04/08/25

Computer Networks

→ components :-
→ sender
Receiver
Medium
Data

Books - C.N. - Taenbaum
C.N. → U. Black
complete Net → William Stanly.

Transmission Modes :-

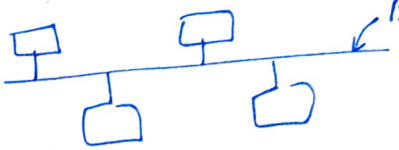
(i) Simplex :- Transmission only in one direction, Ex :- Television
(ii) Half Duplex :- Transmission in both direc" but only one at a time, Ex :- Walki, Toki
(iii) Full Duplex :- Data transmission in both direc" simultaneously

Types of Network Connections :-

(i) Point to Point (Peer to Peer)

→ The nodes are connected through a dedicated line

(ii) Multipoint

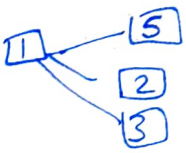


→ A link shared among 2 or more devices

Types of Transmission :-

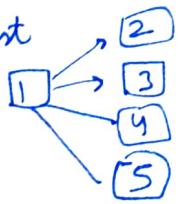
(i) Unicast 1 → 2 → one to one

(ii) Multicast



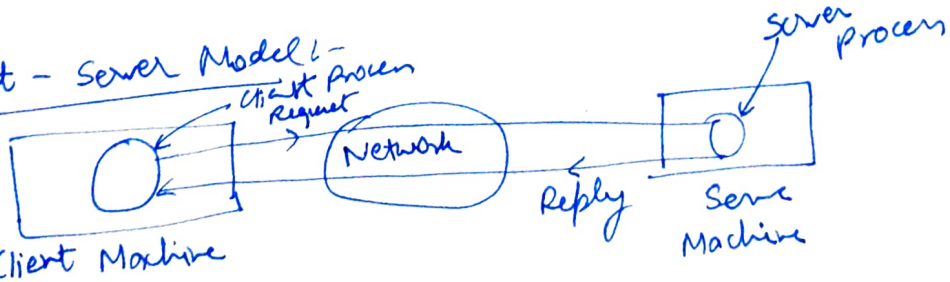
(Not 4) → one to many (but limited)

(iii) Broadcast



→ one to many (everyone)

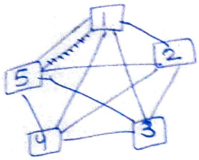
Client - Server Model :-



Robust Property :- States that if some node on the network does not work than it should not effect the entire communication.

Network Topology :-

① Mesh Topology



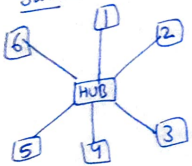
Pros
- Point to Point
- Robust in nature
- Security & Privately
- Direct transmission

Cons
- Costly
- High-Maintaence
- Fails in large network
- Recovery time

→ No. of I/O ports at each node ⇒ $(n-1)$ req^d to connect n devices

→ no. of links req^d → $\frac{n(n-1)}{2}$

② Star :-

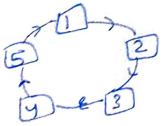


Pros
- Point-to-Point
- Robust

Cons
- Hub can fail whole network down

No. of I/O ports at each node → 1
No. of links → N

③ Ring Topology :- unidirectional

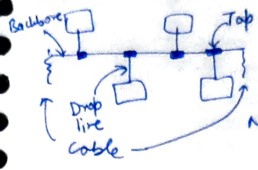


- Point to Point
- Not Robust

No. of I/O ports at each node ⇒ 2

No. of links → N

④ Bus Topology

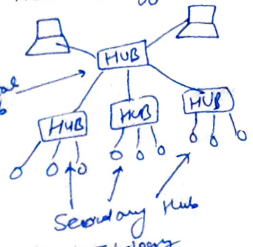


- Multipoint
- Non-Robust

No. of ports → 1
No. of links → N+1

⑤ Hybrid Topology

⑥ Tree Topology



→ Hierarchal Topology
→ Point-to-Point
→ Robust
Ex → Delivery

Categories of Network :-

→ Network

	PAN	LAN	MAN	WAN	Wireless	Home Network	Enterprise Network
	Personal Area Network	Local Area Network	Metropolitan Area Network	Wide Area Network			
Ex:	Hotspot	Building	City	India (.in)			
~~total~~ 1 sq m.		10m, 100m 1Km	10Km	100 Km 1000 Km			
		Room, Building	City	country / continent			

Connection Oriented & connection Less Service :-

→ In connection oriented service in order to exchange info. a connection is established between the sender & the reciever

Steps :- (i) Establish the connection
(ii) Data Transfer
(iii) Release the connection

→ **Connection Less Service :-** No need to establish connection in order to exchange info.

Ex - Postal Service

→ Each service can be categorized by the quality of service (QoS)

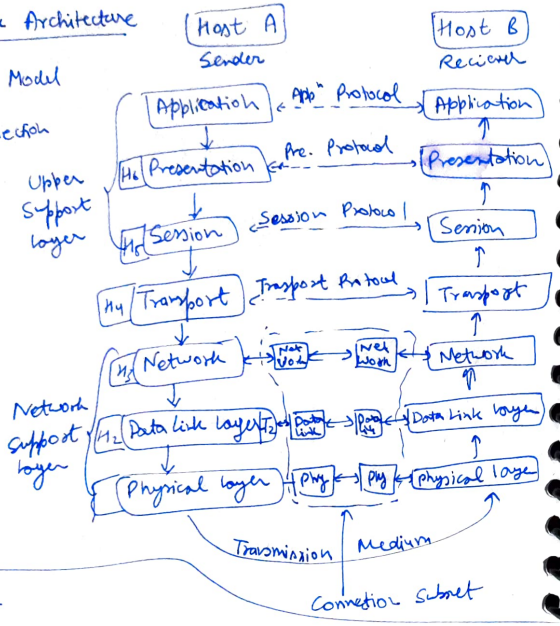
→ Reliable Service (Connection Oriented) or Unreliable Service (connection Less)

→ Cause we dont know the delivery of data

Ex :- App. like digital voice traffic emails

05/08/25 Network Architecture

(1) OSI Reference Model
↑
OPEN System Interconnection
(Given by ISO)



Application Layer :-
↳ Top most layer
↳ Provides services like manipulation of info, re-transmission of files & emails, etc.
↳ It acts as an interface b/w the computer system and the user
↳ Has various protocol, most common - HTTP

Presentation Layer :-
↳ It is concerned with the system & semantics of the data being transmitted.

Encryption/Decryption
Presentation Layer
Compression/Decompression
Translation
converts the data into a common format in order to a common range

Session Layer :-
Session Layer
Session Management
Synchronization
Dialog control

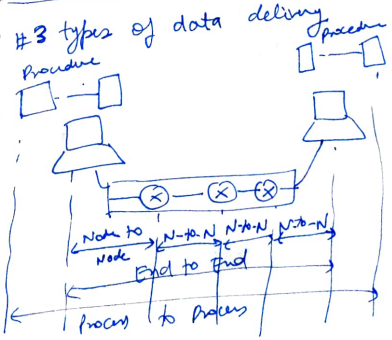
→ The job of the session layer is to establish, maintain & synchronize the interaction b/w communicating system

→ Session layer allows users on diff. machines to establish session b/w them

→ Dialog control keeps track of who's turn it is to initiate or transmit the data and how the initiation is done

→ Session layer allows addition of checkpoints i.e. synchronization points into a stream of data being transmitted. In case of system failure, the data is already saved.

#3 types of data delivery



Type of Addressing

(1) Physical Address (MAC address) — 6 bytes/48 bits
(2) Logical Address (IP)
(3) Port Address
- 16 bits
- IPv4 — 32 bits
- IPv6 — 128 bits

Transport Layer

Transport Layer
- segmentation/reassembly
- process to process comm
- Port Address
  - ↳ Succesful delivery of msg.
- Error Control
- Flow Control
- Congestion Control

→ Data Units at Transport layer are called as TPDU (Transport Protocol Data Units)

→ Flow control is done to handle speed / flow mismatch between user & sender

→ It is req^d when port info generation by sender but now recieving

→ Congestion control is traffic control one transmission medium

→ The header in transport layer Hy includes a service point address to deliver a specific protocol from source to destination

Network Layer

responsible for delivery of data from the original source to destn network

Network Layer → Routing (→ Path Finding)
→ End to End
→ Packets
↳ Logical Address
↳ Congestion Order

Data Link Layer :-

Data Link Layer →
- Framing
- Physical Address
- Protocols
- Logical address
- Node-to-Node comms.
- Error Control
- Access Control
- Flow Control

H_2 → Source & Dest Addr

T_2 → Error control related bits

→ Error Control → A trailer added to the data link layer at the end of the frame to achieve error control)

→ It also uses a mechanism to prevent duplication of frames.

→ Access Control → Data link layer protocol determines which of the has control over the shared link at any given time. A special sublayer of data link layer media access control (MAC) sublayer deals with this problem.

Physical layer :-

Physical layer →
- Actual trasmission of info
- Synchronization
- Conversion of frms to symbol.

like coding

Synchronization → b/w sen / recieve.

11/05/25

TCP/IP Reference Model

Layers →
- App^n
- Transport
- Network → or Internet
- Host-to-Net work

→ Network/Internet layer :- The func^n of layer is to allow hosts to inject packets into the network and then have them travel independently to the destination. However the order of recieving the packets can be diff from the sequence in which they were sent.

↳ It is the job of the higher layers to rearrange them if in order delivery is req.

↳ Some of the protocols used :-
→ IP → Internet protocol.
→ ICMP → Internet control message protocol.
→ ARP → Address resolution protocol.

Transport Layer :

↳ There are two main protocols :-

(i) TCP → Transmission control Protocol.
(ii) UDP → User Datagram Protocol.

(i) TCP :-
- reliable connection-oriented protocol
- performs sequencing & segmentation of data.
- Fragments byte stream into discrete messages and
- passes each one on to internet layer.
- At destination reassembly

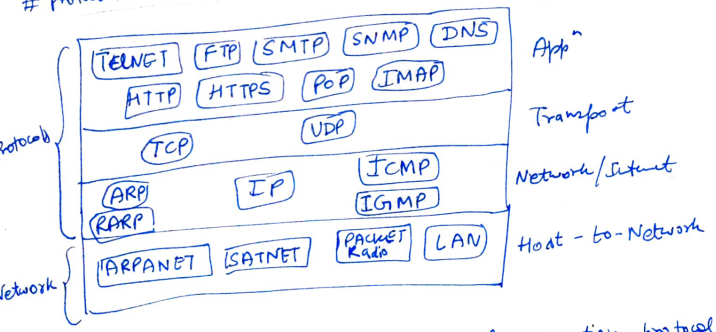
(ii) UDP :-

↳ unreliable → connectionless protocol.
↳ Only concerned with delivery of message and not saying it delivered or not.
eg: livestream, broadcasting, etc.

Application layer :-

↳ It is responsible for session management and controls user interface specifications.
↳ It contains the func" of three top most layer of OSI model.
↳ It consists of higher level protocols such as
~~TELNET~~ TELNET (Virtual Terminal connection),
HTTP, FPP, etc.

Protocols and Networks in TCP/IP Model :-



(i) TELNET :- It is a virtual terminal connection protocal used to login to remote computer on internet

(ii) FTP :- It is used to trasfer data/files from one host to another

↳ Establishes two diff. connections :-
(i) For data transfer → through Port 20
(ii) For control info → " Port 21

(iii) HTTP :- It is a communication protocol it defines a mech for communication b/w the browser and the web server.

(iv) SMTP (Simple mail transfer protocol) ; Std. protocol to trasfer emails efficiently and reliably over the internet

↳ connection-oriented protocol.
↳ handles exchange of messages b/w email servers over the TCP/IP network.
↳ In case the msg is not delivered, an error report is sent to the sender which makes SMTP a reliable protocol.

(v) POP :- (Post Office Protocol)

↳ Latest version → POP3
↳ Generally used to support a single client.
↳ ~~Host~~ In-order to access the message it is necessary to download it.
↳ There is no search facility.

(vi) IMAP (Internet Message Access Protocol) :

↳ Latest Versia - IMAP4
↳ Allows client to manipulate email messages on the server without downloading them on the local computer.
↳ Also enables us to delete email without reading them.
↳ Can search email.

(vii) Simple Network Managent Protocol (SNMP) :-

↳ used to collect info from and configure network devices such as servers, hubs, switches etc.

(viii) DNS :- (Domain Name System) :-

↳ provides an name or URL to logical address

(ix) TCP

(x) UDP :-

↳ Transmits data in form of UDP datagram.

Source port	Destination Port
Length	Checksum
Data	

(xi) ~~ICMP~~ IP :-

↳ connection-less & unreliable protocol

↳ Host-to-Host network layer delivery protocol for the internet.

↳ Data is transmitted in form of IP datagram.

(xii) ARP :- (Add. Resol. Protocol)

↳ used for association or mapping IP address-to-mac address.

(xiii) RARP :- (Reverse ARP)

↳ Mac address to IP address.

(xiv) ICMP :- (Internet Control Message Protocol) :-

↳ Network Management & administration.

(xv) IGMP :- (Internet Group Management Protocol)

↳ Manages group membership like ~~email~~/domain → helps multicast routers to create and update list of members related each router interface

18/08/25

Physical Layer

→ conversation of frame to signal

→ Actual ~~frame~~ transmission

→ Synchronization

Signal → Analog, Digital

Analog ↳ continuous signal which varies smoothly & continuously with time.

↳ ∞ ~~trans~~ no. of voltage levels

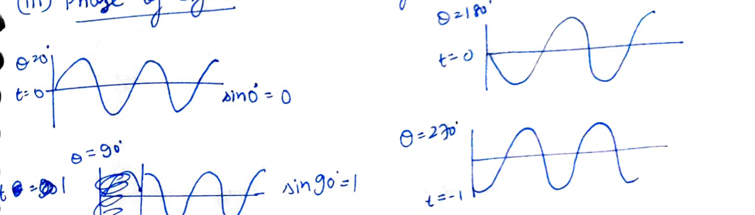
→ Properties of Analog Signal :-

(i) Amplitude → value of voltage or level of signal at time t.

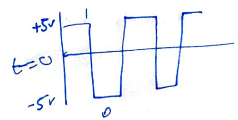
(ii) frequency :- No. of cycles completed in one second

$(f = \frac{1}{t}\ Hz)$

(iii) Phase of Signal :- Position of wave formed at time t=0.



Digital Signals → Discrete in nature



~~Signals~~ # Periodic & Aperiodic Signals

Note :- In data communication, we prefer periodic analog signal during analog transmission of data and Aperiodic digital Signal during digital transmission of data.

Analog & Digital data transmission :-

→ Analog data transmission :- It is means of transmitting analog signals regardless of content it carries, The data may be analog data (voice) or digital data (binary) in either case the signal will become weaker after a certain distance (attenuation)

To overcome this, amplifiers are used to boost the signal

→ Drawback → noise during transmission, in digital → errors.

Analog data (voice) ⟷ Telephone ⟷ Analog Signal

Digital data (binary voltage pulses) ⟷ Modem ⟷ Analog Signal.

Digital data transmission :-

→ In contrast to analog data tran. digital data tran. is concerned with the content of the signal.

→ A digital signal can be transmitted only to a certain limited distance before attenuation, noise, etc. distorts the integrity of the data.

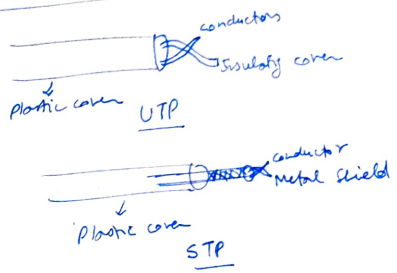
→ To achieve greater distance - repeaters are used

Analog data ⟷ Codec ⟷ Digital Signal

Digital data ⟷ Digital Transceiver ⟷ Digital Signal.

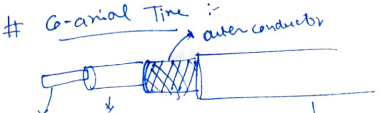
Physical Layer transmission media :-

- Guided or wired transmission Media (Point to Point)
  - Twisted pair
    - copper wires
    - can run upto several kilometers without amplification
    - Unshielded Twisted Pair
      - Type 3
      - Type 5 → high speed
    - Shielded Twisted pair
  - coaxial cable
  - Fiber optics
- Unguided or Wireless Transmission Media
  - Radio wave
  - Microwave
  - Infrared



→ The most common UTP connector RJ45
↑ Registered Jack.

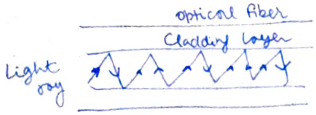
Co-axial Line :-



→ Thick & Thin ethernet
→ Cable TV.
→ Digital Transmission.

Connector → ~~BNC~~ BNC (Bayone - Neil - Concelman)
- Types
  - BNC connector
  - BNC-T
  - BNC Terminator.

Fiber Optics :-



Connector 3 types:-

(i) Subscriber channel connector

(ii) Straight Tip (ST) connector.

(iii) MT - RJ connector

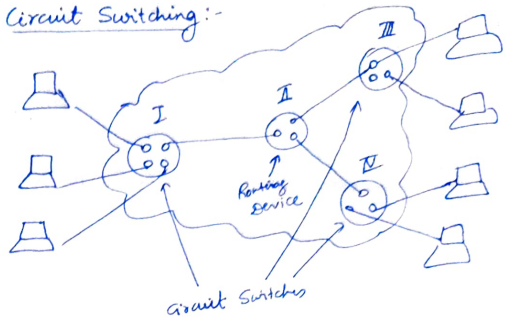
Switching Methods

↳ Techniques that can determine how connection are made & how data movement is handled from source to dest in a network.

→ Three Techniques.
- ↳ Circuit Switching
- ↳ Message Switching
- ↳ Packet Switching.

19/08/25

(i) Circuit Switching :-

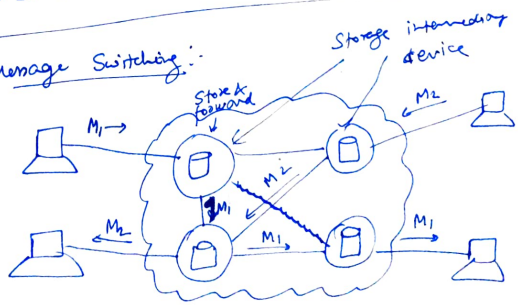


→ connection oriented
→ two-way ~~connection~~ comms. Eg: Public Networks → telephone line.
→ dedicated path.
→ reliable

→ In circuit switch the routing decision is made when the path is set up ~~~~ across the network, after the link has been set up b/w sender & receiver, the info is forwarded continously over the link. After completion the link is released.

→ disadvantage
- ↳ cost & maintenance
- ↳ cross connection.
- ↳ line breakage.
- ↳ bandwidth should be high.

(ii) Message Switching :-



Advantage
↳ Link breakage is not as bad.
↳ better congestion control
↳ high bandwidth not req^{d}
↳ No dedicated path

Disadvantage
↳ more storage → more cost

Eg: Emails

→ Each message is an ~~independent~~ unit & includes it's own source & dest^{n} address.

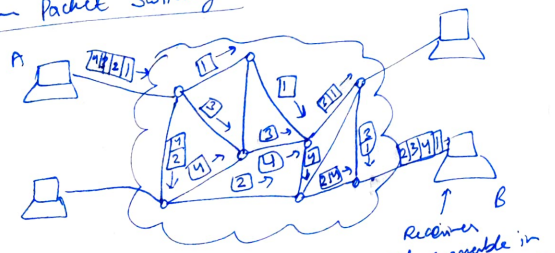
→ Message switchg network is also called as store & forward network.

(iii) Packet Switching :-
↳ 2 methods → Datagram Packet Switchg (Connectionless)
↳ Virtual-Circuit Packet switchg (Connection oriented)

→ Packet switchg, messages are broken up into packets each of which includes a header with source, dest^{n} & intermediate node address info.. Individual packets may take diff. routes to reach the dest^{n}.

→ Adv -
↳ Efficient use of bandwidth
↳ No dedicated path.

→ Datagram Packet Switching :-



(ii) Virtual-Circuit Packet Switchg :-
↳ Establishes a logical connection b/w the sending & receiving devices called virtual circuit.
→ Sending & Receiving devices communicate & agree upon the communication parameters such as max. message size, ~~max~~.
& the network path to be taken.
→ once this virtual circuit is established all the packets travel through the logical connection established b/w sending & receiving devices.

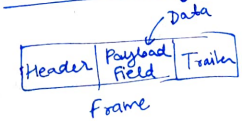
Advantage
↳ ~~Message~~ Efficient bandwidth usage

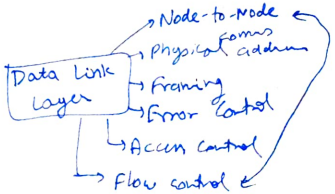
Disadvantage
↳ More RAM req^{d} → more cost
↳ Designated path

ISDN → Integrated Service Digital Network.
↳ eg: Integration voice & video in calls.

Unit -2

Data Link Layer





Design Issues :-
(i) Services provided to network layer
(ii) Framing
(iii) Error Control
(iv) Flow Control.

(i) Service provided to network layer by DLL

↳ Trasfering data from sender's network layer to receiver's network layer.

Types of Services

(i) Unacknowledged Connectionless Service :- eg:- Postal Service

(ii) Acknowledged " " eg:- Email, wifi, IEEE-802.11

(iii) Acknowledge - connection oriented " Eg:- Frame relay

(ii) Framing :-

↳ In order to provide service to the network layer the raw bit stream provided by the physical layer must be broken down into discrete frames & also compute checksum for each frame is done by DLL.

→ Four Methods of Framing :-

(i) Character Count

(ii) Flag byte with byte stuffing

(iii) Starting & Ending flags with bit stuffing

(iv) physical layer coding violation

(i) Character Count :

eg:

5	1	2	3	4	5	6	7	8	9	8	0	1	2	3	4	5	6	7	8	9	0	1	2	3
Frame 1					Frame 2						Frame 3								Frame 4						

Disadv → If count field is corrupted or lost → receiver cat find boarfy

(ii) Flag bytes with byte stuffing :-

↳ To overcome the problem in the previous method, each frames start & end with a some special byte called flag byte as both the starting & ending delimiter.

→ If the receiver loses sync it can just search for flag byte to find the end of the current frame.

→ Problem :- when the binary data like floating point no are being transmitted the flag bytes bit pattern occurs in the data this will interfere with the framing

Sol" → Sender's data link layer inserts a special escape byte just before the flag byte in the data. The DLL of the recvg end removes the escape byte before the data is transmitted to the network layer, this is called byte stuffing / character stuffing.

Original Stuffing

A	Flag	B
A	Esc	B
A	Esc	Flag	B

After Stuffing

A	Esc	Flag	B
A	Esc	Esc	B
A	Esc	Esc	Esc	Flag	B

Disadv → only used for 8-bit characters

→ Overhead → Extra esc characters

02/09/25

Bit Stuffing :-

↳ Each frame begins & ends with a special bit pattern 01111110, whenever sender's DLL encounters five consecutive one's in the data it automatically stuffs a 0 bit into the ~~outgoing~~ bit stream. This process is called as bit stuffing.

Destuffing → at receiver end.

eg:- 0110111111111111110010 ← original data

→ Bit stuffing ↓

01101111101111101111010010 ←

→ Destuffing → 0110111111111111110010

Disadvantage
↳ Extra overhead due to stuffing
↳ Inc. for size of stuffing

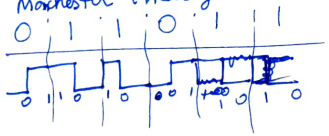
Adv.
→ works well with binary data
→ clear & unambiguous boundaries

Physical layer coding violation :

Bit 1 → High Low Pair (10)

Bit 0 → Low High Pair (01)

Manchester Encoding



Adv.
↳ No need of stuffing
↳ fast

disadv.
↳ Hardware must detect (read) digital data accurately
↳ less flexible

~~Destuffing~~ → ~~If can't find in corrupted bit, receiver can't destuff find borders~~

Physical Code ⇒ 0110100110110

Flow Control :- Speed Mismatch

Two approaches ↳

(i) Feedback based flow control → Receiver's sends back info to the sender giving it permission to send more data or about telling the sender how the receiver is doing.

(ii) Rate based F.C :- Protocol has built-in mechanism that limits the rate at which senders may transmit data without using feedback from the ~~receiver~~ user.

Error Control :

- Ensure reliable delivery
  - ↳ Ack → Positive Ack
    - ↓ Negative Ack
  - Acknowledgement
- Another Problem → Data / Ack - lost
  - ↓
  - Timer → on timer expiry → retransmission
    - ↳ Problem - Duplicate frames
      - ↳ Soln → Seq" no.

Error Detection & Correction :

Error Control
- Error Detection
  - → Parity Bit Code
  - → CRC (Cyclic Redundancy Check)
  - → Checksum
- Correction
  - ↳ Hamming Code

Types of Error :-

(i) Single Bit Error

Eg:- 11100101 (original data)
11000101 ← (received data)
↑ Error

(ii) Burst Error :- more than one bit changed

eg:- 1 1 1 0 0 0 1 0 1 ← original
1 1 0 0 0 1 1 0 1 ← received.

Most significant ↳ ↳ Least significant.

→ The no. of bit position in which two code-words differ is called hamming distance.

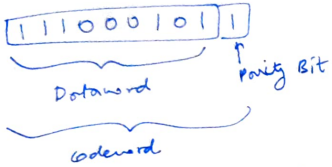
→ Length of the burst Error is the no. of bits from least significant to most significant bit in received data code between two error bits.

so, for previous eg:- Hamming distance = 3

Length of Burst Error = 6

Error detection

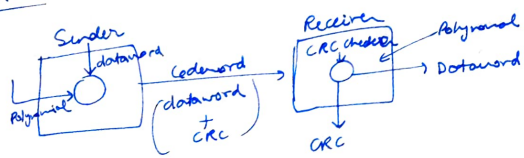
#(i) Parity Bit code :-



→ If even no. of one's → parity bit → 0
If odd " " → parity bit → 1

Note :- In even parity bit code all single bit errors can be detected but burst errors with odd hamming distance only can be detected.

CRC



Steps

→ (i) Input is a polynomial prepare a binary string for it

eg:- 1 0 0 1 for x^3+1

Coefficients $x^3\ x^2\ x^1\ x^0$

(ii) Append 'n' zeroes at the least significant bit within the dataword.

if n = 3 (highest degree in polynomial)

Dataword → 1 1 0 0 1 1 1 0

↳ 1 1 0 0 1 1 1 0 1 0 0 0 (n=3)

Perform binary division operation b/w the outcome of step 1 & step 2

using XOR

A	B	A⊕B
0	0	0
0	1	1
1	0	1
1	1	0

Polynomial

Ex:- x^4+x^3+1

1 1 0 0 1

Data → 1 0 1 0 1 0 1 0

1 1 0 0 1 | 1 0 1 0 1 0 1 0 0 0 0

1 1 0 0 1

0 1 1 0 0 0

1 1 0 0 1

0 0 0 0 1

0 0 0 0 0

1 1 0 1 0

1 1 0 0 1

1 0 0 0 0

1 1 0 0 1

x x x x 0

1 1 0 1 0 0 0 1 0

1 0 0 1 | 1 1 0 0 1 0 1 1 0 0 0 0

1 0 0 1

1 0 1 1

1 0 0 1

0 1 0 0 0

0 0 0 0

1 0 0 1

0 0 0 1

0 0 0 0

0 0 0 1 0

0 0 0 0

1 0 0 0

1 0 0 1

0 0 1 0

0 0 0 0

0 1 0 0

CRC

CRC

Receiver's End

Codeword → 110010110010

Polynomial → x^3+1

```
            110100010
1001 ) 110010110010
       1001
       ----
        1011
        1001
        ----
         0100
         0000
         ----
          1001
          1001
          ----
          0001
          0000
          ----
           0010
           0000
           ----
            0100
            0000
            ----
             1001
             1001
             ----
              0000
              0000
              ----
              0000
```

No Error detected ← 0000

CRC method.
- → convert polynomial
- → Add n zeros if n is highest degree
- → Perform division
- → Replace remainder with those n bits ⇒ codeword

Receiver
Perform division using codeword if 0000 no error.

$$Eff. = \frac{m}{m+r} \times 100$$

Checksum Method :-

Steps

(i) Decompose the given dataword into groups of n bits
↳ length of checksum

(ii) Perform binary addition on it.
↳ if the result has a carry then add carry with the sum & finally take once compliment of it to produce checksum.
↳ if no carry → simply take once compliment of sum

(iii) The created checksum will be appended at the least sig. bit with the dataword to produce codeword.

eg: Dataword → 010001101100010001010100

Let checksum length = 8

Decompose ⇒

01000110 (8) 11000100 (8) 01010100 (8)

Binary addition

```
  01000110
  11000100
  01010100
  --------
carry 1 | 01011110
                 1
  --------
  01011111
```

1's complement

10100000 → checksum

Codeword → 010001101100010001010100 10100000
(Dataword) (Checksum)

Checksum Steps
(i) Break the dataword into K no. of blocks of 'n' bits each
(ii) Sum all k blocks
(iii) Do 1's complement ↳ checksum ⇒ codeword

Receiver
(i) Add all datawords including checksum - if 00—0. no error.

At Receiver's End

```
  01000110
  11000100
  01010100
  10100000
  --------
1 11111110
         1
  --------
  11111111
```

complement ↳ 00000000 — No error

Note: checksum can detect all single bit errors. It can detect all burst errors upto length 'n'!

08/09/25

Hamming Code:

if Dataword = m bits
then codeword = (m+r) bits
r → redundant bits

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

r → min integer value that satisfies this relation

dataword m	redundant bits (r)	Codeword (n)
1	2	3
2	3	5
3	3	6
4	3	7
5	4	9
6	4	10
7	4	11
8	4	12
9	4	13

Q. Compute the hamming code / codeword for dataword = 11010101 when there is even parity, using hamming code technique.

if m = 8 then r = 4 → from above table.

m + r = 12 bits → codeword

$2^0 = 1$
$2^1 = 2$
$2^2 = 4$
$2^3 = 8$
} reference positions

codeword → 1 1 0 1 r_8 0 1 0 r_4 r_2 r_1
12 11 10 9 8 7 6 5 4 3 2 1

	r_8	r_4	r_2	r_1
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0

r_1 = 1, 3, 5, 7, 9, 11
r_2 = 2, 3, 6, 7, 10, 11
r_4 = 4, 5, 6, 7, 12
r_8 = 8, 9, 10, 11, 12

r_1 = 1, 3, 5, 7, 9, 11
r_1 = 1 0 0 1 1 → $r_1 = 1$

r_2 → 1 1 0 0 1 → $r_2 = 1$

r_4 → 0 1 0 1 → $r_4 = 0$

r_8 → 1 0 1 1
$r_8 = 1$

codeword → 1 1 0 1 1 0 1 0 0 1 1 1

Error bit

At receiving end

codeword received : 1 1 0 1 1 0 1 0 0 0 1 1
12 11 10 9 8 7 6 5 4 3 2 1

& Even parity

Let checkbits → c

1 → c_1 = 1, 3, 5, 7, 9, 11 = 10011 ⇒ $c_1 = 1$ { odd parity ∴ 0
2 → c_2 = 2, 3, 6, 7, 10, 11 = 101001 = $c_2 = 1$
4 → c_3 = 4, 5, 6, 7, 10, 11 = 00101 ⇒ $c_3 = 0$ { even parity ∴ 1
8 → c_4 = 8, 9, 10, 11, 12 = 11011 ⇒ $c_4 = 0$

$c_4 c_3 c_2 c_1$ = 3 → position where error occured.
0 0 1 1
Decimal

correct codeword → 1 1 0 1 1 0 1 0 0 1 1 1

Elementary Data Link Protocols :-

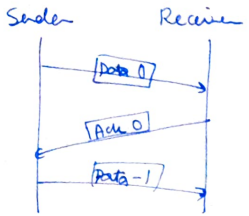
(i) Unrestricted Simplex Protocol
(ii) Simplex stop & wait protocol
(iii) Simplex protocol for a noisy channel
(PAR or ARQ)
Positive Acknowledgement with retransmission
Automatic Repeat Request

Assumption

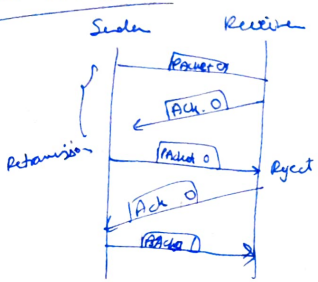
(i) Phy, DLL, NL → Independent

(ii) A → B ; reliable, connection-oriented service

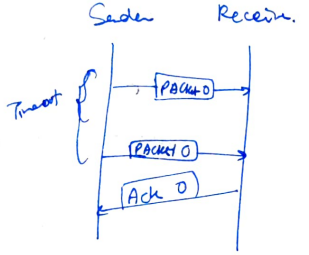
(iii) No crashes of any machine



When Ack is lost



When data is lost



04/04/25

Sliding window Protocol :-

Aim

↳ 1st way → 2 separate ckts
Forward (For data)
Reverse (For Ack)

↳ 2nd way → same ckt
↳ for both A & B
↳ Kind field → To differentiate b/w data & Ack.

↳ 3rd way → same ckt

A —D0→ B

A ←D1— B
A —A0→

Piggybacking → sending Ack. with data → Disadvantage
↳ waiting for data from the received end.

Advantage
↳ Efficient usage of bandwidth

→ SWP → Sender SW
↳ Receiver SW

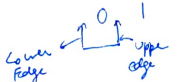
Each outgoing frame → seq no. → 0 to $2^n - 1$

eg: $n = 3 \Rightarrow 2^n - 1 = 7$

∴ seq no. → 0 to 7

sliding window size = 1

0 1 2 3 4 5 6 7

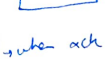


SW size = 1

↳ Advancing (Growing) when ack. comes.

when ack of 0 has not come yet,

when ack is done (of 0)



① One-bit sliding window protocol → Based on stop & wait ARQ

n = 1

seq no. → 0 to $2^n - 1$ } seq no. → 0 or 1
0 to 1

Steps :-

(i) NL → DLL
packet

(ii) DLL Packet → frames

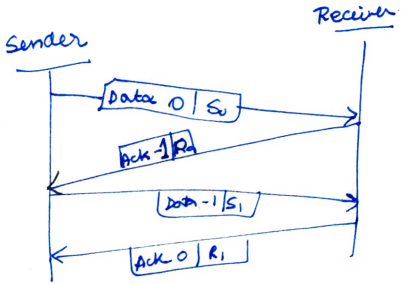
(iii) Receiver DLL → check duplicates

(iv) if dup → discard else accept & send to NL

→ Both data & acknowledgement frames are strictly alternative,
i.e 0 & 1.

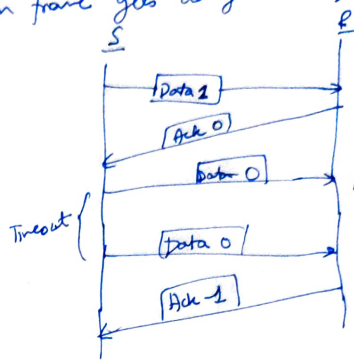
* → Seq no. of frames in stop & wait ARQ is always modulo 2.

frame →	1	2	3	4	5	6	7	8	9	10
Seq no →	1	0	1	0	1	0	1	0	1	0

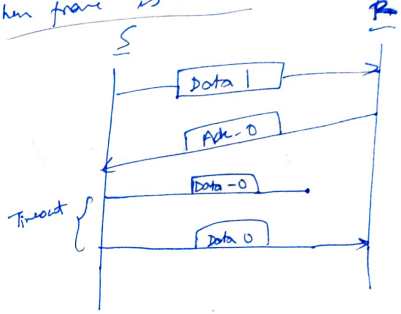


* Retransmission → 3 times

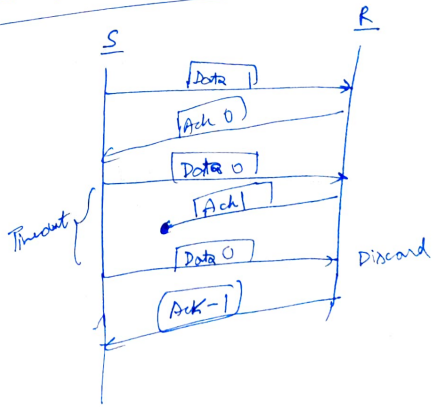
① when frame gets damaged → Timer is introduced



② When frame is lost



③ When Ack is lost



Advantage → Data received in sequential order

Disadvantage : Takes more time.

Ex. Satellite comm.

→ Go back- N -SWP

↳ It continously transmits frames until a nyatic ack. is not received by the sender.

Sender window size = $2^n - 1$, Receiver window size = 1

n → seq. no. of binary bits in repeat cycle.

so, if n=2, total no. = 0,1,2,3 (i.e 4)

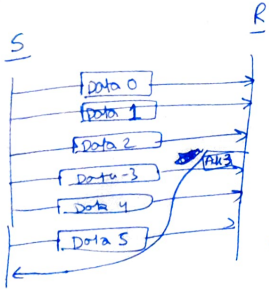
eg: n = 3

Sender window size = $2^3 - 1 = 7$ (0 to 6)

& Seq no. → 0 to $2^n - 1$ (0 to 7)

Frame no.	0	1	2	3	4	5	6	7	8	9	10	11	12
Seq no.	0	1	2	3	4	5	6	7	0	1	2	3	4

window



0 1 2 3 4 5 6 7 0 1 2 3

Sender's window

0 1 2 3 4 5 6 7 0 1 2 3

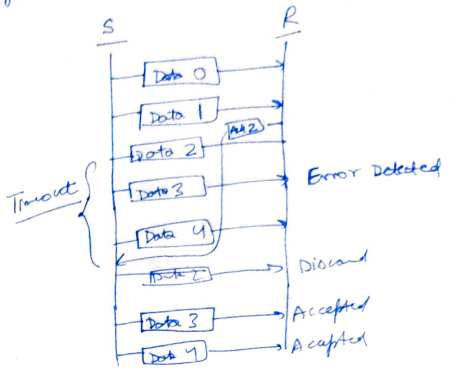
delivered.

#→ The sending window size shrinks when the sender's send the frame & increases when the sender recieves the ack for receives.

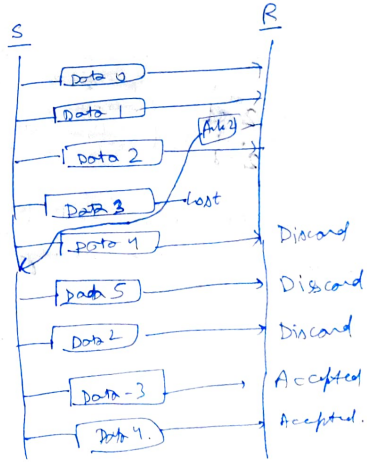
→ The receiver window size is rep. by a pointer which indicates which data frame it is expecting.

→ Once the receiver receives that frame, then the pointer moves to the next frame

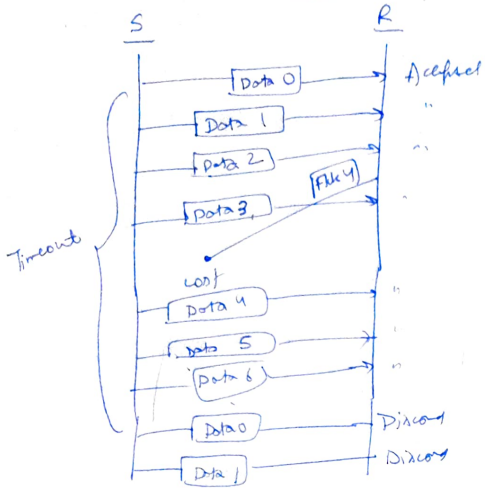
1st case of retransmission : when frame gets lost/damaged



2nd case



(3)



→ Advantage :

↳ If one ack is lost or frame gets damaged / lost we have to resist all data.

Disadvantage

Doesn't accept out of order packets ie. the receiver window will discard all frames until it sees that lost frame.

waits for a particular frame if that frame got lost, and while other frames after it reach it, receiver will discard all frames until it sees that lost frame.

Selective Repeat SWP :-

↳ Sender window size = 2^{n-1}

Receiver window size = 2^{n-1}

→ In this protocol, the transmitter doesn't wait for the acknowledgement for transmission of next frame it sends frame continously till it receives a negative ack, ↦ (Nak where frame is not received)

→ In this protocol, only specified / damaged or lost frame are being transmitted

→ Retransmitter must contain a searchy mechanism. that allows it to find & select only the requested frame for retransmission

→ Receiver can perform sorting as it is able to store frames receipt after a negative ack. has been sent, until the damaged frame is retransmitted replaced.

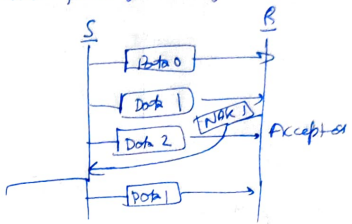
eg: n = 3

$2^{n-1} = 4 \rightarrow (0, 1, 2, 3)$ → window size

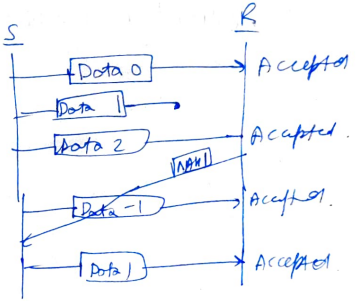
Seq no. → 0 to $2^n - 1$ = 0 to 7

Retransmission cases

(1) When frame get damaged.



(2) When frame is lost



(3) same as previous case 3 of go back-N SWP

→ IEEE LAN Protocol (IEEE 8002) → Deep

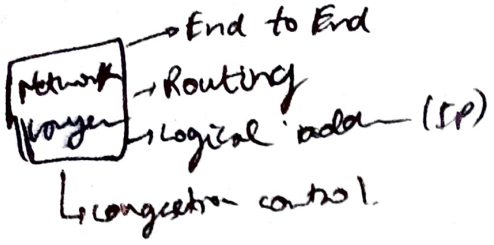
→ FDDI, Token ring

↳ Brief

11/10/25

Unit 3

Network layer



Routing :-

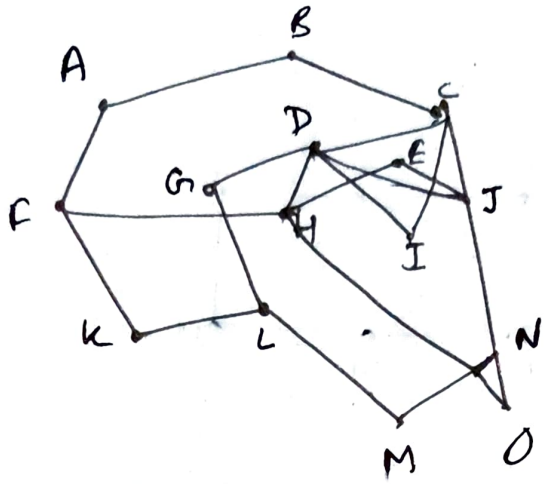
- Non-Adaptive (Static)
  - Shortest path algo
    - (Dijstra algo)
  - Flooding
- Adaptive (Dynamic)
  - Distance vector routing
  - Link State routing
  - Hierarchical routing.

→ optimality principle :- It states that if router 'j' is on the optimal path from router 'i' to 'k', then the optimal path from j to k also falls along the same route.

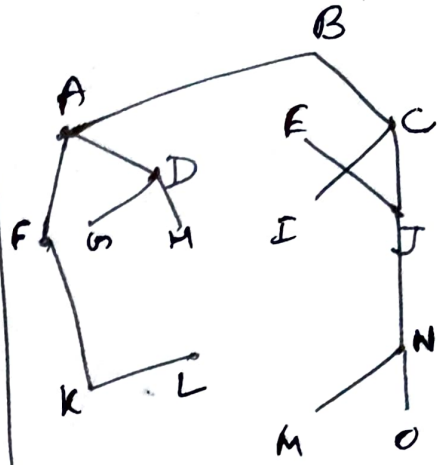
If a route better than route j to k existed, it can be concatenated with route i to j, to improve the route from i to k.

→ A set of optimal routes from all sources to a given dest^n forms a tree rooted at the dest^n, such a tree is called as sink tree; where the dist^n metric is no. of hops.

sink tree
↳ no loops
↳ not unique



Subnet



Sink tree for router B

↑
To reach from B to everywhere optimally

→ Non-adaptive Algo → Dijkstra Algo → DIY

→ Flooding → In this algo, every incoming packet is sent out on every outgoing line, except the one it arrived on. Ex → wokie-toki, wireless network.

Disadv → multiple duplicate packets

→ corrective measures

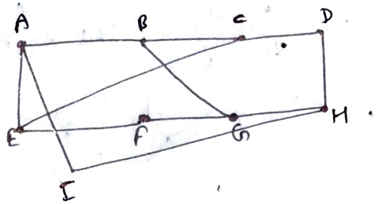
(i) maintaining hop counter;
↳ is contained in the header of each packet which is decremented at each hop with the packet being discarded when the count reaches 0.

(ii) Keep track of which packets have been flooded, to avoid this source router adds a seq^n no. in each packet.

(iii) Selective flooding → The routers do not send every incoming packet out on every line, only on those lines that are going approx. in right direct^n.

→ Dynamic Alg.

(i) Dist^n vector routing → In this algo, each router maintains a table i.e. a vector which gives the best known distance to each dest^n, and which line to use to get there and it is also called as Bellman-Ford routing algo or Ford-Fulkerson algo.



	A	H	New routing table for router I (delay)	Via
A	0	20	8	A
B	10	81	18	A
C	27	19	31	H
D	28	8		
E	12	30		
F	29	19		
G	16	6	18	H
H	19	7		
I	9			

IA delay = 8

IH delay = 12

vectors received from I's four neighbours

new routing table for router I

Adv. → good in small networks

Drawbacks

(i) It is slowness in converging to the correct answer.

● This is due to the ~~problem~~ count to infinity problem.

↓ sol^n

Split horizon algo.

(iii) This algo doesn't take live bandwidth when choosing the route

Link State Algo :-

↳ each router periodically updates it's routing table,

i.e each router shares the info of it's neighbour to all other router over network using flooding algo

Five operation

(i) Each router should discover it's neighbours & learn their network address.

(ii) Measure the delay or cost to each of it's neighbour.

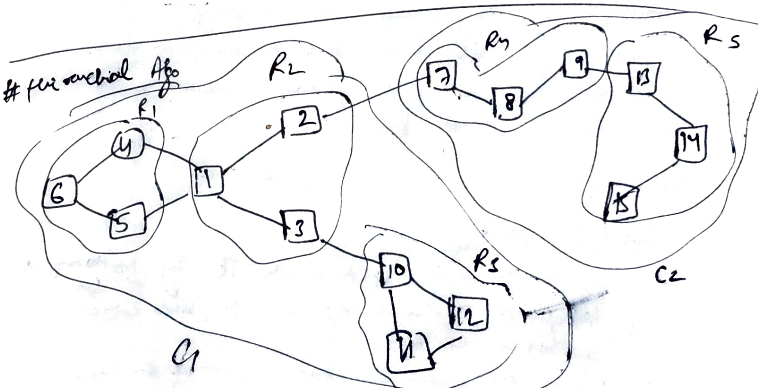
(iii) Construct a packet telling all it has learned i.e. network address and delays of all neighbours.

(iv) Send this packet to all other routers

(v) Compute the shortest path to every other router.

Adv. → No count to ∞ problem | Dis :- cost expensive.
→ Fast

Hierarchical Algo



Groups → Level 5
Zones → Level 4
Clusters → Level 3
Regions → Level 2
Routers → Level 1

14/10/25

Congestion Control Algo :- (Prevent or Remove)

Principal of congestion control –

Sol^n of congestion problem - two catgories
- Open loop sol^n
- Closed loop sol^n

(i) Open loop sol^n

↳ It tries to solve the congestion problem by preventing it from happening

↳ Exercised by using methods such as deciding when to accept new packets, when to discard, which packets are to be discarded & making scheduling decision at various points

(ii) Closed loop control → based on feedback

↳ Three steps

(i) detect the congestion & locate it by monitoring the system.

(ii) transfer the info about congestion to places where actions can be taken.

(iii) Adjust the system operation to correct the congestion.

→ Congestion control in virtual circuit subnets :-

→ One widely used technique to keep congestion that has already started from getting worst is admission control.

→ Once the congestion has been detected don't setup any more virtual circuits until the congestion is cleared.

→ Congestion control in datagram subnets

(i) Choke packets (ii) Hop by hop control (iii) Load shredding

(i) In this technique each router associates a real variable with each of it's output line

(ii) This real variable say 'u', has the value between 0 & 1, indicating % utilization of that line. If the value of u goes beyond the threshold then the line will enter into warning state

(iii) The router will check each newly arrived packet to see if it's output line is in warning state, if it is then the router will send back a choke packet signal to the sending host then the sender might not generate another packets.

(ii) Load Shredding
- (i) Wine Policy
- (ii) Milk Policy
- (iii) Intelligent discard policy

Load shredding states that when routers are being flooded by packets that they cannot handle they should simply discard the packets away.

→ The policy for dropping a packet depends on the type of application

Eg: For file transfer
↳ old packet more imp.

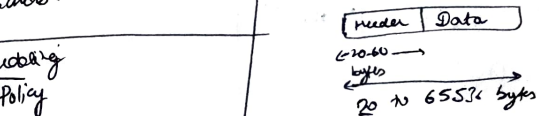
for multimedia
↳ new packet more imp

Policies :-

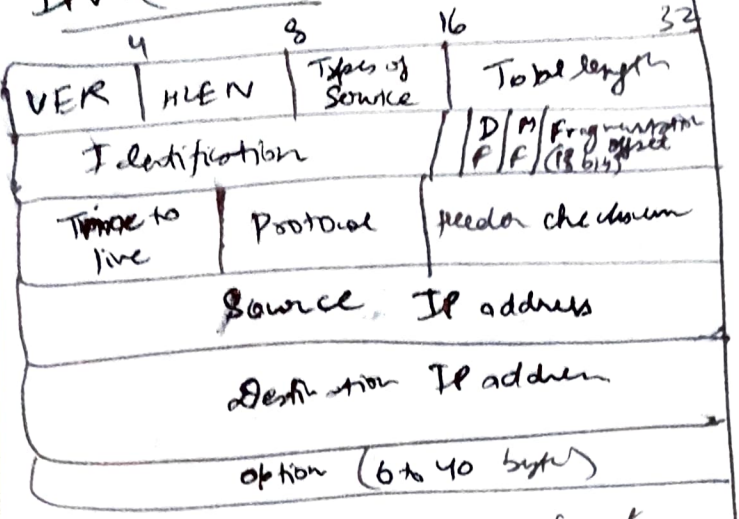
③ Jitter Control :-
↳ variation in delay for packets arrival time belongs to the same flow

Protols in Network layer
- ↳ ICMP (Error messages)
- ↳ IGMP
- ↳ IP
- ↳ ARP
- ↳ RARP

Internet Protocol
- ↳ IP Datagram
- ↳ IPv4



IPv4 (header)



DF → Don't fragment (of data)

MF → More fragment (of data)

IP addressing :-

- classful Addressing
- classless Addressing

IP Addressing

- IPv4 (32 bits)
  - classful
    - class A (0 - 127)
    - " B (128 - 191)
    - " C (192 - 223)
    - " D (224 - 239)
    - " E (240 - 255)
  - classless
- IPv6 (128 bits)

→ IP address notation

- Binary notation
- Dotted decimal notation

Eg :- 01110100 | 10101010 | 00001111 | 10101111

⇒ 116 . 170 . 15 . 175

(116 - class A)

03/11/25

Computer Networks

Transport Layer — Unit IV

Segment/reassembly | Multiple delivery | Port add | process to process | Flow cont | Error control | Congestion

Design Issues :-

(i) Establishing maintaining & releasing connection
(ii) Addressing (iii) Data Transfer (iv) Flow control
(v) Error Control (vi) Congestion control.

In Transport layer is performed end to end rather than node to node as in DLL.
→ Transport layer uses SLP to perform flow control.

(i) Connection Management
(1) Establish (2) Maintain (3) Release

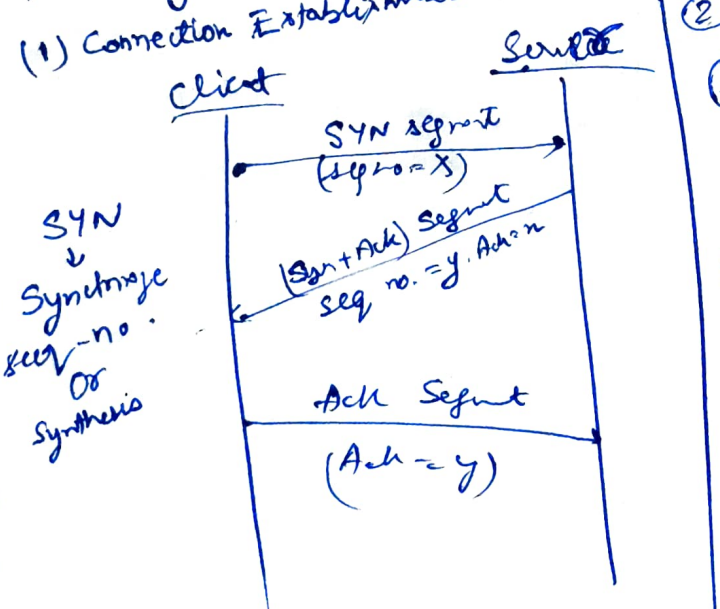
(ii) Addressing →
Socket Address → IP address + Port No.

(v) Error Control :-
Types of Error are due to :-
(i) Damaged bits (ii) Non-delivery of TPDU
(iii) Duplicate Delivery of TPDU (iv) Delivery of TPDU to wrong destination

→ TL Services :-
(i) Connection oriented service → TCP & SCTP (Stream control Transmission protocol)
(ii) Connectionless service → UDP

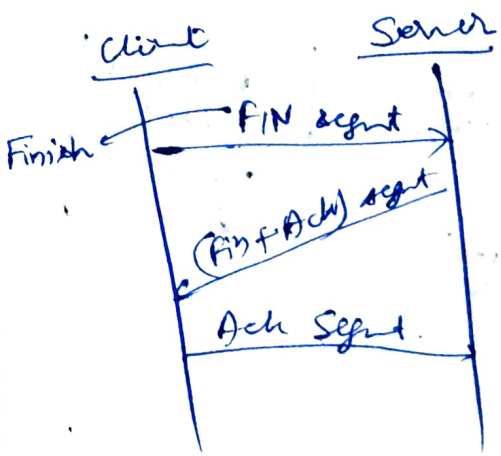
→ 3 way handshaking mechanism (to establish & release the connection)

(1) Connection Establishment



(2) Data Transfer

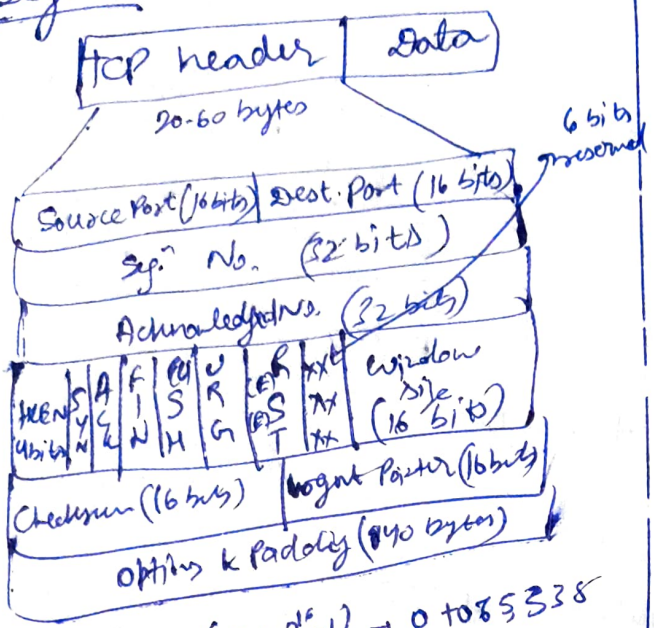
(3) Connection Release



TCP

↳ Stream Data Transfer reliability
↳ Efficient Flow control
↳ Full-duplex operation
↳ multiplexing

Segment



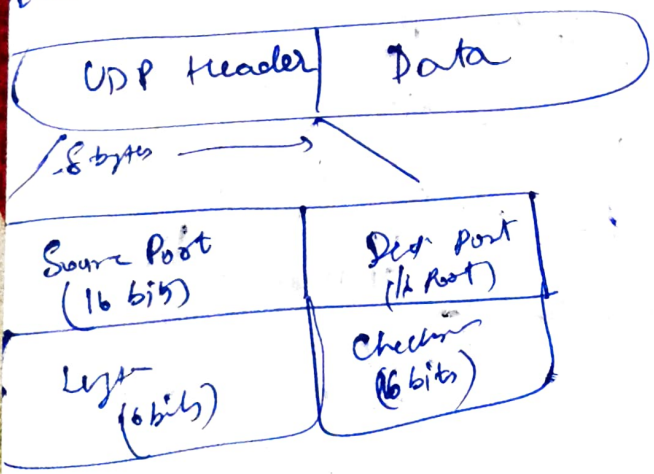
Ports No. ($0 \to 2^{16}-1$) → 0 to 65535

HTTP → 80, SMTP → 25

FTP → 20 & 21

UDP

↳ TL Protocol → connectionless → unreliable
Error control → Process to process → Flow &
→ Data unit → Datagram



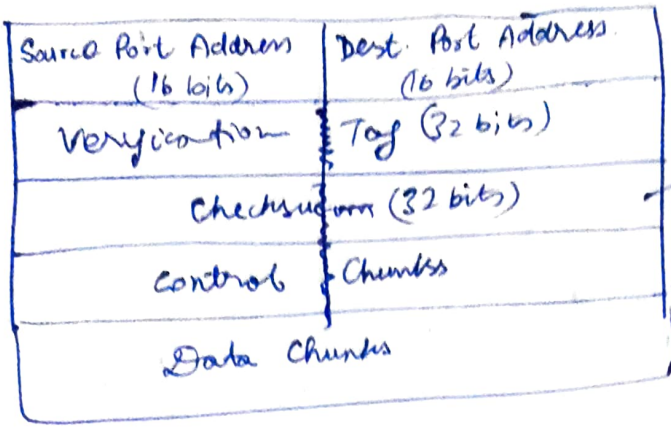
4/11/25

SCTP

↳ reliable
↳ TL message-oriented protocol
↳ connection-oriented
↳ multi-streaming
↳ Full-duplex service
↳ Flow-control
↳ congestion control
↳ Process to process comm.
↳ Multihoming Service
↳ can be connected to multiple networks with multiple IP addresses

→ TCP has one source & one dest. IP address, this means even if the sender or receiver is a multihomed host only one of these IP address per end can be utilised during the connection
→ Sending and receiving hosts can define multiple IP address in each end for an association/connection. It is for fault tolerance, that is, when one path fails another interface can be used for data delivery without interruption.

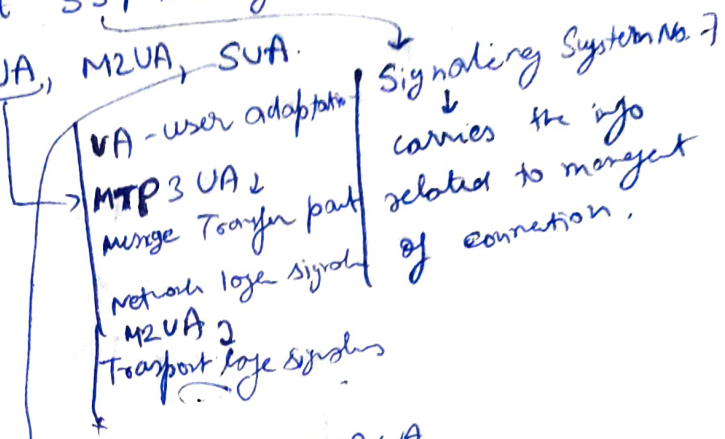
SCTP Packet :-



App" :-

(i) Telephony comm" (Eg. whatsapp calls)

(ii) Mobile Networks → Transport SS7 messages for 3G/4G/5G via M3UA, M2UA, SUA.



(iii) Roaming Security

(iv) Reliable Transport

Session layer

- ↳ sery mangent
- ↳ Sync
- ↳ Dialog control.

Design Issues :-

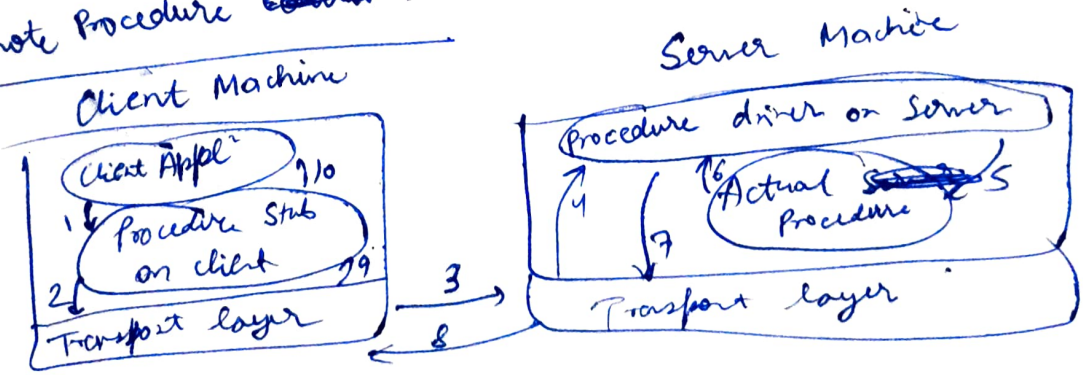
(i) To allow machines to establish sessions b/w them seamlessly

(ii) Provide enhanced services to user

(iii) Manage dialog control

(iv) To provide services like token management and synchronisation.

Remote Procedure call :



→ RPC tries to give the programer an illusion that a program on one machine can call a procedure located on another machine. The main goal is to hide the existence of the network from the programer.

Steps

(i) When a client calls a remote procedure the call is actually made to a local stub (procedure) by passing the parameters.

(ii) The client stub marshalls the parameters into a message. marshaling converts the representation of the parameters into a std format & copying each parameter into the message

(iii) client stub passes the message to transport layer which sends it to the remote server machine.

(iv) When RPC server message arrives the transport layer of server passes the message to the server driver.

(v) The parameters are unmarshalled by the driver and calls the desired server routine user regular procedure call mechanism

(vi) when the server procedure completes it returns to the server stub or driver which marshalls the return values into a message

(vii) Server → driver

(viii)

(ix)

(x) Client stub from unmarshalls the return parameters and the execution return to the caller.

Design Issue for RPC → Transparency.

Unit -(V)

Presentation layer :-

→ Data compression :-

- Lossless Data compression
  - Run length encoding
  - huffman coding
  - Lempel Ziv encoding
- Lossy Data compression
  - JPEG
  - MPEG
  - MP3

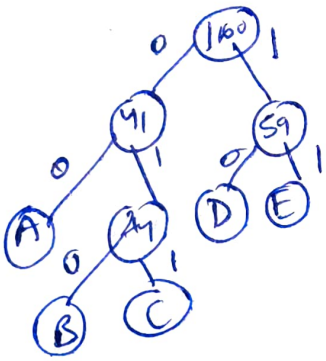
→ (i) Run-length encoding :-

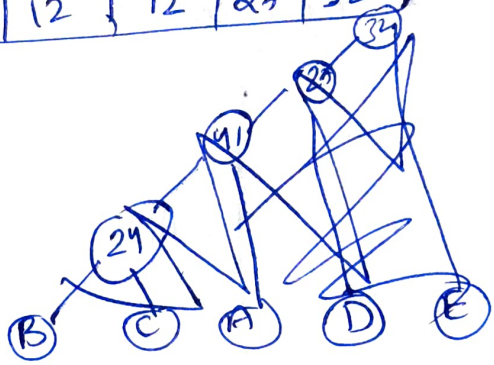
Original : BBBBBBBBAAAAAAAAAAAAMNNNNNNN

Data :- B08A10M01N06

(ii) huffman Coding

Character	A	B	C	D	E
Freq.	17	12	12	27	32

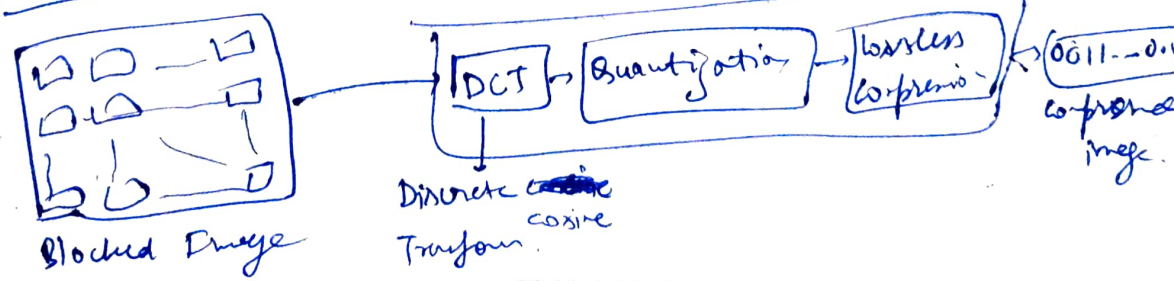


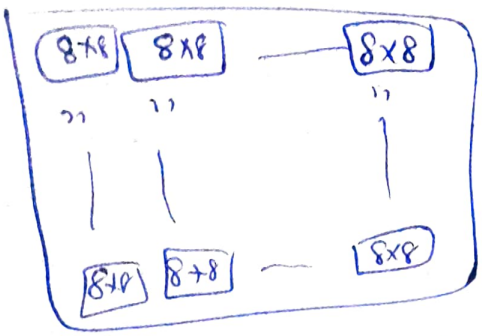


A : 00
B : 010
C : 011
D : 10
E : 11

(iii) Lempel ziv. (from whatsapp)

JPEG :- Compress procedure :-





JPEG grayscale - 640×480p

Grayscale image → 307,200 pixels

↓

24,57,600 bits.

color picture → 72,72,800 bits

10/11/25

→ Encryption / Decryption :-

Plain text → Ciphertext and vice versa.

Key :- Public key & Private key.

Symmetric Asymmetric

Algo ↓ ↓

RSA, Diffie-Hellman.

eg: DES, 3DES, AES ↓ Adv. DES. ↓ Reverse Shared

↓ ↓

Data Encrypⁿ Std. Slayered DES

5 features :-

(i) Confidentiality

(ii) Authentication

(iii) Integrity

(iv) Non-Repudiation → Sender / Receiver cannot deny doing sending or receiving data.

(v) Access - control

Unit - 5

App Layer

Protocols :- HTTP, HTTPS, FTP, SMTP, IMAP, POP Telnet,

DHCP → Dynamic Host config protocol

↓

→ Assigns the IP address, subnet masks, default gateways, DNS servers, etc to users or ~~network~~ login to network.

App layer :- Models

Client/Server Model (more secured)
- Client
- Server
- Because we can establish many security protocols easily

Peer to Peer Model (P2P) (no server)

Server Daemon

→ Server relies on a service called Server Daemon, it runs in the background and listens for request for that service. It can then exchange messages appropriately and send the requested data.

→ Common Port nos.

(1) TCP

FTP → 20 & 21, Telnet → 23, SMTP → 25

DNS → 53 (Both TCP & UDP)

(2) UDP → DHCP → 67 & 68, POP → 110

FTAM :- File Transfer, Access & Management

→ OSI std.

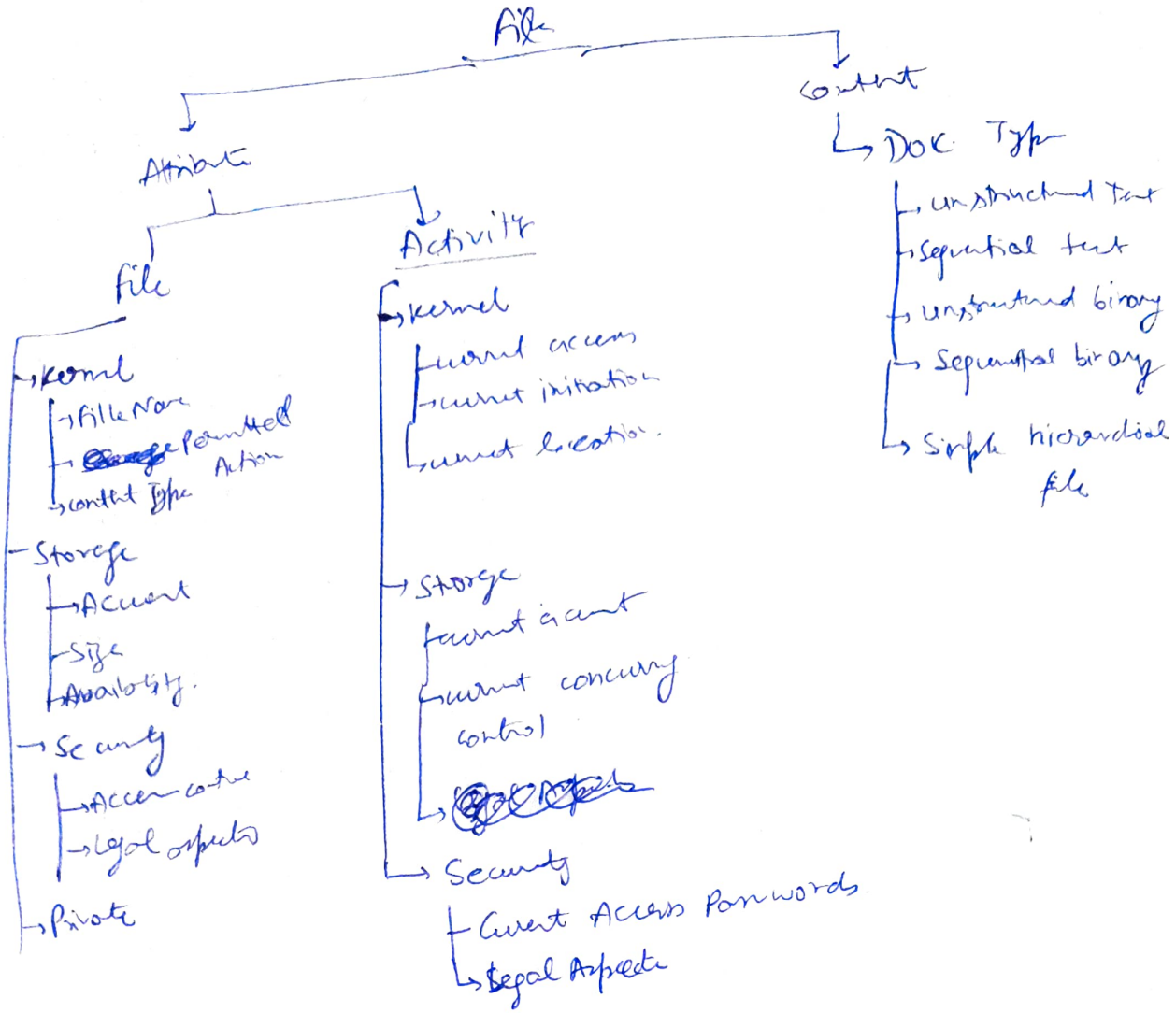
→ Attributes

4 group of attributes

- Kernel Group → Same for every file like extensions,
- Storage " → Size, etc
- Security " → Access, etc
- Private " → For additional attributes

→ FTAM Architecture

File
- Attributes
  - File
  - Activity
- Content

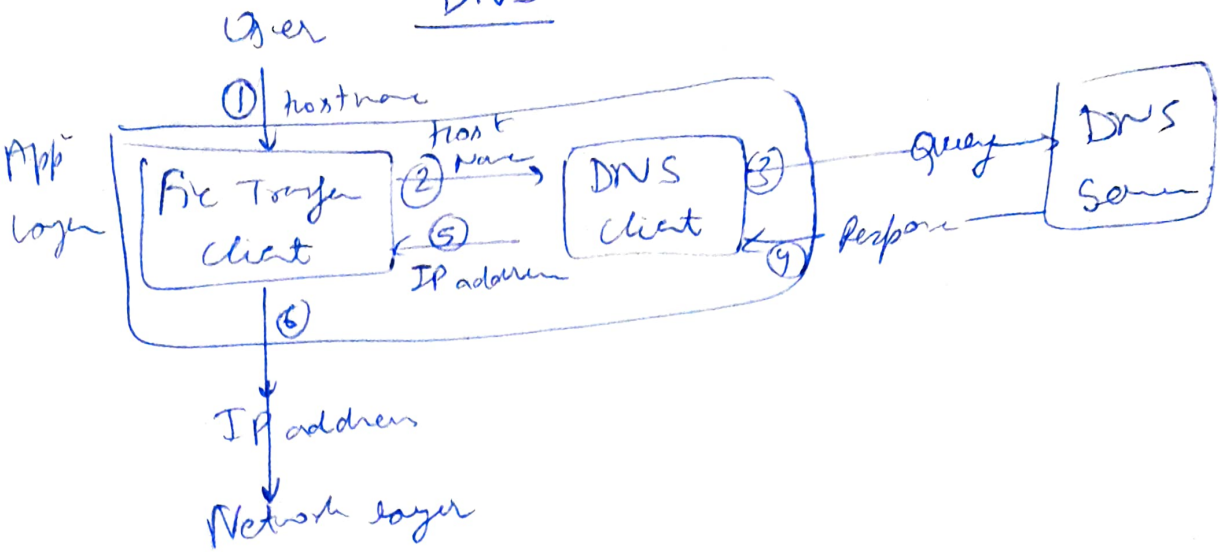


→ In virtual filestore, data record is called data unit (DU) and a node may/may not have a dataunit associated with it

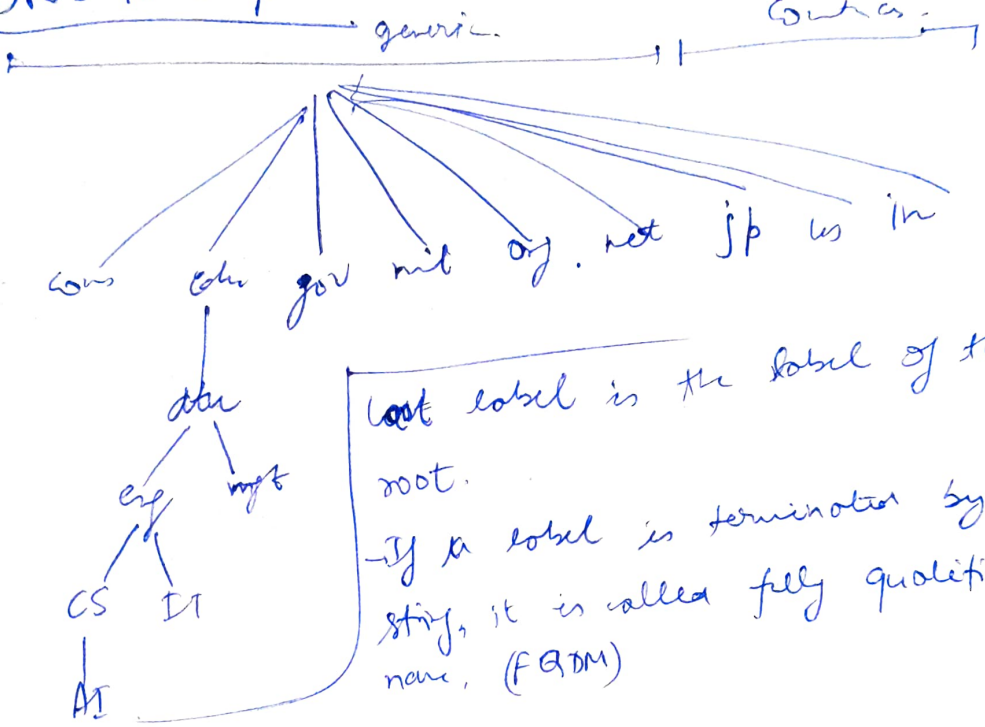
→ The DU are related to eachother through a hierarchical structure called file Access Data Units (FADU)

→ DU is considered to be the smallest data that can be accessed.

DNS



1. DNS Name Space :



Last label is the label of the root.

If a label is terminated by a null string, it is called fully qualified domain name. (FQDN)

App protocol,