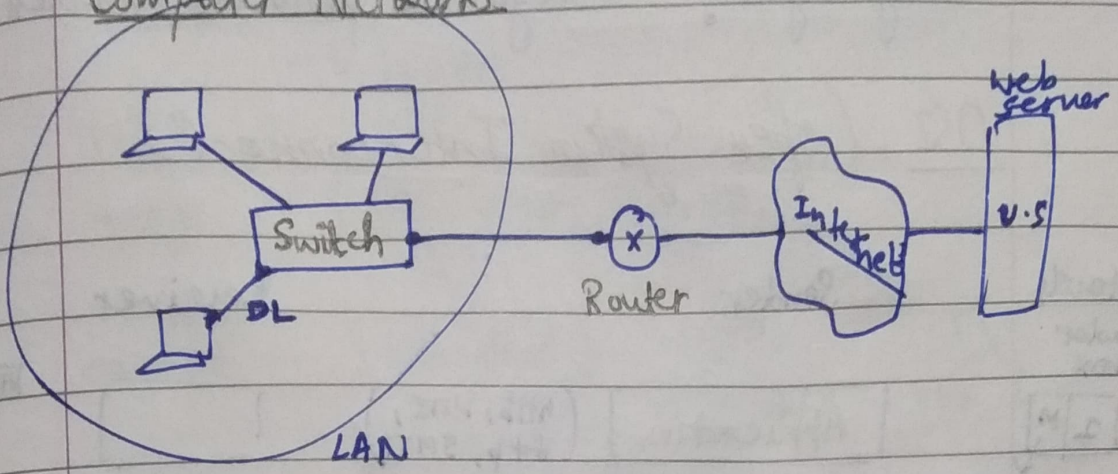
Part 1

Page No. _____
Date

Computer Networks



① **Switch**
- connects computer in small network (LAN)

② **Router**
- connects 2 or more networks

- not good bandwidth utilization algo.
- form single network
- i.e one broadcast to all

Switch	③ Network hub
- connects computer	- bandwidth gets split b/w computers
- x	- slow
- hence fast	- broadcast data packet to all connected devices.

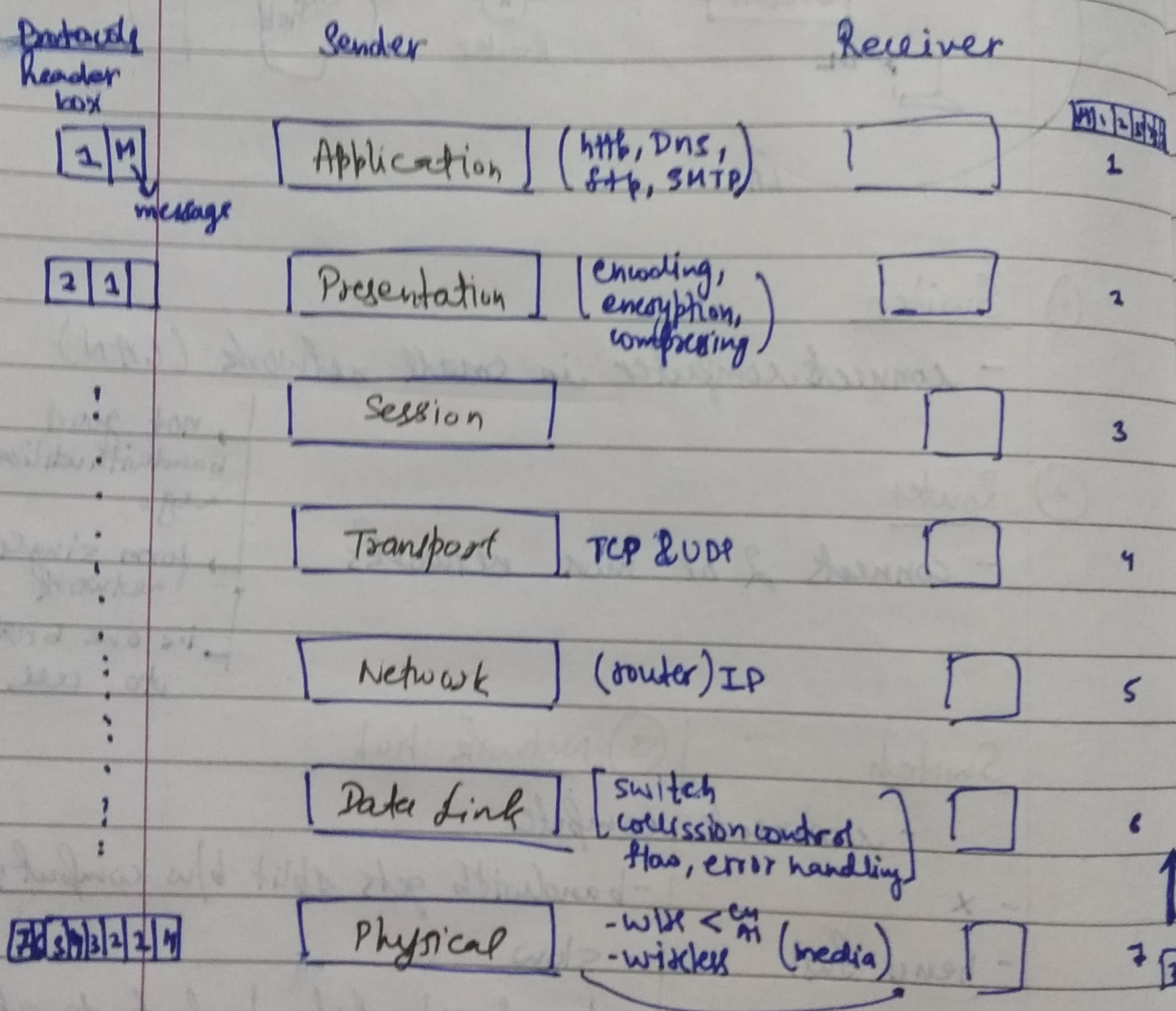
OSI model (theory)

TCP/IP model (used)

Page No.
Date

- Why layers? → organize protocols/header

OSI (open System Interconnect)



- 7 layer

Page No. _____
Date

TCP/IP

- 4 layers

Unit		
message	Application	http
segment	Transport	UDP → DNS, DSCP ; TCP → http, FTP
Packet	Network	IP
frame	Link layer	

Page No.
Date

(I) Data Link Layer

- implemented in NIC (network interface card)
- every NIC has 48 bit MAC Address / Ethernet address
  (Media Access Control) (unique)
- OS provides drivers for NIC

Functionalities of DL

(1) Framing → Bit Stuffing
↳ Byte Stuffing
↳ ARP (Address Resolution Protocol)

(2) Error Detection - ignores frame with error & retransmits [Cyclic Redundancy Check]
- Parity - CRC - checksum

(3) Error & flow Control
↳ Stop & wait
↳ Go Back N
↳ Selective Refeat
(Sliding Window) — ARQ (Automatic Repeat Request)

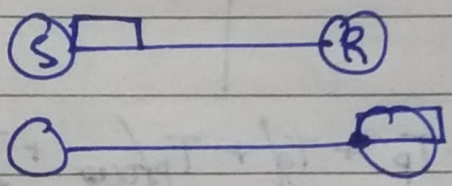
(4) Multiple Access - for no collission
↳ ALOHA
↳ CSMA/CD [Carrier Sense Multiple Access Collission Detection]

Delays

① Transmission Delay
- ↳ link bandwith (B)
- ↳ packet length (L)

$$\text{delay} = L/B$$

② Propogation delay



- ↳ velocity
- ↳ distance

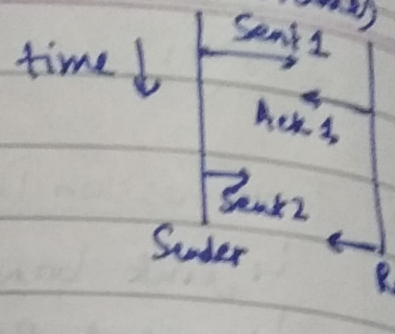
$$\text{delay} = d/v$$

③ Queuing delay
- ↳ depends on traffic

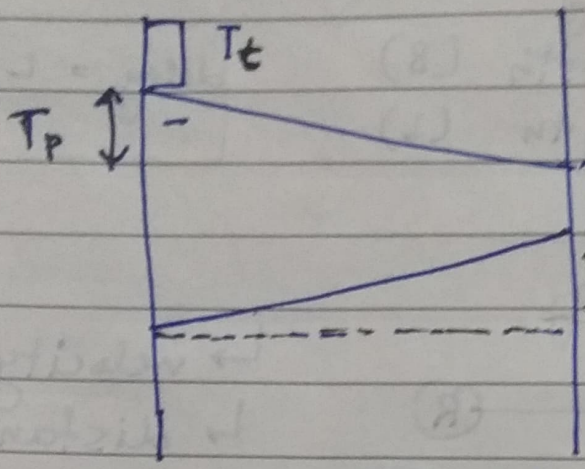
④ Processing Delay
- ↳ depends on loading of router & processing power.

ARQ

Error & flow Control



① Stop & wait



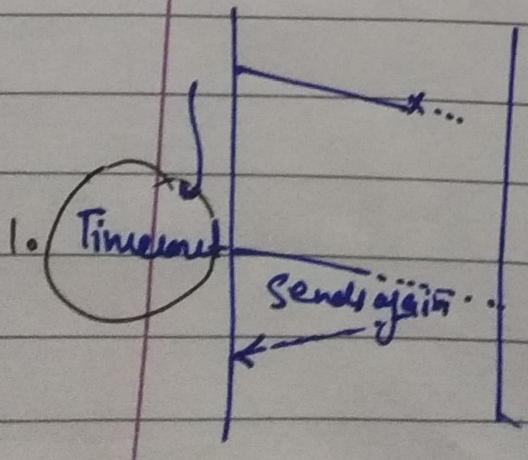
$$Total = T_t + T_p + T_q + T_{process} + T_{Ack} + T_p$$

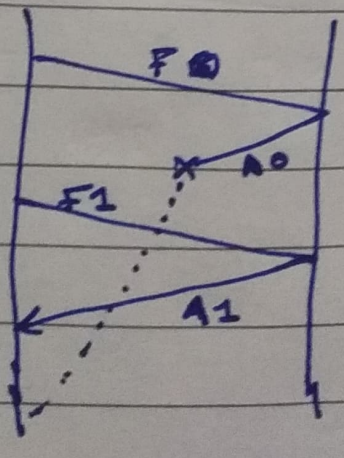
$$Total = T_t + 2T_p$$

$$\text{efficiency} = \frac{T_t}{T_t + 2T_p} = \frac{1}{1+2a} \qquad a = \frac{T_p}{T_t}$$

$$\text{Throughput} = \frac{L}{T_t + 2T_p}$$

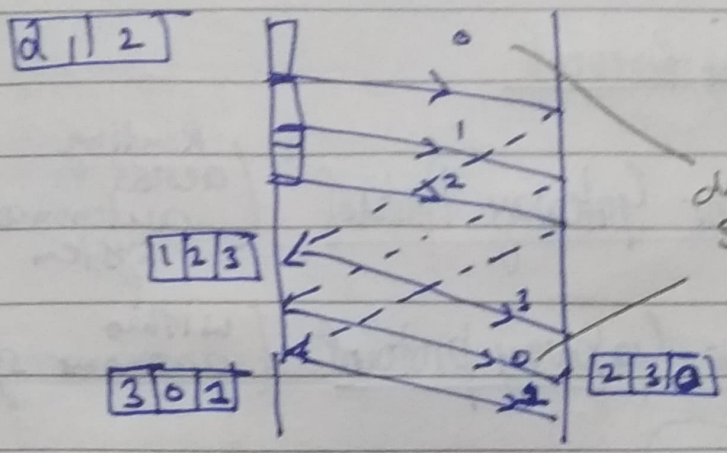
Alternating Bit protocol – Frame Seq No. & Ack Seq No. 0 & 1





Page No.
Date

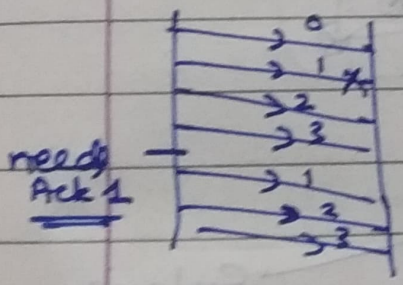
(2) Go Back N – maintains a copy in window Buffer



$$eff = \frac{N}{1+2a}$$

* N → no. of frames

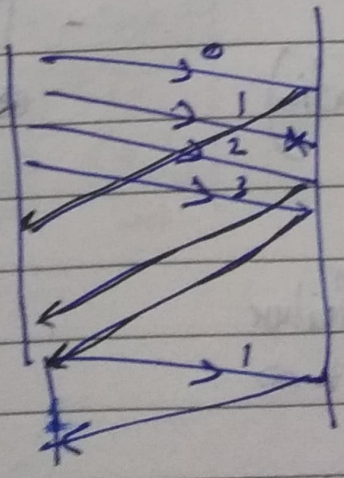
* **Problem** : If a frame is lost. then all frames after that are sent again.



(3) Selective Repeat – complex (more work)
– when channel is too noisy

$$eff = \frac{N}{1+2a}$$

* N → window size



Static (manual)
Dynamic

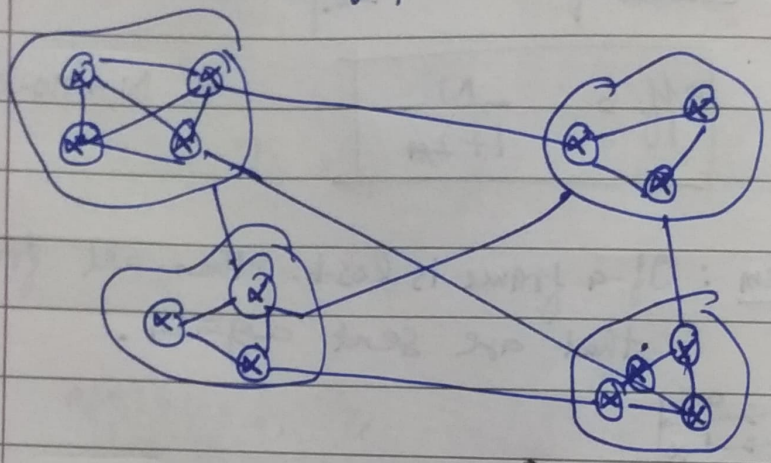
Page No.
Date

(II) Network layer (Internet Protocol) IP

Routing Protocols

(1) Exterior Gateway Protocol (Routing across autonomous system)

(2) Interior Gateway protocol (within autonomous system)



Which path to travel?

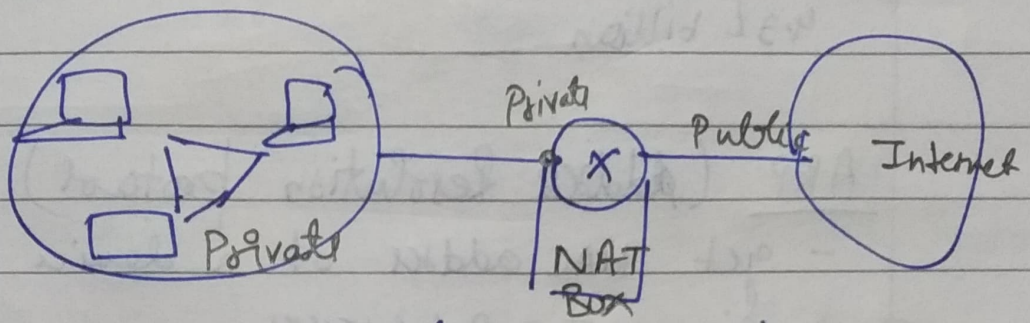
- Interior
  - Distance Vector
    - oldest (ARPAnet)
    - Bellman for algo
    - RIP (Routing Information Protocol)
    - (used more)
    - Problem: In case of failure (slow) count to ∞
  - Link State
    - Dijkstra
    - flooding (neighbour info)
- Exterior
  - Path Vector (BGP)

Page No. _____
Date

NAT (network Address Translation)

IP Address ⟨ Public
Private

- multiple IP Address is connected to single Public address



(do mapping)
- use TCP/UDP port no.(s)

10. 0. 0. 0 ←
10. 255. 255. 255 → broadcasting

Classfull Addressing

Class A	B	C	D	E
(1/2)	(1/4)	(1/8)	(1/16)	(1/16)
0 ——	10 ——	110 —	1110 ——	1111 ——
N/W | host	N/W | host			
8 | 24	16 | 16			
2^7 network	2^{24} host −2	2^{14} network	$2^{16}-2$ host		

Page No.
Date

IP Addressing

IPV4
(32 bit)
2^{32} possible addresses
4.3 billion

IPV6
(128 bit)

ARP (Address Resolution protocol)

- get MAC address of a device in LAN, given IP Addresses
- mapping IP Address → MAC Address

RRP (Reverse Addres Resolution protocol)

- MAC → IP

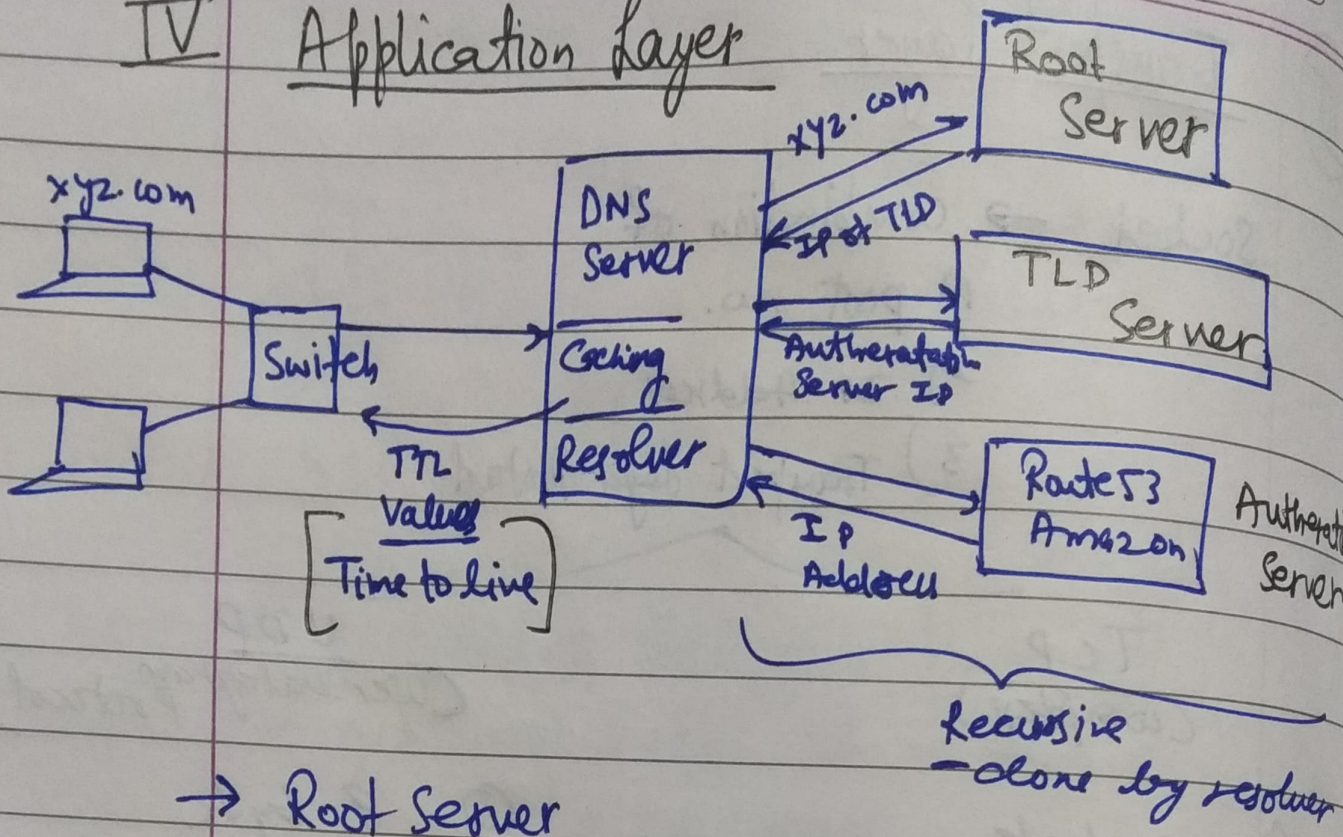
Page No. _____
Date

(III) Transport Layer

Socket → combination of
1) port no.
2) IP Address
3) Transport layer Protocol

TCP (complex)	UDP (User Datagram Protocol)
① 20 byte	① 8 byte
② Reliable	② not-guaranteed
③ error handling	③ Basic error correct
④ order of packet maintained	④ ∝
⑤ http, https, FTP, SMTP, SSH.	⑤ DNS, DHCP, online games.
	⑥ Broadcasting / Multicast

IV Application Layer



→ Root Server

use Any-Casting (Iterative Lookup)