flexibility: It is a flexible protocol that can be used in a wide range of network environments, including wired and wireless networks.

→ low cost : It does not require expensive hardware or software making it a cost-effective solution for network communication.

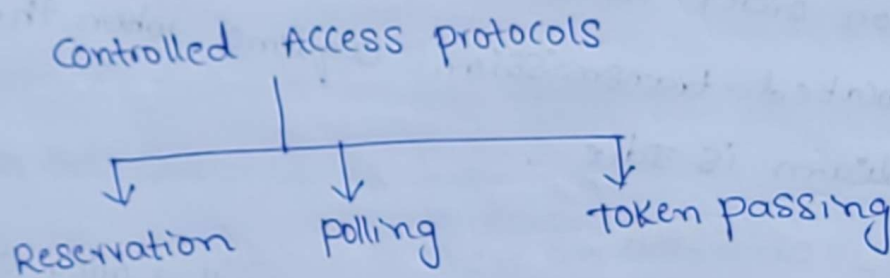
→ CSMA/CD (Carrier Sense Multiple Access with Collision Detection)

→ In this method, a station monitors the medium after it sends a frame to see if the transmission was successful. If successful the transmission is finished, if not the frame is sent again.

→ CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance)

→ The basic idea behind CSMA/CA is that the station should be able to receive while transmitting to detect a collision from different stations.

→ Controlled Access protocols :



→ Controlled Access protocols (CAP) in computer networks control how data packets are sent over a common communication medium. These protocols ensure that data is transmitted efficiently, without collisions, and with little interference from other data transmissions.

→ In controlled access, the stations seek data from one another to find which station has the right to send. It allows only one node to send at a time to avoid the collision of messages on shared medium.

→ The three controlled access methods are :

1) Reservation

2) polling

3) token passing

1) Reservation: In the reservation method, a station needs to make a reservation before sending data.

→ The timeline has two kinds of periods ;

→ Reservation interval of fixed time length ,

→ Data transmission period of variable frames ,

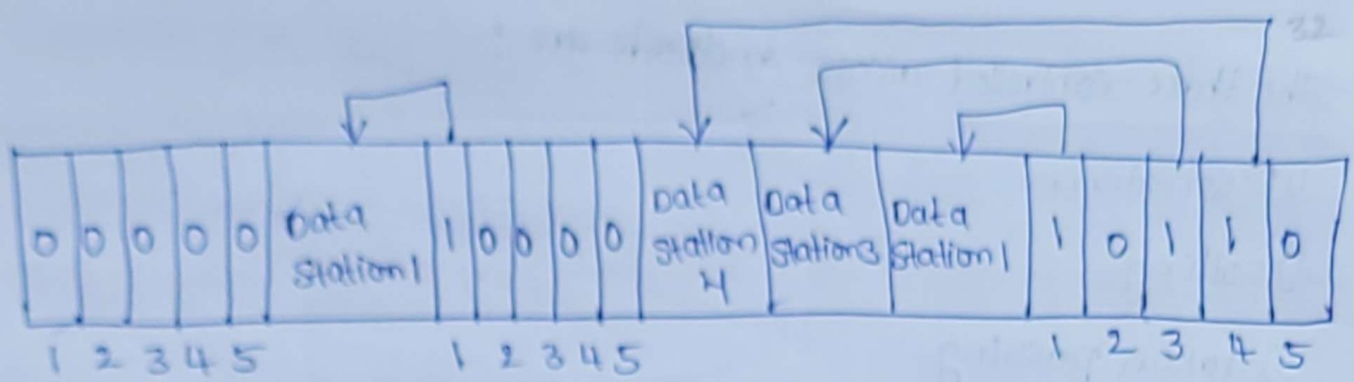
→ If there are M stations, the reservation interval is divided into M slots and each station has one slot.

→ Suppose if station 1 has a frame to send, it transmits 1 bit during the slot 1. No other station is allowed to transmit during this slot.

→ In general, i th station may announce that it has a frame to send by inserting a 1 bit into i th slot. After all N slots have been checked, each station knows which stations wish to transmit.

→ The stations which have reserved their slots transfer their frames in that order. After data transmission period, next reservation interval begins.

→ Since every one agrees on who goes next, there will never be any collisions.



Reservation Frame

The fig shows a situation with five stations and a five-slot reservation frame. In the first interval only stations 1, 3 & 4 have made reservations. In the second interval, only station 1 has made a reservation.

Advantages of Reservation:

The main advantage of reservation is high rates and low rates of data accessing time of the respective channel can be predicted easily. Here time and rates are fixed.

- priorities can be sent to provide speedier access from secondary.
- Reservation-based access methods can enable more efficient use of available bandwidth, as they allow for time and frequency multiplexing of different reservation request on the same channel.

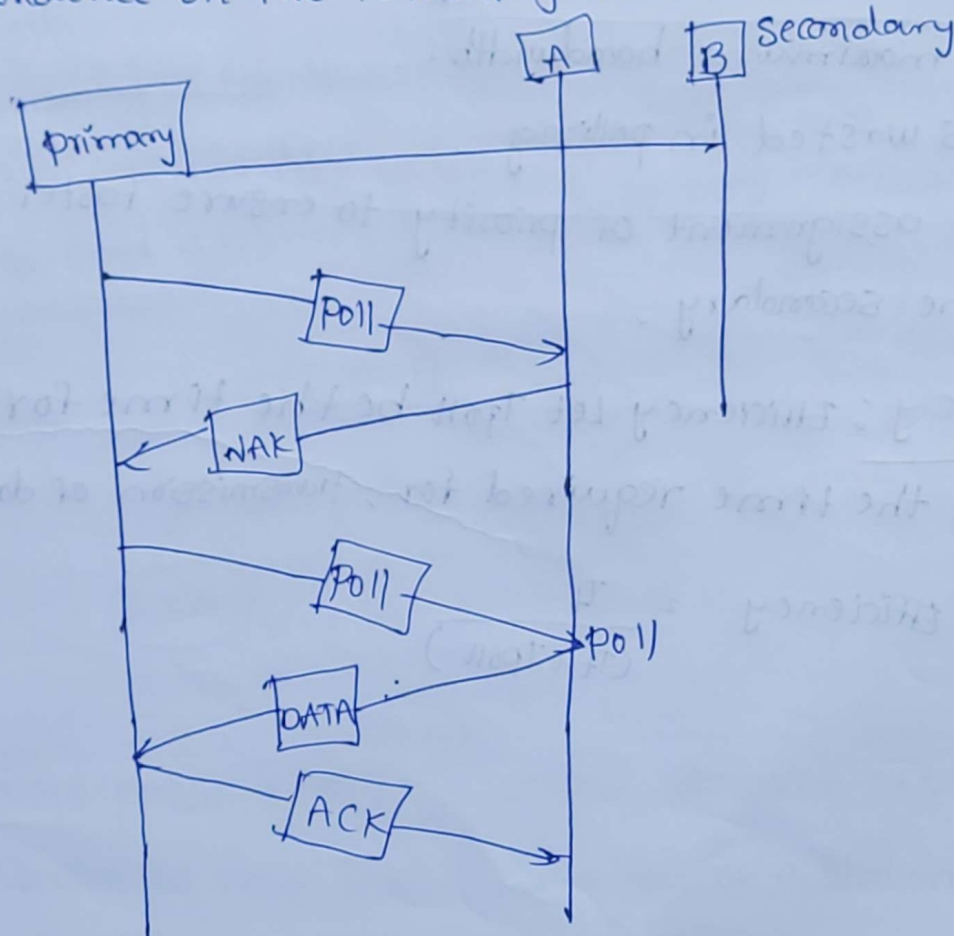
2) polling: polling process is similar to the roll-call performed in class. Just like the teacher, a controller sends a message to each node in turn.

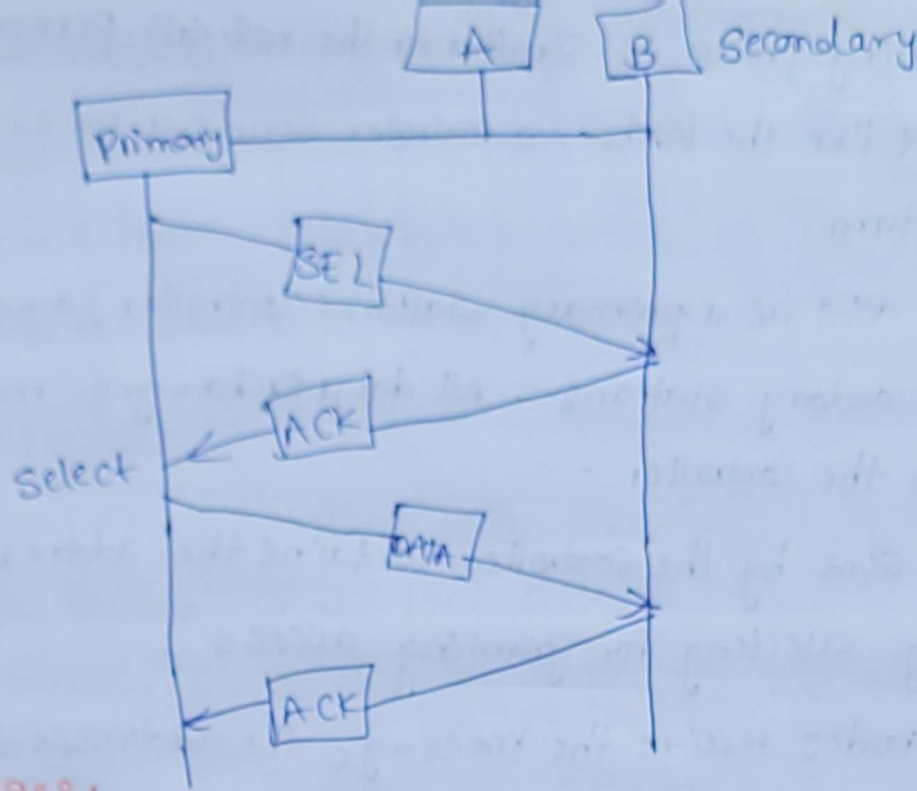
→ In this, one acts as a primary station (controller) and the others are secondary stations. All data exchanges must be made through the controller.

→ The message sent by the controller contains the address of the node being selected for granting access.

→ Although all nodes receive the message, the addressed one responds to it and sends data if any. If there is no data, usually a poll reject (NAK) message is sent back.

→ Problems include high overhead of the polling messages and high dependence on the reliability of the controller.





Advantages:

- The maximum and minimum access time and data rates on the channel are fixed predictable.
- It has maximum efficiency.
- It has maximum bandwidth.
- No slot is wasted in polling.
- There is assignment of priority to ensure faster access from some secondary.

Efficiency: Efficiency Let T_{poll} be the time for polling and T_t be the time required for, transmission of data, then,

$$\text{Efficiency} = \frac{T_t}{(T_t + T_{poll})}$$

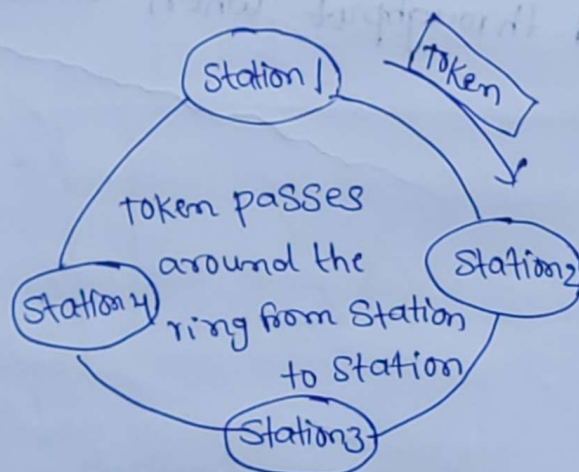
3) Token passing: In token passing scheme, the stations are connected logically to each other in form of ring and access to stations is governed by tokens.

→ A token is a special bit pattern or a small message, which circulate from one station to the next in some predefined order.

→ In token ring, token is passed from one station to another adjacent station in the ring whereas in case of token bus, each station uses the bus to send the token to the next station in some predefined order.

→ In both cases, token represents permission to send. If a station has a frame queued for transmission when it receives the token, it can send that frame ~~que~~ before it passes the token to the next station. If it has no queued frame, it passes the token simply.

→ After sending a frame, each station must wait for all N stations to send the token to their neighbours and the other $N-1$ stations to send a frame, if they have one.



→ There exists problems like duplication of token or token is lost or insertion of new station, removal of a station, which need to be tackled for correct and reliable operation of this scheme.

→ performance of token ring can be concluded by 2 parameters.

Delay: - It is a measure of time between when a packet is ready and when it is delivered. So, the average time (delay) required to send a token to the next station = a/N .

→ throughput, which is a measure of successful traffic.

Throughput, $S = \frac{1}{(1 + a/N)}$ for $a < 1$ &

$S = \frac{1}{\{a(1 + 1/N)\}}$ for $a > 1$.

where N = number of stations

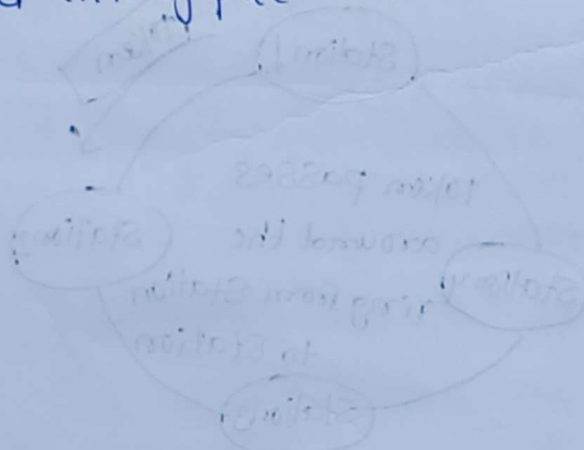
$$a = T_p / T_t$$

T_p = propagation delay & T_t = transmission delay.

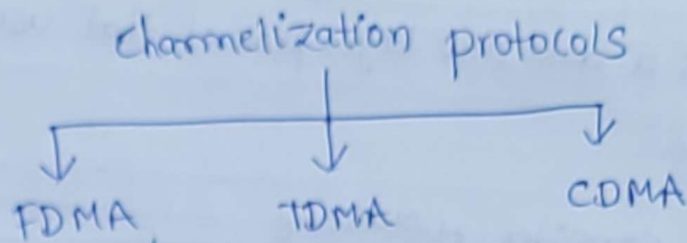
Advantages of token passing:

→ It may now be applied with router's cabling & includes built-in debugging features like protective relay & auto configuration.

→ It provides good throughput when conditions of high load.



3) Channelization protocols:

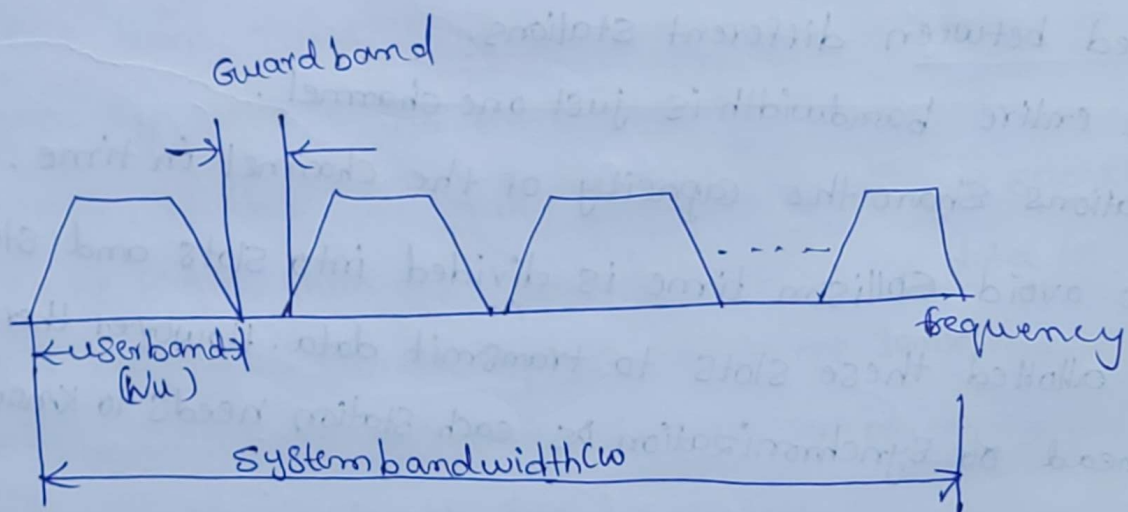


- Channelization is a multiple access protocol that divides a shared communication channel's bandwidth among multiple users, allowing them to transmit data simultaneously without collisions.
- This division can be done using frequency, time or code.
- Common channelization techniques include Frequency-Division Multiple Access (FDMA), Time-Division Multiple Access (TDMA) and Code-Division Multiple Access (CDMA).

1) Frequency Division Multiple Access (FDMA):

- The available bandwidth is divided into equal bands so that each station can be allocated its own band.
- Guard bands are also added so that no two bands overlap to avoid crosstalk and noise.

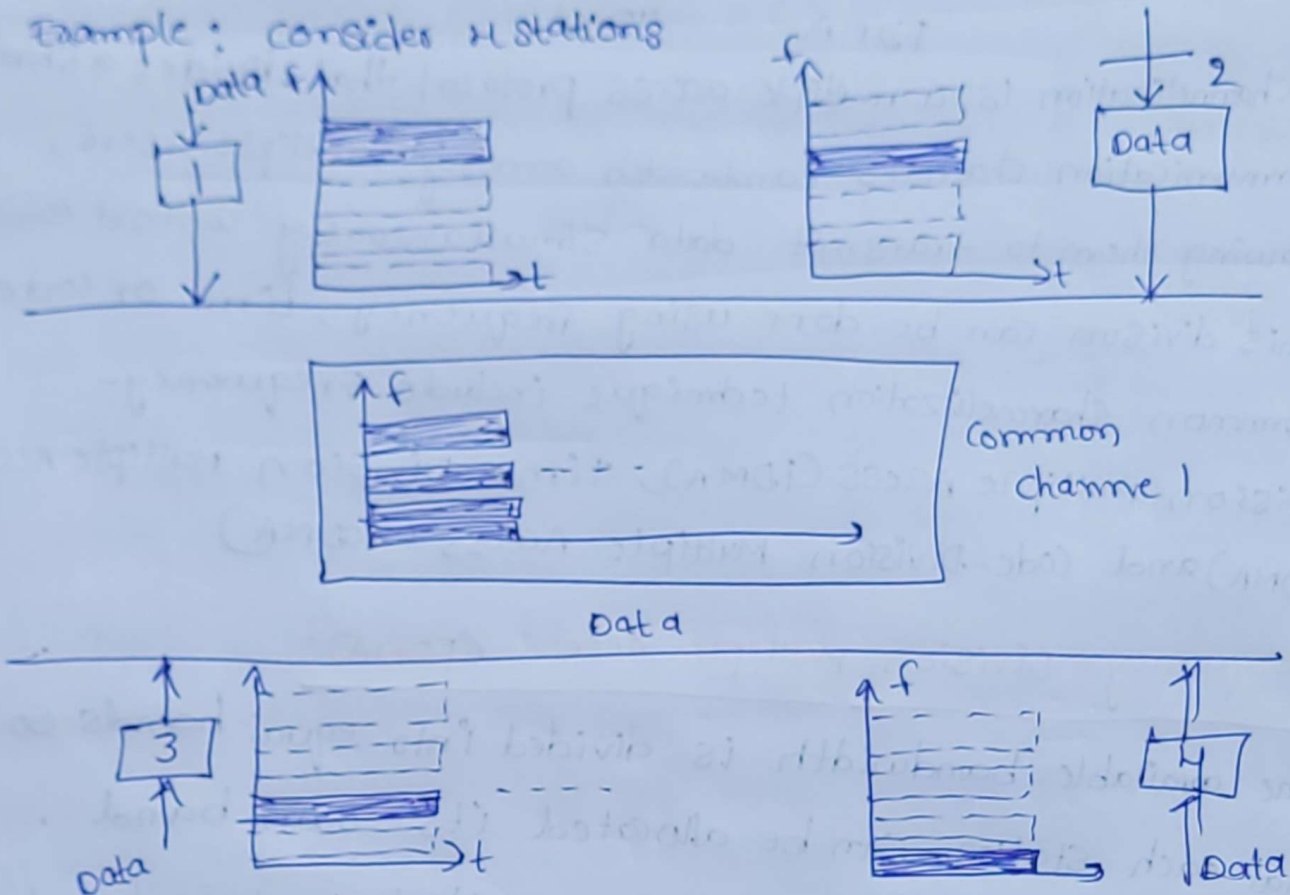
→



→ The available bandwidth is shared by all stations.

→ The FDMA is a data link layer protocol that uses FDM at the physical layer.

Example: Consider 4 stations



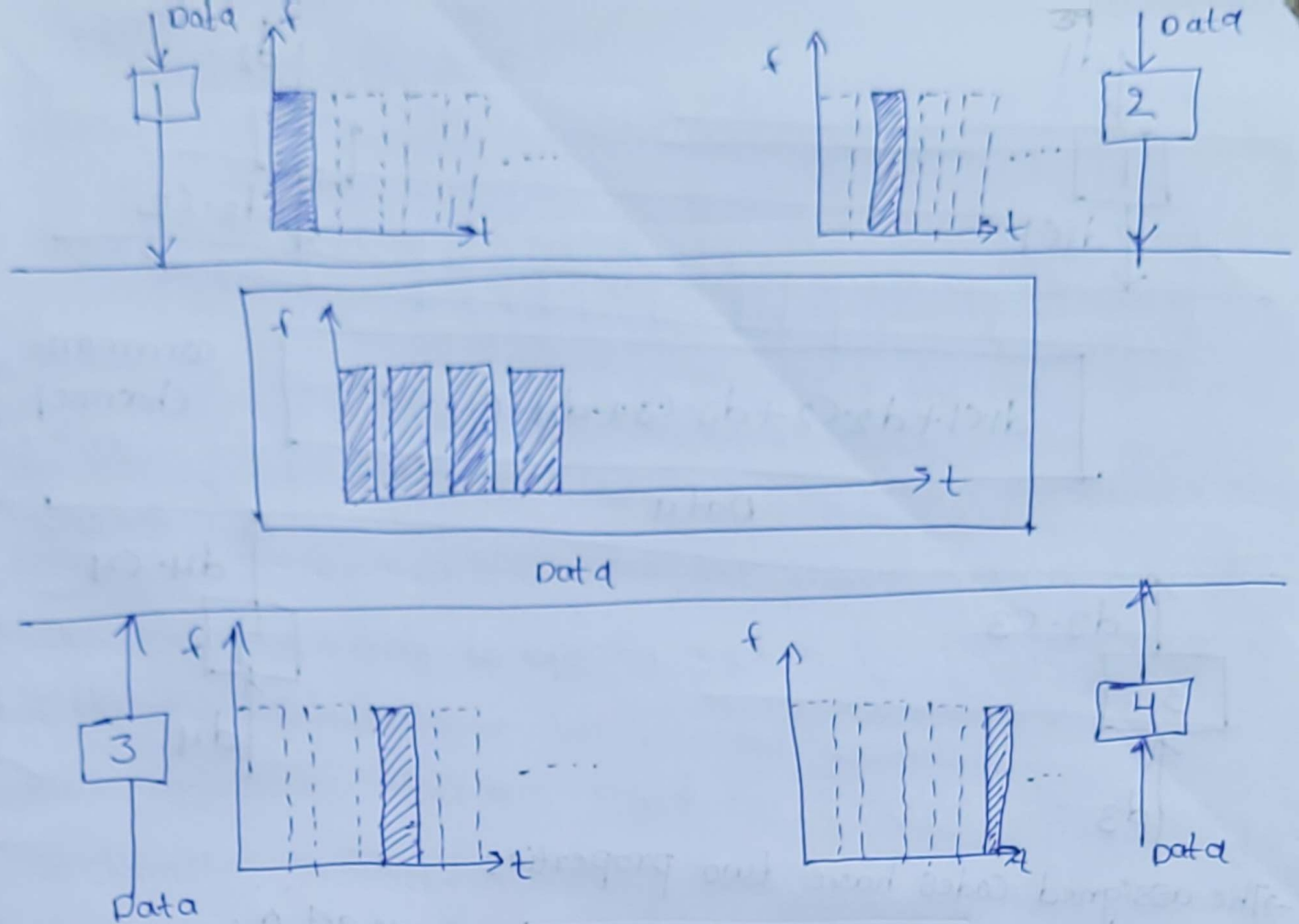
TDMA: (Time Division Multiple Access)

→ In TDMA the bandwidth is just one channel that is time shared between different stations.

→ The entire bandwidth is just one channel.

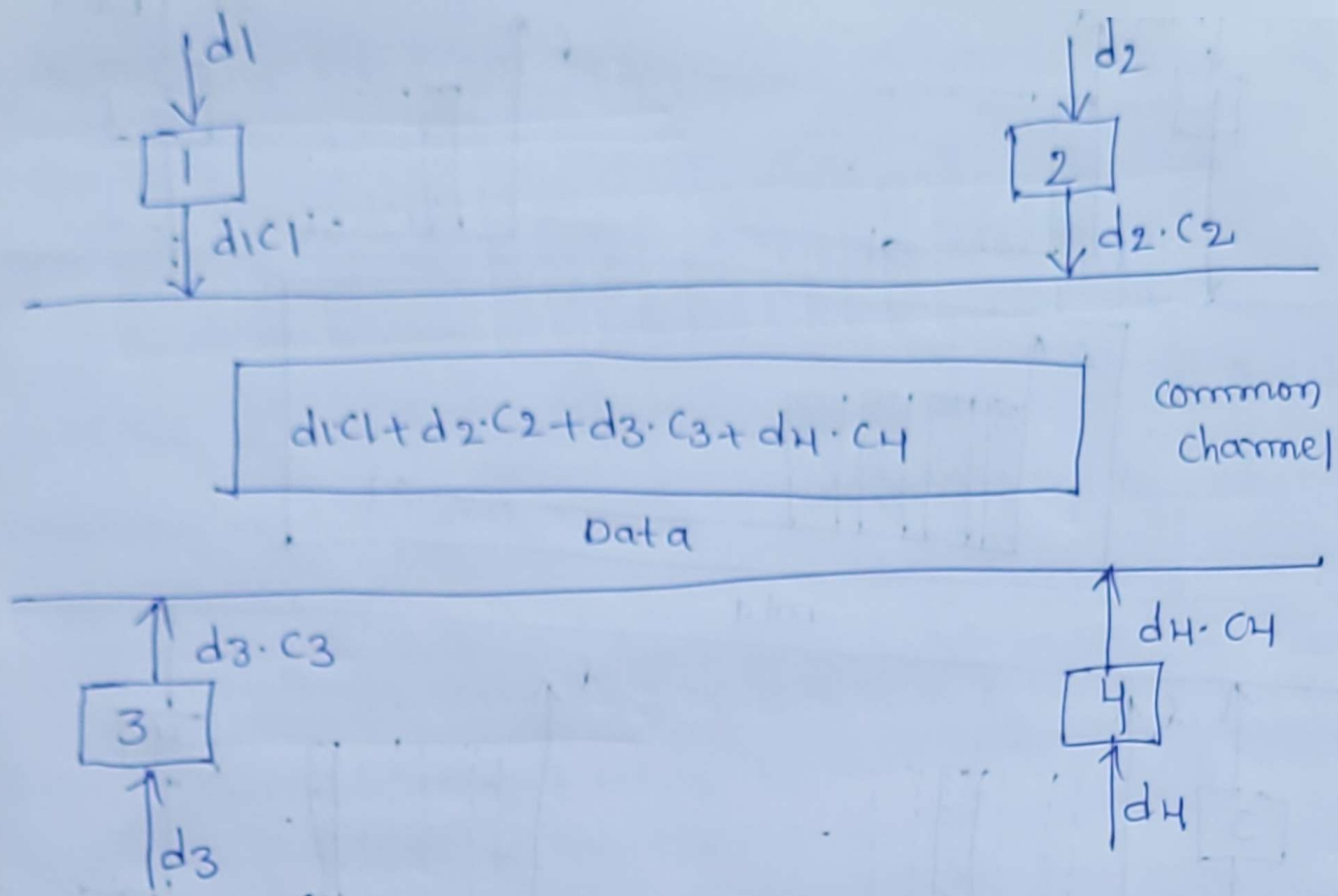
→ Stations share the capacity of the channel in time.

→ To avoid collision time is divided into slots and stations are allotted these slots to transmit data. However there is an overhead of synchronization as each station needs to know its time slot.



→ CDMA: (Code Division Multiple Access)

- In CDMA, one channel carries all transmissions simultaneously.
- CDMA differs from FDMA because only one channel occupies the entire bandwidth of the link.
- It differs from TDMA because all stations can send data simultaneously there is no time sharing.
- For example, If there are many people in a room all speaking at the same time, then also perfect reception of data is possible if only two person speak the same language. Similarly, data from different stations can be transmitted simultaneously in different code languages.



→ The assigned codes have two properties :

- 1) If we multiply each code by another we get 0.
- 2) If we multiply each code by itself, we get 1 (the number of stations).

Example : $\text{Data} = (d_1 \cdot c_1 + d_2 \cdot c_2 + d_3 \cdot c_3 + d_4 \cdot c_4) \times c_1 = 4 \times d_1$.

→ Framming: Grouping the bits together from the physical layer and constructing a frame is called framing. The main responsibility of the data link layer is to detect and correct the errors.

→ The error arises when the bits on the physical layer get scattered while reaching at the destination.

→ In order to perform this, data link layer creates frames. The frame is created by adding header and a trailer to the packet which specify the source & the destination address.

Framing are two types :

i) Fixed-size framing (Static framing): A fixed size framing does not require any frame boundaries, because the given size itself acts as a delimiter. Since, the frames are of fixed size. There is no need of specifying the start of frame.

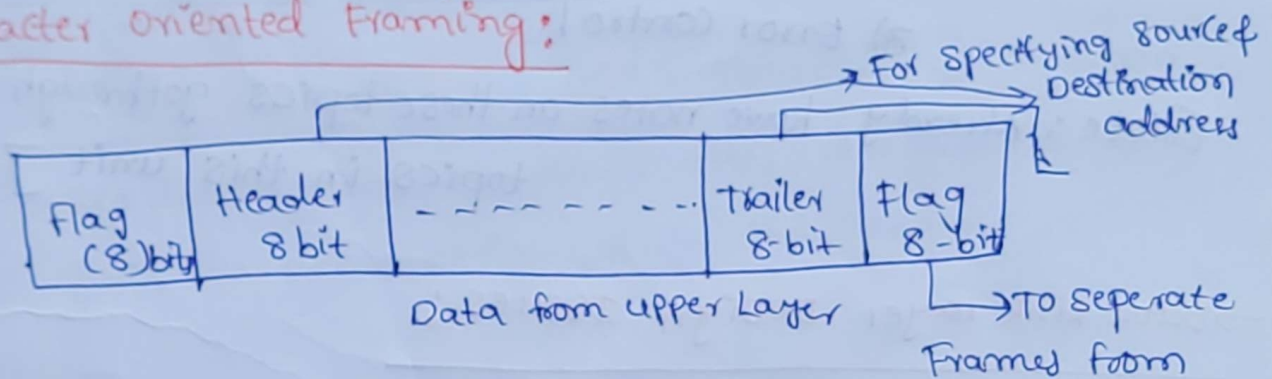
ii) Variable-size Framing (Dynamic Framing): A variable size framing is a framing in which the size of frames changes. Here it is necessary to specify end of the frame and start of the next frame.

→ two approaches that are used for variable size framing.

a) character oriented framing (or) byte oriented framing

b) Bit oriented framing.

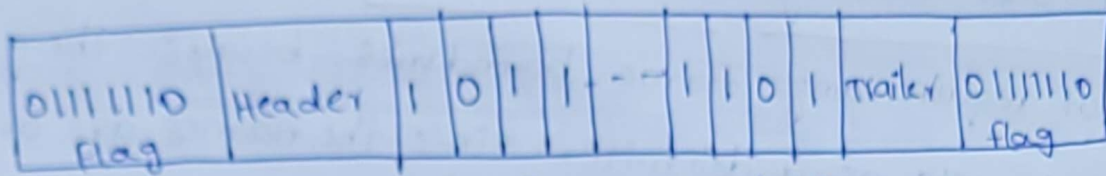
a) Character oriented Framing:



→ It is used only for text exchange b/w layers.

→ Information like graphics, audio and videos cannot be done because of the flags. Flags use character which may be part of the data. This results in misinterpretation and as it assumes end of the frame. To avoid this another strategy called character (or) byte stuffing is used.

(b) Bit-oriented Framing:



Data from upper layer

→ In bit oriented framing, a framed data is a collection of sequence of bits that are to be interpreted by the upper layer. This framed data can be a text, graphic, audio or video.

→ Data Link Layer ^{control} Services: It is a service that ensures a reliable communication between the adjacent nodes. It includes the following functions.

1) Framing

2) Flow Control &

3) Error Control

[Note: Already have notes on these topics go through the topics in this unit]

→ Data Link Layer Design Issues:

1) Services provided ~~by~~ Network Layer

2) Framing

3) Flow Control &

4) Error Control.

Services provided to Network Layer:

→ The major task of data link layer is to provide services to Network layer.

→ we have 3 types of services

(i) Unacknowledged, Connectionless Service

(ii) Acknowledged Connectionless Service

(iii) Acknowledged Connection oriented.

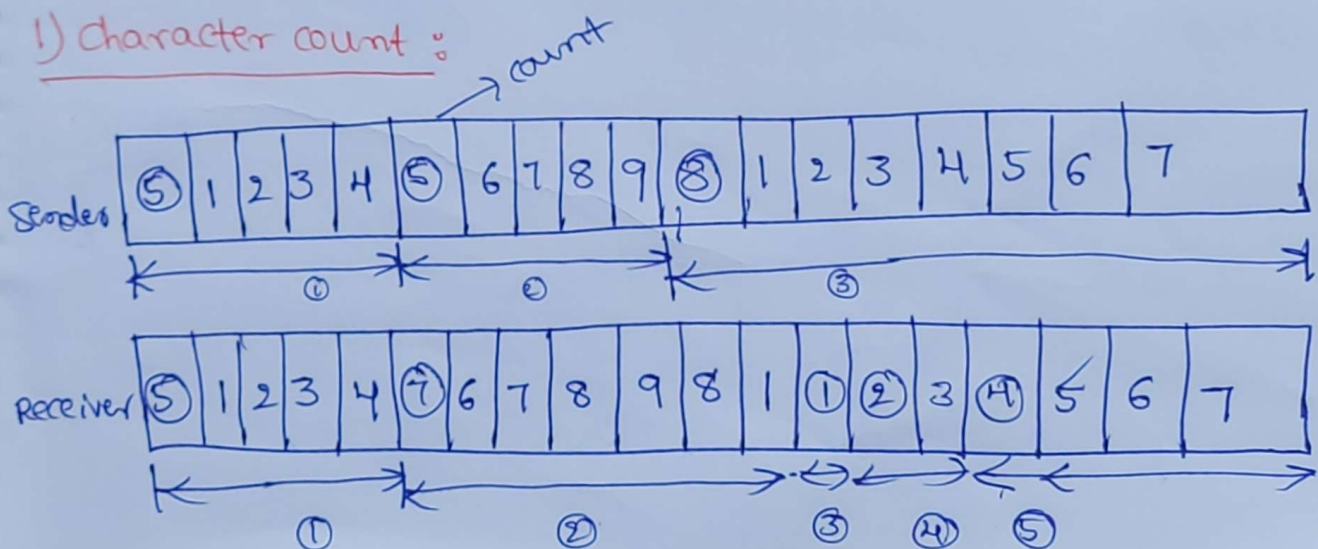
(i) Unacknowledged connectionless service: There is no connection between source & destination, and also no acknowledgment from the client machine.

(ii) Acknowledged Connectionless service: There is no connection between source & destination, and here the receiver receives the frame and give acknowledgment to sender.

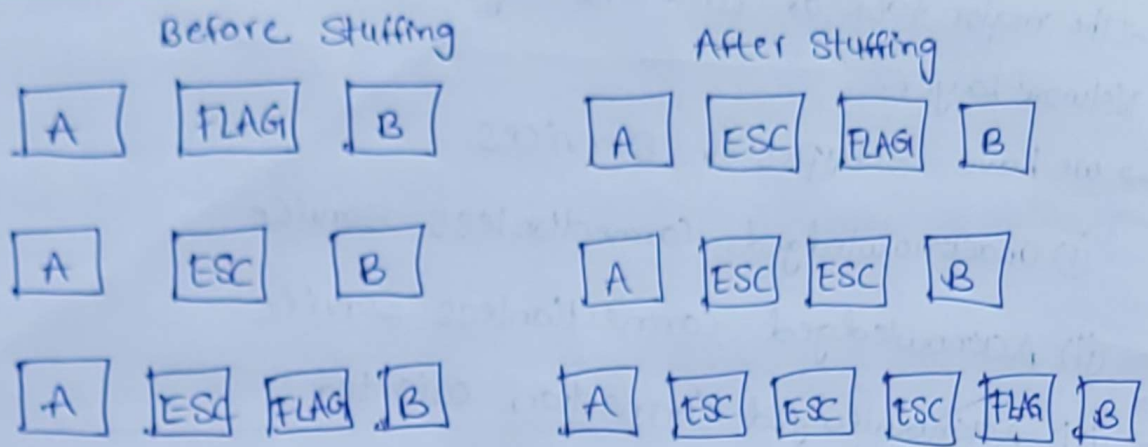
(iii) Acknowledged Connection oriented: there is a connection between sender & receiver, as well as after receives the frame receiver gives the acknowledgment to the sender.

Framing:

1) Character count :



2) Character / Byte stuffing :



3) Bit stuffing :

Ex: ^{Sender}
00111111011110110
^{Receiver}
0011111010111100110

5 consecutive 1's are
 there we stuff with
0.



Unit - III

①

The Network Layer

Introduction: The network layer, also known as Layer 3 in the OSI model, is a crucial part of Computer Network responsible for routing and forwarding data packets between different networks. → It enables communication between devices across networks by assigning logical addresses and finding the optimal path for data transmission.

Functions:

Logical Addressing: The network layer assigns logical addresses (like IP addresses) to devices, allowing them to be identified and located within a network.

Routing: It determines the best path for data packets to travel from the source to the destination, navigating through various networks and routers.

Forwarding: Based on the routing information, the network layer forwards data packets the next appropriate hop (another router or the destination device).

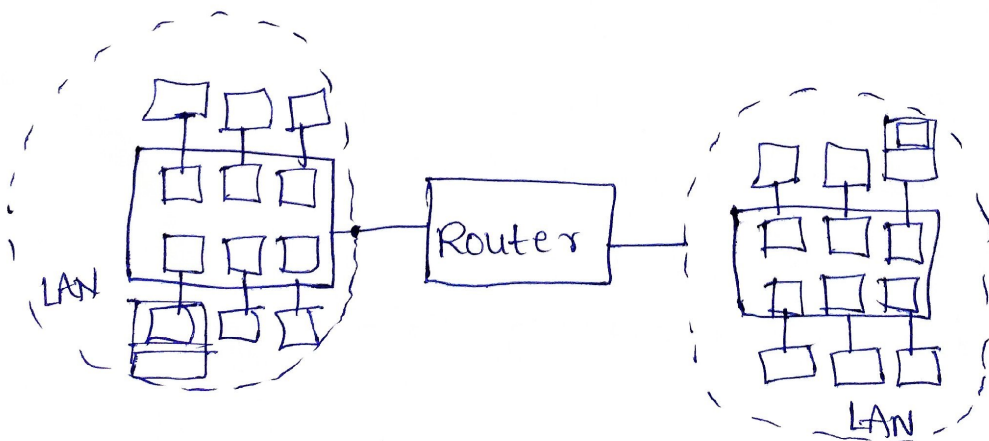
Forwarding:

→ Forwarding is a mechanism of sending packet from a source host to destination via its route. A routing table is essential requirement of forwarding, therefore a router or a host must maintain a routing table prior to forwarding the packet.

→ If a host has a packet which is to be transmitted over the network then the host must initially look into routing so as to discover the route (path) of sending the packet to the desired destination.

→ In the same way if a router receives a packet which has to be forwarded then it initially look within the table so as to check the final destination route.

Routing: Routing is the key feature of Internet. It is a process of selecting a best path for a packet to reach the destination. Every Intermediary Computer performs routing of packets from one computer to another computer until it eventually reaches the destination. Its path is calculated using routing table.



Routing

→ Generally, there exist many networks which many want to communicate with other Network. This is also shows that there may be many routers too.

→ For example consider different networks A to G, which are connected to each other with the help of routers, irrespective of their size and topologies.

Routing / Forwarding table :-

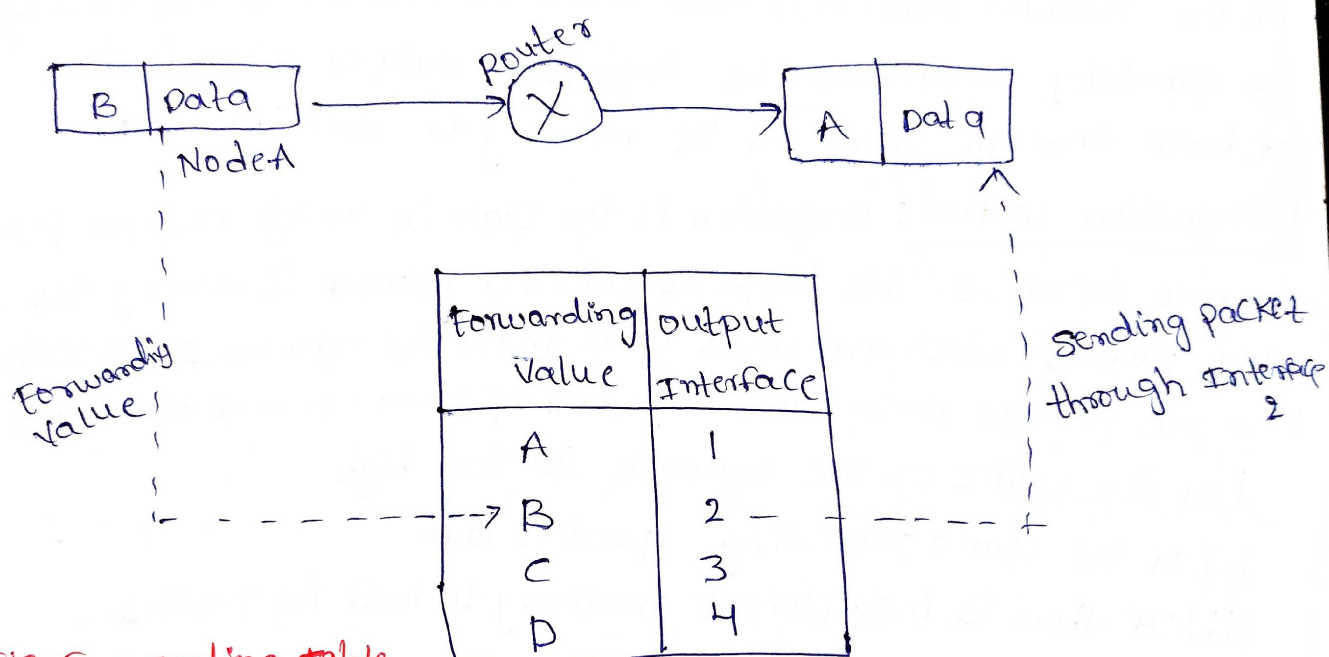


Fig: Forwarding table

→ A routing table consists of various data field that holds path information according to which the data packets are transmitted to reach the destination.

→ The destination can be a specific internetwork or a node in an internetwork. If the path information of a particular destination is not available then the table stores a default path for it.

→ The process of a routing a packet in the router ^{using} table is shown in above table.

→ Services provided by Network Layer:

Some of the services provided by Network layer are as follows

1) Guaranteed Delivery: Network Layer ensures that the data will reach its destination.

2) Guaranteed Delivery within a Specified time: Network layer ensures that the data will reach its destination within a specified time.

3) Flow Control: Flow control means controlling the overflow of data on the receiver side. It is used when the receiver is not having the capacity to process the incoming data or when it does not have enough space in its memory to store the data.

4) Congestion Control: Congestion is the state in which network performance decreases. This happens when a network is holding too many packets which are much more than the network's capability. There are various reasons for congestion to occur. Few of them are

i) If the traffic on the network is very high

ii) If the CPU's processing speed is low.

iii) If there is insufficient memory to hold the packets.

5) Quality of Services: (QoS): It is defined as the ability to provide different priority to different applications, users or data flows or to guarantee a certain level of performance to a data flow.

6) Security: - The Internet consists of tens of thousands of networks which are interconnected without any boundaries. Such environment requires high network security due to the fact that organizational network is accessible through any computer from any part of the world.

→ Network Service Models:

Network service Model can be defined as a model which describes the characteristics of transfer of packets from source to destination in an end to end network.

Example: consider ATM Service Models (ie) CBR (Constant Bit Rate) and ABR (Available Bit Rate).

1) CBR (Constant Bit Rate) ATM Service Model:

CBR Model is used to make the data packets flow at a constant bit rate. It follows a criteria similar to the early telephone system. The flow of the packets is provided using virtual pipes which offer fixed bandwidth transmission.

2) Available Bit Rate (ABR) ATM Service Model:

ABR Model is used to offer best-effort service over Internet. It also offers an additional feature of sending feedback to the sender in the form of a notification.

The feature and service comparison of CBR & ABR is tabulated below

Feature/Service	CBR	ABR
Bandwidth	Guaranteed with constant bit rate	Guaranteed Minimum
NO-LOSS	Guaranteed	No-Guaranteed

⑥

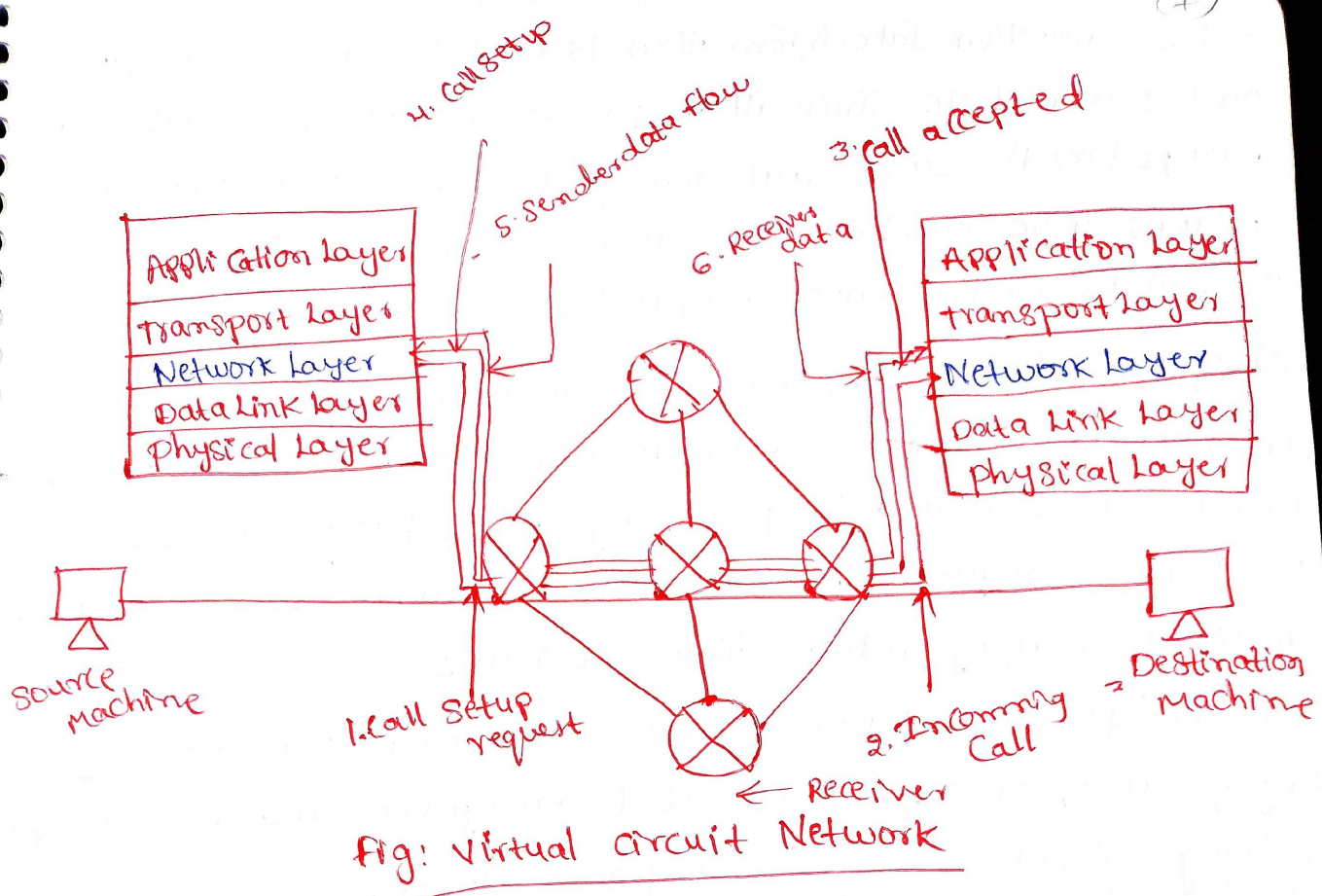
Feature/Service	CBR	ABR
ordering	ordered	ordered
time based	Yes	No
congestion	No congestion	Notifies if occurs

→ Virtual circuit and Datagram Networks

Virtual circuit Network:

- A connection in the context of connection-oriented organization of subnet is called virtual circuit (VC).
- A virtual circuit packet network is analogous to the telephone networks. Example ATM (Asynchronous Transfer mode)
- virtual circuit there are three phases.

1) VC setup: During this phase, the source machine requests the network layer to setup a virtual connection between it and the destination machine. The network layer finds and setup a route between the sender and the receiver. The route consists of the set of transmission lines and packet switches that will be followed by all packets of the same VC. The network layer may also allocate resources such as bandwidth for the VC.



2) Data transmission :

once the VC setup, the data transfer begins. All the data flows along the path of the VC.

3) VC Termination :

After All the data have been sent, the network layer terminates the VC on the desire from the sender (or) receiver. It also updates routing tables at each of the packet switches along the path to indicate that the VC is terminated.

→ In VC subnets each VC is numbered and every router maintains a table with an entry for each of the currently open virtual circuits passing through it.

⑧
→ This connection state information is needed to forward the packets on the correct VC. Since all the packets of a given virtual circuit always take the same route through the subnet. Each packet's header contains a VC number. The router uses this number to forward the packets on the correct output line.

Datagram Network: In datagram subnet, no routes are set up in advance even if the network layer service is connection-oriented. Each packet is routed independently of its predecessors, if any. Therefore subsequent packets from the same source to same destination may follow different routes.

→ The independent packets in the context of connectionless organization of subnet is called datagram and the corresponding subnet is called datagram subnet.

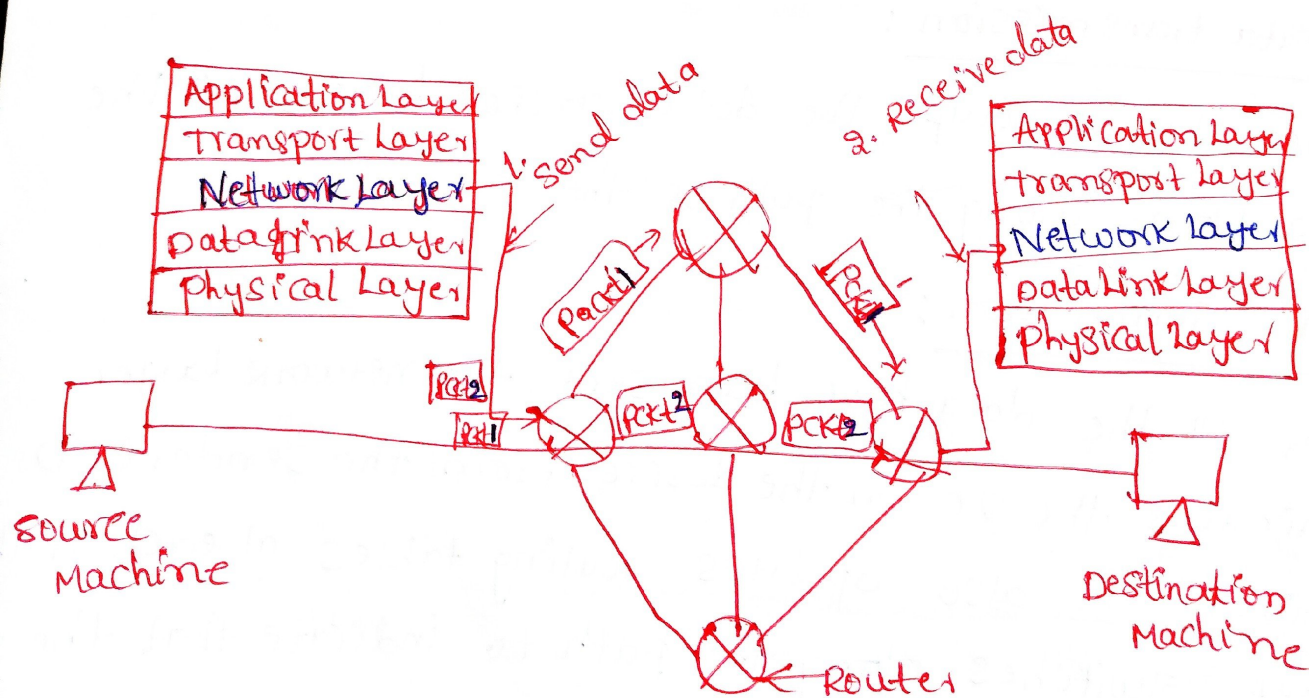


Fig: Datagram Network

- In datagram subnet the routers do not maintain any state information about VCs because no VCs are setup. Instead, they have a table with a pair of the destination number router address and the outgoing time to be used for the destination.
 - Each time the source machine wants to send packets to destination machine, it puts the full destination address in the packet's header and then the packet is entered into the network.
 - When the packet arrives at a router, the router examines the packet's destination address and uses its routing table to forward the packet in the direction of its destination. Since the routing table are updated, the series of packets from the same source to the same destination may follow different paths through the network. They may also arrive out of order at the receiver.
 - The destination address in a packet's header can be quite long for a large network. The datagram subnet are more robust and easily adapt to failures, the congestion control is also difficult.
- Example of Datagram Network is Internet's IP protocol.

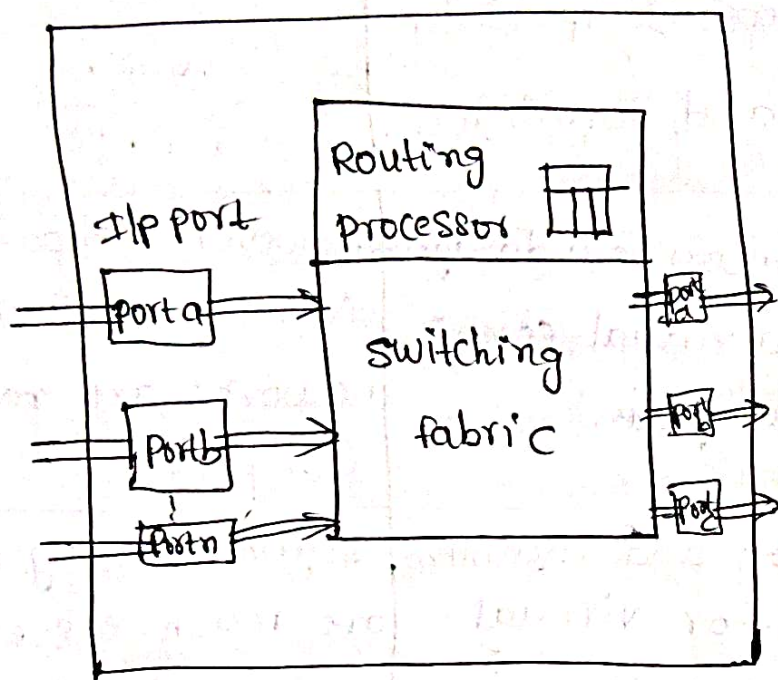
→ Differences between virtual circuits and Datagram Networks ⁽¹⁰⁾

Key	Virtual circuit	Datagram Networks
Definition	virtual circuit is a connection oriented service in which there is an implementation of resources like buffers, cpu, bandwidth etc, used by virtual circuit for a data transfer session	Datagram networks are a type of connectionless service where no such resources are required for data transmission.
Connectivity	It is connection oriented	It is connectionless.
Path	In these networks, the path taken by the initial data packet between the source and destination nodes is allocated. All subsequent data packets sent between them will take the same path	In these each datagram is treated independently. There is no fixed dedicated path for data transfer. Two subsequent packets from the source may travel entirely different paths to their destination.
Header	Since they are all part of the same virtual circuit, all packets that make up a message have the same header information.	Even though they are part of the same communication, the separate datagram packets have different header information.

Phases	transmission is performed in three steps Setup, data transfer and termination.	there are no such communication phases.
complexity	In comparison to a datagram network, a virtual circuit is less complex.	however compared to the virtual circuit, datagram networks are more complex.
cost	Installation and maintenance costs for virtual circuit are higher.	Network using datagrams are much easier to setup and maintain.
Reliability	More reliable data transmission.	Less Reliable than VC.
Addressing	during setup the addressing and route are decided. AS a result each packet just includes the vc number.	The complete source & destination addresses are included in each datagram packet.
Application	virtual circuits are implemented in networks using (ATM) Asynchronous transfer mode communications, such as when placing phone calls.	User datagram protocol (UDP) typically serves as the framework for datagram communication. In IP networks, they are used.

→ Inside a Router :

12

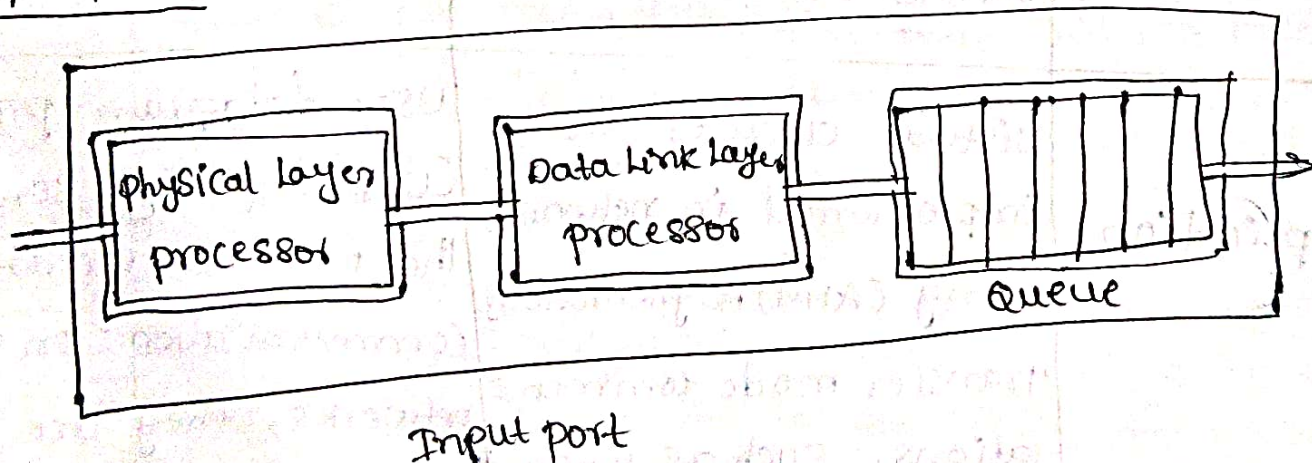


Architecture of Router

The four components of router are

- 1) Input port
- 2) Output port
- 3) Routing processor
- 4) Switching fabric

1) Input port :



→ This component performs the physical and datalink functions of router. Once the bits are created from the received signal, the packet is decapsulated from the frame. The detection and correction of error is carried out allowing the network layer to route the packet.

→ The input port contains buffers to queue the packets prior sending them to the switching fabric along with a physical layer processor and a data link processor.

2) output port :

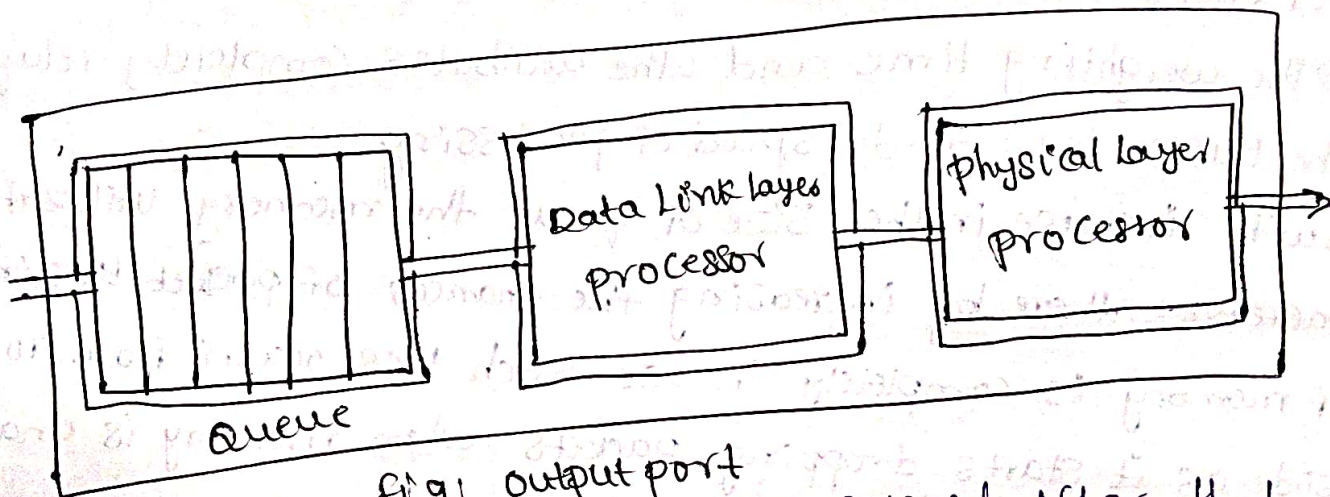


fig: output port
→ In output port, the packets are first queued. After that the packets are encapsulated in a frame and then sent by implementing the functions of physical layer on frame. The functions performed by output port are reverse of functions performed by the input port.

3) Routing processor: This component is responsible for carrying out the network layer functions. It simultaneously finds the address of the next hop as well as output port number from which the packet arrived. The activity of choosing next hop for the packet is some times called table lookup. The functionality of routing processor in latest switches is handed over to input ports to make the process fast and efficient.

4) Switching fabric: This component is responsible for moving the packets from the input queue to the output queue. The speed of this process not only impacts on the size of input/output queue but also on the total delay in packet delivery.

→ The transfer of packets from input queue to output queue is the most difficult task in the router.

Queueing: Packet queueing is possible at both input and output ports similar to the cars waiting near round about for their turn to cross it.

→ The waiting time and other attributes completely rely on the traffic load and speed of processing.

→ With increase in the size of queue the memory utilization increases thereby increasing the chances of packet loss in case if memory is completely in use. Such loss occurs from router side as it starts dropping packets when memory is unavailable.

→ Similar situation can arise at output ports when multiple packets are sent simultaneously to a single port. This leads to the creation of queue as output port can forward only a single packet at a time. Moreover the queued packets need to be forwarded based on certain criteria.

→ This criterion is imposed using packet scheduler. Two of such criteria include FCS (First-come First Served) and weighted fair queueing (WFQ).

→ The use of packet scheduler helps in ensuring quality of service.

→ Similarly on the input ports, various approaches like AQM (Active Queue Management) and RED (Random Early detection) are adopted to make a decision regarding the packets arriving in the queue.

→ Routing Control plane :

The routing plane exists near the location where the decisions are made by routing processor. This makes it decentralized as the routing process involves different aspects of entire network. However modern routers integrate both hardware data plane and software control plane to generate an integrated product.

→ The Internet protocol :

→ Internet protocol is the third layer in TCP/IP. The functionality of this layer is to transmit data over the Internet. The responsibility of this layer is to send IP packets from source to destination independently. This means that each packet should contain the full address of the desired destination. It is an connectionless protocol, where is no connection exists between the end points.

Components of IP : The two important components of IP are forwarding & Routing.

Note : Explain here about forwarding & Routing

Datagram Format of IPv4:

Datagram is nothing but the packets in IPv4. The datagram format of IPv4 is as follows -

Version 4-bits	Header Length 4-bits	Service 8-bits	Total Length 16-bits
Identification 16-bits		Flag 3-bits	Fragmentation Offset 13-bits
Time to Live 8-bits	Protocol 8-bits	Header checksum 16-bits	
Source IP address			
target IP address			
option			

← 32-bits →

Fig: IPv4 Datagram Format

→ A datagram is a variable length packet having header and a data. Length of the header is about 20 to 60 bytes containing the information necessary for routing and forwarding. Usually, a 4-bytes header sections are shown in TCP/IP. The following are the fields of a datagram.

17
1) Version (VER): It is a 4-bit field defining the session of IPv4 protocol. Currently available version is 4. In future, version 6 is supposed to replace version 4. Version (VER) field is responsible for informing the software in the processing machine about the version in use (i.e.) version 4.

2) Header Length (HLEN): It is a 4-bit field defining the entire length of the datagram header in 4-byte words. This field is required because header length is variable between 20 and 60 bytes.

3) Services: It is a 8-bit field whose name and interpretation is changed by IETF (Internet Engineering Task Force). From service type to differentiated services.

4) Total Length: It is a 16-bit field that contains header as well as data. It has a maximum length of 65,535 bytes. But, when larger datagrams are required. The field contains length greater than the maximum length.

5) Identification field: It is a 16-bit field, which specifies the identify of a fragment (i.e.) to which datagram it belongs to.

6) Flag bit: It is a 3-bit field which is used to control the fragment. Here, first bit is zero and it is reserved. Second bit is 1 which represents Don't Fragment (DF). Third bit is 2, which represent More Fragment (MF).

7) Fragment Offset: It is a 13 bit field, which specifies the location of the fragment in the datagram. The maximum length of fragment offset is one byte more than the —
— total length field (i.e.) 65,536 bytes.

8) Time-to-live Field: It is a 8-bit field, which refers to a counter used for limiting the lifetime of a packet. The maximum lifetime of a packet is 255 Sec. At each hop, the counter is decremented and when it becomes zero, the packet is removed and a warning packet is transmitted to the source.

9) Protocol: It is a 8-bit field. It specifies to which transport layer protocol (TCP or UDP), the datagram is to be given.

10) Header Checksum: It is a 16-bit field, which checks the header part so as to detect the errors occurred during the transmission of packet. At each hop, the checksum must be recomputed because one or more fields of the header changes regularly.

11) Source IP address and target IP address: Each of these fields is of 32 bit length. The source field specifies the port number of the sender host that sends the fragment whereas, the destination field specifies the port number of the receiver host that receives the fragment.

12) Options Field: It is of variable length; options may occupy space at the end of IP header and their lengths are multiple of four bytes. It is generally used to provide more information that wasn't covered in the IP header.

→ IPv4 Addressing:

IPv4 address: IPv4 (IP-version 4) defines 32 bit IP address for hosts and routers connected to the Internet.

→ Each host and router has a unique and universal IP address. This means that no two hosts connected to the Internet can have same IP address at the same time.

→ Any host that wants to establish connection on the Internet must accept the addressing system. Hence, IPv4 address are unique and universal.

Address space: An address space is defined as the total number of addresses that are used by the protocol. IPv4 protocol uses 32-bit addresses. Hence the address space of IPv4 is 2^{32} or 4,294,967,296 (i.e. more than 4-billion).

→ If limitation is not given then these large number of devices can be connected on the Internet. The IPv4 address makes use of three generated notations.

They are:

1) Binary notation: Binary notation is a popular notation to present an IPv4 address which is displayed as 32 bits. An IPv4 address can be defined as 32-bit address or a 4 byte address where each octet (8bit) refers to one byte.

Example: The example of IPv4 address in binary notation is: 11011101 00101101 01001110 00111000

2) Dotted Decimal Notation: Dotted Decimal Notation is another notation that specifies the IPv4 addresses. Generally Internet addresses are written in decimal form which are more easier to look through. In decimal forms, each octet (1 byte) is separated by decimal point (dot) and its value ranges from 0 to 255.

Example: The dotted decimal notation for the IPv4 binary notation is given below.

11011101	00101101	01001110	00111000	← Binary notation
↓	↓	↓	↓	
221	45	78	56	← Dotted decimal notation

3) Hexadecimal Notation: Hexadecimal notation is another notation which is used rarely to specify the IPv4 address. Here, every set of four bits is equal to single hexadecimal digit. That is, in an address of 32-bit there are total 8-hexadecimal digits. This type of notations are mostly used in network programming.

Example:

	11011101	00101101	01001110	00111000
	↓	↓	↓	↓
	221	45	78	56
	↓	↓	↓	↓
Hexadecimal notation →	DD	2D	4E	38

Internet Control Message Protocol (ICMP):

ICMP has been designed to overcome the problems posed by IP protocol. It is used in IP network management and administration. Its importance is seen in IP implementations. It is said to be a control protocol because it does not hold the data instead it keeps the report of data. The latest version of ICMP is ICMPv4.

Types of ICMP messages:

ICMP messages are categorized into following two types -

1) Error - reporting messages

2) Query messages.

1) Error Reporting Messages: These messages report about the issues encountered by host/router while processing an IP packet.

2) Query Messages: these messages help the network manager in retrieving specific information about the network from host/router.

Format of ICMP message:

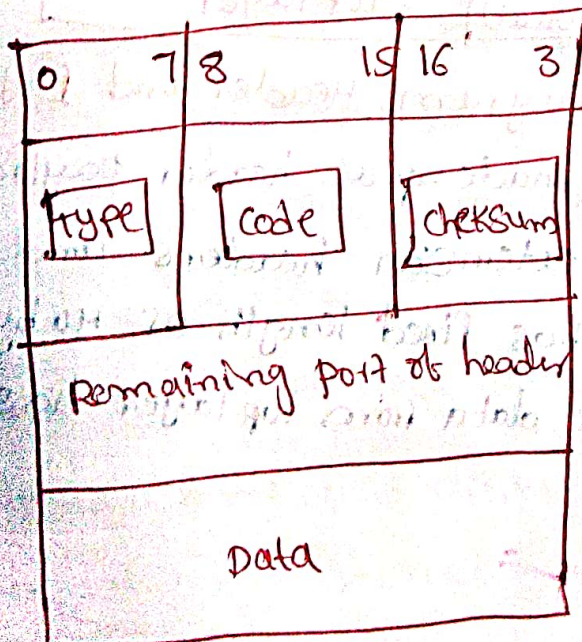


Fig: ICMP Message Format

1) Type: This is 1-byte field that specifies the message type.

2) Code: This is 1-byte field that specifies the reason for including a particular type of message.

3) Checksum: This is 2-byte field that is used for detecting the errors within the message.

4) Remaining part of Header: This is 4-byte field that contains information specific to the type of message specified in "type" field.

5) Data: This is variable length field that contains data pertaining to respective message type. If "error messages" are specified in "type" field.

IPv6

→ IPv6:

IPv6 Header format:

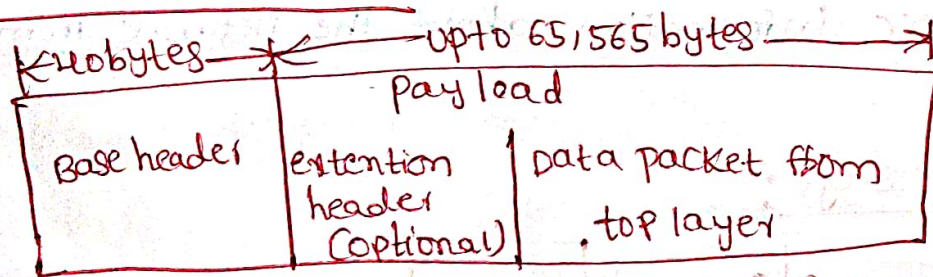


Fig: IPv6: Datagram Header and Payload

This IPv6 packet is made up of header payload. payload consists of optional extension headers and data from top layer. The base header has fixed length of 40 bytes but the extension header and data from top layer has approximately 65,535 byte of data.

Base header:

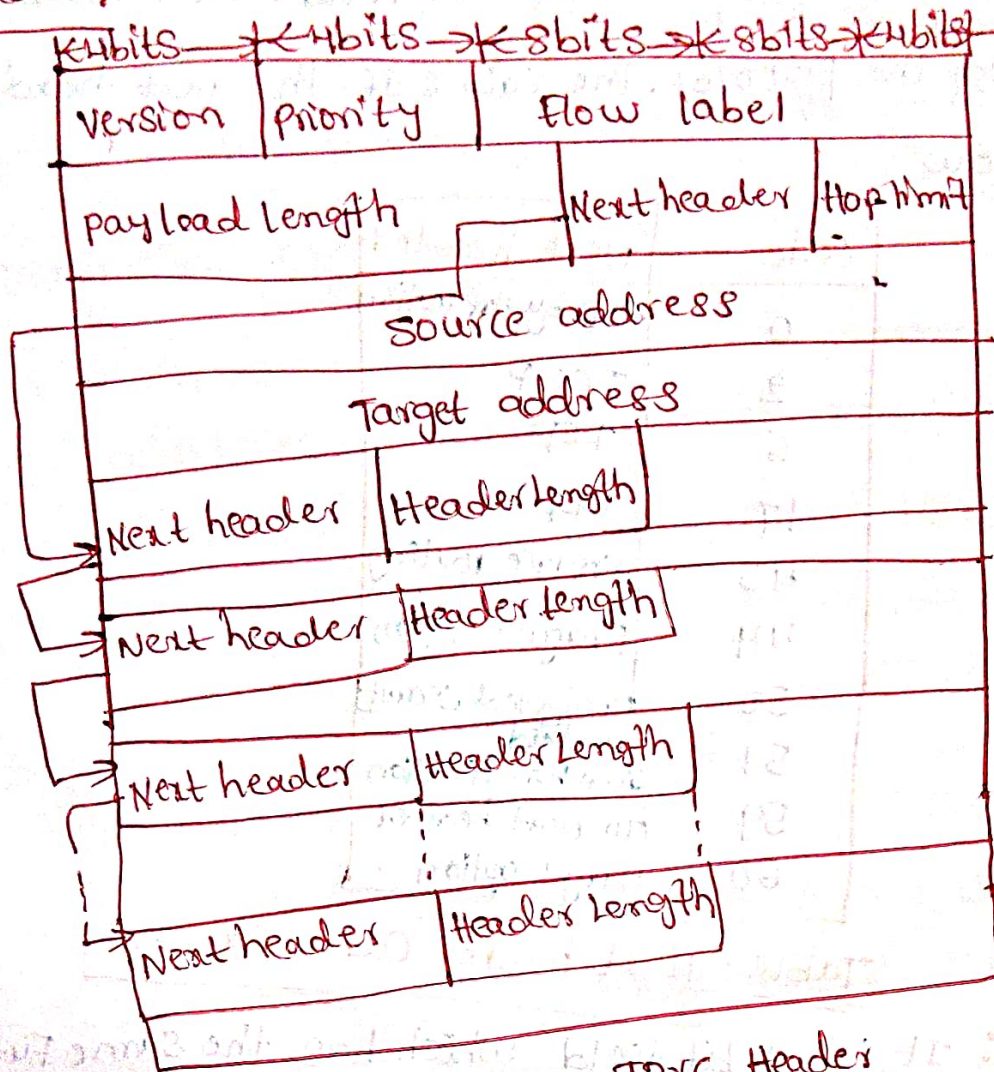


Fig: Format of an IPv6 Header

(i) Version: It is a 4-bit field defining the version number of the Internet protocol.

Example: IPv6 has the value 6.

(ii) Priority: It is a 4-bit field defining the priority of the packet related to traffic congestion.

(iii) Flow label: It is a 3-byte field used for defining IP datagram length. handling flow of data.

(iv) Payload length: It is a 2-byte field defining IP datagram length.

(v) Next Header: It is a 8-bit field defining the header after the base header in the datagram. The next header can be an optional extension headers used by IP or the header of an encapsulated packet like UDP or TCP.

every extension header include this field. In version 4, this field is called the protocol. The values of the next headers are tabulated below.

Code	Next Header
0	Hop-by-hop option
2	ICMP
6	TCP
17	UDP
43	Source routing
44	Fragmentation
50	Encrypted security payload
51	Authentication
59	To next header
60	Target option

Table: Next header codes for IPv6

(vi) Hop Limit: It is 8-bit field which has the same function as that of TTL field in IPv4.

(vii) Source Address: It is 16 byte Internet address field identifying the actual source of the datagram.

(viii) Target Address: It is a 16-byte Internet address field identifying the final target of the datagram. In case of source routing, this field includes the address of the next router.

→ comparison of IPv4 and IPv6

25

IPv4 Headers

- 1) IPv4 is the ^{4th} version of IP.
- 2) Fig: Draw IPv6 Header Here
- 3) IPv4 allows 32-bit source and target addresses.
- 4) IPv4 has service type field which represents priority of the packet.
- 5) In IPv4, fields are not renamed.
- 6) In IPv4, Identification, flag and offset fields are present.
- 7) IPv4 has a TTL field and protocol field.
- 8) IPv4 has a header checksum field.
- 9) IPv4 has an options field.

IPv6 Headers

- 1) IPv6 is the 6th version of IP.
- 2) Draw IPv6 Header [Note]
- 3) IPv6 allows 128-bit source and target addresses.
- 4) IPv6 do not have service type field. Its functionality is handled by priority and flow label fields.
- 5) In IPv6, some of the fields are renamed.
Ex: Type of service to traffic class and total length to payload length etc..
- 6) In IPv6, Identification, flag and offset fields are removed from the base header and added in the fragmentation extension header.
- 7) In IPv6, TTL field is replaced by hop limit and protocol field is replaced by next header field.
- 8) In IPv6 header checksum field is removed because checksum is carried out by its upper layer protocols.
- 9) In IPv6, options field is moved under extension header.

→ IPv6 Address: The IPv6 addresses were introduced to overcome the problems related to,

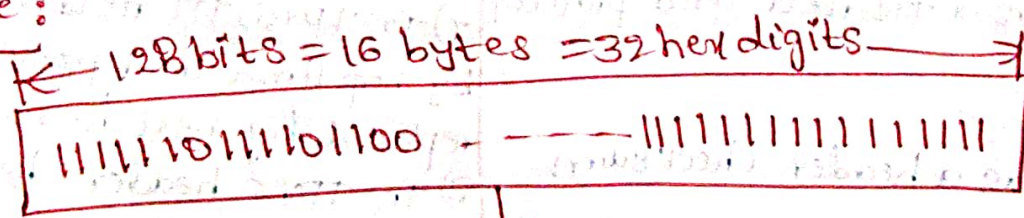
- (i) Address depletion for the Internet
- (ii) Lack of accommodation for real time audio and video transmission
- (iii) Data encryption and authentication

Structure: An IPv6 address is 128 bits long (i.e. 16 bytes).

The structure of IPv6 address is based on hexadecimal colon notation and abbreviation that enables the IPv6 address to be more readable and abbreviated.

(a) Hexadecimal Colon Notation: Hexadecimal colon notation of IPv6 address makes them more reliable. In hexadecimal colon notation 128 bits are divided into eight sections where each section of 2 bytes, which requires four hexadecimal digits. This implies that IPv6 addresses comprises of 32 hexadecimal digits. The arrangement of these digits is made in such a way that a colon is placed as a separator between every four digits.

Example:



↓

FDEC : 0074 : 0000 : 0000 : 0000 : 80FF : 0000 : FFFF

Fig: Hexadecimal Colon Notation.

(b) Abbreviation: The hexadecimal notation results in a very long format consisting of numerous zero digits. In such cases, the abbreviation technique can be performed only once for every address. That is there are two sections of zeros, only one of them is abbreviated. The re-expansion of the abbreviated address is performed easily just by aligning the unabbreviated parts and inserting zeroes so as to get the original expanded address.

Hexadecimal notation

Example: FDEC:0074:0000:0000:BOFF:0000:FFFO

After Abbreviation FDEC:74:0:0:BOFF:0:FFFO

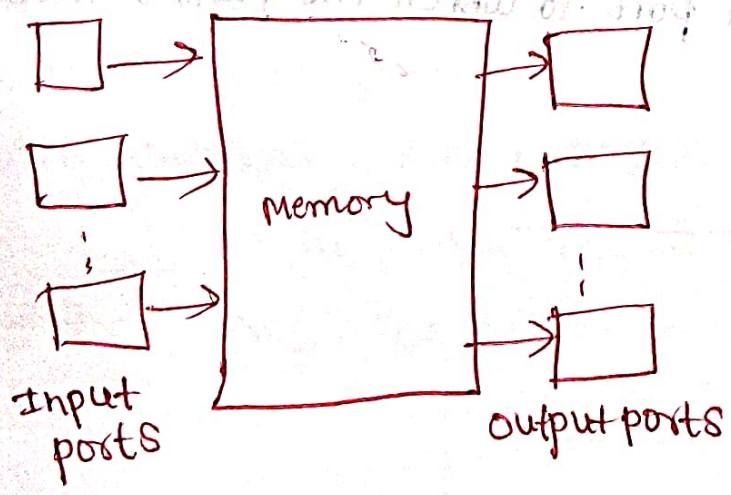
More abbreviation: FDEC:74::BOFF:0:FFFO

Switching

Switching: The three switching techniques are:

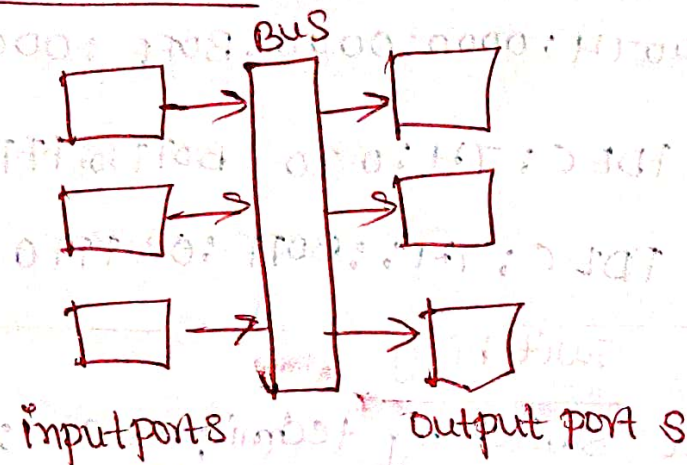
- 1) Switching via memory
- 2) Switching via bus
- 3) Switching via crossbar switch

1) Switching via memory



- Traditional routers use routing processor as CPU for the purpose of switching among input and output ports. Here, input/output ports act as input/output devices for such routing processors.
- Modern routers use memory for the purpose of switching.
- These routers perform like a shared memory multiprocessors in various aspects. These routers work by saving the packets in memory while using input line cards for processing.

2) Switching via a Bus :



→ This type of processing makes use of a bus that directly forwards the packets received from input ports to the output ports. For this reason, it does not require assistance from routing processor.

→ However, the bus needs to be shared among all the available ports. Here the input port includes a header value which indicates the output port to which the packet needs to be forwarded.

3) Switching via crossbar switch :

29

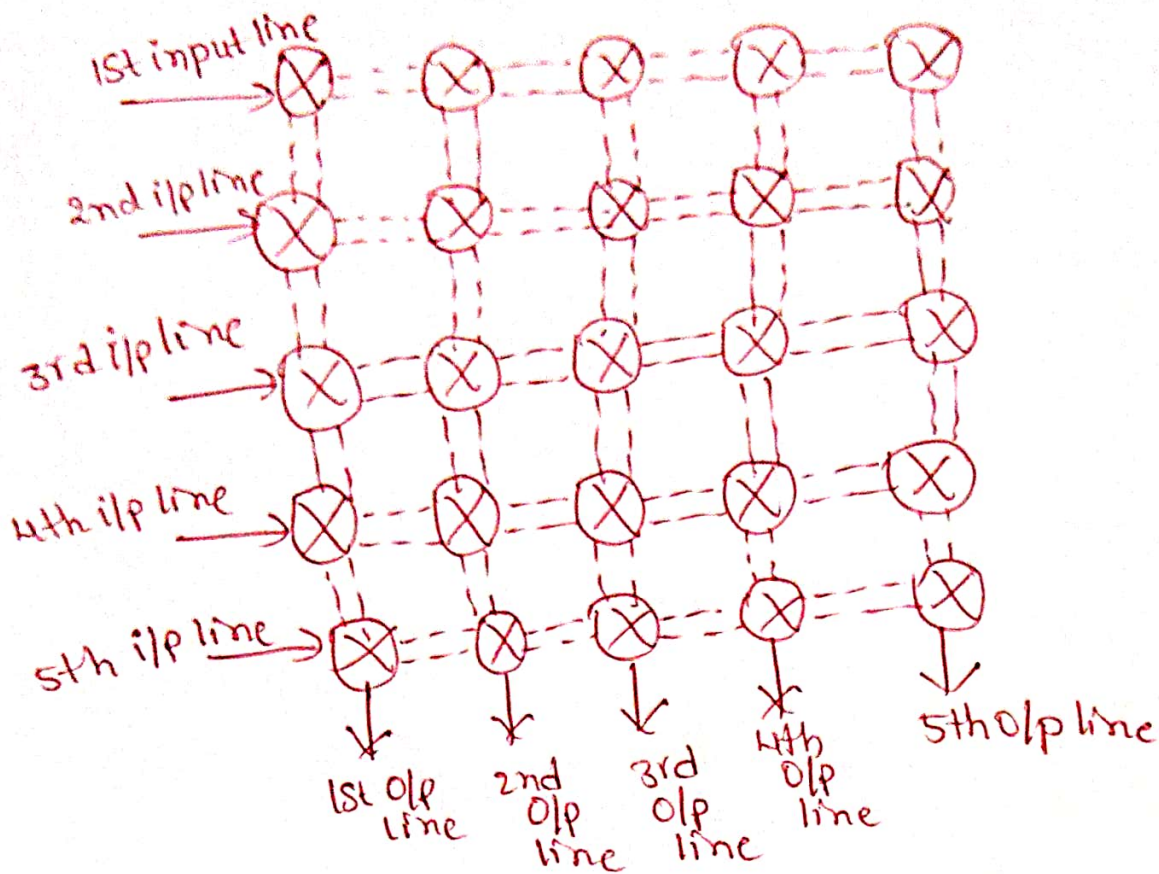


fig : crossbar switch

→ Crossbar switch consists of series of points called cross points which handles inputs and outputs in the switch. Depending on the number of inputs given to the cross points, output will be given by the cross points.

→ Example, if 5 inputs of data are given, then 5 outputs of data will be the result given by the cross points.

(ie) $5 \times 5 = 25$ crosspoints.

→ The fig shows a crossbar switch, which is built from two important entities, transistor or gates and switch which consists of 25 cross points. The switch is in the form of grid or a square.

Unit-4

Transport layer

Introduction: Transport layer is the second layer in the TCP/IP model and the fourth layer in the OSI model.

- It is an end-to-end layer used to deliver messages to a host. It is termed an end-to-end layer because it provides a point-to-point connection rather than hop-to-hop between the source host and destination host to deliver the services reliably.
- The unit of data encapsulation in the transport layer is a segment.

Working of transport layer:

- Transport layer takes services from the application layer and provides services to the network layer.
- Transport layer ensures the reliable transmission of data between systems.

At the sender's side: The transport layer receives data (message) from the application layer and then performs segmentation, divides the actual message into segments, adds the source and destination's port numbers into the header of the segment and transfers the message to the network layer.

At the receiver's side: The transport layer receives data from the network layer, reassembles the segmented data, reads its header, identifies the port number, and forwards the message to the appropriate port in the application layer.

Responsibilities of a Transport Layer:

- Process to process delivery
- End-end connection between Hosts.
- Multiplexing and Demultiplexing
- Congestion control
- Data Integrity and Error correction
- Flow control

Transport Layer protocols:

The two main transport layer protocols are:

- 1) TCP (Transmission Control Protocol)
- 2) UDP (User Datagram Protocol)

1) TCP (Transmission Control Protocol): It provides reliable communication between two Hosts.

Transmission Control Protocol (TCP) is one of the most important protocols of Internet Protocol Suite. It is most widely used protocol for data transmission in communication network such as Internet.

2) UDP (User Datagram Protocol): It provides unreliable communication between two Hosts.

Multiplexing and Demultiplexing in Transport Layer:

→ Multiplexing and Demultiplexing services are provided in almost every protocol architecture ever designed. UDP & TCP perform the demultiplexing and multiplexing jobs by including two specific fields in the segment headers the source port number field and destination port number field.

Multiplexing: Gathering data from multiple application processes of the sender, enveloping that data with a header, and sending them as a whole to the intended receiver is called multiplexing.

Demultiplexing: Delivering received segments at the receiver side to the correct app layer process is called demultiplexing.

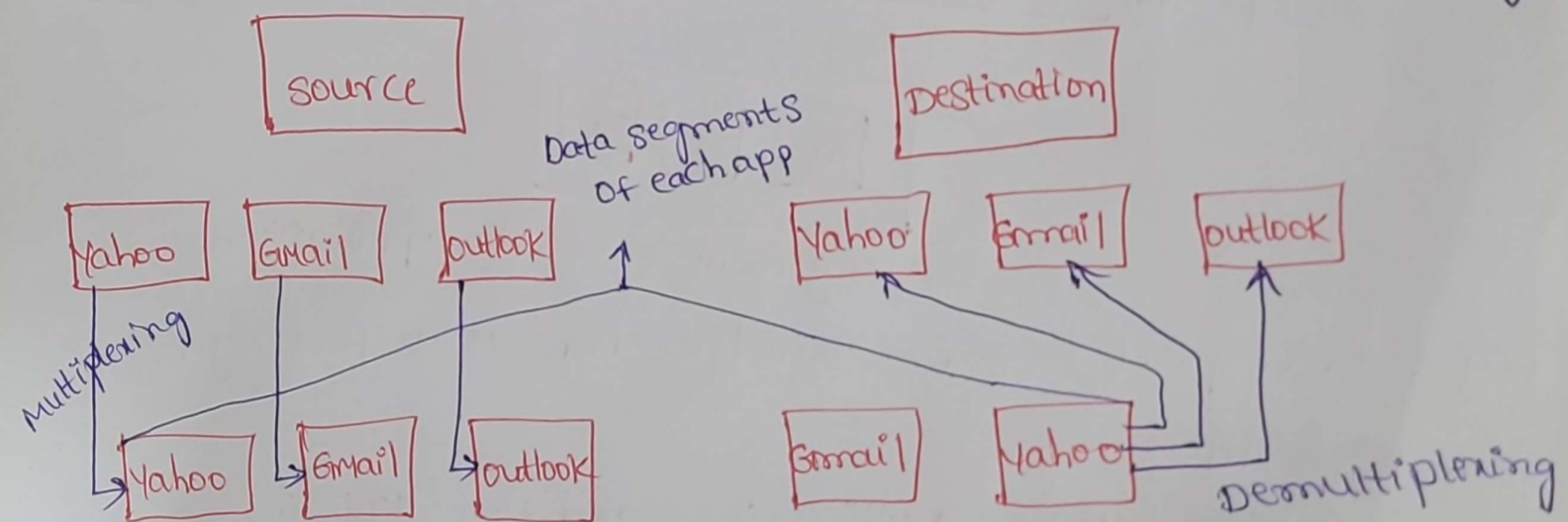


Fig: Abstract view of multiplexing & demultiplexing

Multiplexing and Demultiplexing are the services facilitated by the transport layer of the OSI model.

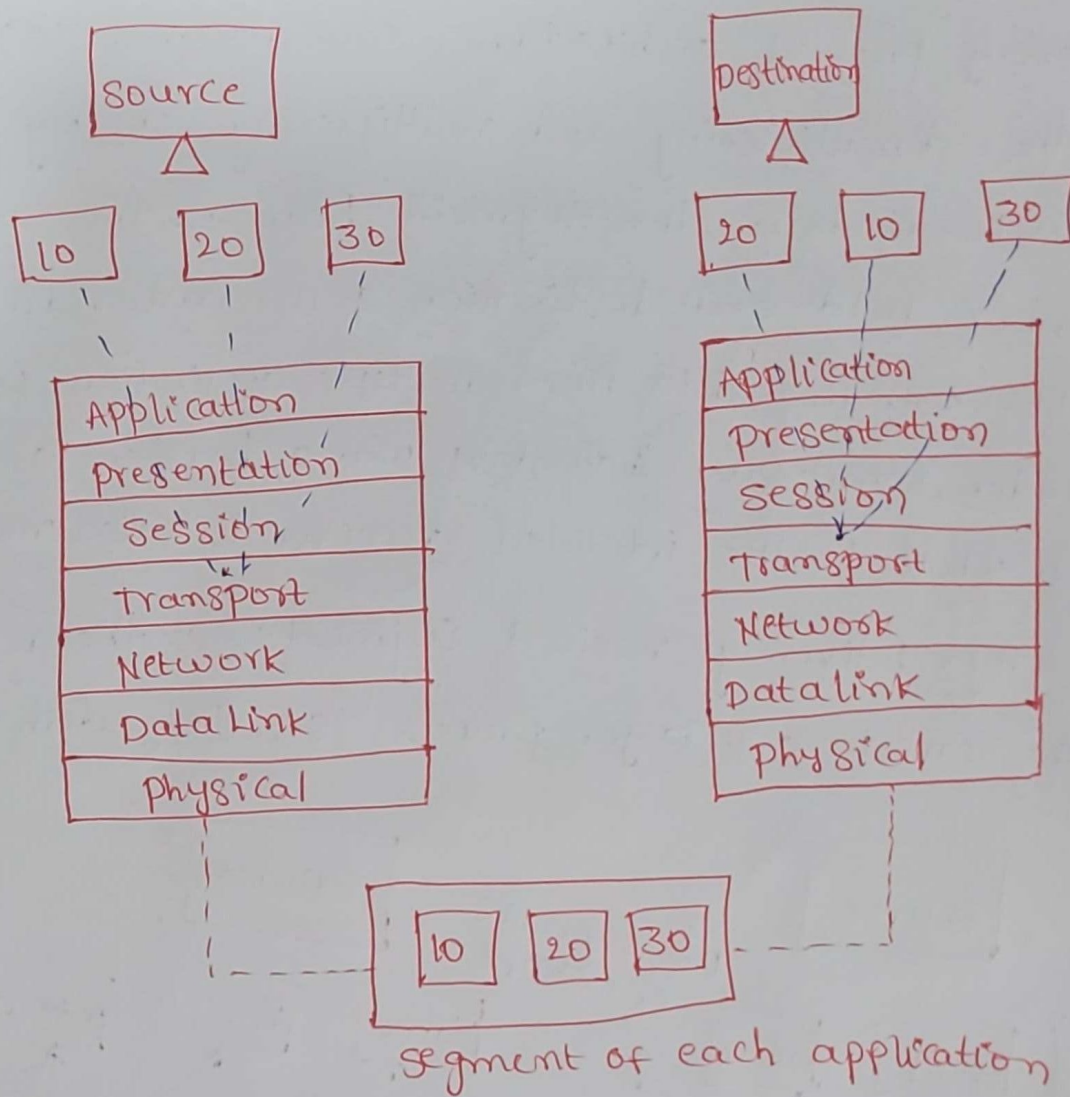


Fig : transport layer - junction for Multiplexing & Demultiplexing

→ There are two types of Multiplexing and Demultiplexing

- 1) connectionless multiplexing and demultiplexing
- 2) connection-oriented multiplexing and demultiplexing

Sender (X)

Source IP : X
Destination IP : Y
Source port No : 40
Destination port No : 73

Receiver (Y)

Source IP : Y
Destination IP : X
Source port No : 56
Destination port No : 23

Fig : Transfer of packet between Application of sender & Receiver

3
→ For sending data from an application on the sender side to an application at the destination side, the sender must know the IP address of the destination and port number of the application (at the destination side)

→ connectionless Transport UDP: (User Datagram protocol):

→ Relationship Between Transport and Network Layer : 4

→ The network layer handles host-host delivery using IP addresses to route packets between networks, while the transport layer provides process-process delivery for applications using ports to direct data to the correct application on a host.

→ The network layer relies on the transport layer for reliable ordered data flow between end systems, and the transport layer depends on the network layer to route data packets to the correct destination host.

Network Layer's Role : Ex: IP

Host-Host Delivery : Responsible for addressing and routing data packets from a source host to a destination host, even if they are on different networks.

Packetization : Divides the data received from the transport layer into packets (or datagrams) for transmission.

Routing : Uses IP addresses (like IPv4 or IPv6) to determine the path for data packets across various networks.

Devices : Routers are key devices operating at this layer for forwarding packets between networks.

Transport Layer's Role : Ex: TCP/UDP

process-to-process Delivery : Manages communication between specific applications (processes) on the source and destination hosts.

Port numbers : Uses port numbers in conjunction with IP addresses to identify and deliver data to the correct application on a host.

Segmentation / Reassembly : Breaks large data messages into smaller segments at the sender and reassembles them in the correct order at the receiver.

Reliability & Flow control : provides services like error checking, retransmission [Ex: TCP], and congestion control to ensure reliable data delivery.

Protocols : Utilizes protocols like TCP for reliable, connection-oriented delivery and UDP for faster connection-less delivery.

Relationship and Dependency :

Layers work together : The two layers form a functional partnership, with the transport layer sitting above the network layer.

Upstream and Downstream : The transport layer takes data from the application layer, prepares it for the network layer & sends it down. The network layer then routes these packets.

Mutual Reliance : The transport layer needs the network layer to get data packets to the correct destination host, and the network layer relies on the transport layer to provide the necessary services for end-to-end communication between applications.

→ Transport Layer in the Internet :

→ Internet uses two Transport Layer protocols called TCP & UDP.

The data packets of transport layer protocols for TCP are referred as segments. The transport layer using Internet IP, offers the following services.

Note: write remaining answer that (ie) Services / Responsibilities of transport layer.

Services/Responsibilities of Transport Layer:

1) Service-point Addressing: Transport Layer introduces a type of address called service point address. This address is included in header of Transport Layer.

→ Computers are capable of running several programs simultaneously. The term source to destination delivery doesn't simply mean delivery of data from one system to the other, but it also means delivery of data from one specific process to the another. Therefore in this case it becomes necessary to include a service point address in the header of transport layer.

2) Segmentation and Reassembly:

→ Transport Layer divides a single message into various transmittable segments with wherein every segment contains a sequence number. This sequence enables the layer to reassemble the message once it reaches the destination. It also enables the layer to identify the lost packets during transmission and to replace them as well.

3) Connection Control:

A transport layer can be either of the following

i) Connection Less: This layer considers segment as an individual packet. These packets are then individually delivered to the destination.

(ii) Connection oriented: This layer initially establishes a connection with the transport layer of destination machine. It then transfers the entire data and terminates the connection after transmission.

4) Flow control: Flow control of this layer is similar to the flow control of data link layer. But the only difference is that, in data link layer, flow control is performed across a single link whereas in transport layer end-end flow control is performed.

5) Error control: Error control of this layer is similar to the error control of data link layer. But the only difference is that in data link layer error control is performed across a single link whereas in transport layer, process-process error control is performed. While sending the data, transport layer of sender's machine ensures that complete data reaches the transport layer of receiver's machine. In case if an error occurs, the entire data is retransmitted.

UDP: User Datagram protocol (UDP): It is a Transport Layer protocol.

UDP checksum: UDP checksum covers the UDP Header and UDP data.

→ Both UDP and TCP include a pseudo-header with the UDP datagram just for the checksum calculations.

UDP Checksum Example:

$$\begin{array}{r}
 (\quad 1110011001100110 \\
 \quad 1101010101010101 \\
 + \quad 1111111111111111 \\
 \hline
 \quad 1101000101110111 \\
)
 \end{array}$$

$$\begin{array}{r}
 1110011001100110 \\
 1101010101010101 \\
 \hline
 1101110111011101 \\
 \boxed{}
 \end{array}$$

$$\begin{array}{r}
 \\
 \\
 \\
 + 1 \quad \leftarrow \text{carry} \\
 \hline
 1011101110111100 \text{ (sum)}
 \end{array}$$

's complement of sum.

0100010001000011

→ UDP checksum is defined as a safety feature in which its value represents an encoding of a datagram. Initially its value is calculated by the sender and then by the receiver. If the given value of the sender does not match with the receiver's data, then the UDP helps in detecting the error by using checksum algorithm. The checksum value of the UDP is mandatory for the sender but not to the receiver because if the given value is checksummed then automatically the receiver gets the correct data sent by the sender.

Real time application of UDP :-

- 1) Streaming media applications such as movies.
- 2) online multiplayer games.
- 3) Voice over IP (VoIP)
- 4) UDP is used with the RTOS (real time operating systems).
- 5) UDP is used in aircraft controls and flight instruments.

Principle of Reliable Data transfer protocol:

- Transport layer protocols are central piece of layered architectures these provides logical communication between application processes.
- These processes uses the logical communication to transfer data from transport layer to network layer and this transfer of data should be reliable and secure.
- The data is transferred in the form of packets but the problem occurs in reliable transfer of data.
- The problem of transferring the data occurs not only at the transport layer, but also at the application layer as well as in the link layer.
- This problem occurs when a reliable service runs on an unreliable service. Ex: TCP is a reliable data transfer protocol that is implemented on top of an unreliable layer (ie) internet protocol (IP) is an end to end network layer protocol.

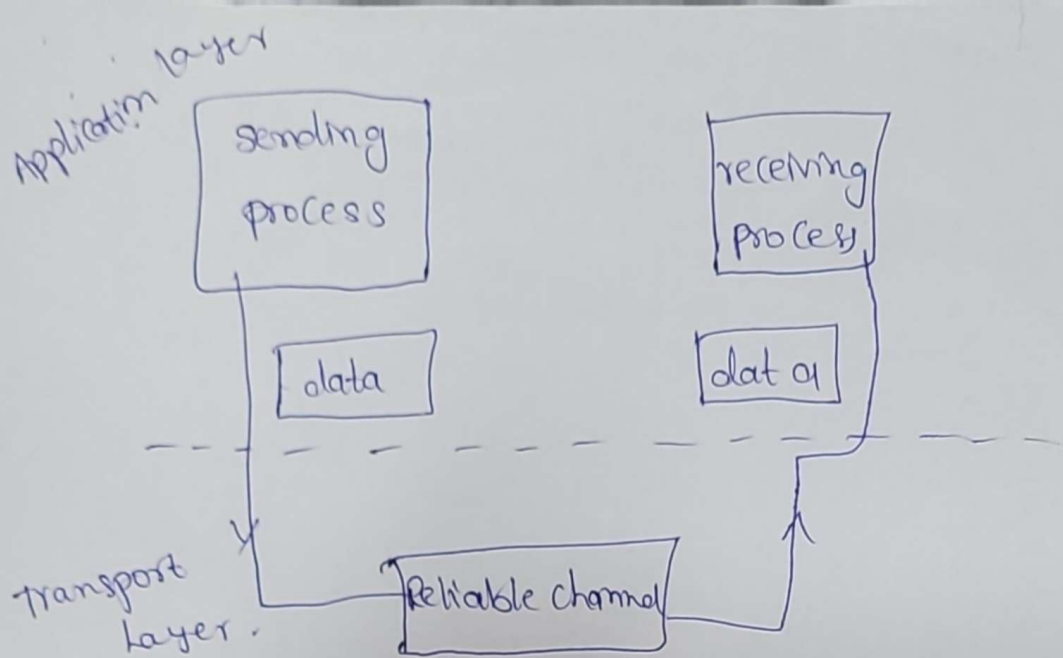


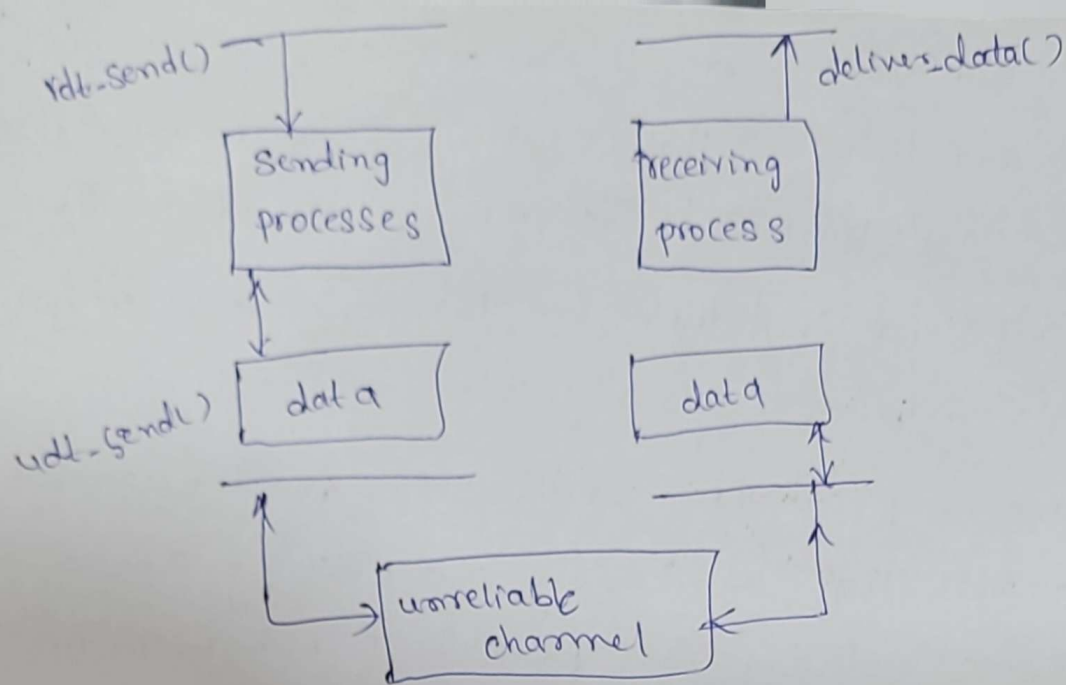
Fig. Study of Reliable Data Transfer

⇒ In this model we have design the sender and receiver sides of a protocol over a reliable channel.

→ In the reliable transfer of data the layer receives the data from the above layer breaks the message in the form of segment and put the header on each segment and transfer.

→ Below layer receives the segments and remove the header from each segment and make it a packet by adding to header.

→ The data which is transferred from the above has no transferred data bits corrupted or lost, and all are delivered in the same sequence in which they were sent to the below layer this is reliable data transfer protocol. This service model is offered by TCP to the internet applications that invoke this transfer of data.



Study of unreliable Data transfer

Similarly in an unreliable channel we have design the sending and receiving side.

→ sending side of the protocol is called from the above layer to `rdt_send()` then it will pass the data that is to be delivered to the application layer at the receiving side (here `rdt_send()` is a function for sending data where `rdt` stands for reliable data transfer protocol and `_send()` is used for the sending side).

on the receiving side, `rdt_rcv()` (`rdt_rcv()` is a function for receiving data where `_rcv()` is used for receiving side). will be called when a packet arrives from the receiving side of the unreliable channel.

→ when the `rdt` protocol wants to deliver data to application layer, it will do so by calling `delivers_data()` where `delivers_data()` is a function for delivering data to upper layer).

- in reliable data transfer protocol, we only consider the case of unidirectional data transfer, that is transfer of data from the sending side to receiving side (ie only in one direction).
- in case bidirectional (full duplex or transfer of data on both the sides) data transfer is conceptually more difficult.
- Although we only consider unidirectional data transfer but it is important to note that the sending and receiving sides of our protocol will need to transmit packets in both directions.
- in order to exchange packets containing the data that is needed to be transferred the both (sending and receiving) sides of rdt also need to exchange control packets in both direction (ie, back and forth), both the sides of rdt send packets to the other side by a call a `udt_send()`.
- `udt_send()` is a function used for sending data to other side where udt stands for unreliable data transfer protocol.

→ Pipelined Reliable Data transfer protocols:

→ A pipelined reliable data transfer protocol allows the sender to start sending a second data packet before receiving an acknowledgment for the first.

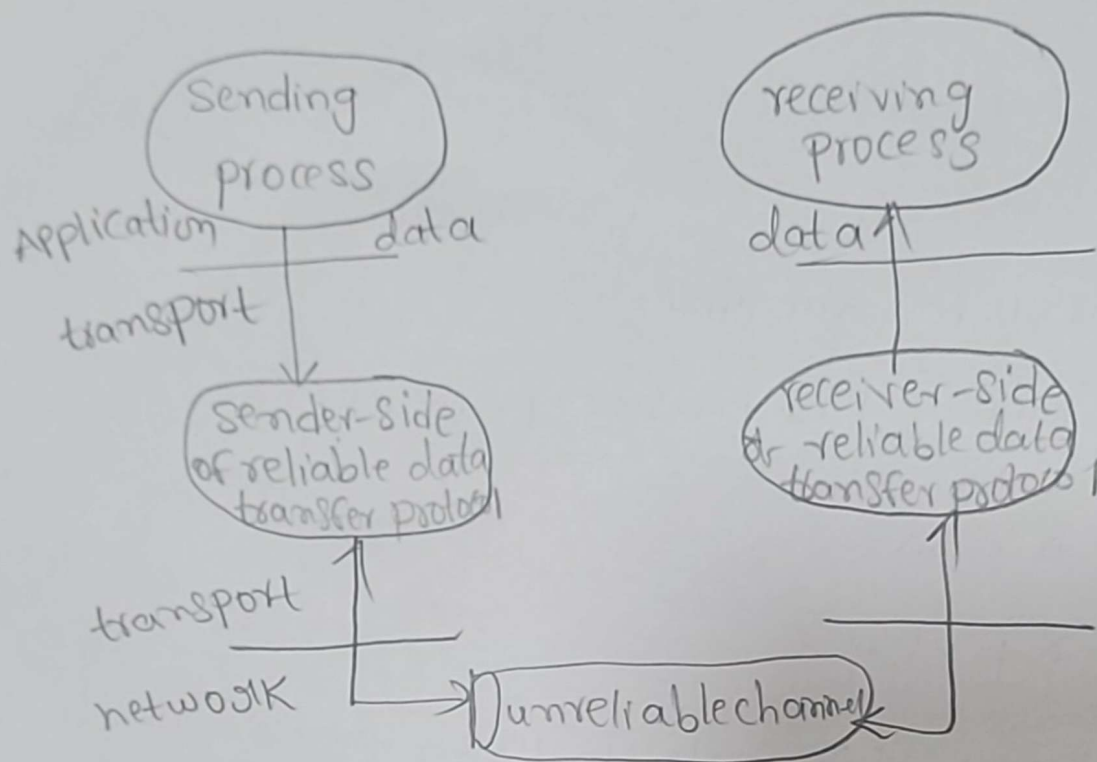
Two types of reliable data transfer protocols:

- 1) Go-Back-N (GBN)
- 2) Selective Repeat (SR)

Go-Back-N (GBN): This protocol allows the sender to send multiple packets without waiting for an acknowledgment.
 → The sender's window is of size n , and the receiver's window size is always 1.

Selective Repeat: This protocol avoids unnecessary retransmissions by having the sender retransmit only those packets that it suspects were received in error.
 → The size of the sender and receiver window must be 1.
 → The sender can send a number of frames specified by a window size without waiting for individual ACK from the receiver.

→



reliable service Implementation

→ pipelining has several consequences for reliable data transfer protocols -

→ The range of sequence numbers must be increased, since each in-transit packet (not counting retransmissions) must have a unique sequence number and there may be multiple, in-transit, unacknowledged packets.

→ pipelining can significantly increase transmission rates.

→ Flow control - Flow control is a technique in the transport layer of the OSI model that regulates the rate at which data is transferred between a sender and receiver.

UDP (User Datagram Protocol) : It is a transport layer protocol

UDP is a part of the Internet protocol suite. Unlike TCP it is an unreliable and connectionless protocol. So there is no need to establish a connection before data transfer. The UDP enables process-to-process communication. UDP is a communication protocol used across the Internet for time-sensitive transmissions such as video playback.

UDP segment structure : A UDP segment used in the User Datagram Protocol, consists of a header and a data payload

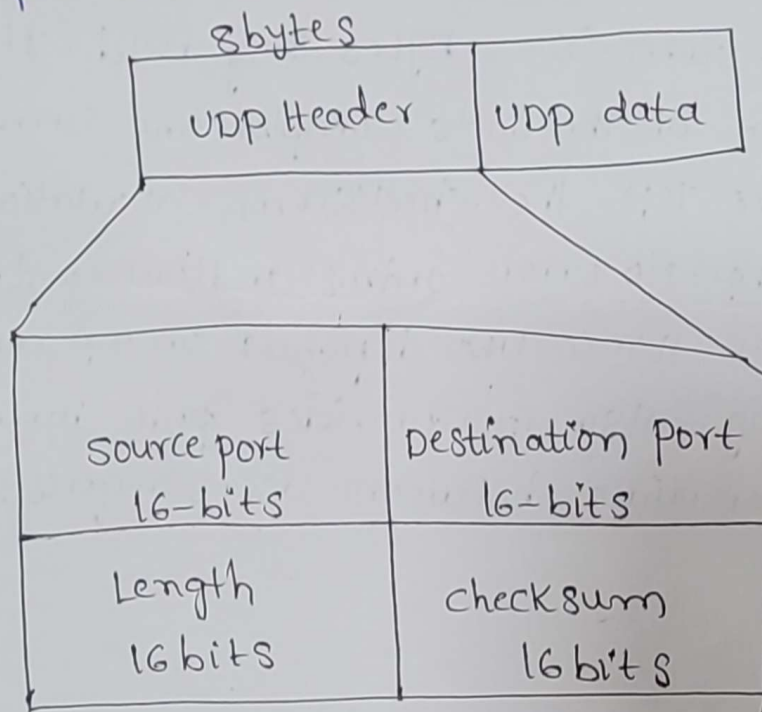


Fig: UDP Header.

UDP Header is an 8-byte fixed and simple header. While for TCP it may vary from 20 bytes to 60 bytes. The first 8 bytes contain all necessary header information and the remaining part consists of data.

→ UDP port number fields are each 16-bit long, therefore the range for port number is defined from 0 to 65535.

→ port number 0 is reserved. port numbers help to distinguish different user requests or processes.

Source port: Source port is a 2 Byte long field used to identify the port number of the source.

Destination port: It is a 2 Byte long field, used to identify the port of the destined packet.

Length: Length is the length of UDP including the header and the data. It is a 16-bits field.

Checksum: checksum is 2 Bytes long field. It is the 16-bit one's complement of the one's complement sum of the UDP header. Unlike TCP, the checksum calculation is not mandatory in UDP. No error control or flow control is provided by UDP. Hence UDP depends on IP and ICMP for error reporting. Also UDP provides port numbers so that it can differentiate between users requests.

Transmission control protocol: (TCP)

→ The transmission control protocol (TCP) is one of the most important protocols of Internet protocols suite.

→ It is most widely used protocol for data transmission in communication network such as Internet.

Features:

→ TCP is reliable protocol. That is, the receiver always send either positive or negative acknowledgement about the data packet to the sender, so that the sender always has bright clue about whether the data packet is reached the destination or it needs to resend it.

→ TCP ensures that the data reaches intended destination in the same order it was sent.

→ TCP is connection oriented. TCP requires that connection between two remote points be established before sending actual data.

→ TCP provides error-checking and recovery mechanism.

→ TCP provides end to end communication.

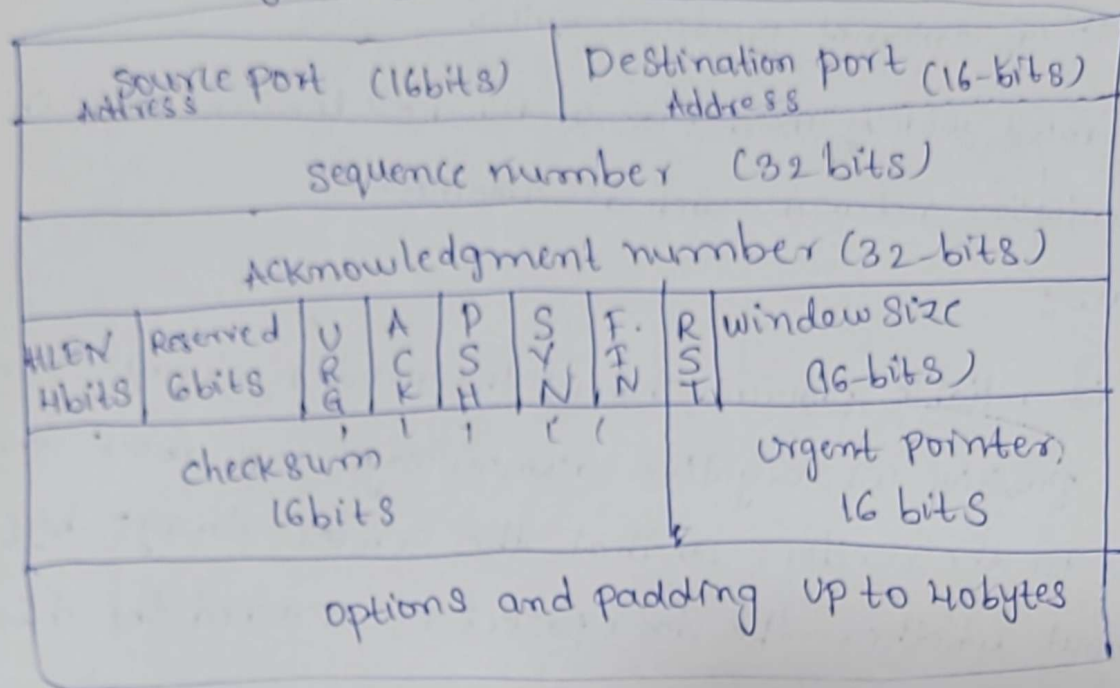
→ TCP provides flow control and quality of service.

→ TCP operates in client/server point to point mode.

→ TCP provides full duplex server (ie) it can perform roles of both receiver and sender.

Header:

The length of TCP header is minimum 20 bytes long and maximum 60 bytes. ^{20-60 bytes}



→ 40 bytes are for options. If there are no options, a header is 20 bytes else it can be of upmost 60 bytes.

Header fields:

→ Source Port Address: - A 16-bit field that holds the port address of the application that is sending the data segment.

→ Destination Port Address: - A 16-bit field that holds the port address of the application is the host that is receiving the data segment.

Sequence Number:

A 32-bit field that holds the sequence number (ie) the byte number of the first byte that is sent in that particular segment. It is used to reassemble the message at the receiving end of the segments that are received out of order.

Acknowledgement number: - A 32-bit field that holds the Acknowledgement number (i.e.) the byte number that the receiver expects to receive next. It is an acknowledgement for the previous bytes being received successfully.

→ Header length (HLEN): This is a 4-bit field that indicates the length of the tcp header by a number of 4-byte words in the header. (i.e.) if the header is 20 bytes (min length of tcp header), then this field will hold 5 (because $5 \times 4 = 20$) and the maximum length: 60 bytes, then it'll hold the value 15 (because $15 \times 4 = 60$). Hence, the value of this field is always between 5 & 15.

Control flags: These are 6 1-bit control bits that control connection establishment, connection termination, connection abortion, flow control, mode of transfer etc. Their function is:

URG: Urgent pointer is valid.

ACK: Acknowledgement number is valid (used in case of cumulative acknowledgment).

PSH: Request for push

RST: Reset the connection

SYN: Synchronize sequence numbers.

FIN: Terminate the connection.

window size: This field tells the window size of the sending tcp in bytes.

checksum: This field holds the checksum for error control. It is mandatory in tcp as opposed to UDP.

urgent pointer: This field (Valid only if the URG control flag is set) is used to point to data that is urgently required that needs to reach the receiving process at the earliest. The value of this field is added to the sequence number to get the byte number of the last urgent byte.

14

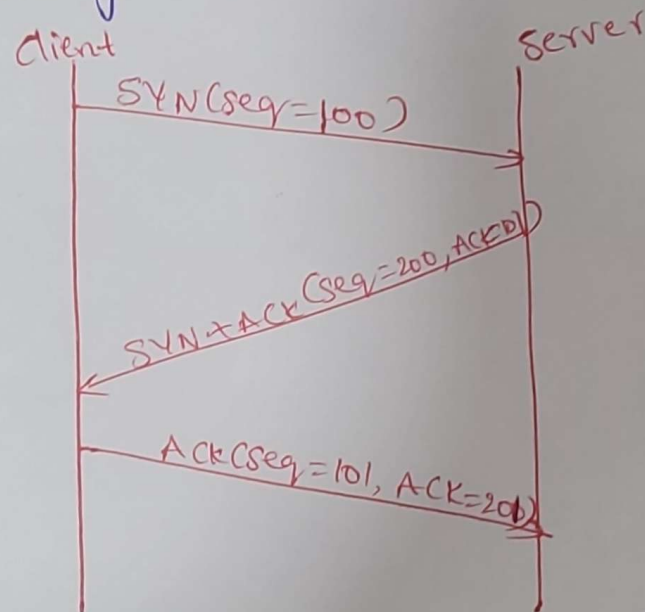
TCP connection Management : TCP connection management mainly involves in 3 phase or stages.

1) TCP connection Establishment

2) Data transfer

3) TCP connection Termination

1) TCP connection Establishment : TCP is Transmission control protocol and it is a connection-oriented protocol. So here connection has to be established ^{btw} client & server before transmitting the data.



→ TCP use full duplex communication (ie) the data is transmitted in both directions simultaneously. TCP connections use the three-way Handshake to create a connection.

(i) client sends SYN segment to the server to establish a connection. Sends initial sequence number.

(ii) SYN+ACK

server acknowledges the client segment & sends SYN segment

Here seq number we are sending 200 $\Rightarrow$ Seq = 200,

ACK = 101 $\Rightarrow$ means previous sequence number + 1 (ie) 101.

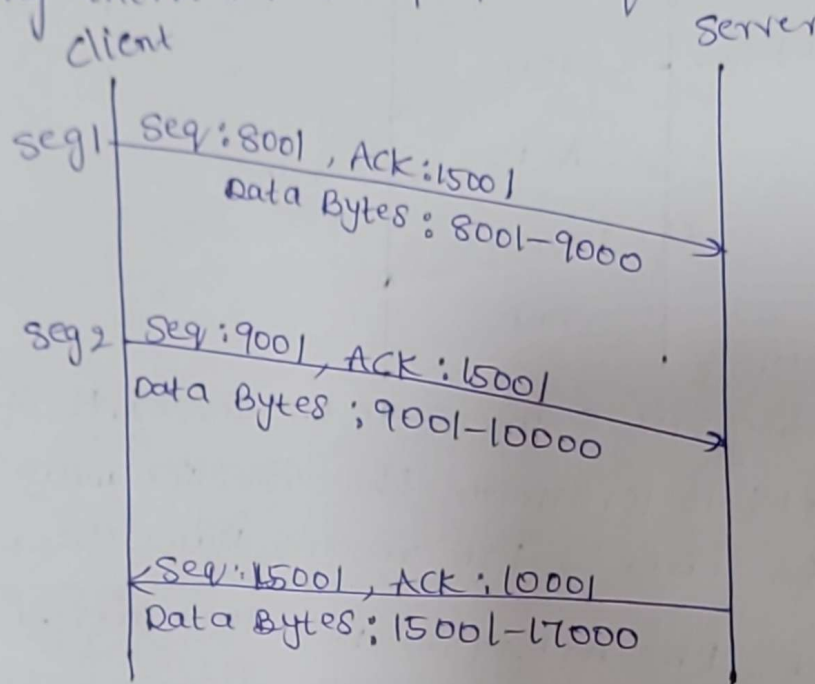
(iii) ACK: Client acknowledges the server. Here Seq = 101,

& ACK = 201. $\Rightarrow$ means sequence number + 1 (ie) 200 + 1 = 201.

$\rightarrow$

2) Data Transfer: once the connection is established, both the client and server can exchange data in both directions simultaneously. Known as full duplex mode!

→ TCP ensures data is delivered in the correct order and without errors by reassembling packets, resending lost packets & reassembling them in the proper sequence at the destination.



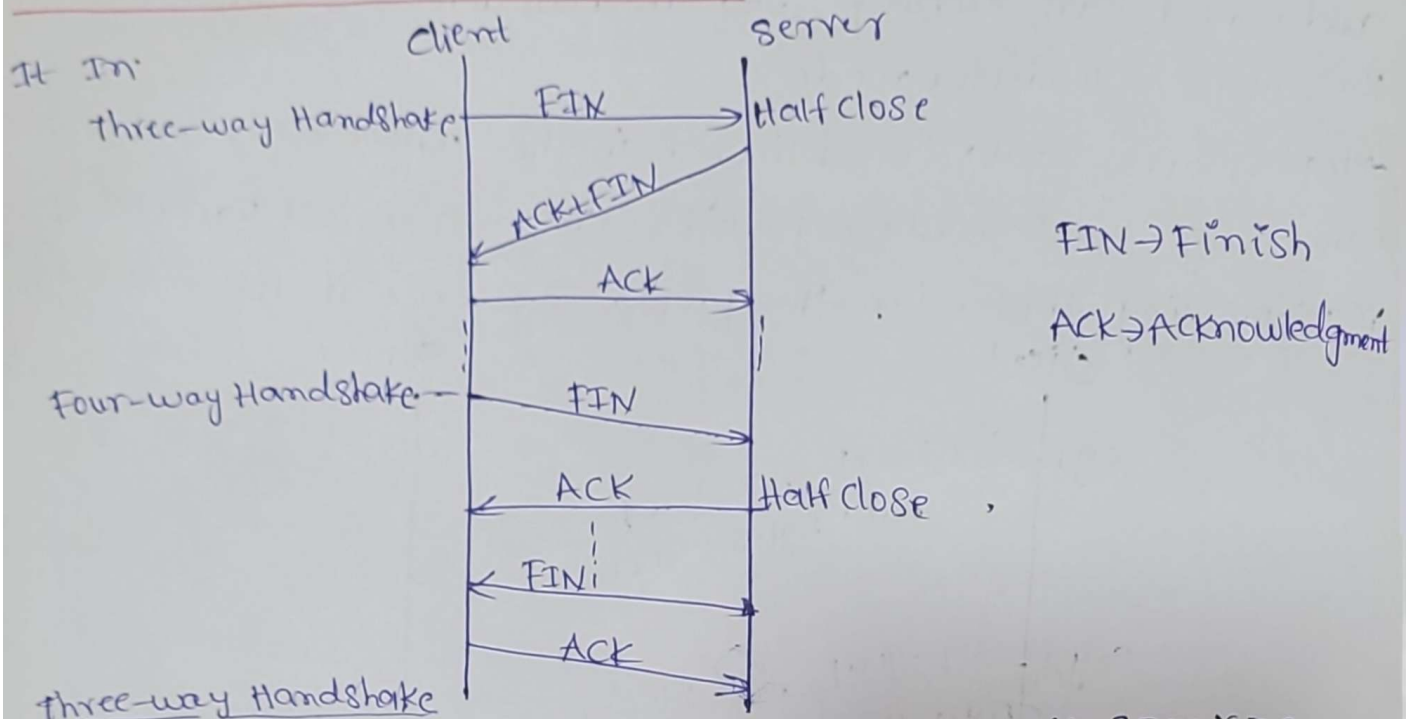
Step: 1
→ Client is sent seg1 & seg2. Here seg1 Data Bytes: 8001-9000.
seg2 Data Bytes: 9001-10,000.

Here 8001 sequence number of segment, & ACK: 15001 specifies the client is expecting the segment ^{from server} the starting byte of segment is 15001.

Step: 2: Sequence number for second segment is 9001, ACK specifies the client is expecting a segment from the server where the starting byte of segment is 15001.

Step 3: server is sending a segment to the client. Here along with the data it is sending acknowledgment (ie) called Piggybacking - Here Data Bytes: 15001-17000. Here segment is 15001

3) TCP connection termination:



Three-way Handshake
→ client is sending FIN segment to the server. It specifies the client wants to terminate the connection with server, then client to server is FIN means Half close.

→ then we get the ACK+FIN from the server. After receiving the client it is giving ACK to the server.

- Steps:
- 1) FIN
 - 2) ACK+FIN
 - 3) ACK

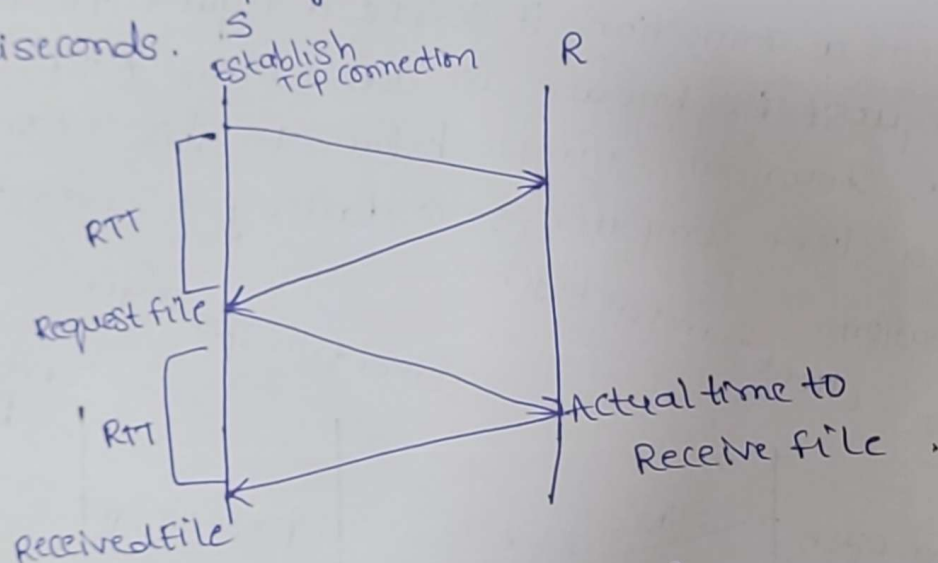
Four-way Handshake: client is sending FIN seg to server.

→ If server has data to send to client, server is going to perform operations. After that only server give FIN seg to the client, then client responds ACK to the server.

Round-Trip Time Estimation and Timeout :

Round-trip time (RTT) is the duration for a network request to travel from its origin to a destination and for the acknowledgment or response to return. A timeout is the maximum time a sender waits for an acknowledgment before retransmitting the data.

→ RTT also called round trip delay is a crucial tool in determining the health of a network. It is the time between a request for data and the display of that data. It is the duration measured in milliseconds.



RTT (Round trip time) Measurement

Applications of RTT :

→ Round trip time refers to a wide variety of transmissions such as wireless internet transmissions and satellite transmissions.

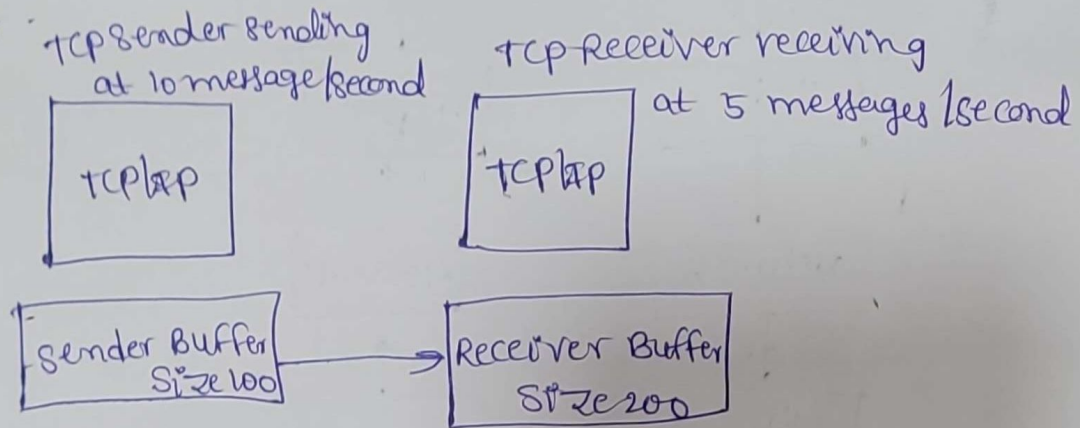
Advantage of RTT :

→ It allows users and operators to identify how long a single will take to complete the transmission.

→ It also determines how fast a network can work and the reliability of the network.

Flow control in TCP: In a communication network, in order for two network hosts to communicate with each other, one has to send a packet while another host has to receive it.

→ For the slow sender and fast receiver, no flow control is required whereas for the fast sender and slow receiver, flow control is important.



→ Here the diagram given, there is a fast sender and slow receiver.

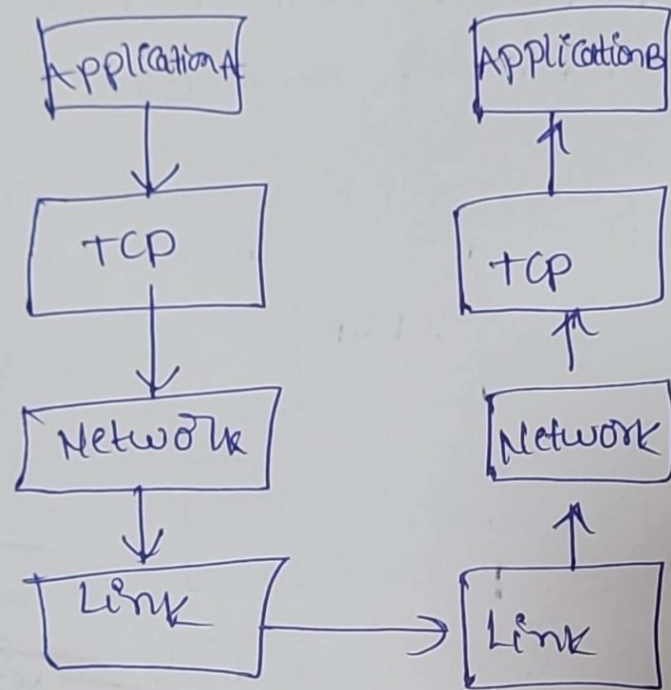
→ Here the receiver is receiving the message sent by the sender at the rate 5 messages per second while the sender is sending the messages at the rate of 10 messages per second.

→ When the sender sends the message to the receiver, it gets into the network queue of the receiver.

→ Once the user reads the message from the application the message gets cleared from the queue and the space becomes free.

→ According to the mentioned speed the sender and receiver the receiver queue will be getting shortened and the buffer space will reduce at the speed of 5 message/second. Since, the receiver buffer size can accommodate 200 messages. Hence in 40 seconds, the receiver buffer will become full.

So, after H_0 seconds, the messages will start dropping as there will be no space remaining for the incoming messages.



→ The sender writes the data to a socket and send it to the transport layer which in tcp in this case. The transport layer will then wrap this data and send it to the network layer which will route it to the receiving node.

Congestion Control

→ TCP congestion control is a method used by the TCP protocol to manage data flow over a network and prevent congestion.

→ TCP uses a congestion window and congestion policy that avoids congestion.

→ Congestion policy in TCP:

1) Slow start phase: starts slow increment is exponential to the threshold.

2) Congestion Avoidance phase: After reaching the threshold increment by 1.

3) Congestion Detection phase: The sender goes back to the slow start phase or the congestion avoidance phase.

Slow Start phase:

Exponential Increment: In this phase after every RTT the congestion window size increments exponentially.

Example: If the initial congestion window size is 1 segment and the first segment is successfully acknowledged the congestion window size becomes 2 segments.

→ If the next transmission is also acknowledged the congestion window size doubles to 4 segments.

→ This exponential growth continues as long as all segments are successfully acknowledged.

Initially $cwnd = 1$

After 1 RTT, $cwnd = 2^1(1) = 2$

2 RTT, $cwnd = 2^2(2) = 4$

3 RTT $cwnd = 2^3(3) = 8$

2) congestion Avoidance phase;

Additive increment: This phase starts after the threshold value also denoted as $ssthresh$.

→ The size of $cwnd$ (congestion window) increases additive after each RTT $cwnd = cwnd + 1$

example: If the congestion window size is 20 segments and all 20 segments are successfully acknowledged within an RTT, the congestion window size would be increased to 21 segments in the next RTT.

If all 21 segments are again successfully acknowledged the congestion window size will be increased by 22 segments and so on.

Initially $cwnd = i$

After 1 RTT, $cwnd = i + 1$

2 RTT, $cwnd = i + 2$

3 RTT, $cwnd = i + 3$

Congestion Detection Phase:

19

Multiplicative Decrement: If congestion occurs the congestion window size is decreased.

→ The only way a sender can guess that congestion has happened is the need to retransmit a segment.

→ Retransmission is needed to recover a missing packet that is assumed to have been dropped by a router due to congestion.

→ Retransmission can occur in one or two cases.

→ When the RTO timer time out or when three duplicate ACKs are received.

Case 1: Retransmission due to timeout - in this case the congestion possibility is high.

(a) $ssthresh$ is reduced to half of the current window size.

(b) $set\ cwnd = 1$

(c) start with the slow start phase again.

Case 2: Retransmission due to 3 acknowledgment duplicates.

The possibility is less.

(a) $ssthresh$ value reduces to half of the current window size.

(b) $set\ cwnd = ssthresh$

(c) start with congestion avoidance phase.

Differences between TCP & UDP

TCP/IP

1) TCP is a connection-oriented protocol.

2) TCP Header length is 20 to 60 bytes.

3) More Reliable

4) Slower Transmission

5) Packets order can be preserved or can be rearranged.

6) Uses three-way handshake model for connection.

7) TCP packets are heavy weight

8) Offer error checking mechanism

9) Protocols like HTTP, FTP, SMTP etc.

HTTP → Hyper text Transfer protocol

FTP → File transfer protocol

SMTP → Simple Mail Transfer protocol.

10) TCP ensure data delivery in order.

UDP

1) UDP is a connectionless protocol.

2) UDP header length is 8 bytes.

3) Less Reliable

4) Faster transmission

5) Packets order is not fixed and packets are independent of each other.

6) No Handshake for establishing the connection.

7) UDP Packets are light weight.

8) No error checking mechanism

9) Protocols like DNS, RTP (Real time Transport protocol), etc.

DNS → Domain Name System

10) UDP does not guarantee data delivery.

V Application layer

→ Application layer of OSI (Open System Interconnection) model is the top layer in this model and takes care of network

communication.

- The application layer provides the functionality to send and receive data from users.
- It acts as the interface between the user and the application.
- The application provides services like file transmission, mail service and many more.
- This layer provides several ways for manipulating the data (information) which actually enables any type of user to access network with ease.
- This layer also makes a request to its bottom layer, which is presentation layer for receiving various types of information from it.

Functions of Application Layer:

Data from user $\rightleftharpoons$ Application layer $\rightleftharpoons$ Data from presentation layer

- Application layer provides a facility by which users can forward several emails and it also provides a storage facility.
- This layer allows users to access, retrieve and manage files in a remote computer.

→ It allows users to log on as a remote host.

→ This layer provides access to global information about various services.

→ This layer provides access to global information about various services.

→ This layer provides services which include: email, transferring files, distributing results to the user, directory services, network resources and so on.

→ Application layer is basically not a function, but it performs application layer functions.

→ Application layer helps us to identify communication partners, and synchronizing communication.

→ This layer allows users to interact with other software applications.

→ Working of Application Layer:

→ At first, client sends a command to server and when server receives that command, it allocates port number to client.

→ Thereafter, the client sends an initiation connection request to server and when server receives request, it gives acknowledgment (ACK) to client through client has successfully established a connection with the server.

→ Therefore, now Client has access to server through which it may either ask server to send any types of files or other documents or it may upload some files or documents on server itself.

Services provided by Application Layer:

→ The services provided by application layer in OSI model are;

→ It provides interface between user and application.

→ It is used for file transfer.

→ File transfer, Access & Management

→ Addressing

→ Mail Services

→ Directory services [means global information]

FTP Transmission Mode:

→ FTP can transfer a file across the data connection by using one of the following three transmission modes.

- 1) Stream mode
- 2) Block mode
- 3) Compressed mode

1) Stream mode:

→ It is the default mode. Data are delivered from FTP to TCP as a continuous stream of bytes. TCP is responsible for chopping data into segments of appropriate size. If the data are simply a stream of bytes (file structure), no end-of-file is needed.

→ End-of-file in this case is the closing of the data connection by the sender.

→ If the data are divided into records (record structure), each record will have a 1-byte end-of-record (EOR) character and the end of the file will have a 1-byte end-of-file (EOF) character.

Block mode: Data can be delivered from FTP to TCP in blocks. In this case each block is preceded by a 3-byte header. The first byte is called the block descriptor, the next 2 bytes define the size of the block in bytes.

3) compressed mode: If the file is big, the data can be compressed. The compression method normally used is run-length encoding. In this method, consecutive appearances of data unit are replaced by one occurrence and the number of repetitions. In a text file, this is usually spaces (blanks). In a binary file, null characters are usually compressed.

Principles of Networking Applications:

The principles of Network Applications are fundamental concepts that govern the design and development of applications that run on a computer network.

→ these principles encompass several key aspects of network applications, including:

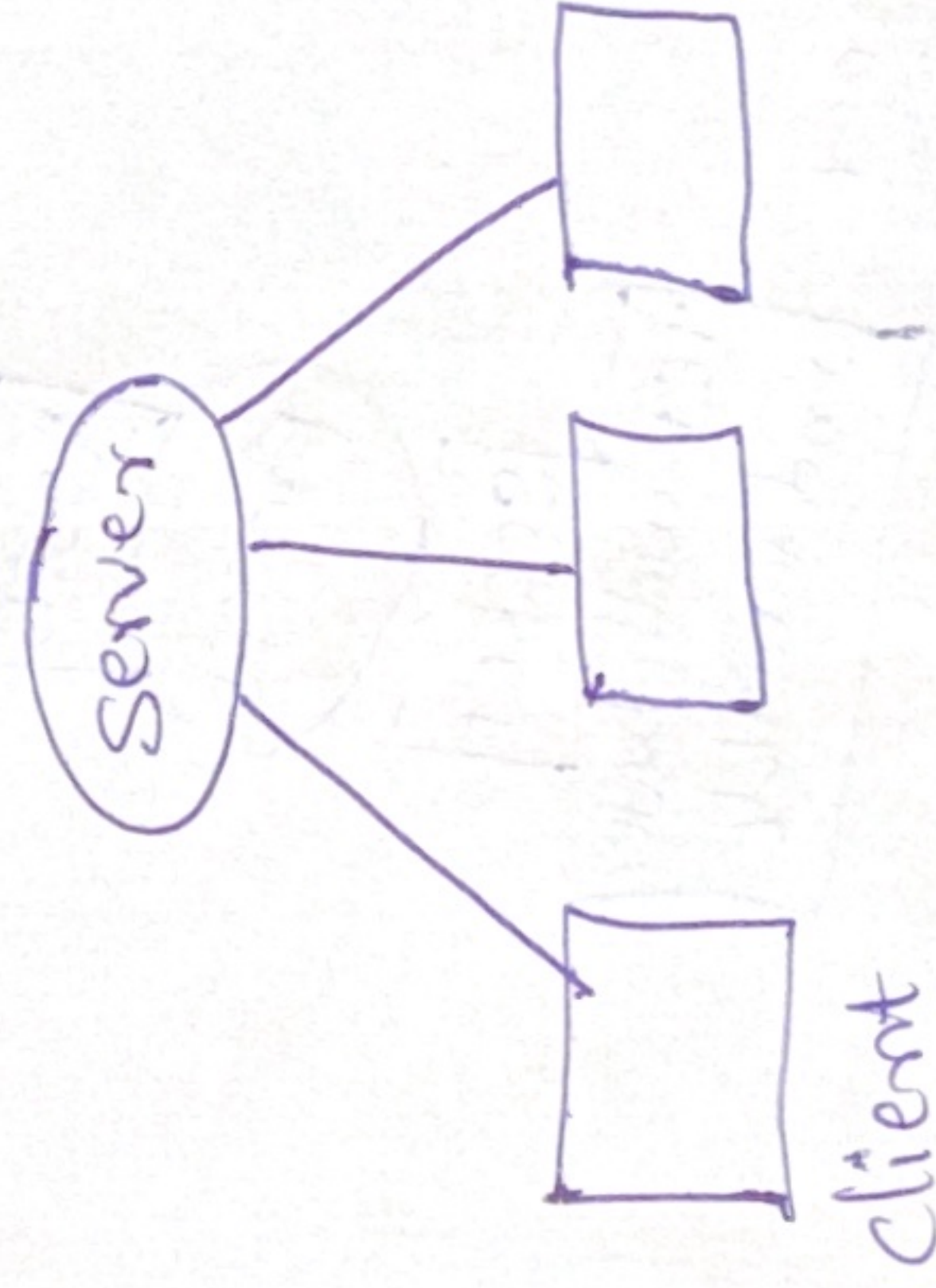
- 1) Network Application Architectures,
- 2) Process Communicating
- 3) Transport Services available to Applications
- 4) Transport Services provided by Internet.
- 5) Application Layer Protocol.

1) Network Application Architectures:

1) Client-Server

2) Peer-to-peer

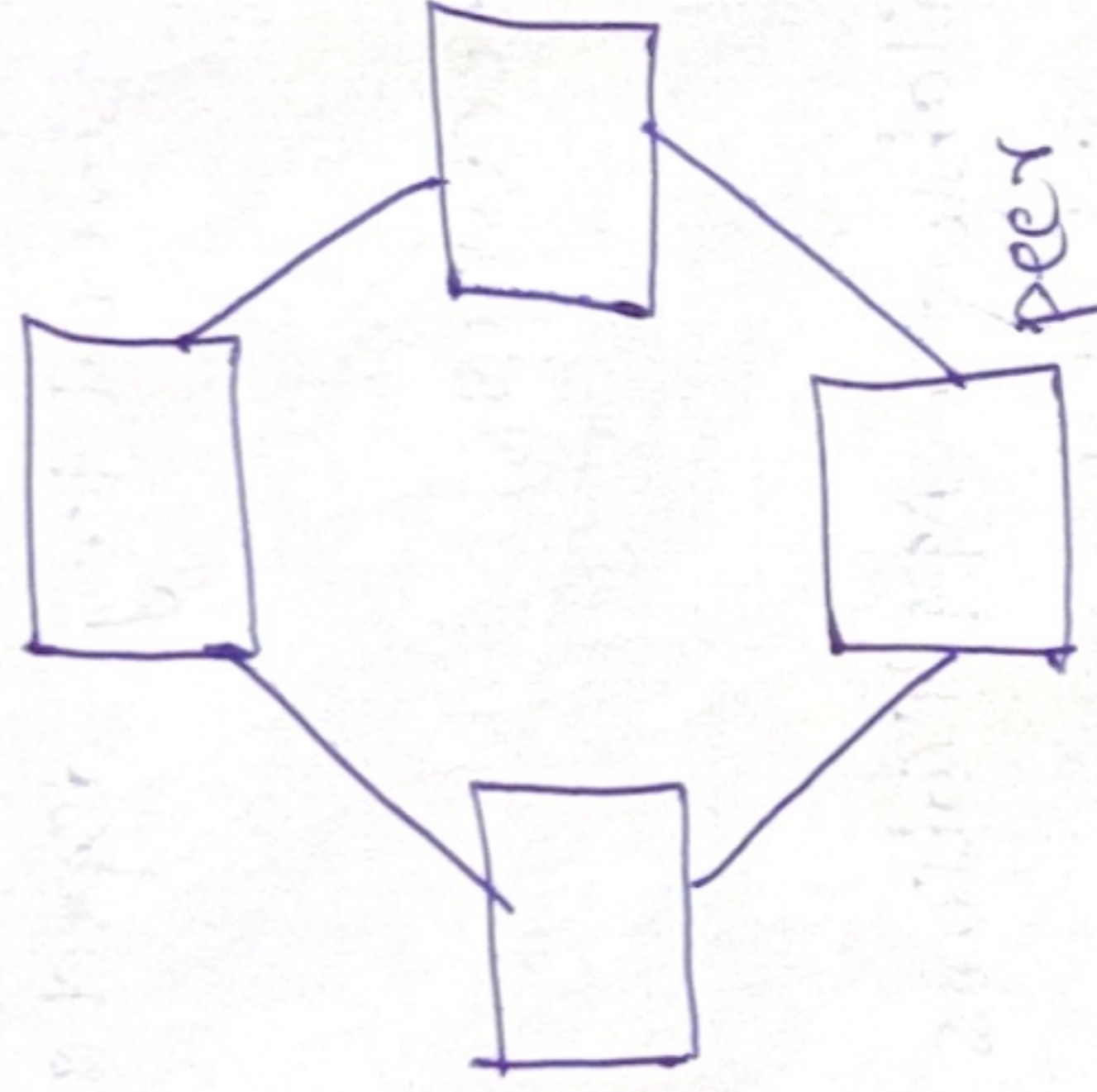
1) Client-server:



→ In this architecture, one component acts as a client and makes request to a server component, which provides the requested services.

→ This architecture is commonly used in web applications, where the client is a web browser and the server is a web server.

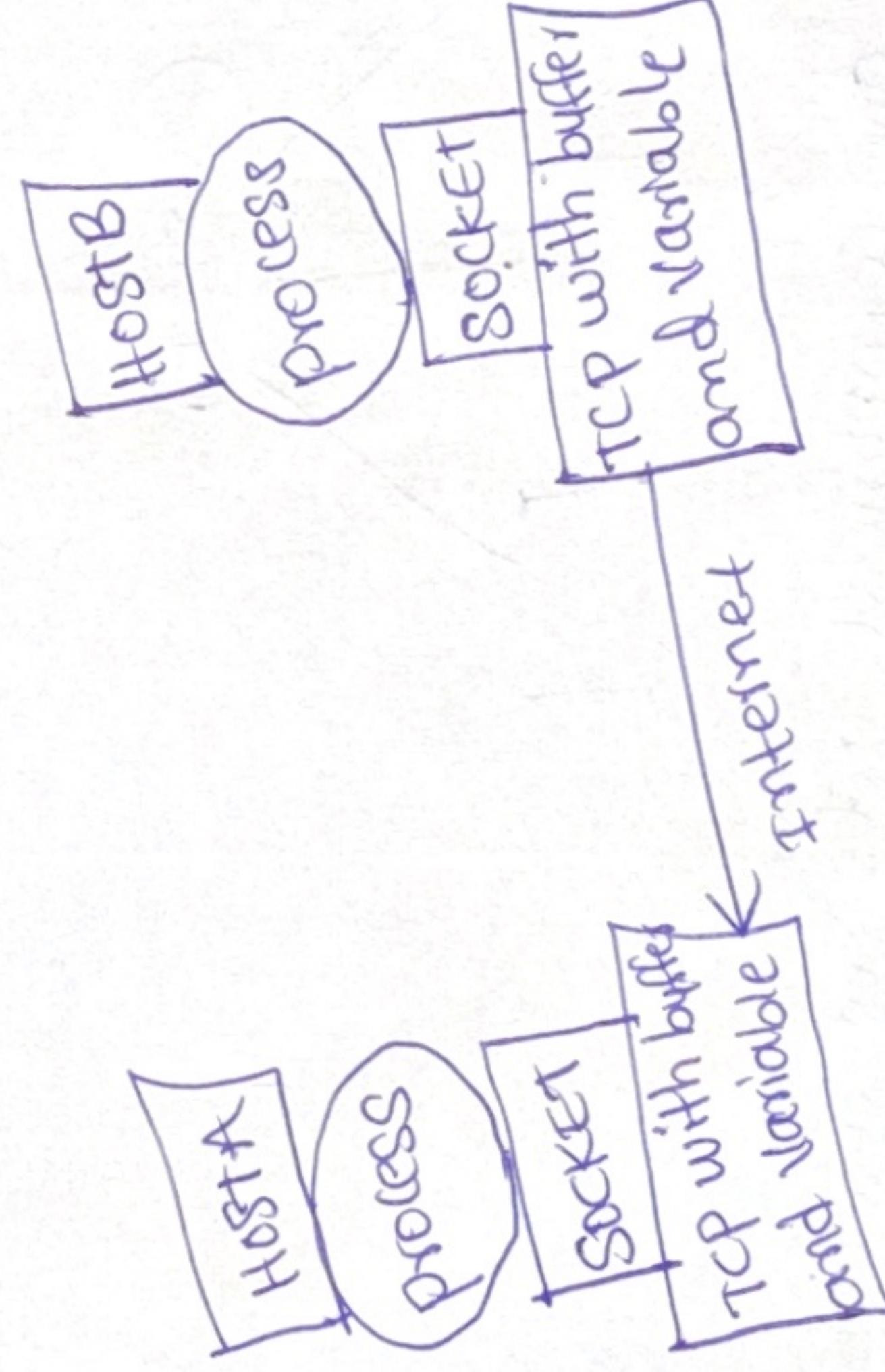
2) Peer-to-Peer Architecture:



→ In this architecture, every component is both a client and a server and each component can communicate directly with any other component. This architecture is commonly used in file sharing applications, where each user's device acts as both a client and a server.

Process Communication: (Interface between process and

Network)



- has
- Host B to send a message to Host A
 - First Identify the hosts by IP address.

→ Network applications consist of processes that communicate by exchanging messages.
 Ex: A web browser.

→ Here Socket Interface is the door between application and the Network.

→ It provides an interface for sending and receiving messages through the network.

→ To communicate, an application needs IP address & port number.

→ Here TCP is used for reliable, connection-oriented, ensures all data arrives in order.

→ Successfully send data from Host B to Host A.

3) Transport Services available to Applications? → that are

1) Reliable data transfer

2) Throughput

3) Timing

4) Security

Reliable data transfer: Ensures that data is delivered without loss, duplication, or error.

→ It uses Acknowledgments (ACKs) and retransmissions for reliability.

Ex: TCP.

Throughput: The amount of data transferred per second.

Throughput is the rate at which data is successfully transmitted from one place to another over a network in given amount of time.

Timing: Within time data transferred from one to another.

Security: End-End encryption or service it is provided.

→ Transport service provided by Internet:

1) TCP (Connection oriented)

2) UDP (Connectionless)

→ TCP: It provides reliable, connection based data transfer. Ensures data is delivered in order, without loss or duplication.

→ UDP: provides fast, Connectionless data transfer with no guarantee of delivery, ordering, or reliability. It is useful for real-time applications.

Application Layer Protocol:

1) Type of message : Identify a code.

2) Syntax of message : which language it written

3) Semantics of message : meaning of message

4) Rules for determining when and how a process sends and

responds to a message.

→

File Transfer: Transferring files from one computer to another is one of the most common tasks expected from a networking or Internetworking environment. The greatest volume of data exchange in the Internet today is due to file transfer.

File transfer protocol (FTP): FTP is the standard mechanism

provided by TCP/IP for copying a file from one host to another. Although transferring files from one system to another seems simple and straightforward.

→ FTP differs from other client/server applications in that it establishes two connections between the hosts.

→ one connection is used for data transfer, the other for control information (commands and responses),

→ separation of commands and data transfer makes FTP more efficient.

→ The control connection uses very simple rules of communication.

→ We need to transfer only a line of command or a line of response at a time.

→ The data connection on the other hand needs more complex rules due to the variety of data types transferred.

→ However the difference in complexity is at the FTP level, not TCP. For TCP, both connections are treated the same.

→ FTP uses two well-known TCP ports: Port 21 used for the control connection and port 20 is used for data connection.

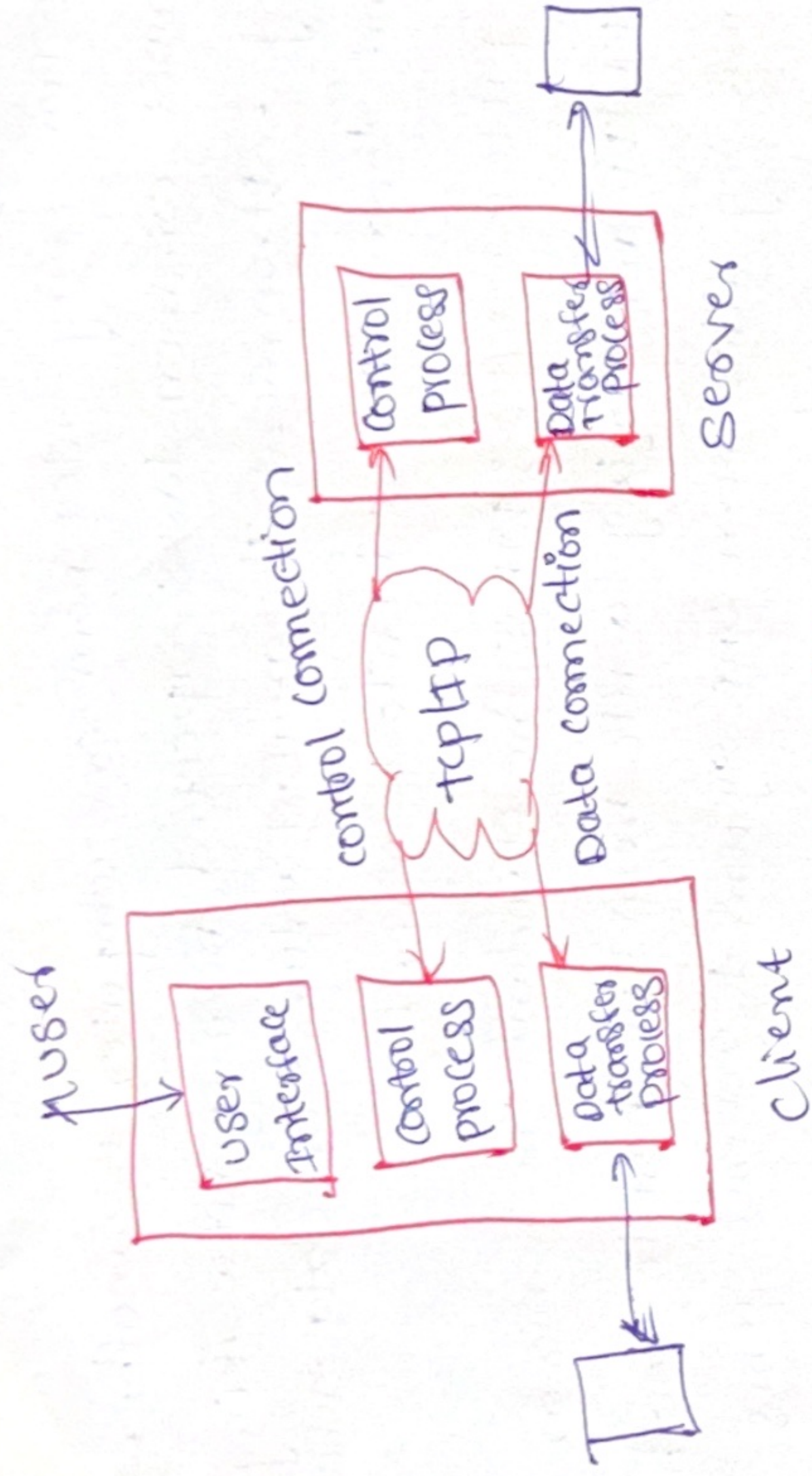


Fig: FTP

Figure shows the basic model of FTP. The Client has three-

Components: User Interface, client control process, and the client data transfer process.

→ The server has two components; the server control process and server data transfer process.

→ The control connection is made between the control processes.

→ The data connection is made between the data transfer processes.

→ The control connection remains connected during the entire interactive FTP session.

→ The data connection is opened and then closed for each file transferred.

→ It opens each time commands that involve transferring files are used, and it closes when the file is transferred.

→ In other words, when a user starts an FTP session, the control connection opens, while the control connection is open, the data connection can be opened and closed multiple times if several files are transferred.

Communication over Control Connection: [Command & Response]

- FTP uses the same approach as SMTP to communicate across the Control Connection. It uses the 7-bit ASCII Character set.
- Communication is achieved through Commands and responses.
- This simple method is adequate for the Control Connection because we send one Command (or response) at a time.
- Each Command or response is only one short line, so we need not worry about file format or file structure.
- Each line

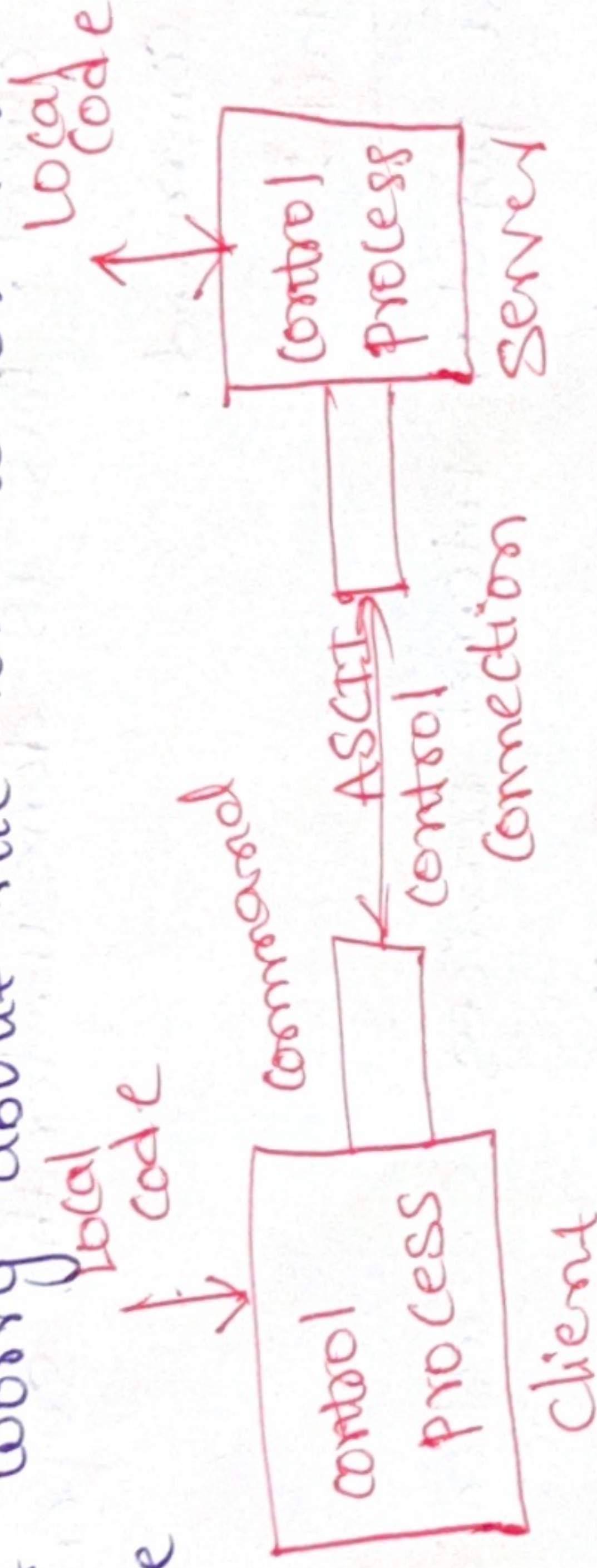


Fig: Using Control Connection

Communication over Data Connection:

- The purpose of the data Connection is data Connection is different from that of the Control Connection, we want to transfer files through the data Connection.
- File transfer occurs over the data Connection under the control of the Commands sent over the Control Connection.
- However, we should remember that file transfer in FTP means one of three things:

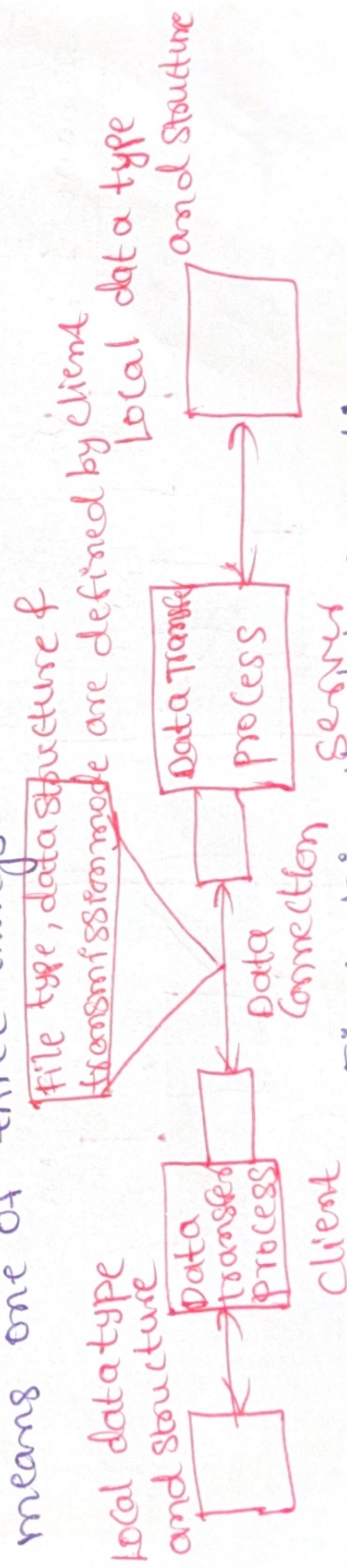


Fig: Using the data Connection

- 1) A file is to be copied from the server to the client. This is called retrieving a file, it is done under the supervision of the ~~STOP~~^{RETR} command.
- 2) A file is to be copied from the client to the server. This is called storing a file. It is done under the supervision of the STOP command.
- 3) A list of directory or file names is to be sent from the server to the client. This is done under the supervision of the LIST command.

→ Electronic Mail:

→ one of the most popular Internet services is electronic mail.

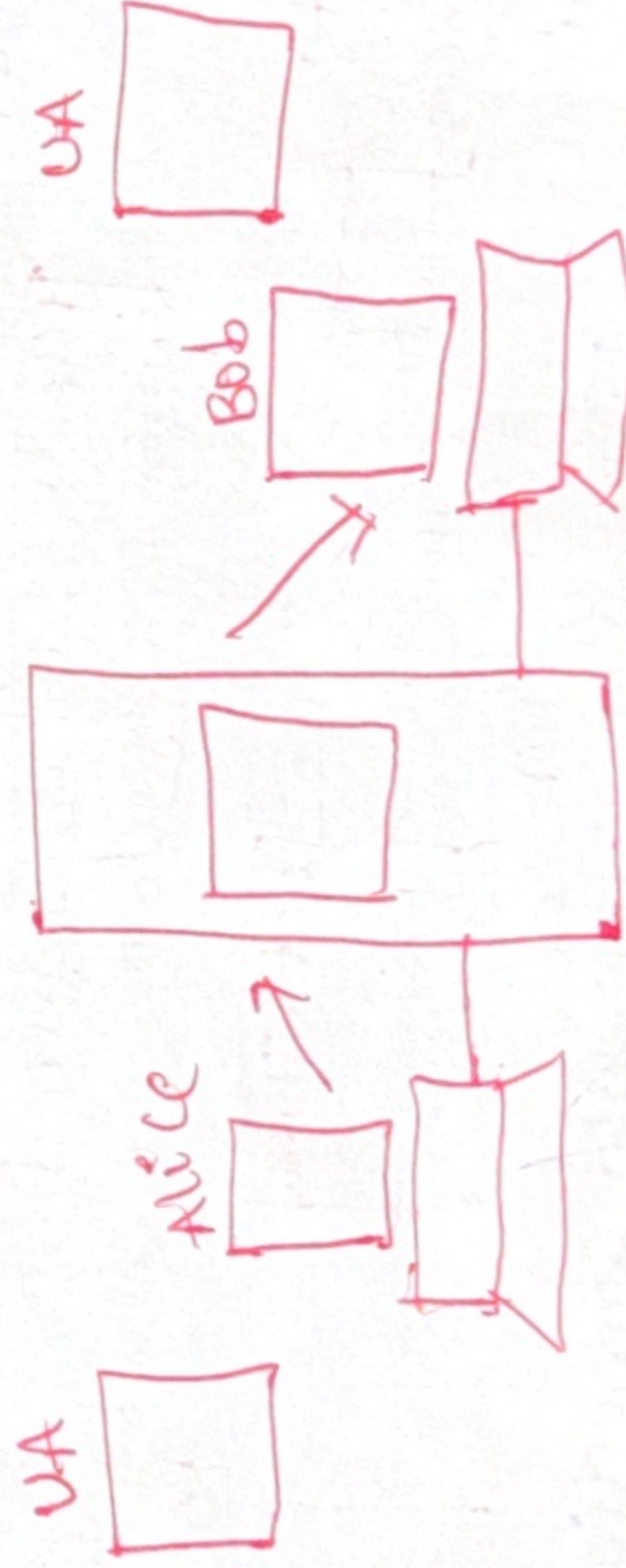
→ At the beginning of the Internet era, the messages sent by electronic mail were short and consisted of text only, they let people exchange quick memos.

→ Today, electronic mail is much more complex. It allows a message to include text, audio, and video. It also allows one message to be sent to one or more recipients.

→ The general architecture of an e-mail system including the three main components: "User agent (UA), Message transfer agent (MTA) and Message Access Agent (MAA)".

E-Mail Scenarios:

Scenario - 1:

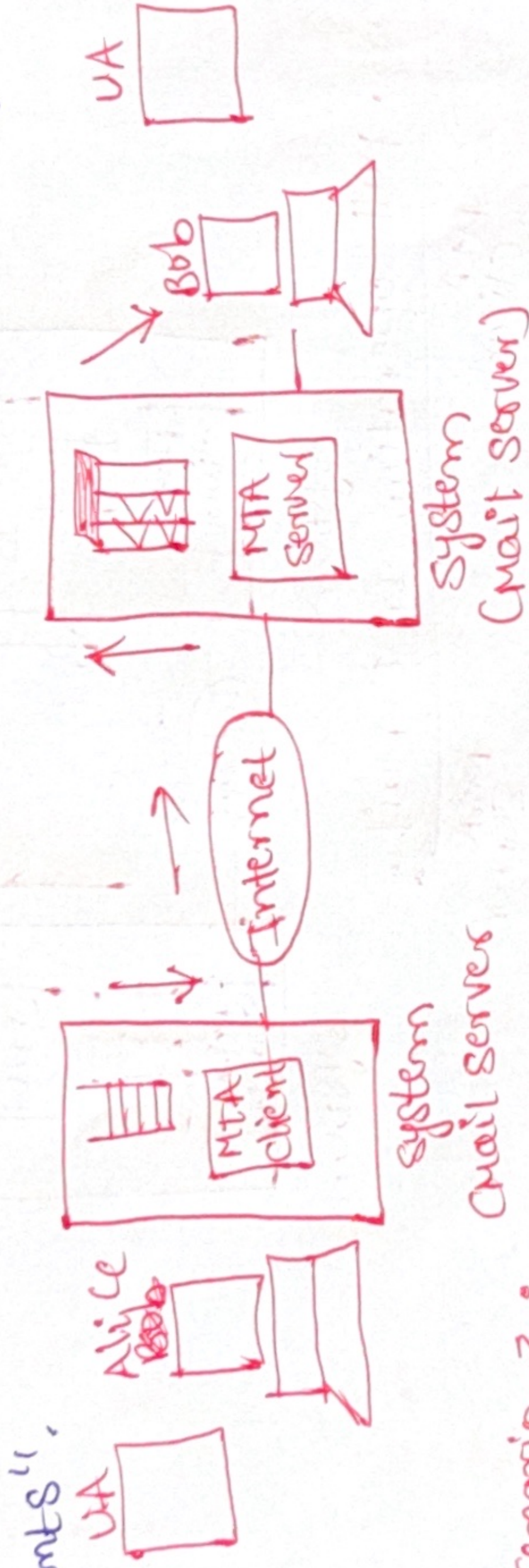


- In this the first scenario, the sender and the receiver of an email users are on the same system.
- They are directly connected to a shared system.
- So it requires two "User Agents".

Scenario-2:

- In the second scenario, the sender and the receiver of the email users are on the different systems.
- The messages need to be sent over the 'Internet'.

- So it requires two "User Agents" and pair of "Message Transfer Agents".

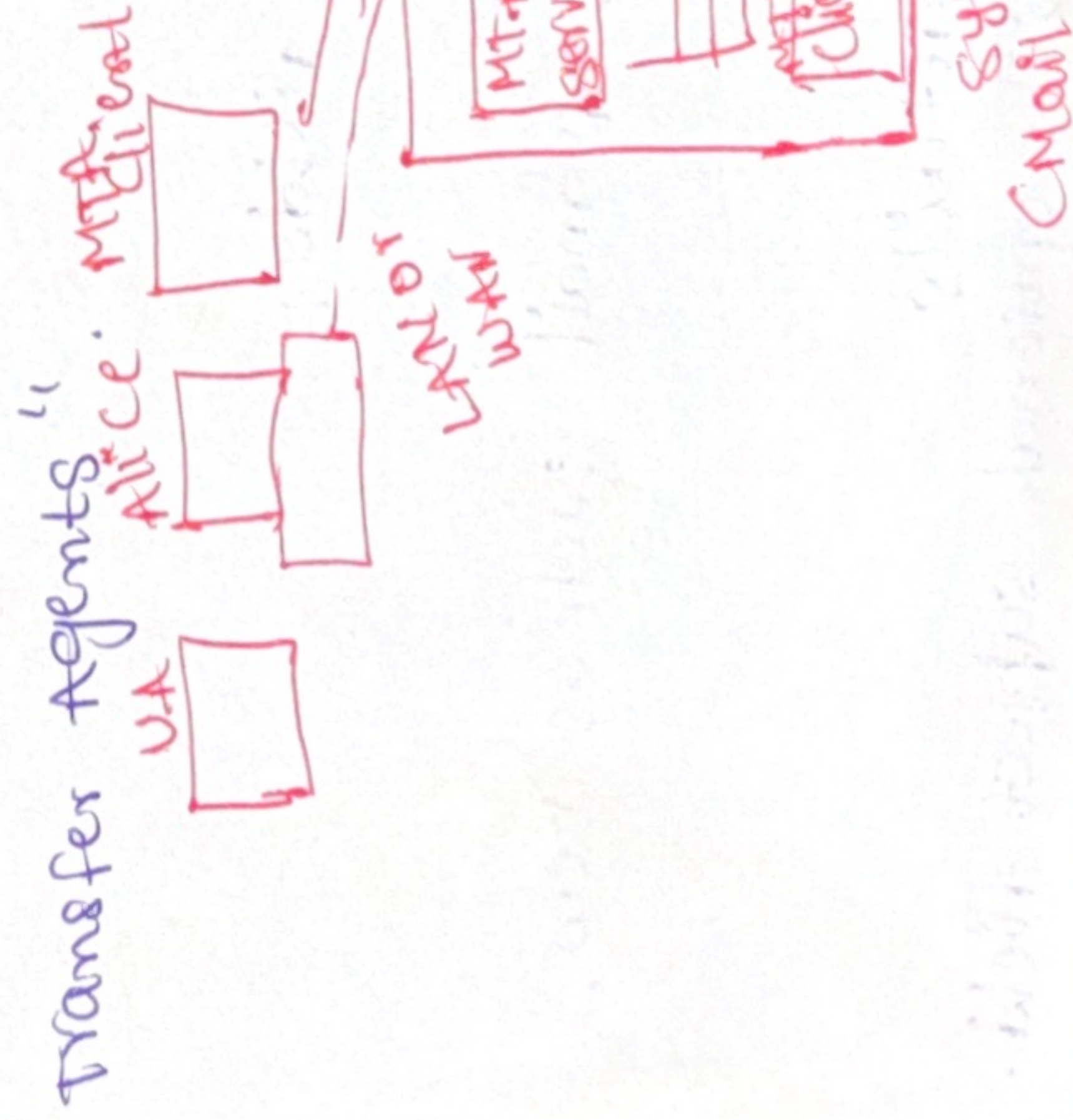


Scenario-3:

- In the third scenario, the sender is connected to the 'mail server' via 'LAN' or 'WAN'.

- So it requires two "User Agents" and pair of 'message

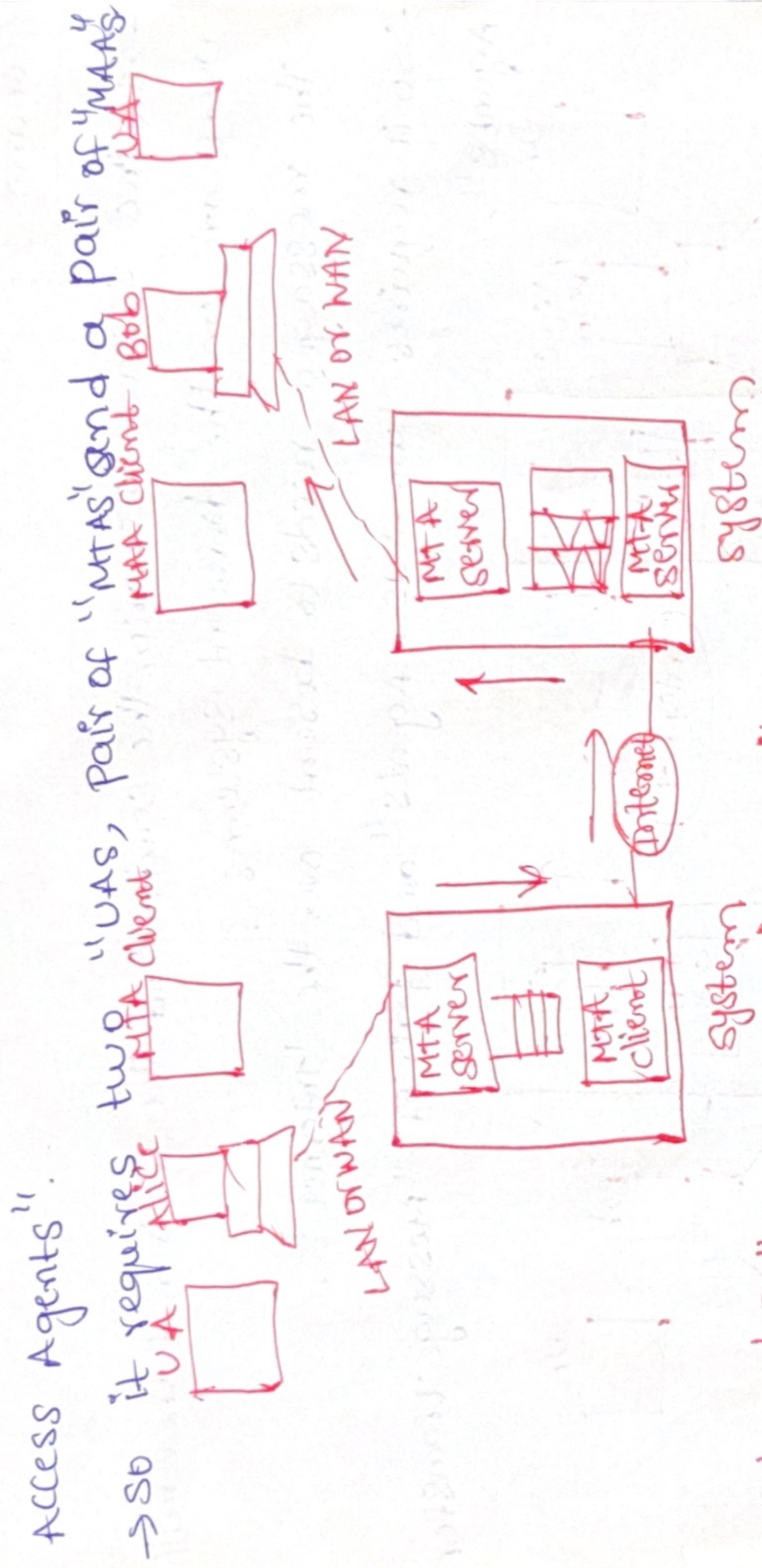
UA → User Agent
MTA → Message Transfer Agent



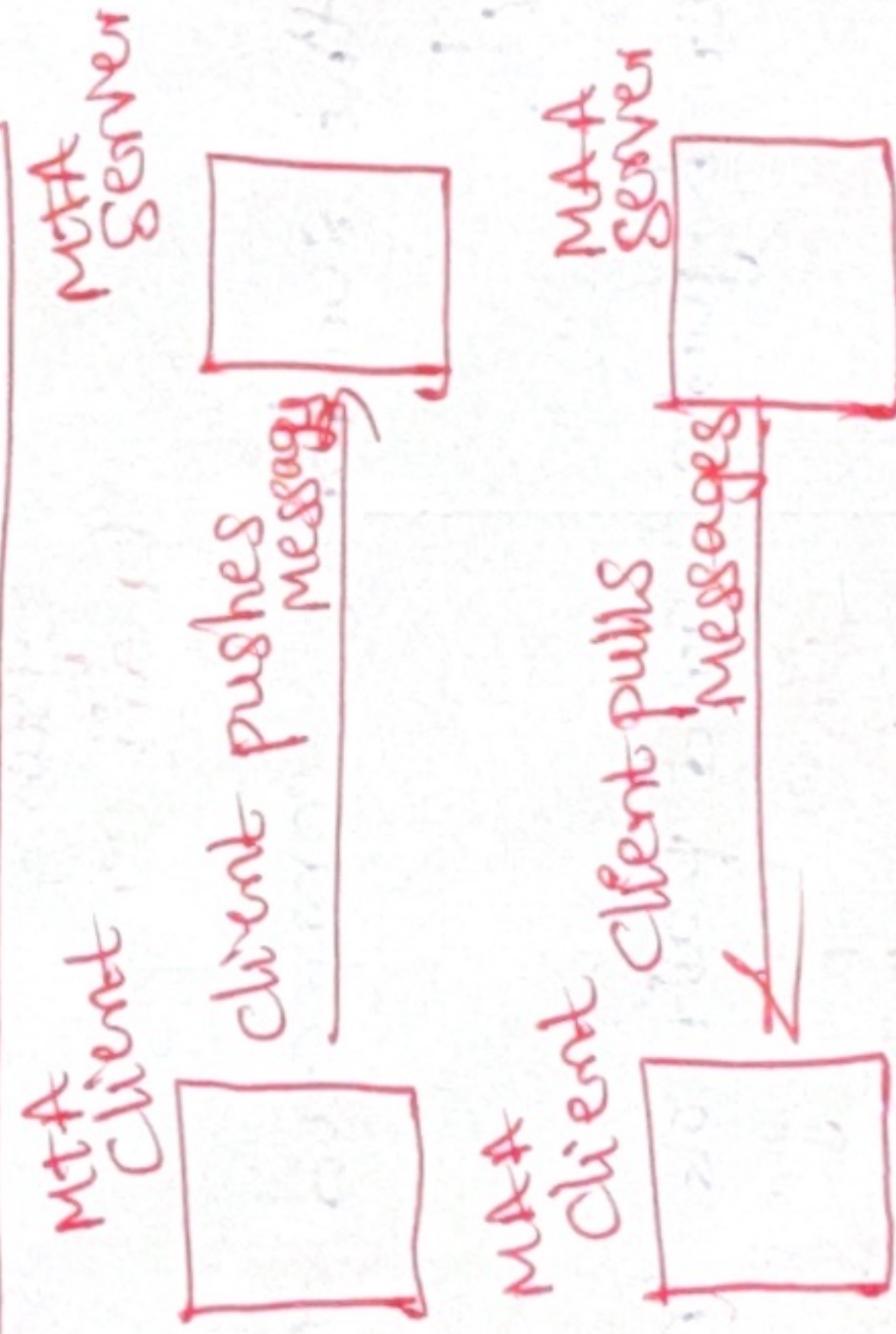
Scenario-4

→ In the fourth scenario, both sender and receiver are connected to the 'mail server' via LAN or WAN.

→ So to retrieve the messages from server we need "message Access Agents".



Push and pull operation in E-mail:



Email Contains :-

Each E-mail contains the following fields:

→ A number field

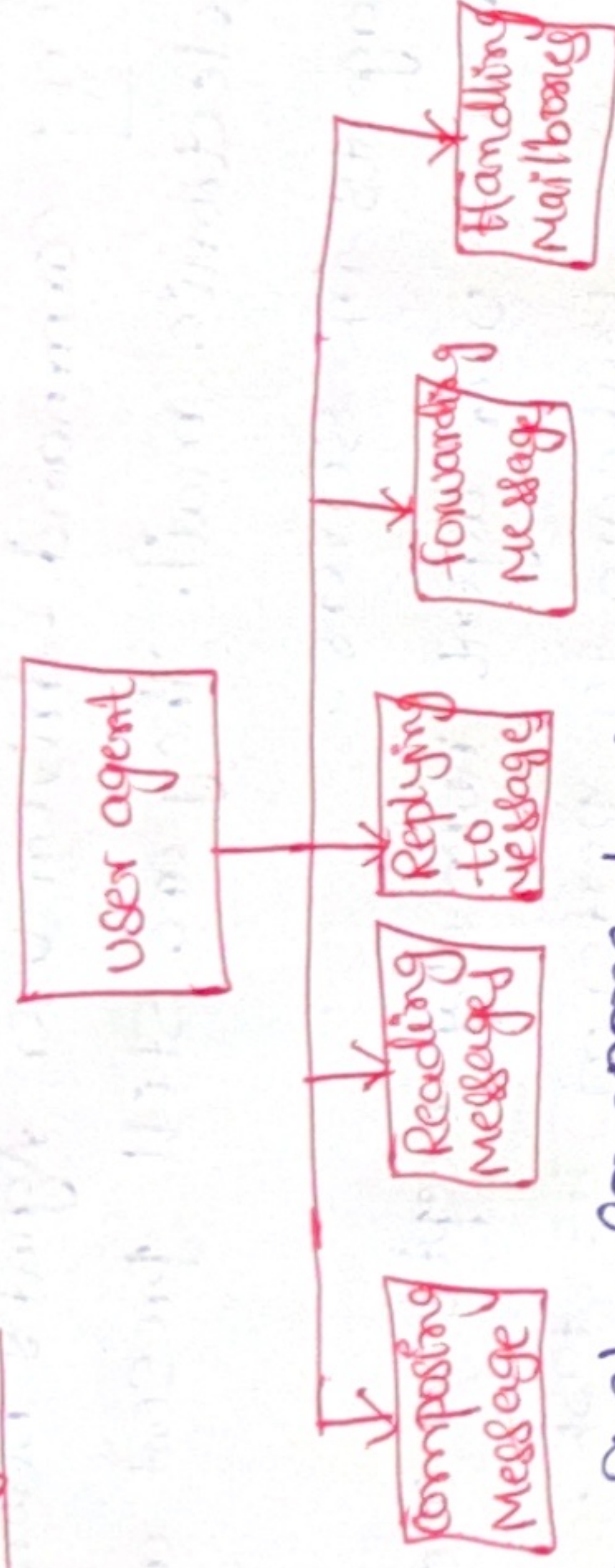
→ A 'Flag' field the shows the status of mail such as new, already read out not replied etc.

→ The size of message → The sender details

→ The optional subject field.

E-MAIL Architecture:

1) User Agent:



→ It is the first component in e-mail system architecture.

→ It provides services to the user for sending & receiving a message easier.

User Agent services:

Composing Messages: A 'User Agent' helps the user to

compose the e-mail message to be sent out.

Reading Messages: The 'User Agent' is to read the

incoming messages by checking the 'inbox'.

Replying to Messages: After reading a message, a user can use

the 'user agent' to reply to message to original sender or reply to all recipients.

Forwarding Messages: Forwarding is defined as sending the message to 'third party'.

Handling Mailboxes: A 'User Agent' creates two mail boxes:

"Inbox" and "Outbox". The 'Inbox' keeps all received messages until they are deleted by the user and the 'Outbox' keeps all sent-mails until user deletes them.

→ User Agent Types:

There are two types of User Agents: Command-driven and GUI-based.

Command-Driven: Command-driven user agents belong to the early days of electronic mail. They are still present as the underlying user agents in servers.

→ A Command-driven user agent normally accepts a one-character command from the keyboard to perform its task.
for example: A User can type the character 'r', at the command prompt, to reply to the sender of the message, or type the character 'R' to reply to the sender and all recipients.

Some examples of Command-driven user agents are mail, Pine, and elm.

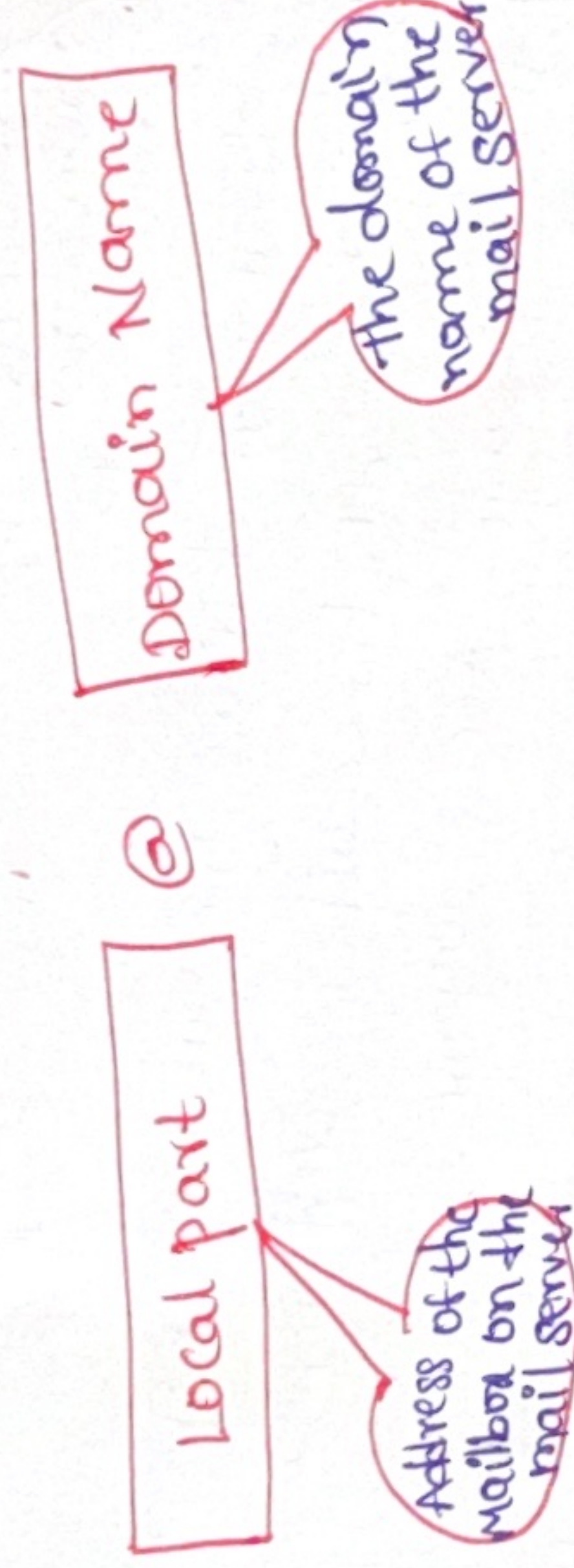
GUI Based: Modern User Agents are GUI based. They contain graphical - user - interface (GUI) components that allow the user to interact with the software by using both the keyboard and the mouse. They have graphical components such as icons, menu bars, and windows that make the services easy to access.

Some examples of GUI-based user agents are Eudora, Microsoft's Outlook, and Netscape.

Format of an E-Mail:



Electronic Mail



E-mail Address

→ Envelope: the envelope usually contains the sender and the receiver addresses.

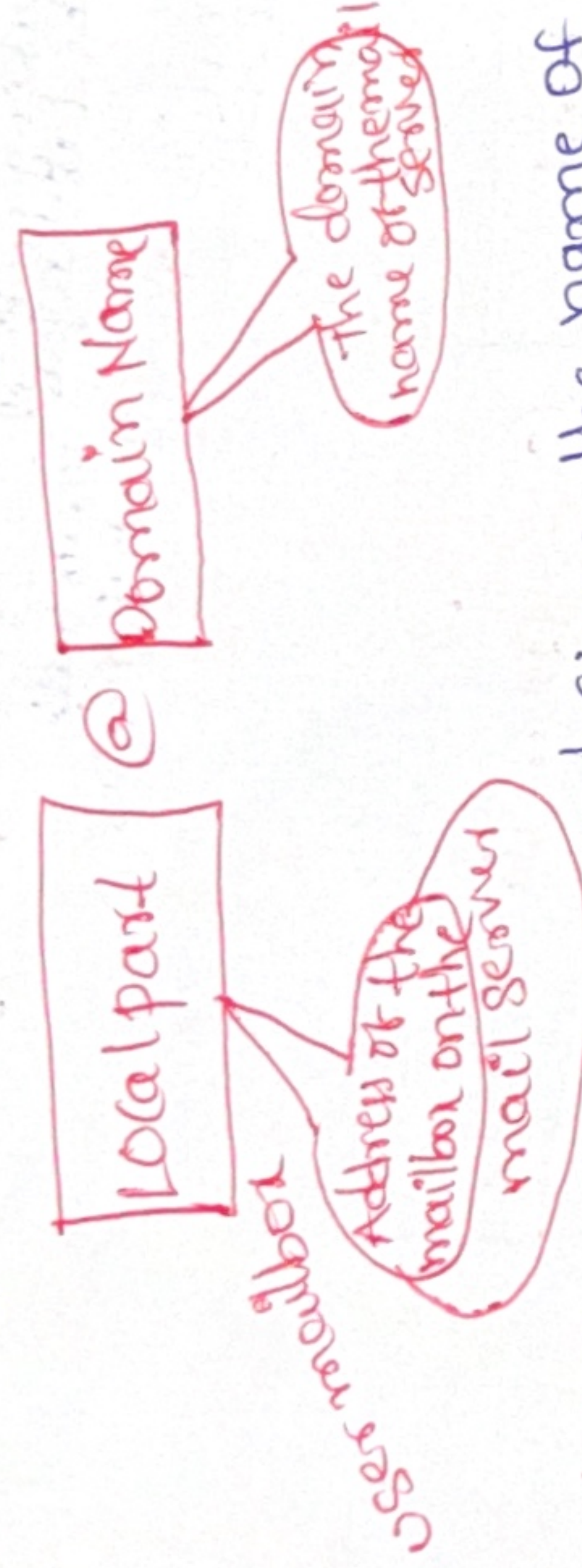
Message: The message contains the header and the body. The header of the message defines the sender, the receiver, the subject of the message, and some other information. The body of the message contains the actual information to be read by the recipient.

Receiving Mail: The user agent is triggered by the user (or a timer)

if a user has mail, the UA (User Agent) informs the user with a notice. If the user is ready to read the mail, a list is displayed in which each line contains a summary of the information about

Particular message in the mail box. The summary usually includes the sender mail address, the subject, and the time the mail was sent or received. The user can select any of the messages and display its contents on the screen.

Addresses: To deliver mail, a mail handling system must use an addressing system with unique addresses. In the Internet, the address consists of two parts: a local part and a domain name, separated by an @ sign.



Local part: The local part defines the name of a special file, called the user mailbox, where all the mail received for a user is stored for retrieval by the message access agent.

Domain Name: The second part of the address is the domain name. An organization usually selects one or more hosts to receive and send e-mail, the hosts are sometimes called mail servers or mail exchangers. The domain name assigned to each mail exchanger either comes from the DNS database or is a logical name.

→ Message Transfer Agent: SMTP

→ The actual mail transfer is done through message transfer agents to send mail, a system must have the client MTA, and to receive mail, a system must have a server MTA. The formal protocol that defines the MTA client and server in the Internet called the Simple Mail Transfer Protocol (SMTP).

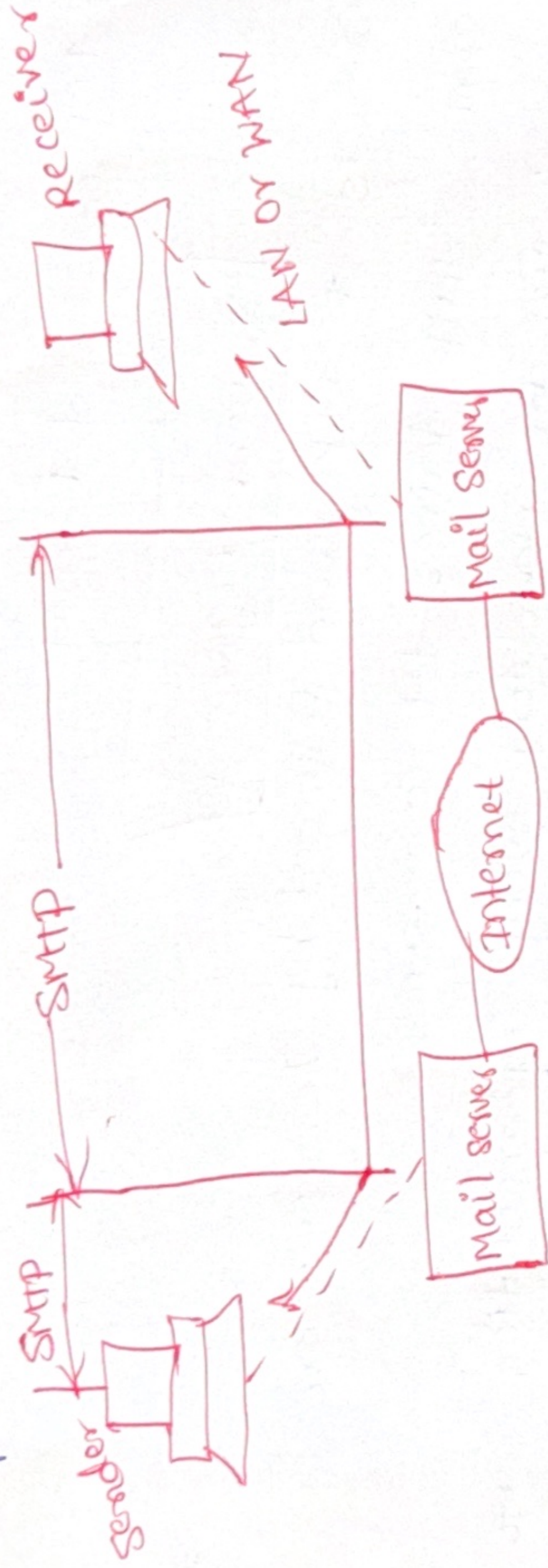


Fig: SMTP Range

→ SMTP is used two times between the sender and the receiver. Sender's mail server and between the two mail servers. As we will see shortly, another protocol is needed between the mail server and the receiver.

→ SMTP simply defines how commands and responses must be sent back and forth. Each network is free to choose a software package for implementation.

Commands and Responses

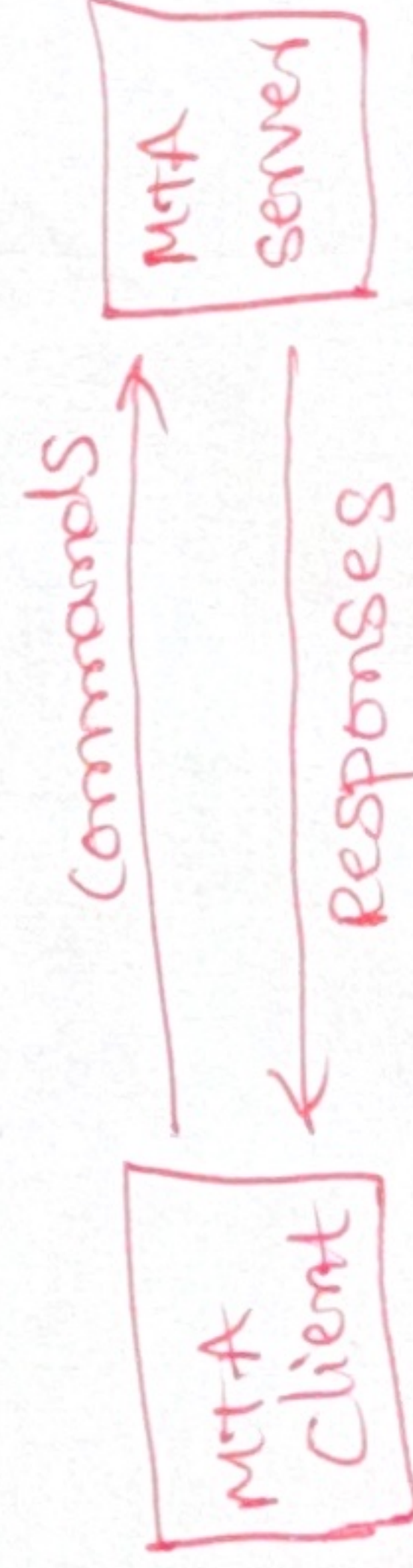


Fig: Commands and Responses

SMTP uses commands and responses to transfer messages between an MTA Client and an MTA Server.

→ Each command or reply is terminated by two-character end-of-line token.

Commands: commands are sent from the client to the server. The format of a command is shown in figure below.

Fig: Command Format

keyword: Arguments

→ It consists of a keyword followed by zero or more arguments. SMTP defines 14 commands.

→ The first five are mandatory, every implementation must support these five commands. The next three are often used and highly recommended. The last six are seldom used.

Commands

keyword	Arguments
HELO	Sender's host name
MAIL FROM	Sender of the message
RCPT TO	Intended recipient of the message
DATA	Body of the mail
QUIT	terminate message
RSET	Reset Connection
VERB	Name of recipient to be verified
NOOP	
TURN	
EXPN	Mailing list to be expanded
HELP	Command Name
SEND FROM	Intended recipient of the message
SMOL FROM	Intended recipient of the message
SMAL FROM	Intended recipient of the message

Responses: Responses are sent from the server to the Client.
 A response is a three-digit code that may be followed by additional textual information.

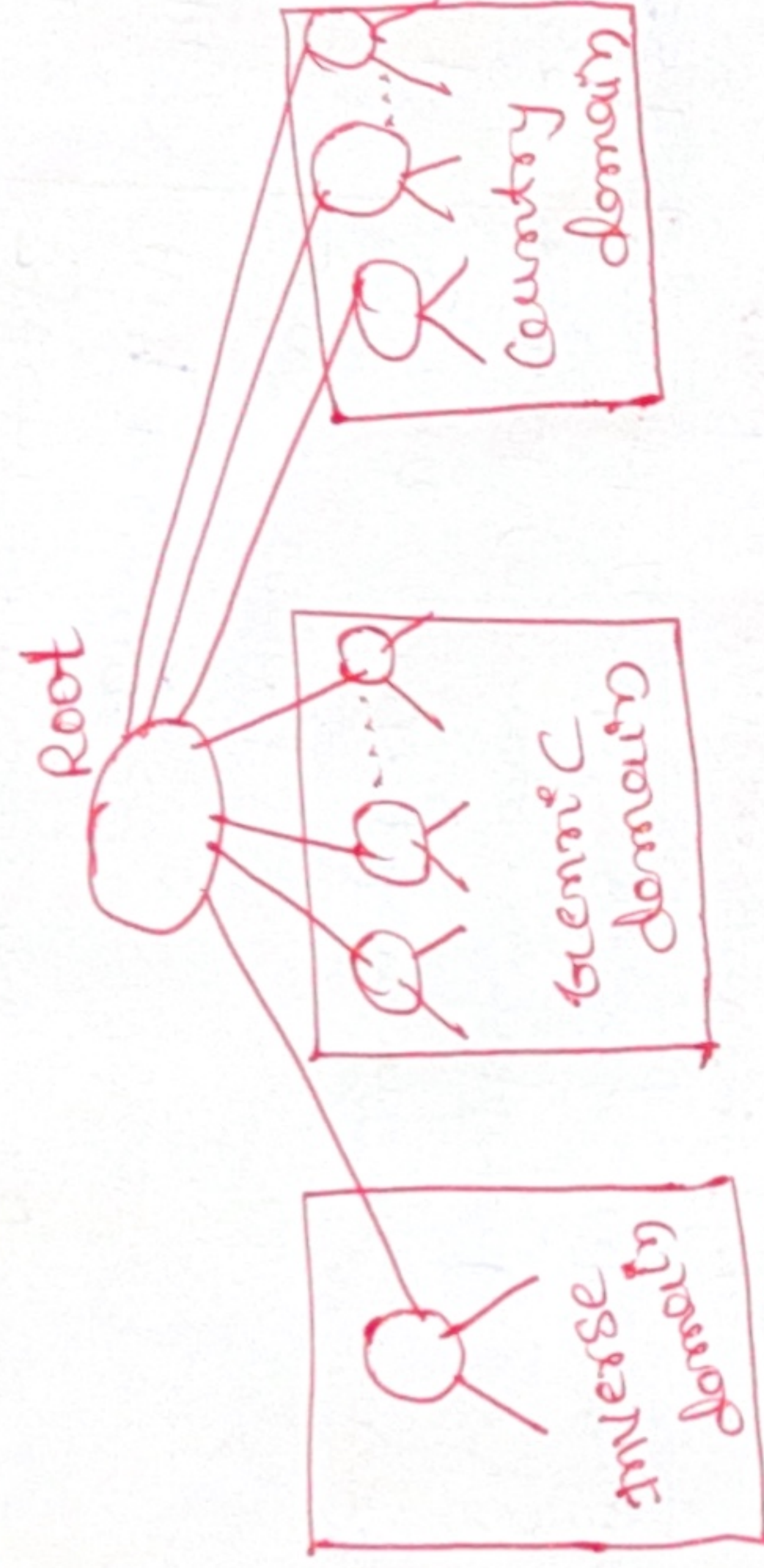
Responses

Code	Description
Positive Completion Reply	
211	System status or help reply
214	Help message
220	Service ready
221	Service closing transmission channel
250	Request command completed
251	User not local; the message will be forwarded
Positive Intermediate Reply	
354	Start mail input
Transient Negative Completion Reply	
421	Service not available
450	Mailbox not available
451	Command aborted: local error
452	Command aborted: insufficient storage
Permanent Negative Completion Reply	
500	Syntax error; unrecognized command
501	Syntax error in parameters or arguments
502	Command not implemented
503	Bad sequence of commands
504	Command temporarily not implemented
550	Command is not executed; mailbox unavailable
551	User not local
552	Requested action aborted; exceeded storage allocation
553	Requested action not taken; mailbox name not allowed
554	Transaction failed.

→ DNS in the Internet :

- DNS is a protocol that can be used in different platforms.
- In the Internet, the domain name space (tree) is divided into three different sections: generic domains, country domains, and the inverse domain.

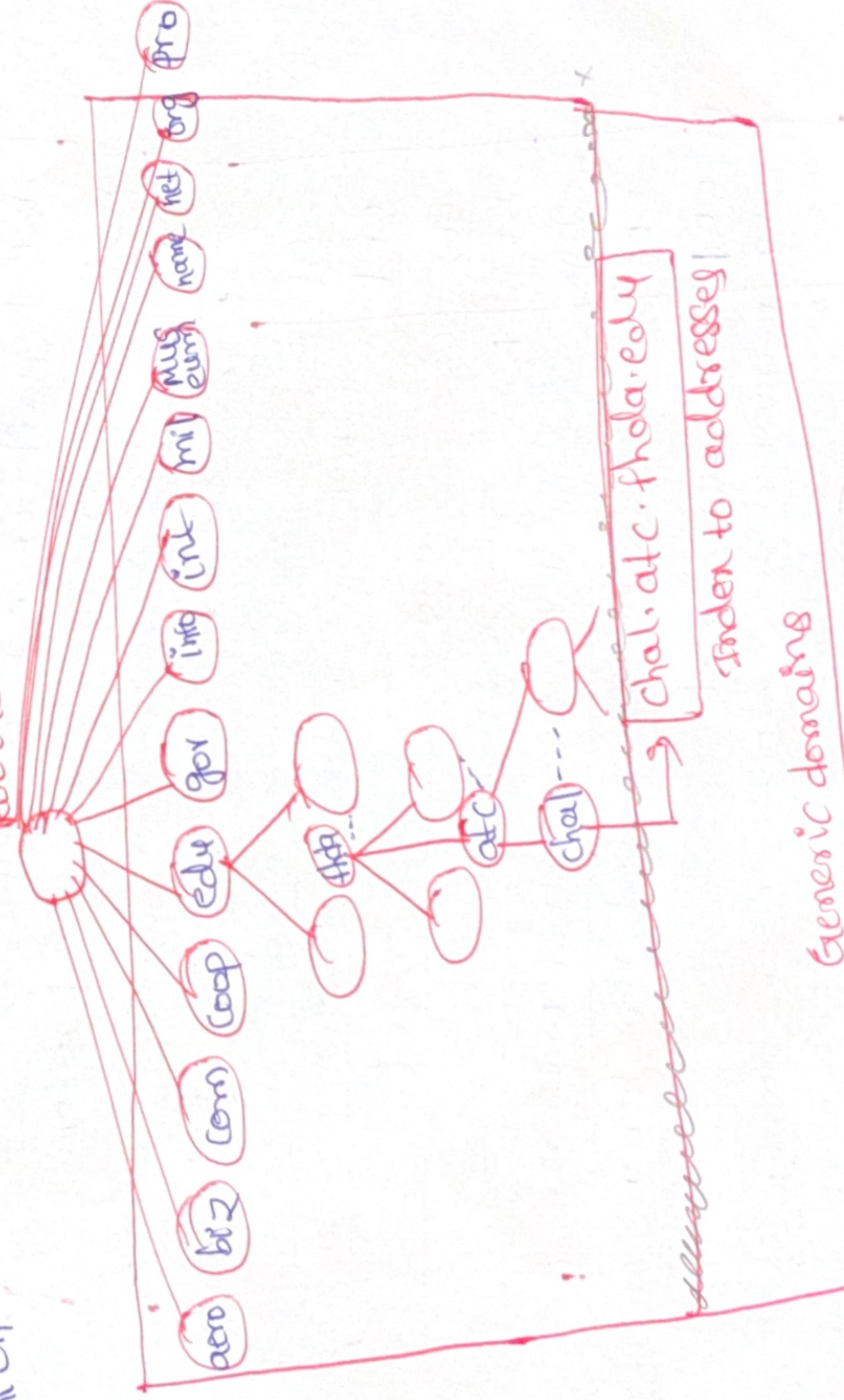
DNS used in the Internet



Generic Domains :

- The generic domains define registered hosts according to their generic behaviour. Each node in the tree defines a domain, which is an index to the domain's name space database.

Root level



Generic Domains: Looking at the tree, we see that the first level in the generic domains section allows 14 possible labels.

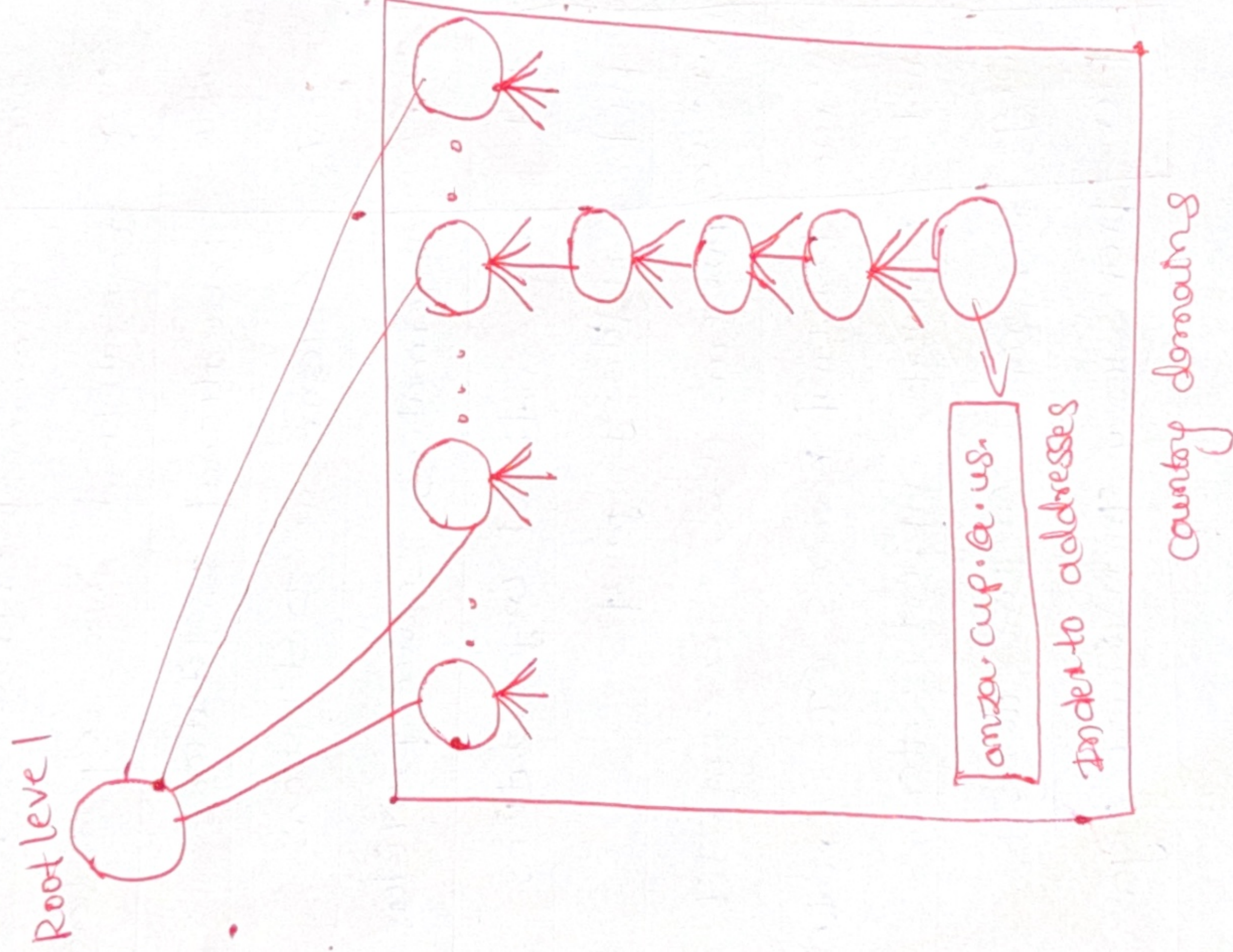
Generic domain labels

Label	Description
aero	Airlines and aerospace companies
biz	Businesses or firms
com	Commercial organizations
coop	Cooperative business organizations
edu	Educational institutions
gov	Government institutions
info	Information service providers
int	International organizations
mil	Military groups
museum	Museums and other nonprofit organizations
name	Personal names (Individuals)
net	Network support centers
org	Nonprofit organizations
pro	Professional individual organizations

Table: Generic domain labels

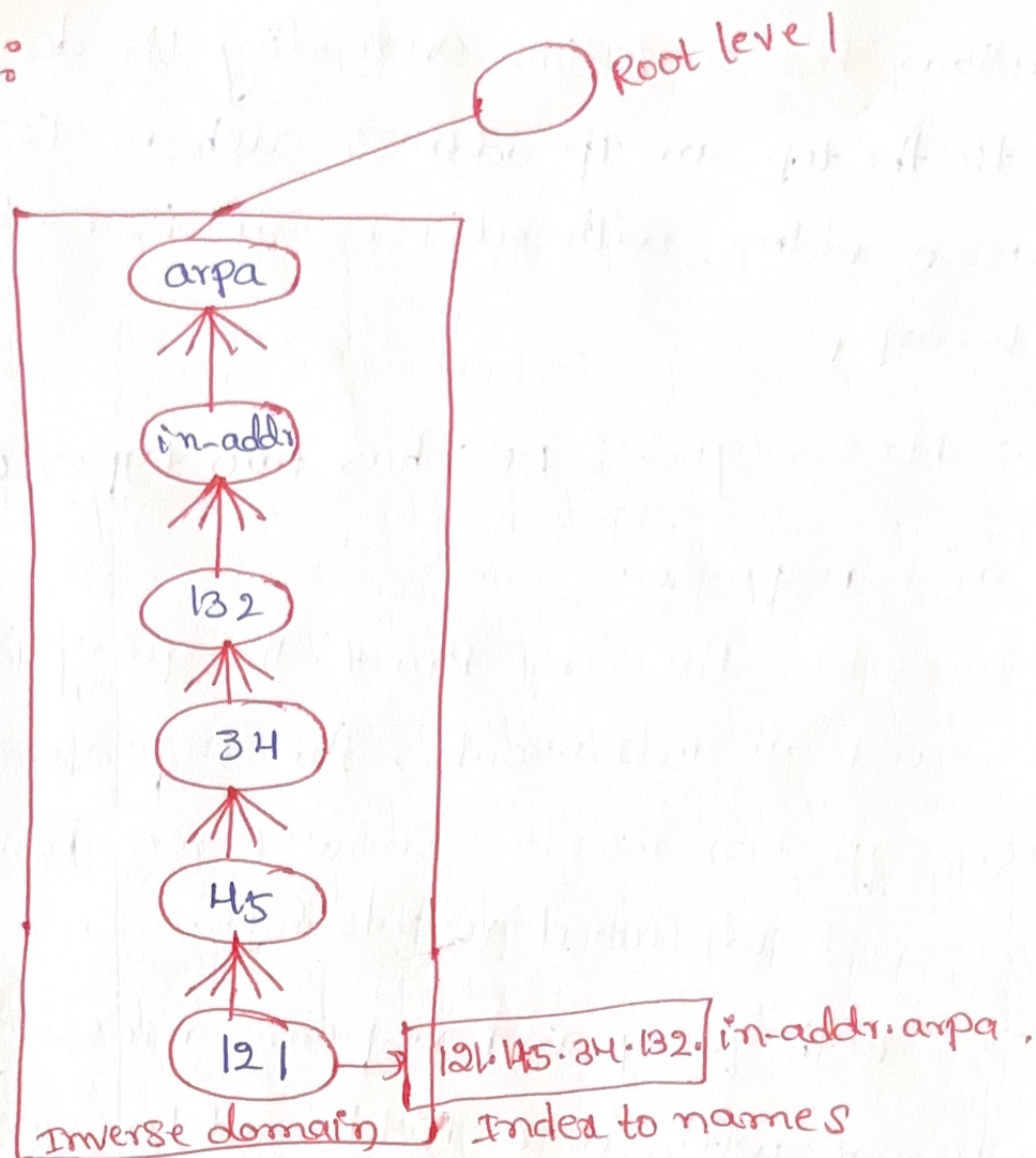
Country Domains:

The country domains section uses two-character country abbreviations (e.g.: us for United States). Second labels can be organizational, or they can be more specific, national designations. The United States, for example, uses State abbreviations as a subdivision of us (e.g., ca.us.).



Inverse Domain :

(13)



- Inverse Domain is used to map an address to a name, this may happen, for example, when a server has received a request from a client to do a task. Although the server has a file that contains a list of authorized clients, only the IP address of the client is listed.
- The server asks its resolver to send a query to the DNS server to map an address to a name to determine if the client is on the authorized list.
- This type of query is called an inverse or pointer (PTR) query. To handle a pointer query, the inverse domain is added to the domain name space with the first level node called arpa. The second-level is also one single node named in-addr (for inverse address). The rest of the domain defines IP addresses.

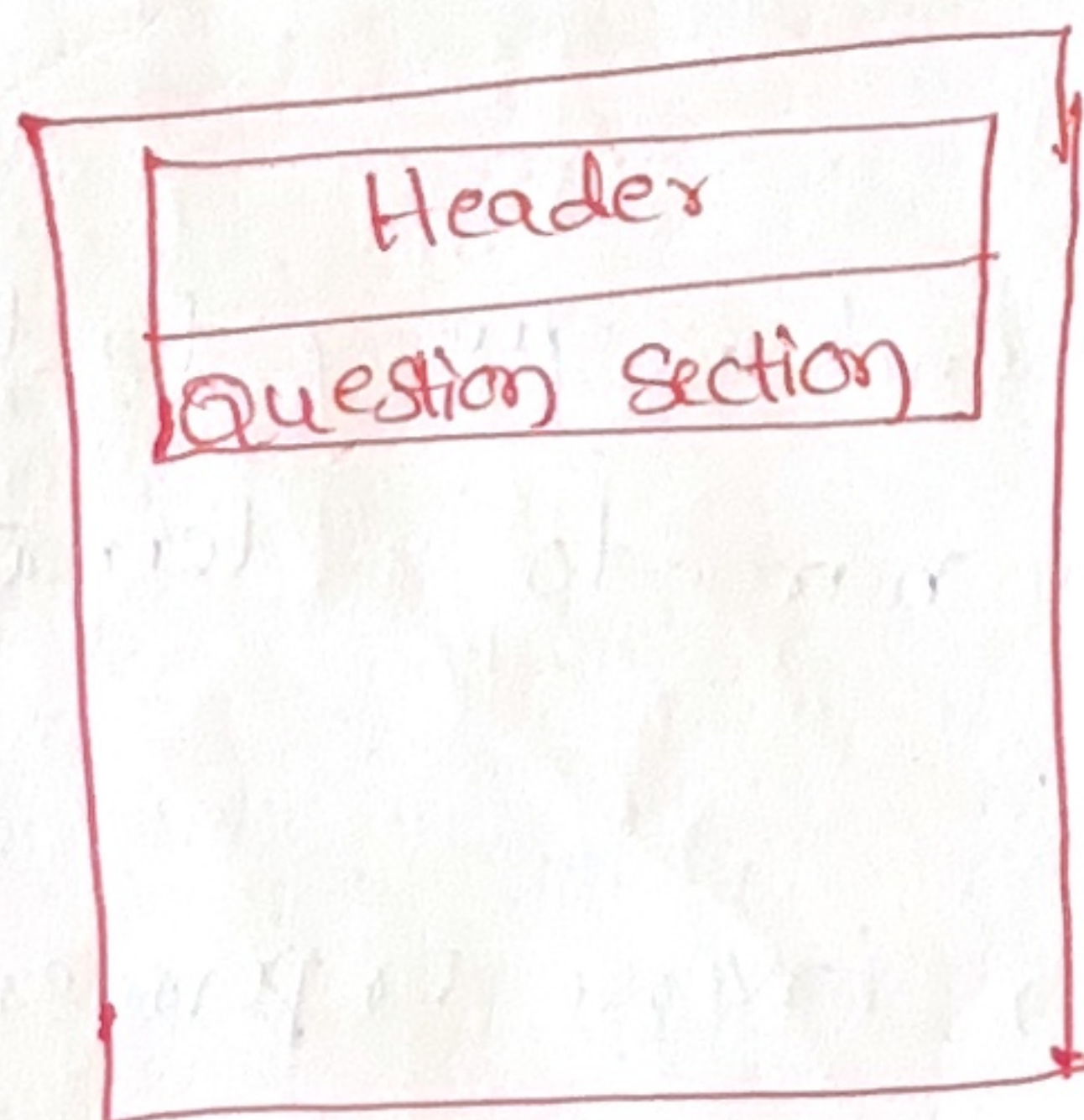
→ to follow the convention of reading the domain labels from the bottom to the top, an IP address such as 132.34.45.121 (a class B address with netid 132.34) is read as 121.45.34.132, in .addr.arpa.

→ DNS Messages: DNS has two types of messages: query and response.

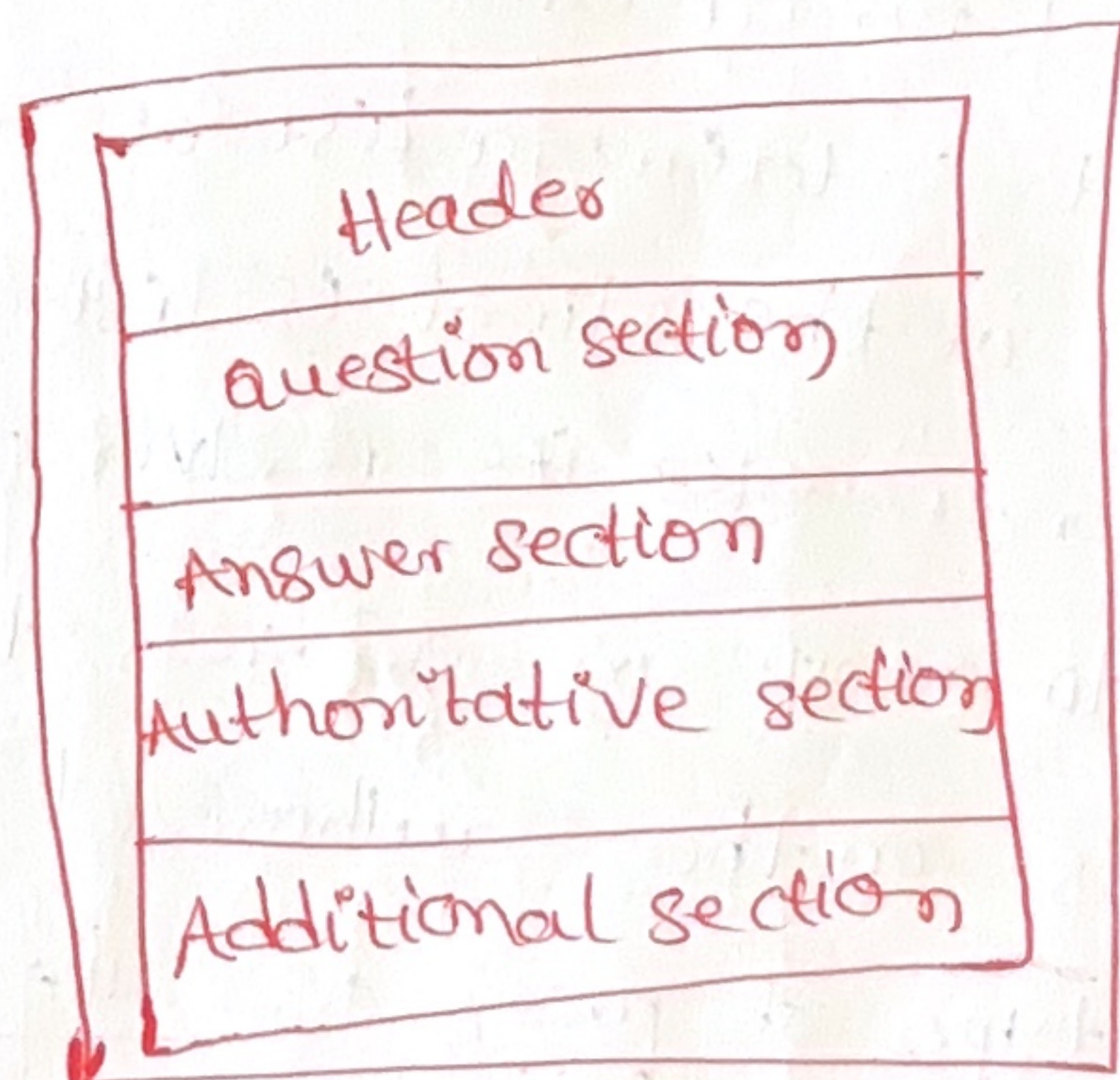
→ Both types have the same format. The query message consists of a header and question records, the response message consists of a header, question records, answer records, authoritative records, and additional records fig 1.

Header: Both query and response messages have the same header format with some fields set to zero for the query messages. The header is 12 bytes, and its format is shown in figure 2.

Fig: 1



a. Query



b. Response

Fig 2: Header Format

(14)

Identification	Flags
Number of question records	Number of answer records (all 0's in query message)
Number of authoritative records (all 0's in query message)	Number of additional records (all 0's in query message)

→ The identification subfield is used by the client to match the response with the query. The client uses a different identification number each time it sends a query, the server duplicates this number in the corresponding response, the flags subfield is a collection of subfield that define the type of the message, the type of answer requested, the type of desired resolution and so on.

→ The number of question records subfield contains the number of queries in the question section of the message. The number of answer records subfield contains the number of answer records in the answer section of the response message.

→ Its value is zero in the query message. The number of authoritative records subfield contains the number of authoritative records in the authoritative section of a response message. Its value is zero in the query message. Finally, the number of additional records subfield contains the number of additional records in the additional section of a response message. Its value is zero in the query message.

Types of Records : There are two types of Records are

used in DNS .

→ 1) Question Record 2) Resource Record

→ The Question records are used in the question section of the query and response messages. The resource records are used in the answer, authoritative, and additional information sections of the response message.

Question Record :

A Question Record is used by the client to get information from a server. This contains the domain name.

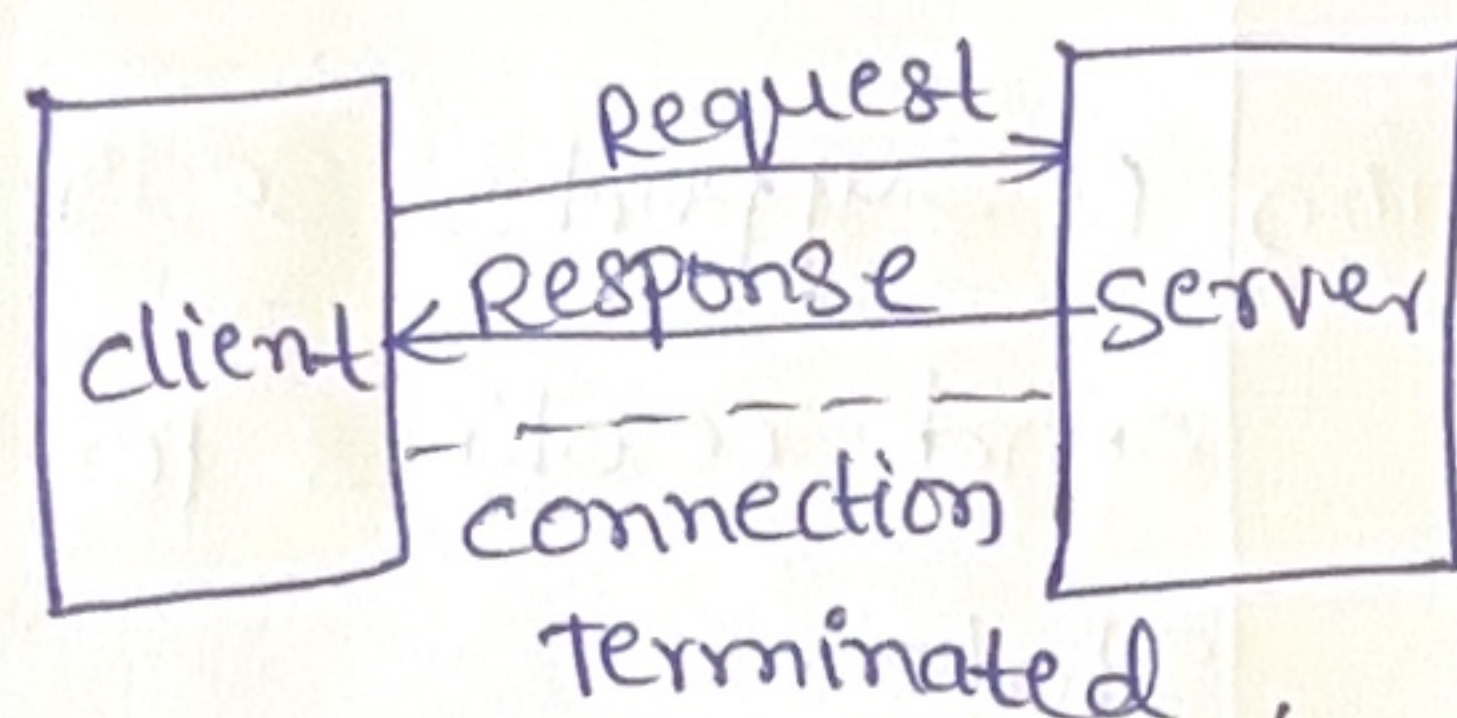
Resource Record :

Each domain name (each node on the tree) is associated with a record called the resource record. The server database consists of resource records. Resource Records are also what is returned by the server to the client.

(5)

HTTP: HTTP is a client-server protocol. It is IP based communication protocol that is used to deliver data from server to client or vice-versa. Any type of content can be exchanged as long as the server and client are compatible with it.

Http Connection



HTTP: Hyper text transfer protocol.

Difference between SMTP & HTTP

SMTP

- 1) SMTP: Simple Mail transfer protocol
- 2) SMTP is used for mail services
- 3) It uses port 25.
- 4) It is primarily a push protocol.
- 5) It imposes a 7-bit ASCII restriction on the content to be transferred.
- 6) SMTP transfers emails via mail servers.

HTTP

- 1) HTTP: Hyper text transfer protocol.
- 2) HTTP is mainly used for data and file transfer.
- 3) It uses port 80.
- 4) It is primarily a pull protocol.
- 5) It does not impose a 7-bit ASCII restriction. Can transfer multimedia, hyperlinks, etc.
- 6) HTTP transfers files between the webserver and the web client.

- | | |
|--|---|
| <p>1) SMTP is a persistent type of TCP connection.</p> <p>8) Uses base 64 encoding for authentication.</p> <p>9) Does not support session management or cookies.</p> <p>10) Has a smaller message size limit compared to HTTP.</p> <p>11) Requires authentication for sending emails.</p> <p>12) Supports both plain text and encrypted communication (SMTPS).</p> | <p>1) It can use both persistent and non-persistent.</p> <p>8) Uses different methods of authentication such as basic, digest, and OAuth.</p> <p>9) Supports session management and cookies to maintain state.</p> <p>10) Has a larger message size limit compared to SMTP.</p> <p>11) Does not require authentication for browsing web pages.</p> <p>12) Supports both plain text and encrypted communication (HTTPS).</p> |
|--|---|