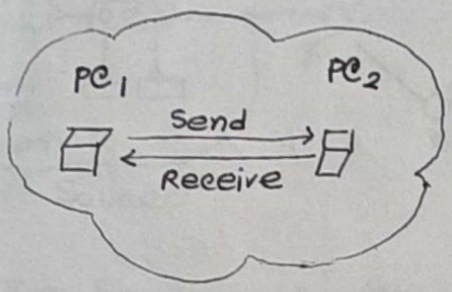
CSE405

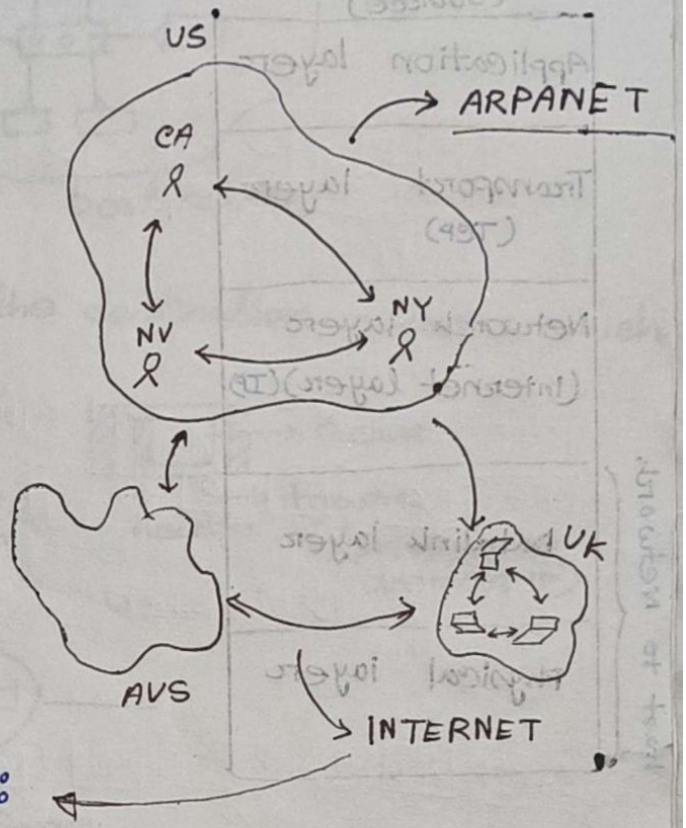
CamScanner

Computer Network

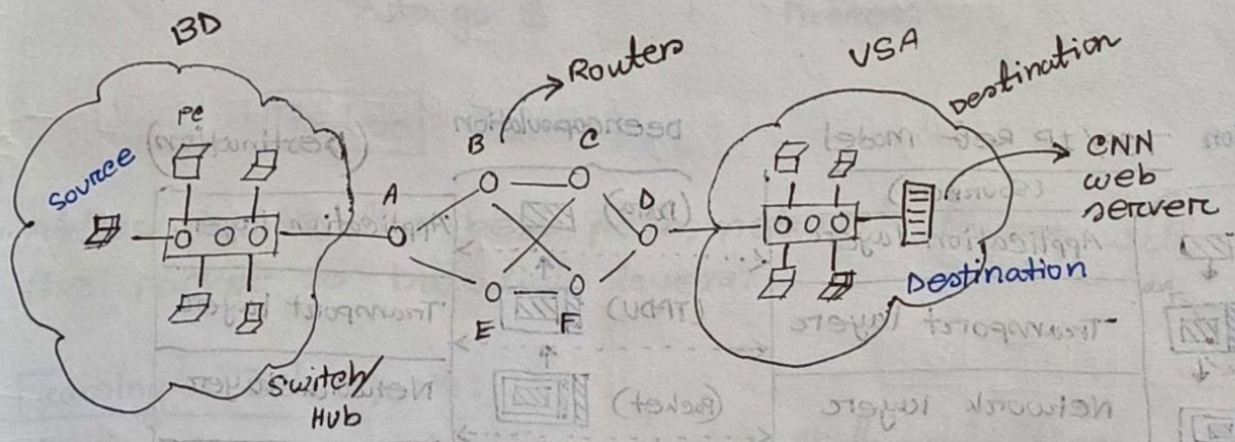
30.01.24 / CSE405
Tuesday / Ch-1

Two or more device can communicate each other in this network.





OSI Ref Model (Open system Inter connection):



OSI Ref Model

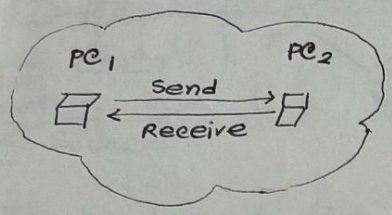
Application layer	→ All types of software debind করে।
Presentation layer	→ It present the syntex of the data.
Session layer	→ It maintains the session. মানে কার Data আগে পাঠাতে কারটা পরে। ভেঙ্গে ভেঙ্গে Data যাওয়ার এই Process ই session এর কাজ
Transport layer	→ মূল থেকে শেষ পর্যন্ত Data কে Supervice করে।
Network layer	→ Router হলো Network layer এর device.
Datalink layer	→ দুইটা Adjucent Device এর মধ্যে How to ~~communicate~~ stublish error free communication.
Physical layer	→ Hardware debind করে এই layer এ

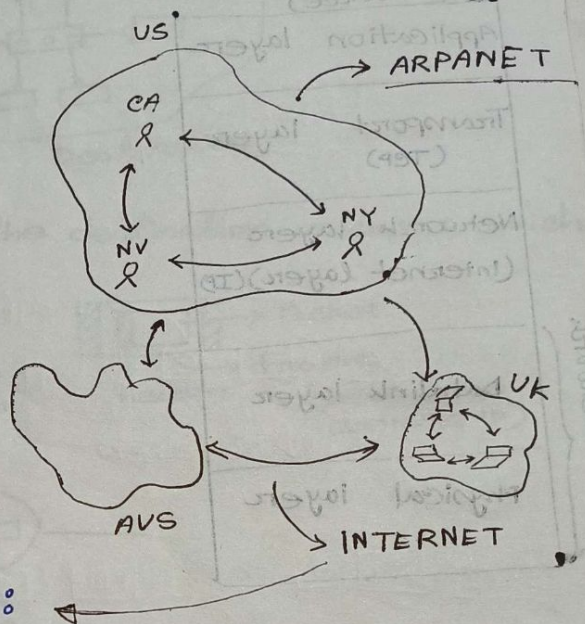
04.02.24 / CSE405
Sunday / Ch-2

Computer Network

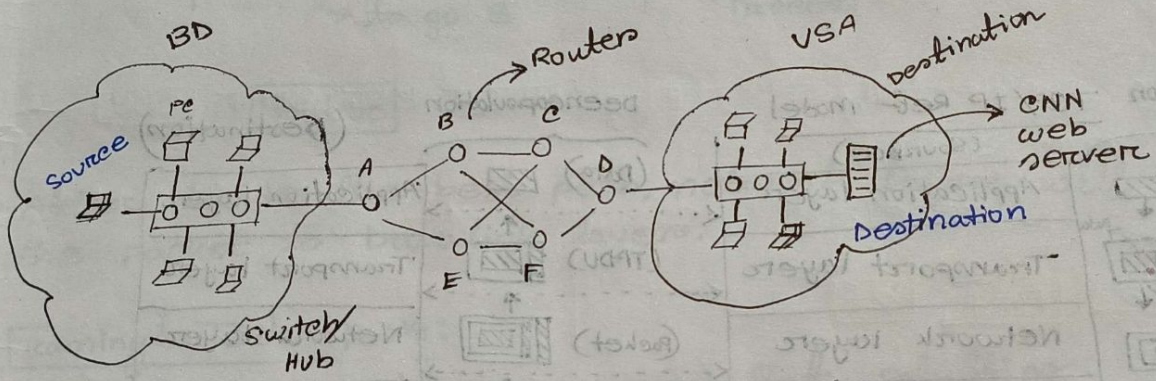
30.01.24 / CSE405 Tuesday Cls-1

Two or more device can communicate each other in this network.





OSI Ref Model (Open system Inter connection):

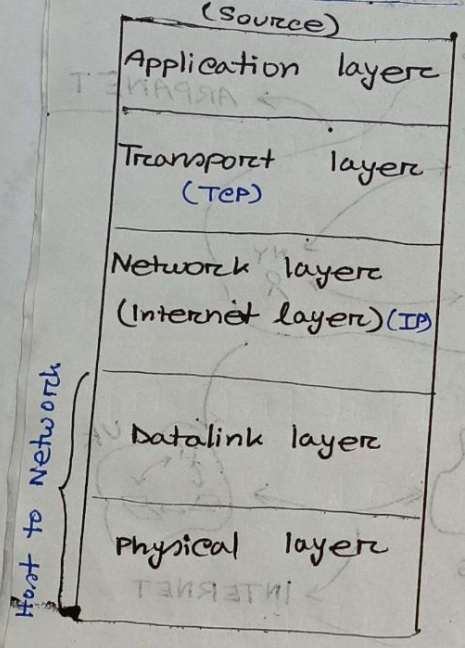


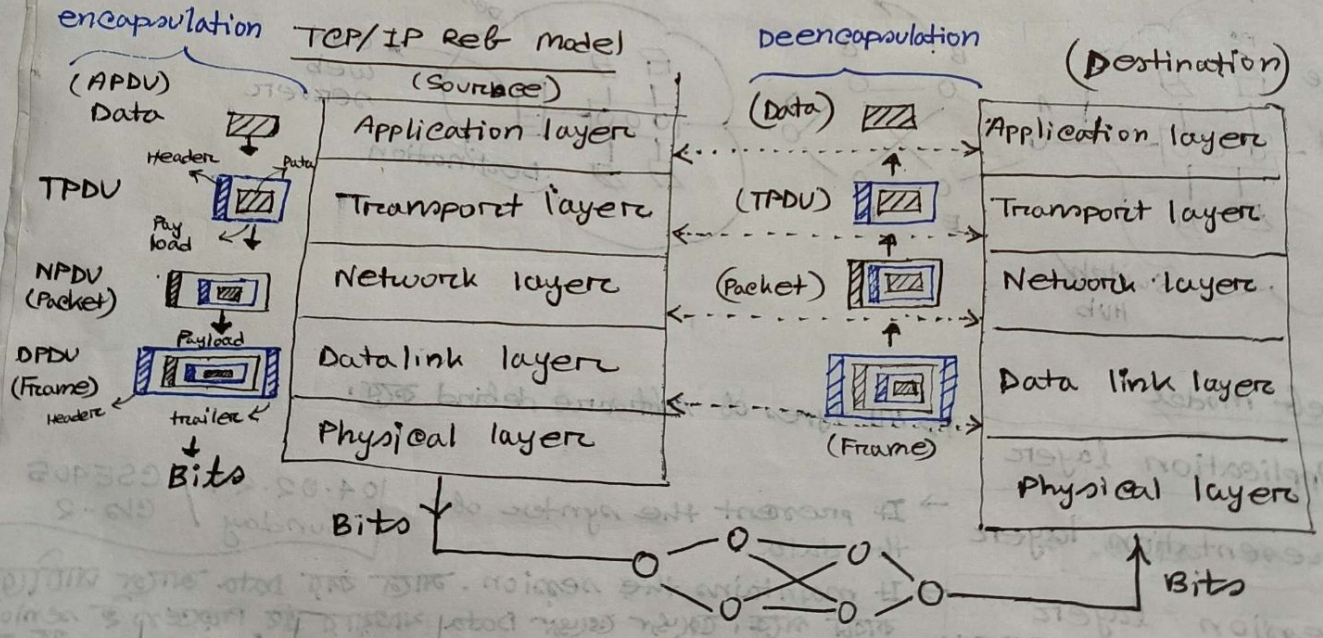
OSI Ref Model

Application layer	→ All types of software debind করে।
Presentation layer	→ It present the syntex of the data.
Session layer	→ It maintains the session. মানে কার Data আগে পাঠাতে কারটা পরে। ভেঙ্গে ভেঙ্গে Data যাওয়ার এই Process ই senion এর কাজ
Transport layer	→ শুরু থেকে শেষ পর্যন্ত Data কে Supervice করে।
Network layer	→ Router হলো Network layer এর device.
Datalink layer	→ দুইটা Adjucent Device এর মধ্যে How to ~~communicate~~ stublish error free communication.
Physical layer	→ Hardware debind করে এই layer এ

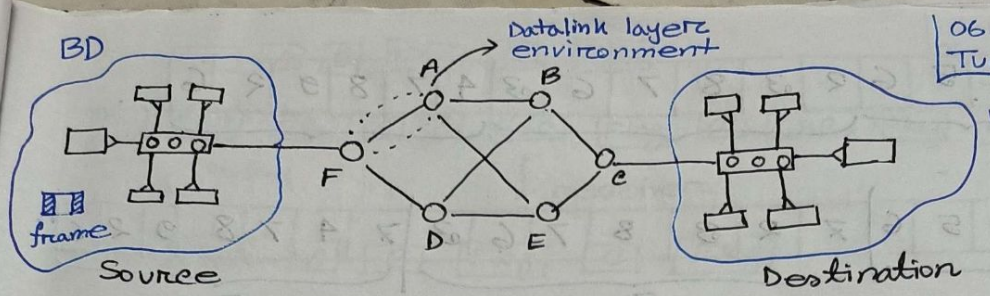
04.02.24 / CSE405 Sunday Cls-2

TCP/IP Ref. Model

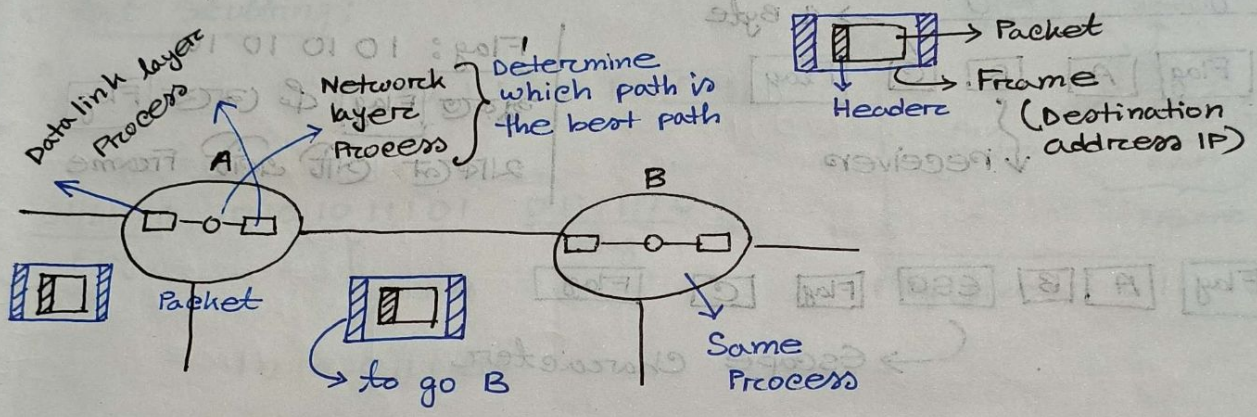




06.02.24/CSE405
Tuesday Ch - 3



→ The header of the packet has the destination address which is an ip address.



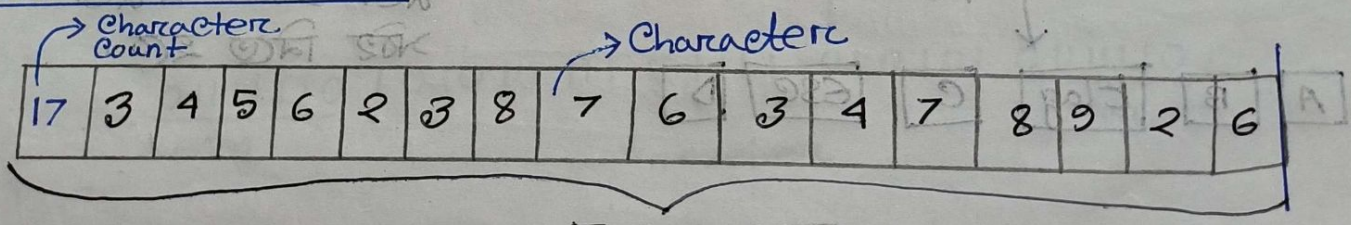
Router

→ After decide the best path, network layer will return the packet to Datalink layer.

Framing Techniques:

i) Character Count
ii) Byte Stuffing
iii) Bit stuffing
iv) Physical layer Coding violation

i) Character Count:



ATM (Asyncronas Transfer Mode)

Frame (cell) 53 byte (max)

Frame ↓ receiving

Ethernet LAN
→ Frame 1460 byte (max)

3	4	5	6	2	3	8	7	6	3	4	7	8	9	2	6

5	3	4	5	6	7	2	3	8	7	6	3	7	4	7	8	9	2

Fr_1 Fr_2 Fr_3

Byte Stubbing:

Flag | A | B | C → 1 Byte | Flag

↓ receiver

Flag: 10 10 10 10

শুরুতে Flag & শেষে Flag থাকলে সেটি একটি Frame

Flag | A | B | ESC | Flag | C | Flag

→ Escape character

→ যতবার accidentally মধ্যে Flag আসলে তার আগে ESC use করতে হবে।

↓ receiver

A | B | Flag | C

ESC: 00001111

Flag | A | B | ESC | Flag | C | ESC | ESC | D | Flag

↓ receiver

A | B | Flag | C | ESC | D

→ Black character গুলো Data. এগুলো সহি নিতে হবে।

* প্রতিটি Accidental ESC, Flag এর আগে একটি করে ESC, Flag বসাতে হবে।

13.02.24 / CSE405
Tuesday / Ch-4

[ESC] [A] [B] [Flag] [C] [Flag]

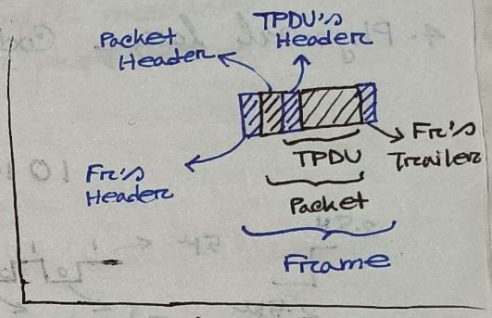
[Flag] [ESC] [ESC] [A] [B] [ESC] [Flag] [C] [ESC] [Flag] [Flag]

↳ stuffed

↓ receiver

[ESC] [A] [B] [Flag] [C] [Flag]

3. Bit stuffing:



Flag (fixed): 01111110

* [Flag] 1010 10111 01 [Flag]

01111110 (Flag) 1010 10111 01 01111110 (Flag)

↓ receiver

1010 10111 01

* 01111110 (Flag) 1010 10111 01111110 101 01111110 (Flag)

↓ receiver

1010 10111 | 01111110

Every 5 consecutive 1's, insert a "0" (Sender)
Receiver will omit a 0 after 5 consecutive 1's. (Receiver)

↓

01111110 (Flag) 1010 10111 01111110 101 01111110 (Flag)

↓ receiver

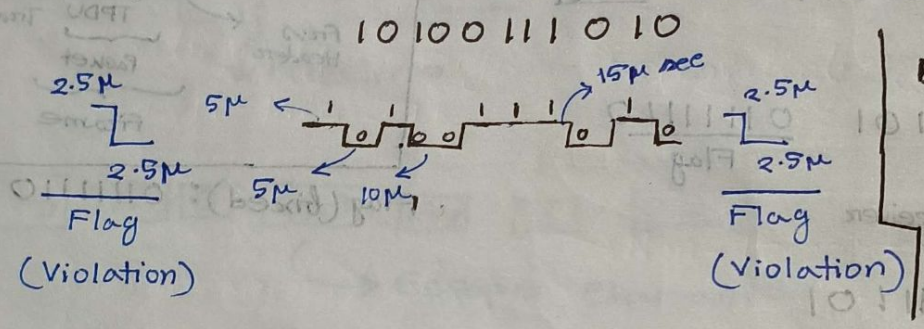
1010 10111 01111110 101 | 01111110 (Flag)

01111110 (Flag) 1010101110111110 10100101 01111110 (Flag)

receiver

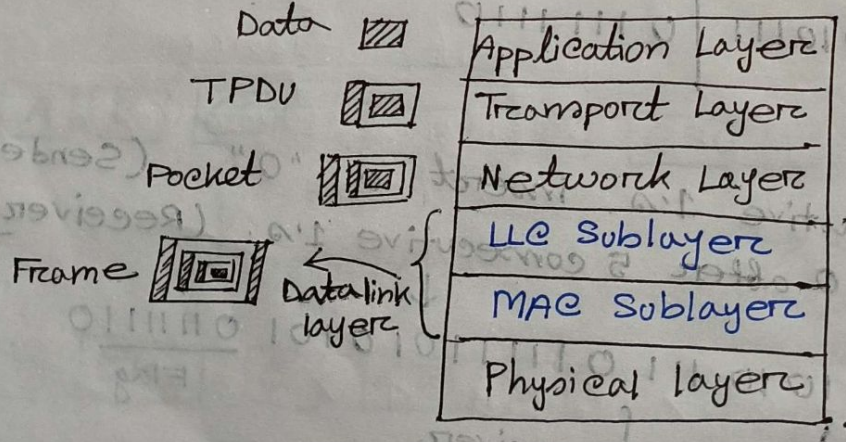
1010101110111101 0100101 | 01111110 (Flag)

4. Physical large Coding violation –



Bit time : 5 μ sec

Violation should be half of the bit time

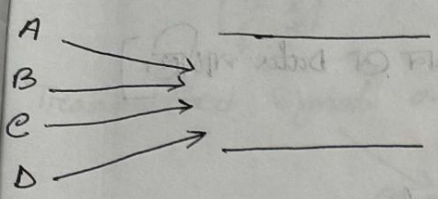


18.02.24 / CSE405
Sunday / Cls - 5

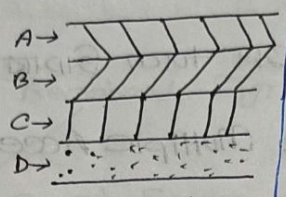
LLC: Logical link Control layer

MAC: Medium Access Control Sublayer

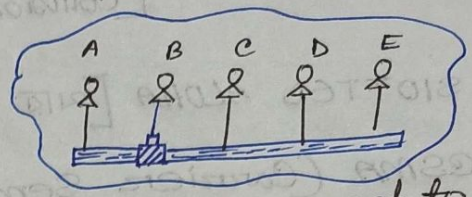
→ কে কখন পাঠাবে এটা Control করে।



Dynamic Allocation ✓



Static Allocation (Bandwidth কমে যাবে)

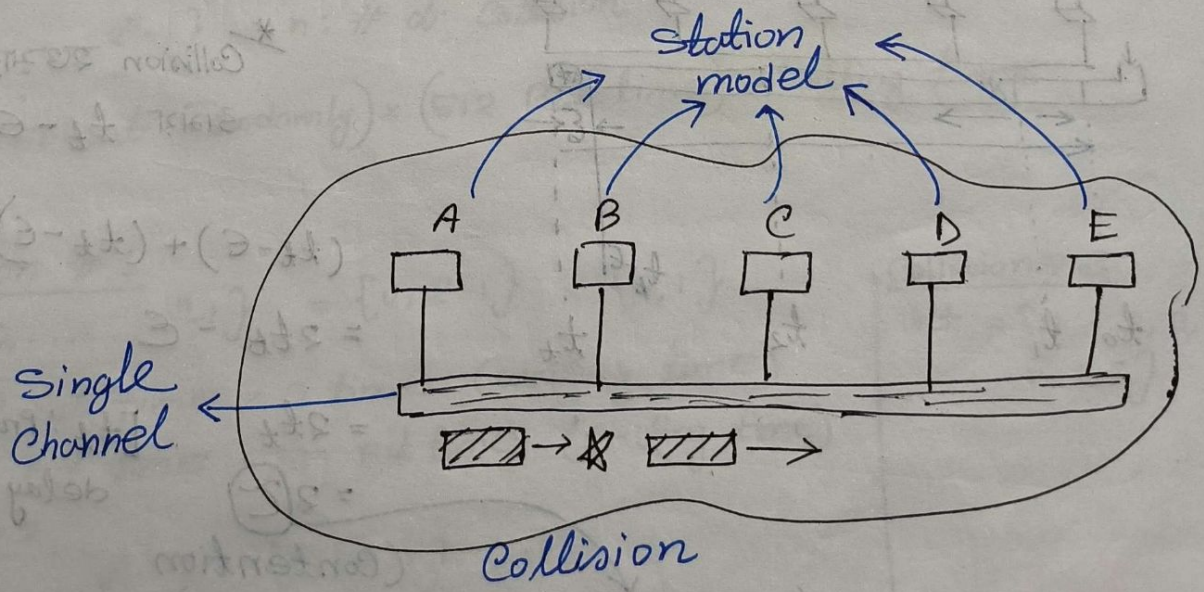


→ Everybody is connect to single medium → User dependend.

Dynamic Channel Allocation Protocals:

Assumptions:

1. Station Model (N number of independent nodes)
2. Single Channel
3. Collision
4. Time — Continious (if there is data then send it)
   — Slotted (wait for your specific turn)
5. Sense — Yes (Carrier or medium কে sense করার Capability)
   — No



Protocols:

- ALOHA → hello

[যখন যার data থাকবে then Send করে দিবে। একটির বেশি data থাকলে পাঠাবে then Collision হয়ে data নষ্ট হবে।]

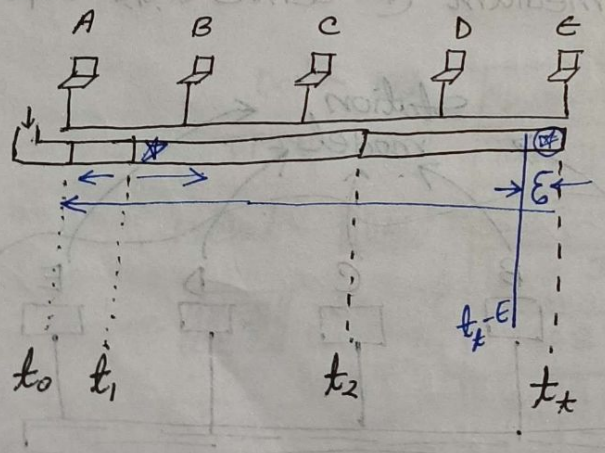
- SLOTTED ALOHA [যার যখন turn আসবে তখন সে Data পাঠাবে।]
- CSMA (Carrier Sense Multiple Access)

→ Carrier Sense → Idle → transmit
→ Busy → Refrain from transmission → * (Persistent) → **

Continiously Sensing কখন Idle হবে। ← * Persistent CSMA

Random amount of time পরপর Sense করবে। ← ** Non-Persistent CSMA

CSMA



* Collision হওয়ার শেষ সময় $t_t - \epsilon$

$(t_t - \epsilon) + (t_t - \epsilon)$

$= 2t_t - \epsilon$

$= 2t_t$ [t_t : Propagation delay (τ)]

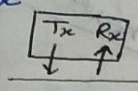
$= 2\tau$ (Contention Period)

(Between 2 farthest Nodes)

03.03.24 / CSE405
Sunday Cls-6

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

Half Duplex

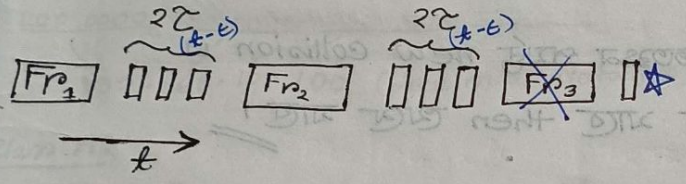


Transmitted Signal and Received Signal

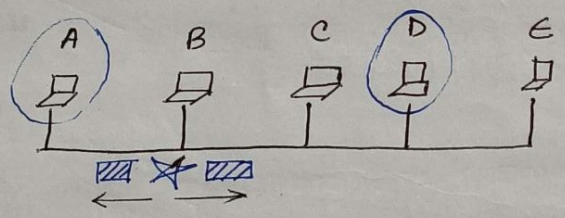
Compare ⇒ Same → No Collision

↓

Different → Collision



BEB (Binary Exponential Backoff) Algorithm –



Set = $\{0, \ldots, 2^n - 1\}$ n : # of Collision

(Pick an element Randomly) × (512 Bit time) = Waiting time

For A

Collision #1

Set = $\{0, \ldots, 2^n - 1\}$ = $\{0, 2^1 - 1\}$ = $\{0, 1\}$

0 × 512 Bit time = 0 time (waiting time)

1 × 512 Bit time = 512 Bit time (waiting time)

Collision #2

Set = $\{0, \ldots, 2^2 - 1\}$ = $\{0, 1, 2, 3\}$

0/1/2/3 × 512 Bit time = (waiting time)

Collision #3

Set = $\{0, 1, 2, 3, 4, 5, 6, 7\}$

For D

Collision #1

Set = {0, 1}

0 × 512 Bit time = 0 time (waiting time)

1 × 512 Bit time = 512 Bit time (waiting time)

50% next collition

যদি দুইটি Same Pick করে এবং Collision হয় Then again Pick করবে।

Collision #2

set = {0, 1, 2, 3}

0/1/2/3 × 512 Bit time = (waiting time)

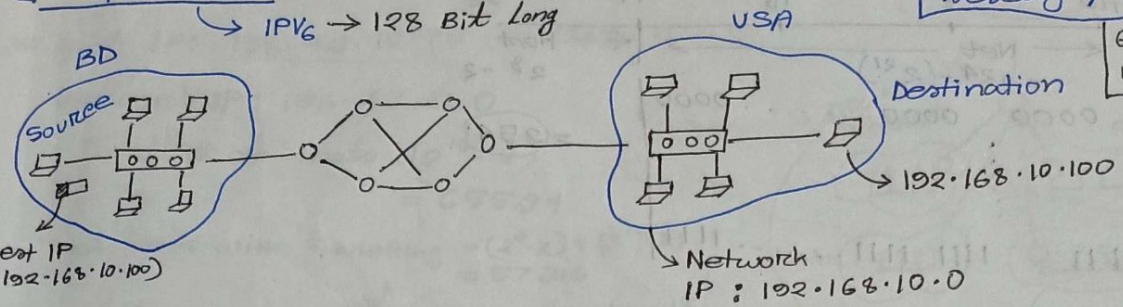
25% next collision

যতক্ষন পর্যন্ত Collision ~~চলতে~~ হবে ততক্ষন পর্যন্ত new collision এ যাবে এভাবে 10th Collision পর্যন্ত যাবে then থেমে যাবে।

05.03.23 / CSE405
Tuesday / Cls - 7

Quiz - 1
12.03.23

IPv4 Address: → 32 Bits long
→ IPv6 → 128 Bit Long



32 Bits long
⇓
2^{32}

যেহেতু Network IP 0 থেকে তাই আমরা device IP 1 থেকে শুরু করি।

(0) 0000 0000 ($2^8 = 256$)
⋮
(255) 1111 1111

1100 0000 ....
→ 192 . 168 . 10 . 100 (decimal Notation)

Class A:

0 — 8 — 16 — 24 — 32 (0.0.0.0 – 127.255.255.255)

Net → $= 2^8$ (2^7)
← Host $= (2^{24} - 2)$

(0) 0000 0000 | 0000 0000
⋮
(127) 0111 1111 | 1111 1111

Class B:

10 — 8 — 16 — 24 — 32 (128.0.0.0 – 191.255.255.255)

(128) 1000 0000 Net → 0000 0000 | ← Host →
⋮ $= 2^{16}$ = ~~2(14)~~ (2^{14}) | ($2^{16} - 2$)
(191) 1011 1111 1111 1111 | 1111 1111

Class C: (192.0.0.0 – 223.255.255.255)

110 | 8 | 16 | 24 | 32

← Net → | ← Host

$= 2^{24} = (2^{21})$

$2^8 - 2$

$= 254$

(192) 1100 0000 0000 0000

⋮

(223) 110 1111 1111 1111 1111

Class D: (224.0.0.0 – 239.255.255.255) → Multi Casting

Class E: (240.0.0.0 – 255.255.255.255) → Future Use

#

Dest IP: $\underbrace{192.168.10}_{\text{Network}}.\underbrace{100}_{\text{Host}}$ → 0000 0000 (0)

Class : C

Network IP Address: 192.168.10.0

Broadcast IP Address: 192.168.10.255

Subnet Mask IP Address:

225.255.255.0

↓

(Network গুলো 1111.... 1111 করে ফেলে এবং Host গুলোকে 0000.... 0000 করে ফেলে)

192.168.10.0 → Network

192.168.10.1 } Device

192.168.10.255 → Broadcast

* Dest IP: $\underbrace{156.45}_{\text{Net}}.\underbrace{15.35}_{\text{Host}}$

Class : B

Network IP Address: 156.45.0.0

Broadcast IP Address: 156.45.255.255

Subnet Mask IP : 255.255.0.0

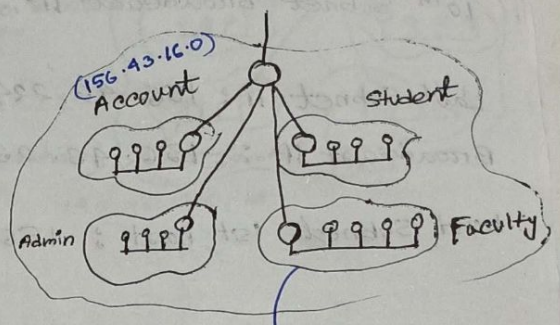
10.03.24/CSE405
Sunday Ch-8

Dest IP: 156.43.10.75 Class: B

Network IP: 156.43.10.0

Total # of hosts: $(2^{16}-2)$
$= 65534$

Total hosts with Subnetting $= (2^4-2) \times (2^{12}-2)$
$= 57316$



Subnet
Net IP: 156.43.0.0
Broadcast IP: 156.43.255.255

Subnet:

156	43	Sub $=2^4$	Host $=2^{12}$	
← Net →		0000	0000	00000000
		0000	0000	00000000
		0001	0000	00000000
		0010	0000	00000000
		1111		

× 1st subnet:
1st subnet ↔ 2nd subnet:
2nd Subnet:
last subnet:

1st Subnet IP: 156.43.0.0 × এই Number কে use করা যাবে না।

∴ 1st Subnet IP: 156.43.16.0

∴ 2nd Subnet IP: 156.43.32.0

Subnet mask

156	43	Sub $=2^4$	Host $=2^{12}$		
← Net →		0000	0000	00000000	
1st Subnet:		0001	0000	00000000	→ 1st Host ×
			0000	00000001	→ 1st host
		0001	1111	11111110	→ Last host
1st Subnet:		0001	1111	11111111	→ Broadcast
10th Subnet:		1010	0000	00000000	
		1010	1111	11111111	→ Broadcast
Last Subnet:		× 1111	0000	00000000	

1st Subnet 1st host IP: 156.43.16.0 ×
156.43.16.1

1st Subnet Last host IP: 156.43.16.254

1st Subnet Broadcast IP: 156.43.31.255

Last Subnet: 1110 | 0000 00000000

Last Broadcast: 1110 | 1111 11111111

∴ 10th Subnet IP: 156.43.160.0

10th Subnet Broadcast IP:

Last Subnet IP: 156.43.224.0

Broadcast IP : 156.43.239.255

∴ Last Subnet 1st host : 156.43.224.1

Example

205.16.15.10

Class: C

Network IP: 205.16.15.0

			Subnet	Host
205	16	15	000	00000
		1st subnet →	001	00001 → 1st host
		2nd subnet →	010	11111 → Broadcast
1st subnet		111..11	111	00000 → Subnet mask IP

subnet mask

1st subnet 1st host IP: 205.16.15.33

2nd subnet Broadcast: 205.16.15.95

Subnet mask IP: 255.255.255.224 (decimal notation)

205.16.15.10/27 (CIDR notation)

Ex

146.25.156.14 → 4 Bits to Creat Subnet

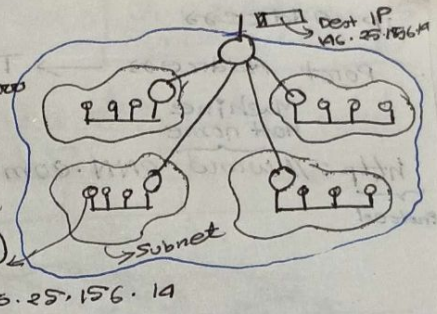
146.25 0000 | 0000 0000000

11.11.11.11 1111 | 0000 (CIDR)

Class B

Subnet Mask IP: 146.25.156.14/20 (CIDR)

255.255.240.0 (Decimal Notation)



AND

Dest IP: 146.25.156.14

Subnet Mask IP: 255.255.240.0

10010010	00011001	10011100	00001110
11111111	11111111	11110000	0000000
10010010	00011001	1001 | 0000	00000000
(146)	(25)	(144)	(0)

9th Subnet

Subnet IP: 146.25.144.0

Types of IP:

Private IP:

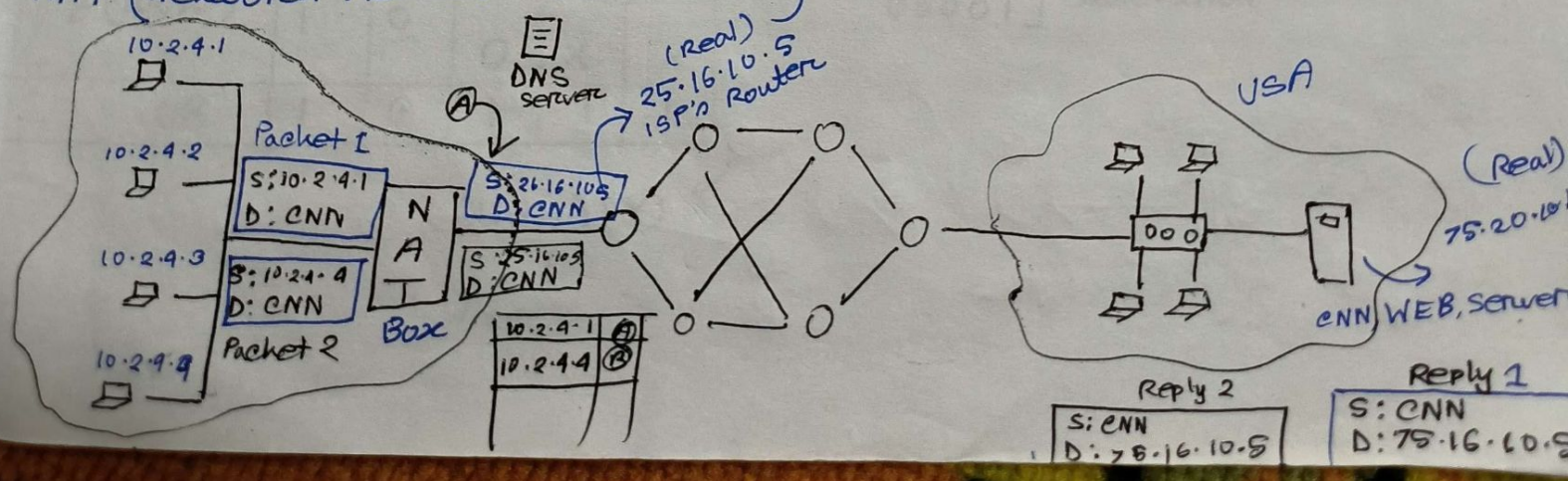
Class A: 10.0.0.0 – 10.255.255.255 ($>16\times10^6$)

Class B: 172.16.0.0 – 172.31.255.255 ($>1\times10^6$)

Class C: 192.168.0.0 – 192.168.255.255 ($>65\times10^3$)

(Public IP) Real IP: (Rest of them)

NAT (Network Address Translation) –



12.03.24 / CSE405
Tuesday / Ch-9

IP Address

Port Address → TPDU Header.

Machine
Host name

http : // www. CNN. com / filename

Protocol

16 Bit

Source Port	Dest Port
000000 000110 ← 10 Bits →	

(A) → 000000 000110

2^{16}

TPDU Header

http → 80
Telnet → 23
Ftp → 21
SMTP → 110

0 - 1023 (Reserved)
Rest of them (free to use)

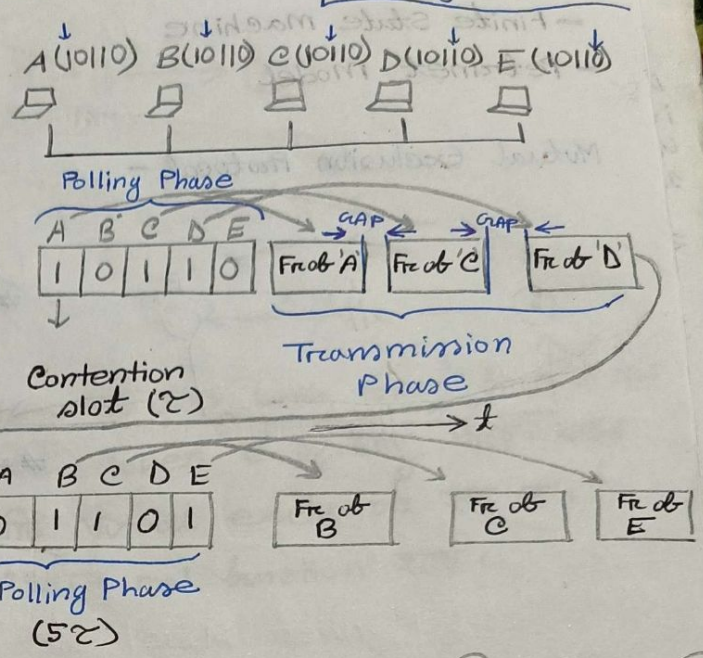
Channel Allocation

Final Start

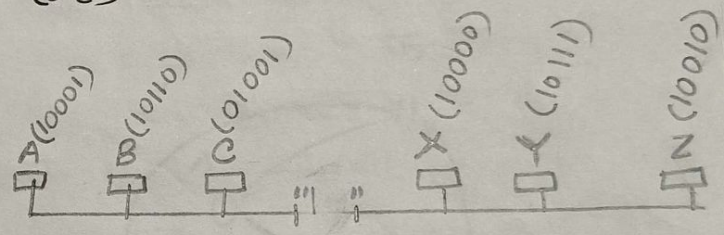
31.03.24 / CSE405
Sunday / Cls-10

Collision Free Protocols:

i) A Bit Map Protocal



ii) Binary Countdown Protocal –



Bit length: 5 Bits long

$\boxed{2^5 = 32}$

A(10001)	1	0	0	x	x
B(10110)	1	0	1	1	0
C(01001)	0	x	x	x	x
X(10000)	1	0	0	x	x
Y(10111)	1	0	1	1	1
Z(10010)	1	0	0	x	x
OR	1	0	1	1	1

(10111) (Y)
↓
Winner

** OR এর সাথে Bit match করলে Continue করবে otherwise out.

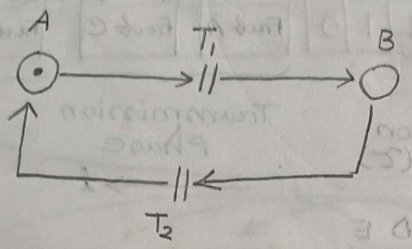
একবার Winner পাওয়ার পরে আবার Same process follow করবে। এবং Previous winner আবার Participate করতে পারবে।

00000 } Starvation
00001

Protocol verification –

- Finite State machine
- Petrinet model

Mutual Exclusion Protocol –



petrinet model notion

O → State without a token

⊙ → State with a token

→|→ → Transition

→ → Arc

#* • → Token না থাকলে Transition Possible না / incomplete model.

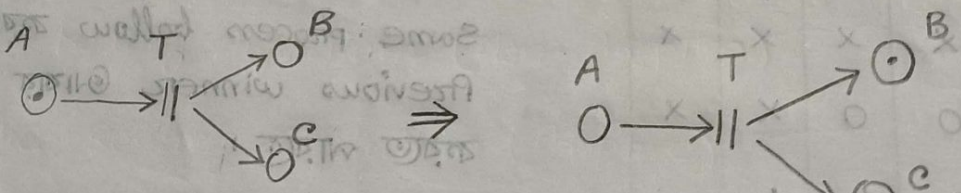
#* Token আবার আগের জায়গায় আসলে Cycle এর মাধ্যমে তাহলে complete model হবে।

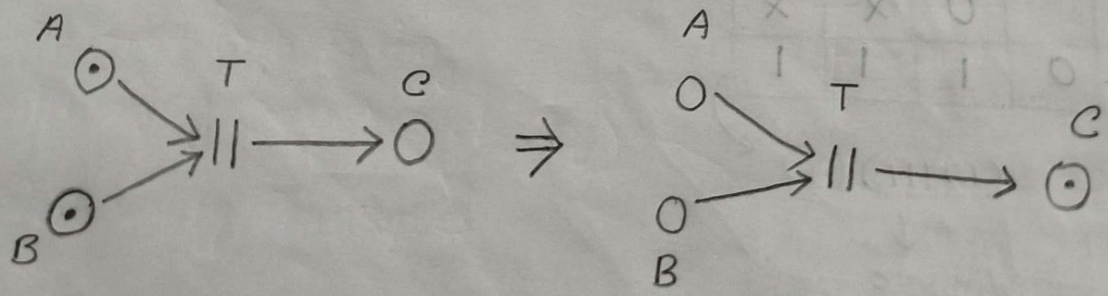
Petrinet model

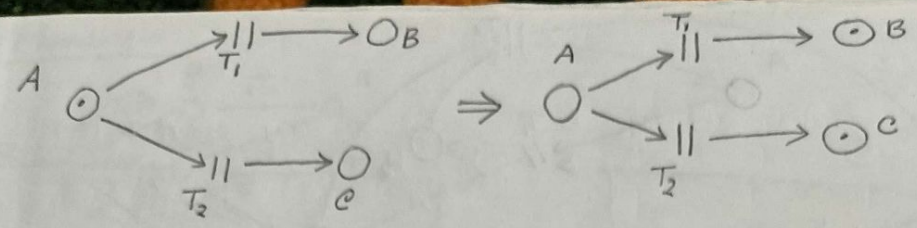
Mutual Exclusion Protocol –

02.04.24 / CSE405 Ch- 11 Tuesday

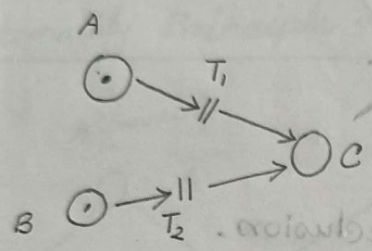
Rules:



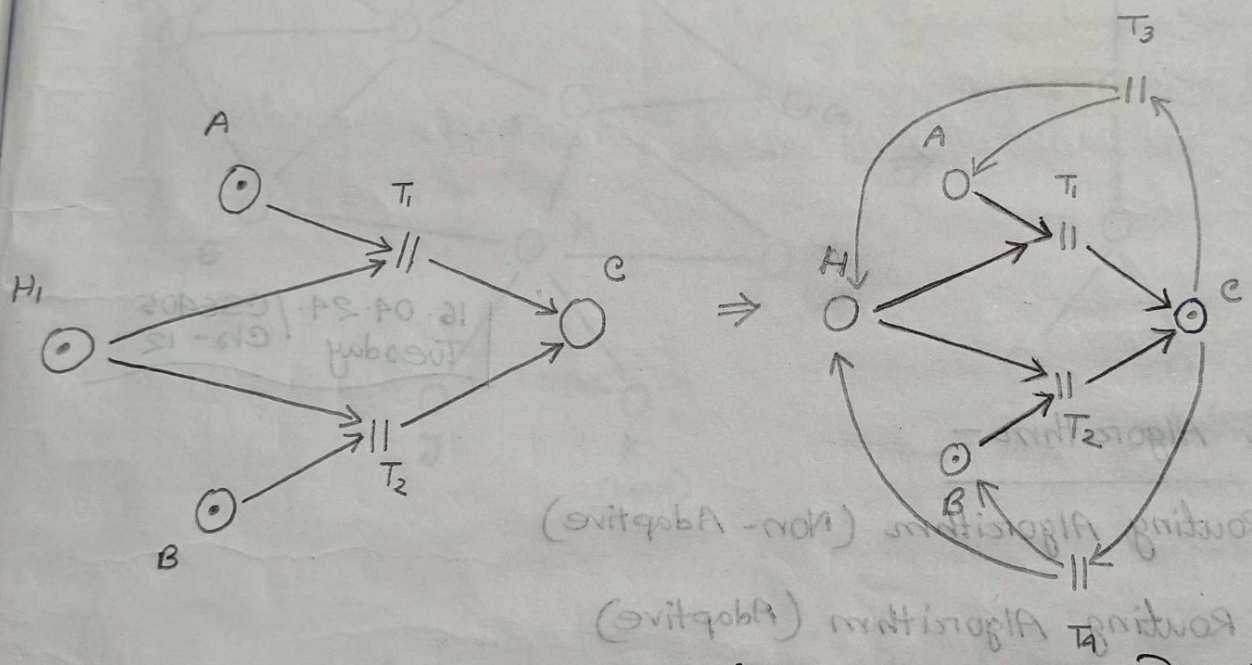




Mutual exclusion-

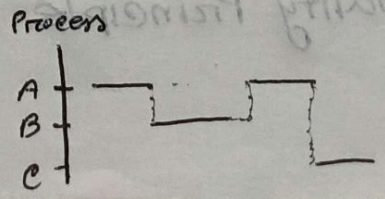


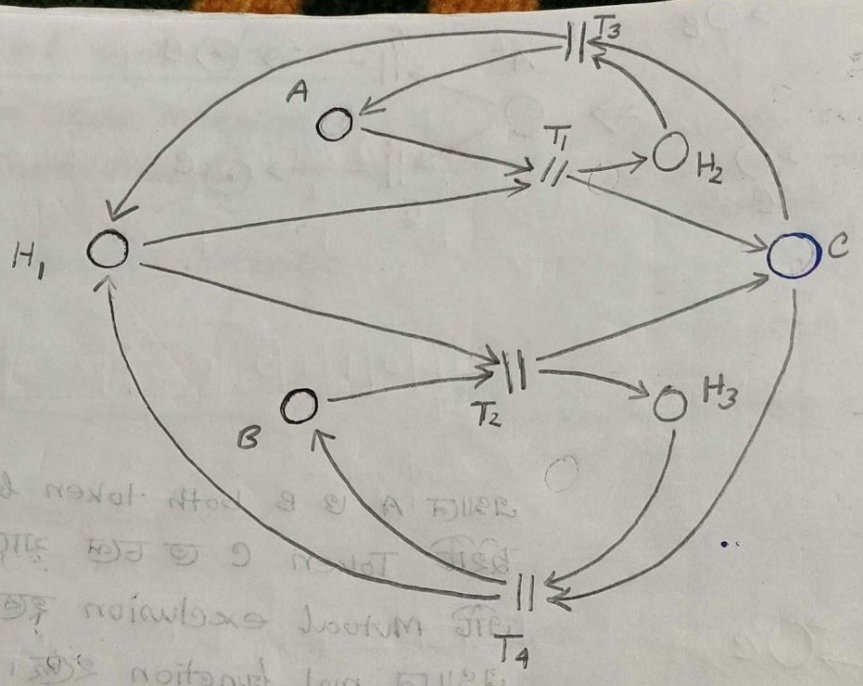
এখানে A ও B both token fire করলে দুইটি Token C তে চলে যাবে। ফলে এটি Mutual exclusion হবে না। এখানে mal function হচ্ছে।



এখানে A এর সাথে H এর Token fire করতে হবে। কিন্তু B যখন Token পাঠাবে তখন H এর কাছে Token থাকবে না তাই B execute হবে না। এটি Mutual exclusion.

Process যেহেতু Parallal way তে হতে পারে না তাই A ও B এক সাথে Token fire করতে পারবে না।





Petrinet Model of Mutual Exclusion.

16.04.24 / CSE405
Tuesday / Ch-12

Routing Algorithm -

- Static Routing Algorithm (Non-Adaptive)
- Dynamic Routing Algorithm (Adaptive)

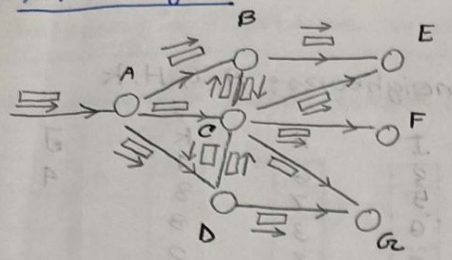
Static

- → Shortest Path Routing
- → Flooding (Selective flooding)
- → Optimality Principle

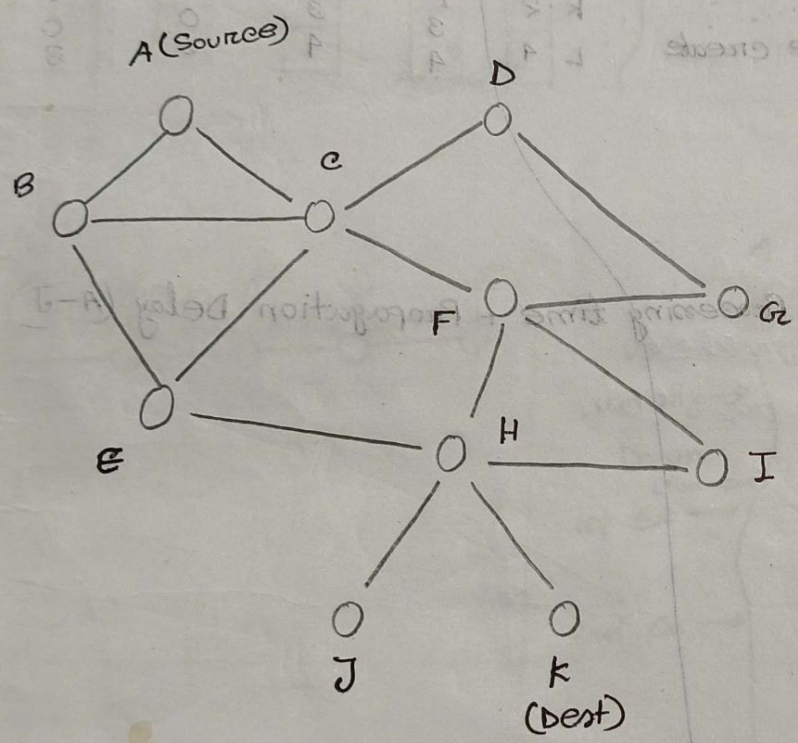
Dynamic

- → Distance vector Routing Algo
- → Link-State Routing Algo

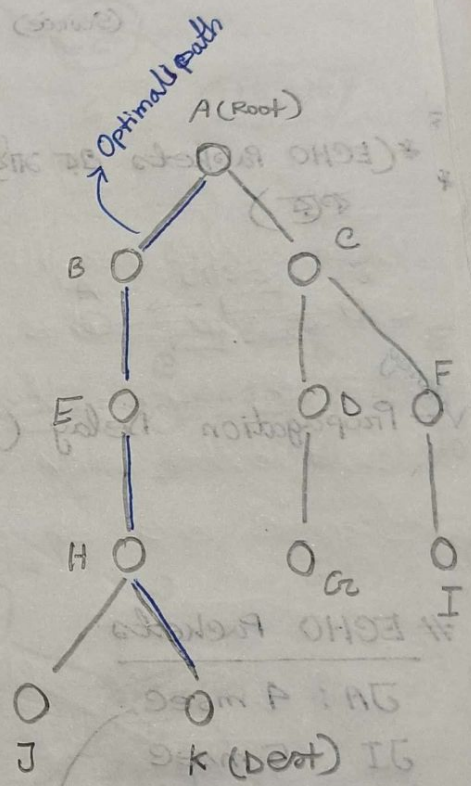
Flooding –



Optimality Principle –

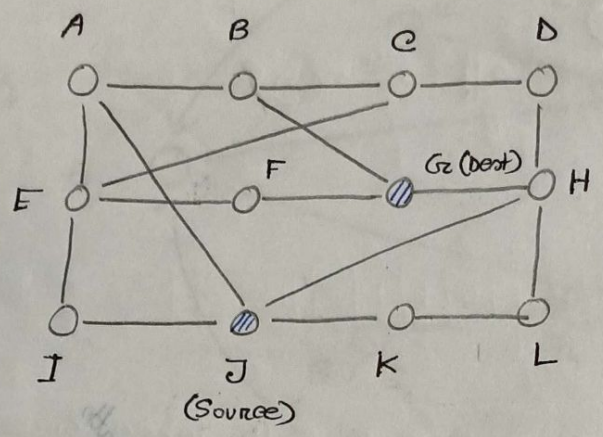


⇒



* যত কম Node traverse করবে তত Optimal হবে। Deapth কম Width বেশি tree বানাতে হবে।

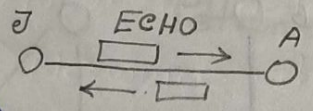
Distance vector Routing Algorithm –



J's neighbors: A, I, H, K

to	A	I	H	K	J
A	0	2	6	4	4
B	3	5	7	8	
C	6	6	3	8	
D	7	3	8	2	
E	3	8	9	1	
F	5	9	2	5	
G	8	2	5	8	7
H	9	8	0	6	6
I	3	0	2	9	5
J	6	7	5	8	0
K	2	3	3	0	3
L	4	4	4	3	

*(ECHO Packets এর মাধ্যমে table create করে)



Latency = Propagation Delay (J–A) + Processing time + Propagation Delay (A–J)

ECHO Packets

JA : 4 msec
JI : 5 msec
JH : 6 msec
JK : 3 msec

JG : JA + AG = 4 msec + 8 msec = 12 msec

JG : JI + IG = 5 msec + 2 msec = 7 msec

JG : JH + HG = 6 msec + 5 msec = 11 msec

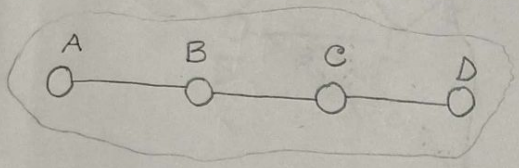
JG : JK + KG = 3 msec + 8 msec = 11 msec

Problem → (Count-to-infinity Problem)

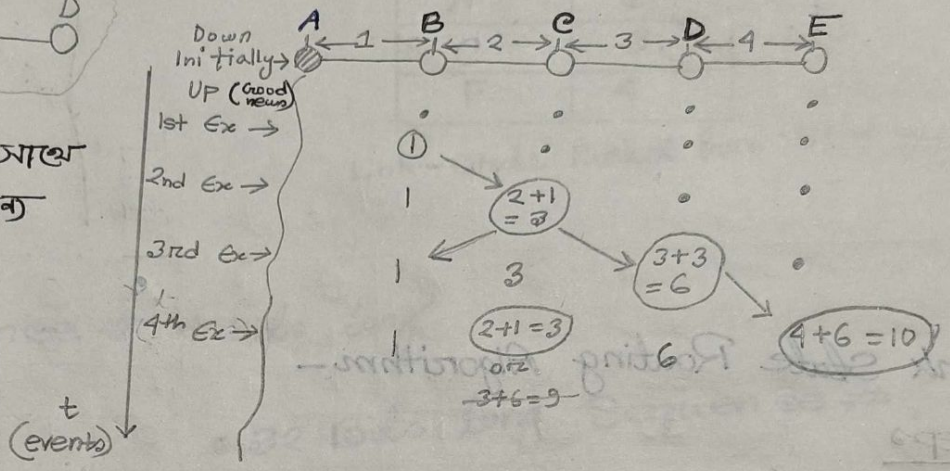
Problem in Distance vector Routing Algorithm

21.04.24 / CSE405 Sunday Cls-13

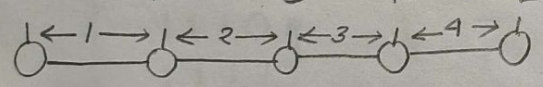
May occure if there exist a "Linear Subnet."

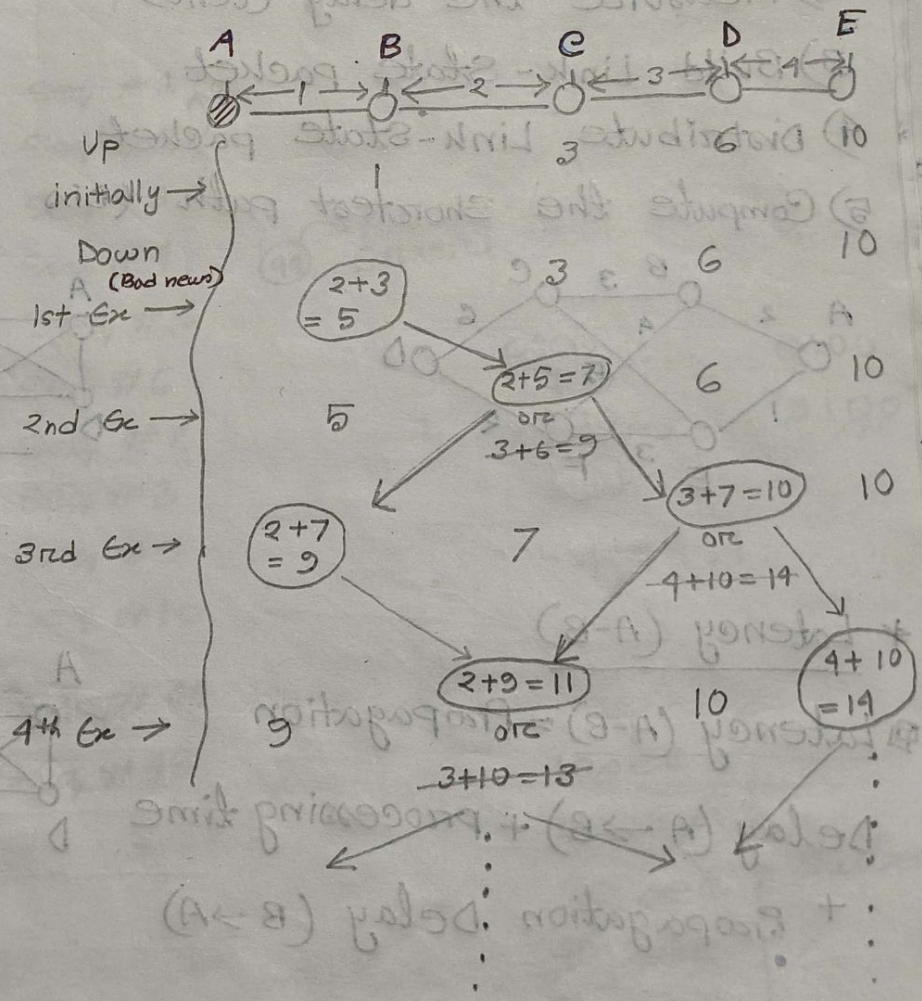


Router up হওয়ার সাথে সাথে Hello packet পাঠাবে অন্য Router দেরকে।

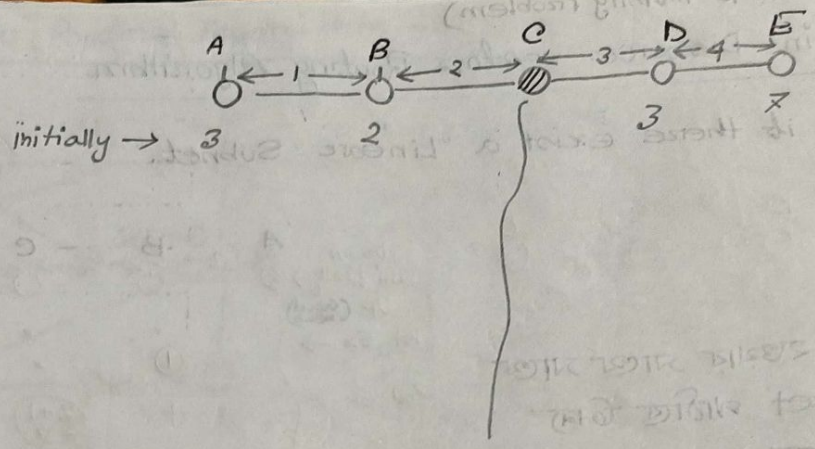


A B C D E





Count-to-infinity

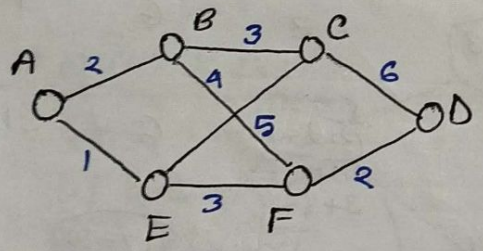


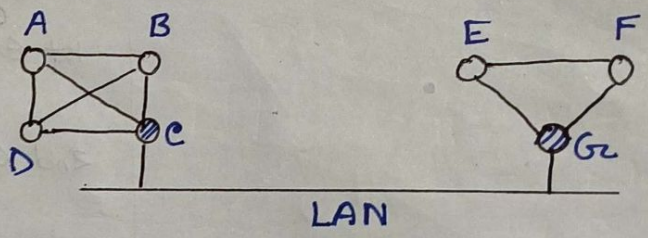
23.04.24 / CSE405
Tuesday / Cls - 14

Link State Routing Algorithm –

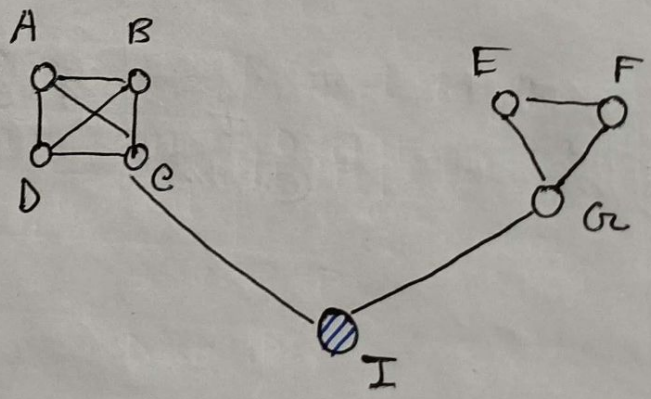
* Steps

1) Know the neighbours (HELLO)
2) Measure the delay (ECHO)
3) Build Link-State packet
4) Distribute Link-State packet
5) Compute the shortest path (Dis





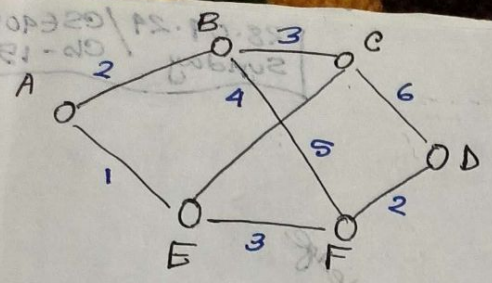
⇓



(Imaginary Node)

* Latency (A-B)

Latency (A-B) = Propagation Delay (A → B) + processing time + Propagation Delay (B → A)



সেটার info বড় সেটা routing table এ যাবে।

(x+1), (x)

'B'	
Sequence #	
Age	
A	2
C	3
F	4

60 sec Link-state Packet for 'B'

Link-State Packet for 'B'

- length of Sequence # 4 bits long

0000
0001
0010
0011
⋮
1110
1111
0000
0010
⋮
1111

- 32 Bits long Sequence #

$2^{32} \rightarrow$ 100 years

precisely 122 years

Seq #1
Seq #2
Seq #3
Seq #4
Seq #5
Seq #6
Seq #7
Seq #8
Seq #9
Seq #10
Seq #44
Seq #45

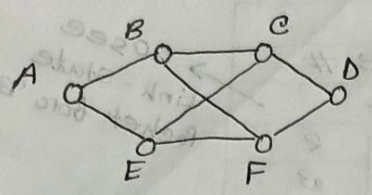
44 → Read Error

0000 0100
↓
1000 0100
↓
444444 .....

10 sec
20 sec
30 sec
40 sec
50 sec
60 sec
} Subtract

28.09.24 / CSE405 Cls-15
Sunday

Distribute Link-state Packet

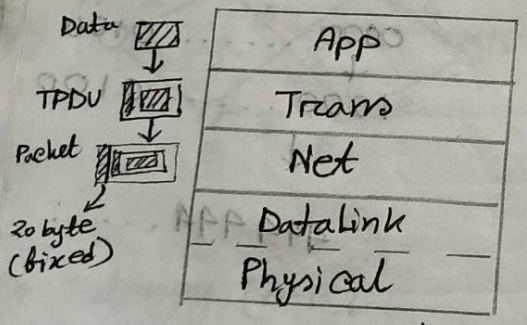


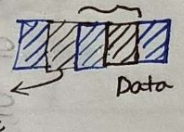
For 'B':

Acknowledge

	Owner	Source	Seq #	Age	Packet Sent A	Packet Sent C	Packet Sent F	Ack Sent A	Ack Sent C	Ack Sent F
	A	A	36	60	0	1	1	1	0	0
	E	A	56	60	0	1	1	1	0	0
	C	F	25	60	1	0	0	0	1	1
×	A	C	21	60						

TCP/IP Ref model





Throughput
↳ কতটুকু Data পাঠানো যাচ্ছে per sec

DF: Don't Fragment
MF: More Fragment

Delay, Throughput, Reliability

3 Bits	D	T	R

Precedence (Priority)

000 (Normal Data Packet)

⋮

111 (Controlling Packet)

20 Byte

Time-to-live

Version

Internet Header Length

16 Bits

Vers	IHL	Type of Service	Total length
Identification			DF / MF / Fragment offset
TTL (128)	Protocol	Header Checksum	
Source IP			
Dest IP			
Options			

IPV4 Header

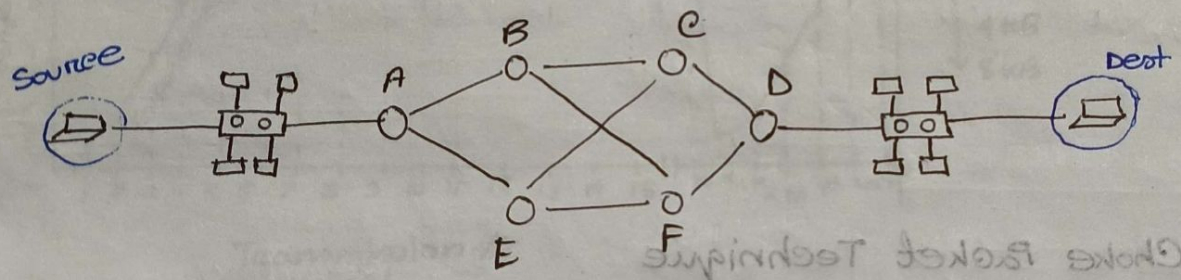
Tells you the Position of the fragment in the packet

Ethernet: 1460 Byte
ATM: 53 Byte

TTL এ Device number দেয়া থাকে। যে কতগুলো Device cross করতে পারবে।

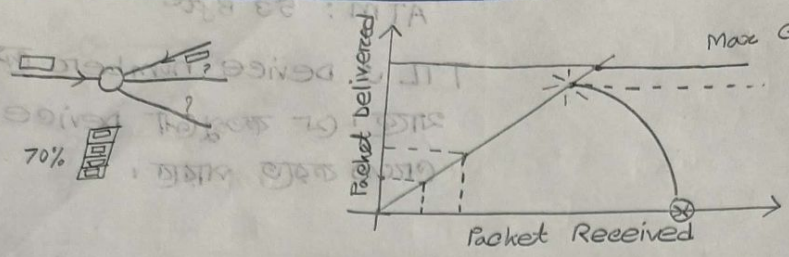
Options -

1) Security → (How secret the packet is)
2) Strict Source Routing → (Gives you the dedicated path from Source to Dest)
3) loose Source Routing → (Gives you the list of routers that can't be missed)
4) Record Route
5) Record Time

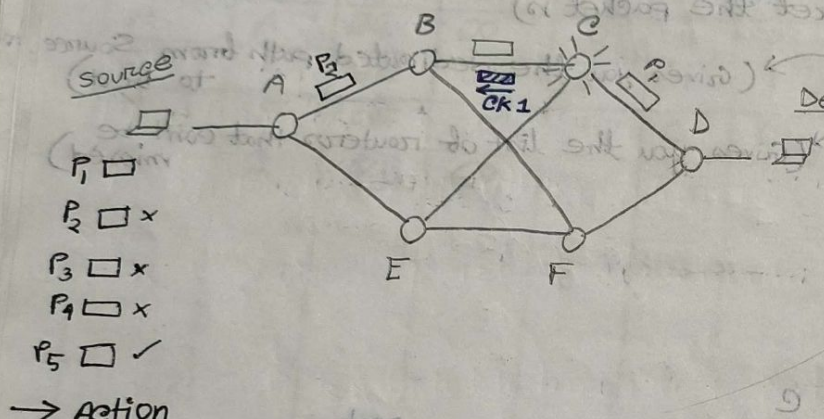


30.04.24 / CSE
Tuesday

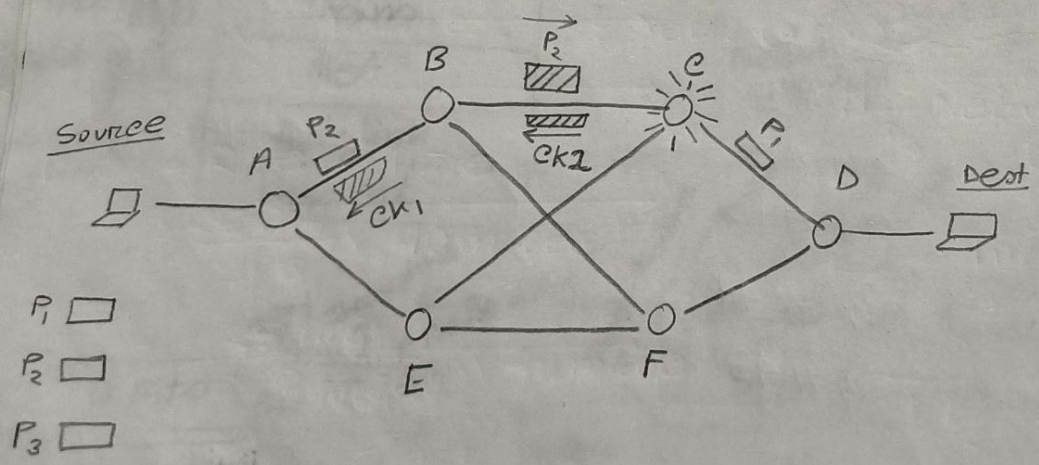
Congestion Control Algorithm –



– Choke Packet Technique –



Hop - by - hop Choke Packet Technique



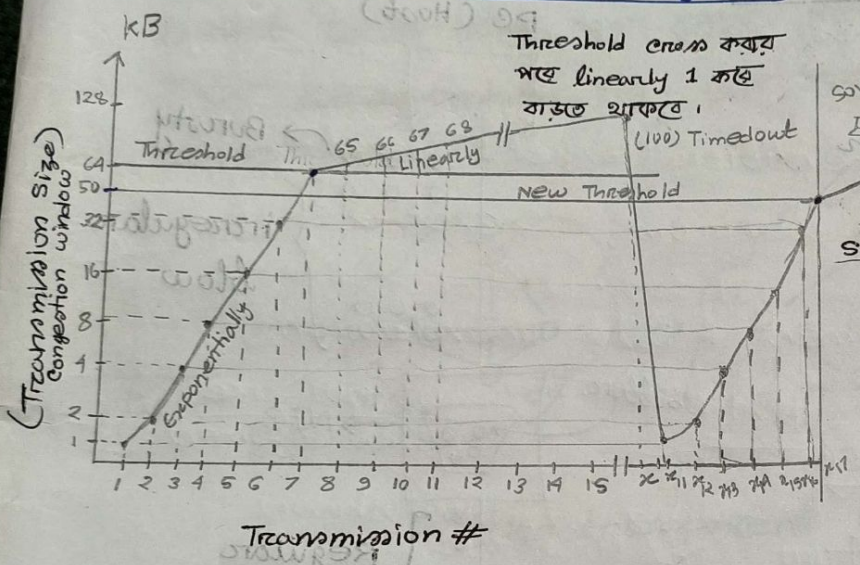
Load Sheding

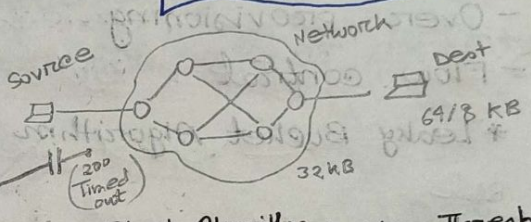
a) Milk Technique (Fresh is good)

b) Wine Technique (old is good)

Congestion Control in Transport Layer -

05.05.24 / CSE405 Sunday / Ch-17





Slow Start Algorithm

1 kB
2 kB
4 kB
8 kB

New Threshold

$= \frac{1}{2} \times$ Timeout

$= \frac{1}{2} \times 100$

$= 50$ kB

যদি নির্দিষ্ট সময়ের মধ্যে Ack না আসে তাহলে আবার Pack send করতে হবে।

1 kB থেকে Transmission শুরু হবে এবং Successfull হলে Transmission এর kB বাড়াবে

* Timeout হয়ে গেলে Direct পরের step এ 1 kB তে Drop করবে এবং New Threshold set করবে যেটা Previous ~~threshold~~ Timedout এর half.

Jitter: Variations in Packets arrival time.

Scenario 1

Source — Delay → Dest

P_1 — 9 msec → P_1

P_2 — 10 msec → P_2

P_3 — 8 msec → P_3

P_4 — 15 msec → P_4

P_5 — 20 msec → P_5

Delay → low

Variation → High

⇓

High Jitter

Scenario - 2

Source — Delay → Dest

P_1 — 90 msec → P_1

P_2 — 89 msec → P_2

P_3 — 90 msec → P_3

P_4 — 91 msec → P_4

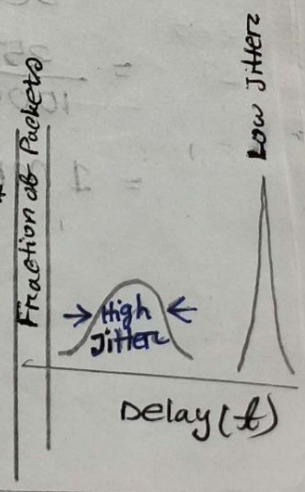
P_5 — 89 msec → P_5

Delay → High

Variation → low

⇓

Low Jitter



Quality of Services (QOS)

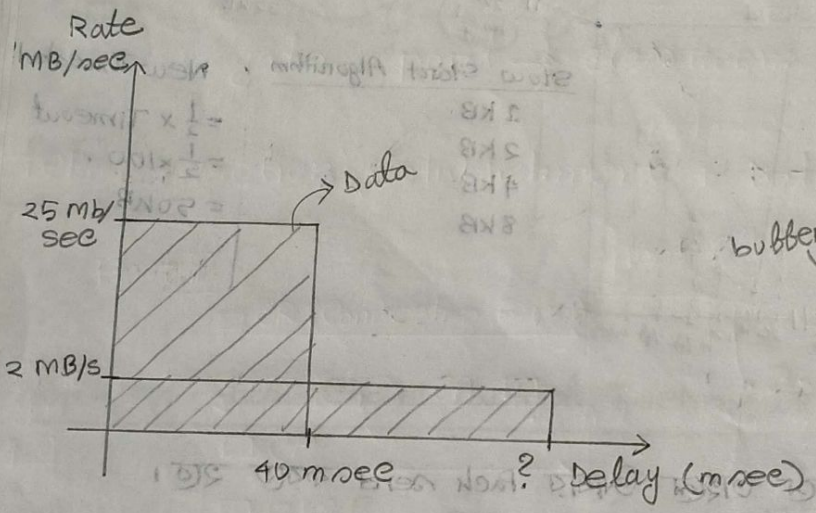
Importance Matrix

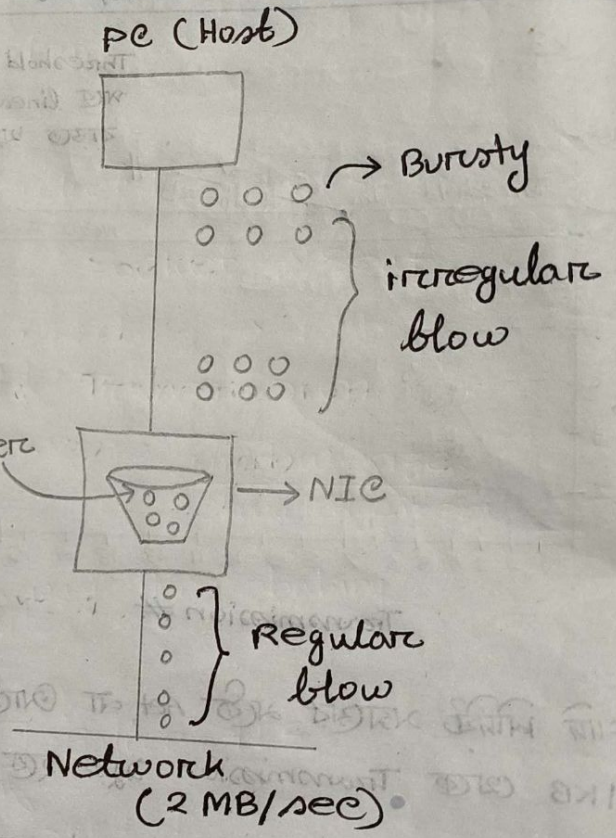
Application (Service)	Delay	Bandwidth	Reliability	Jitter
EMAIL	low	low	high	low
Video on demand	medium	high	low	high
Video Conference	high	high	high / low (medium)	high

Technique to achieve

- Over provisioning
- Flow control

* Leaky Bucket Algorithm





$$\text{Data} = \text{Rate} \times \text{Time}$$

$$= \frac{25\ MB}{sec} \times 40\ msec$$

$$= \frac{25\ MB}{1000\ msec} \times 40\ msec$$

$$= 1\ MB$$

$$\text{Time} = \frac{\text{Data}}{\text{late}}$$

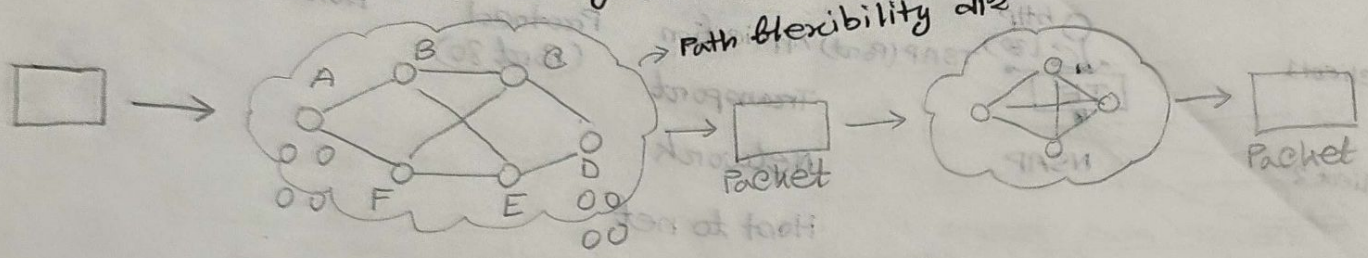
$$= \frac{1\ MB}{2\ MB/sec}$$

$$= \frac{1}{2\ s}$$

$$= 500\ msec$$

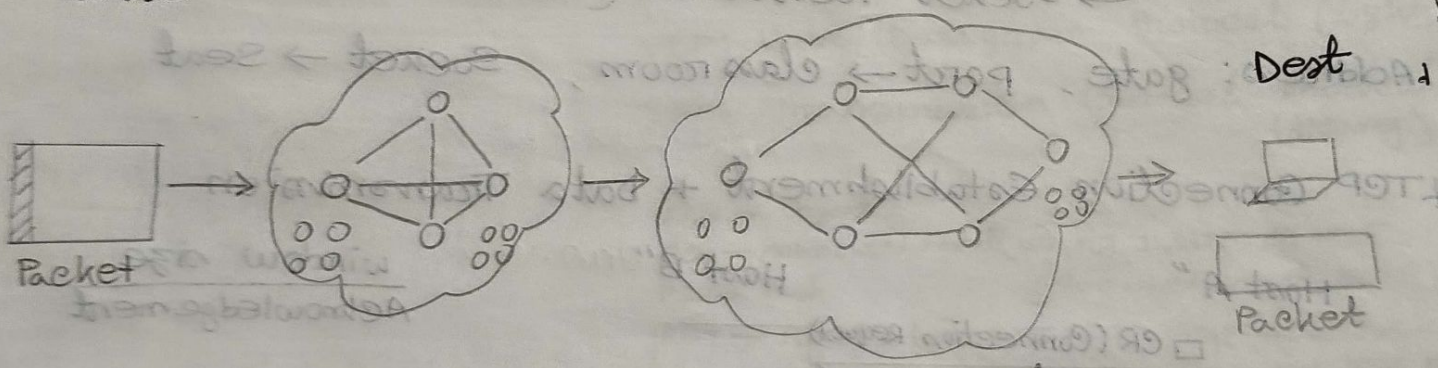
Fragmentation:

- Transparent Fragmentation
- Non Transparent Fragmentation



Transparent Fragmentation

Packet টা 4 ভাগে ভাগ করবে। Network থেকে leave করার সময় reassemble হবে। ভাগ হবে network এর capacity wise



Non transparent Fragment

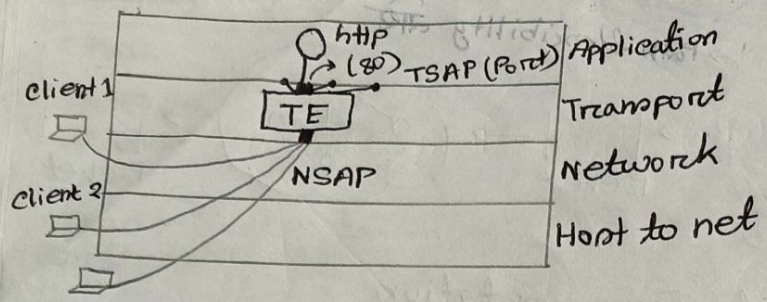
Fragment করার পর reassemble করা লাগে না। Fragment কে আবার Fragment করা যায়। Destination এ পৌঁছানোর পর একবারে reassemble হবে।

14.05.24 / CSE405
Tuesday / Cls-17

Tcp Ref Model:

Host (Ex CNN Web Server)

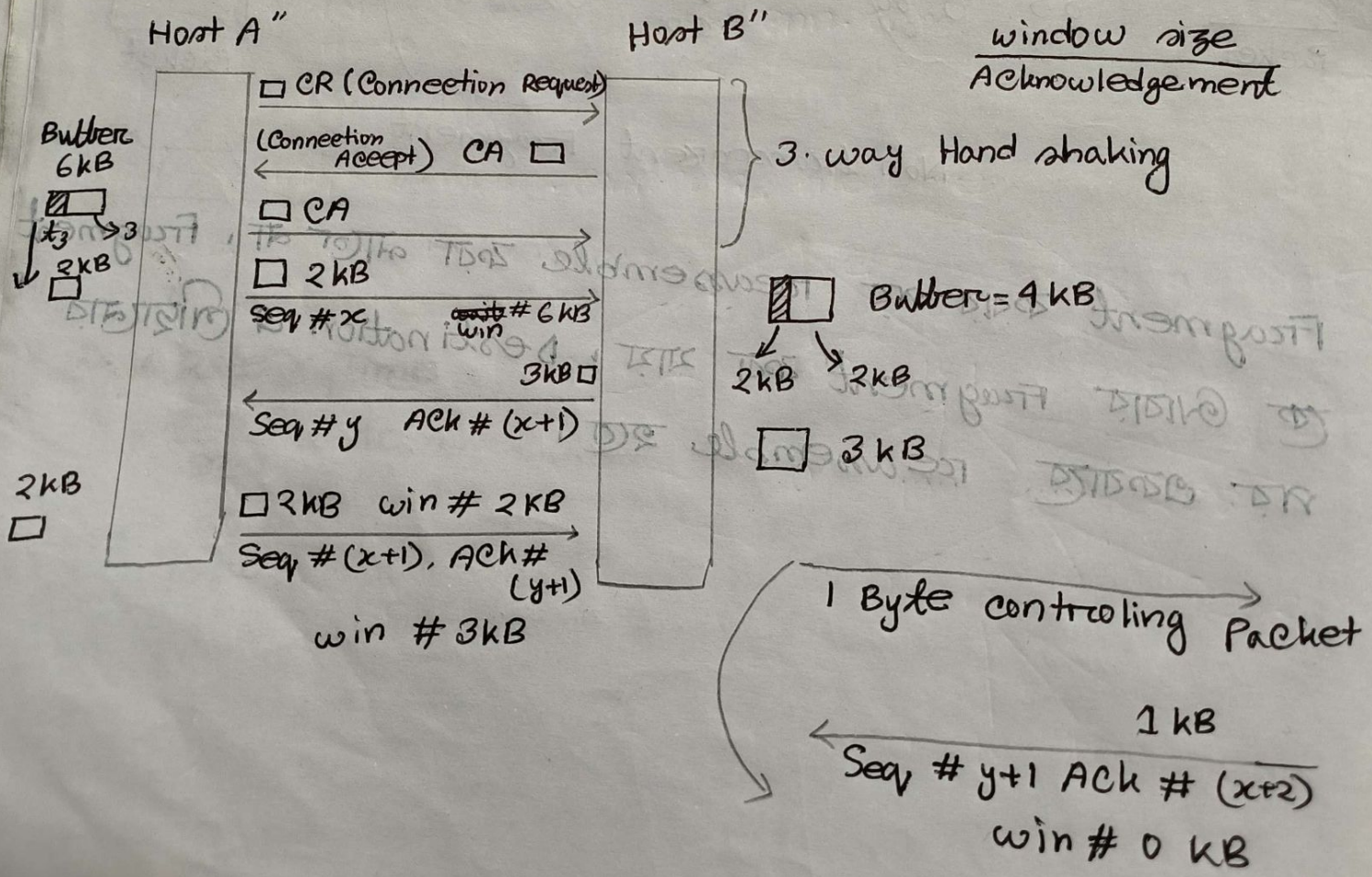
http://www.cnn.com
Protocol (Port 80) — Host name



NSAP: Network Services Access point
TSAP: Transport Services Access Point
END point: IP + Port
(Socket) → Socket Identifier (Socket 1, Socket 2, ...)

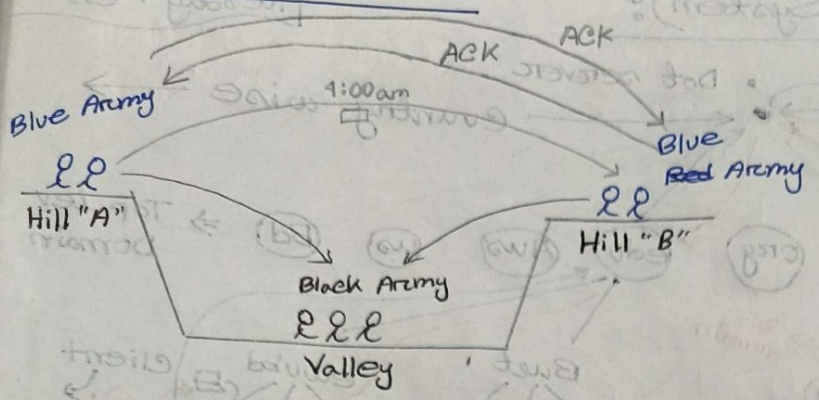
Address: gate, port → class room, Socket → Seat

*TCP Connection Establishment + Data Transmission

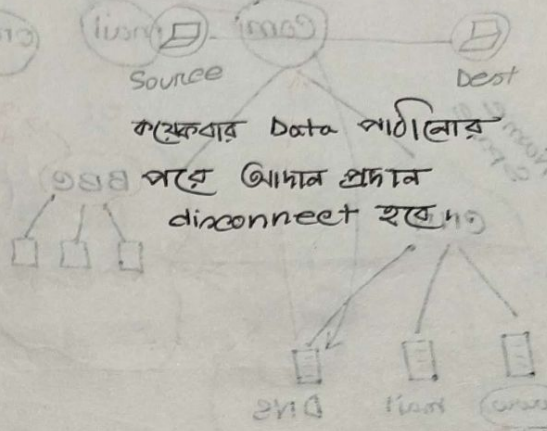


Connection Release

19.05.24 / CSE405
Sunday / Ch-18

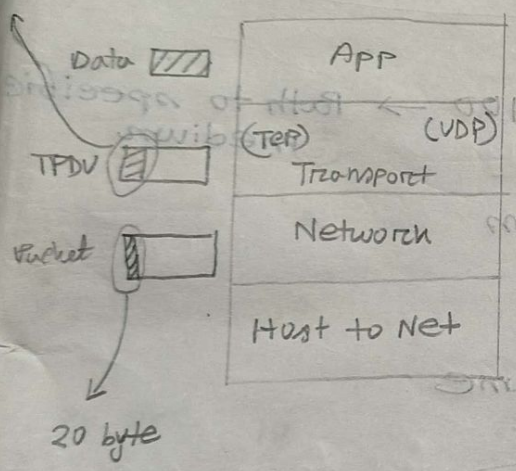


Two - Army Problem



কয়েকবার Data পাঠানোর পরে আদান প্রদান disconnect হবে।

TCP/IP Ref Model -



UDP: User Datagram Protocol (light)

TCP: Transmission Control Protocol (Heavy)

URG → Urgent
ACK → Acknowledgement (যদি এটা 1 থাকে এর মানে Acknowledgement আছে)
PSH → Push
RST → Reset (এটা থাকা মানে আগে একটি Connect ছিলো যেটা reconnect করতে চায়)
SYN → Syne (প্রথম নতুন Connection এর সময় 1 থাকে)
FIN → Finish (এর মানে আর কোনো Data নাই)

UDP Header:

16 Bits

Source Port	Dest Port
UDP length	Check Sum

TCP Header

16 Bits

Source Port									Dest Port
Sequence # x									
Acknowledgement # (x+1)									
TCP Header length		U R G	A C K	P S H	R S T	S Y N	F I N		Window size
Check Sum									Urgent Printer
Options									

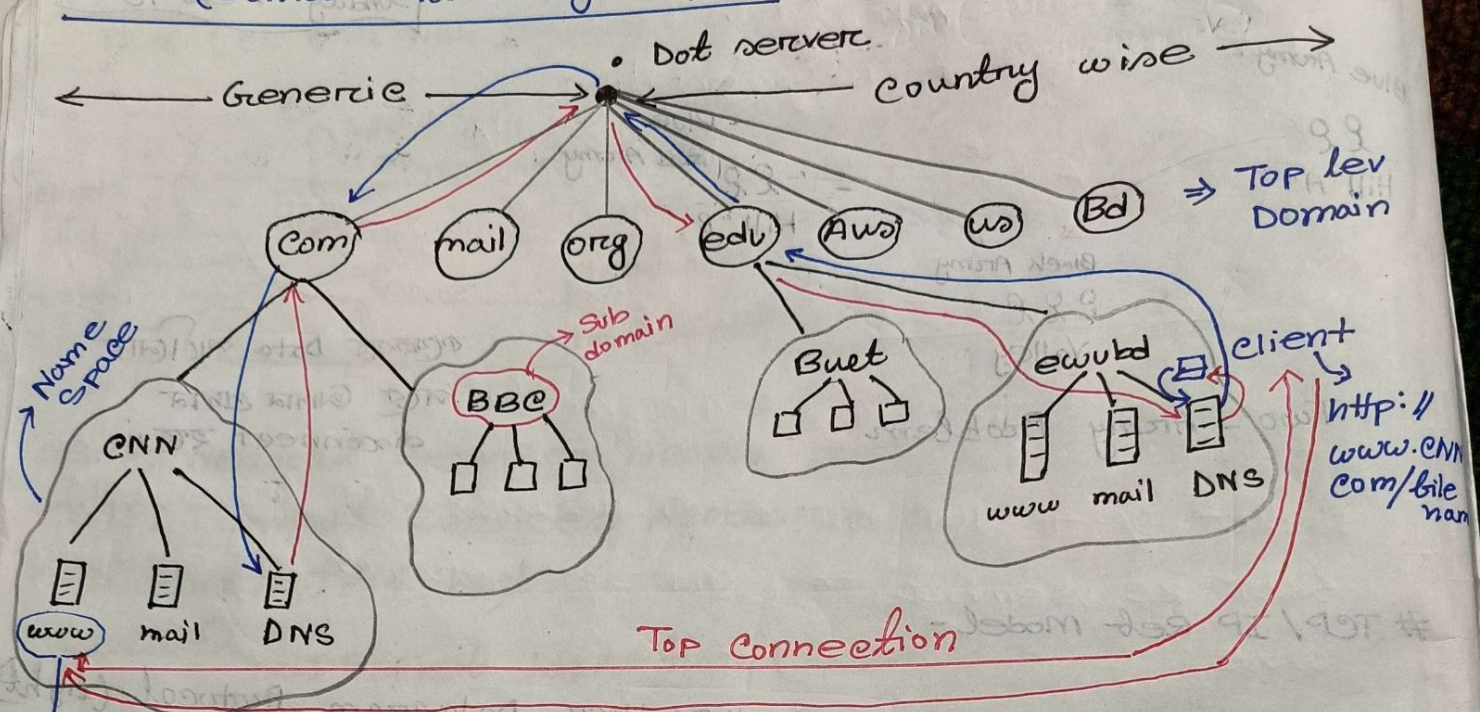
20 Bytes

Sequence and Acknowledgement এর মধ্যের mechanism টা হলো Piggybacking

(Type of Urgency → Urgent Pointer)

21-05-24 / CSE 405
Tuesday / Cls

DNS (Domain Name System):



DNS এর কাছে এটার IP চাবে।

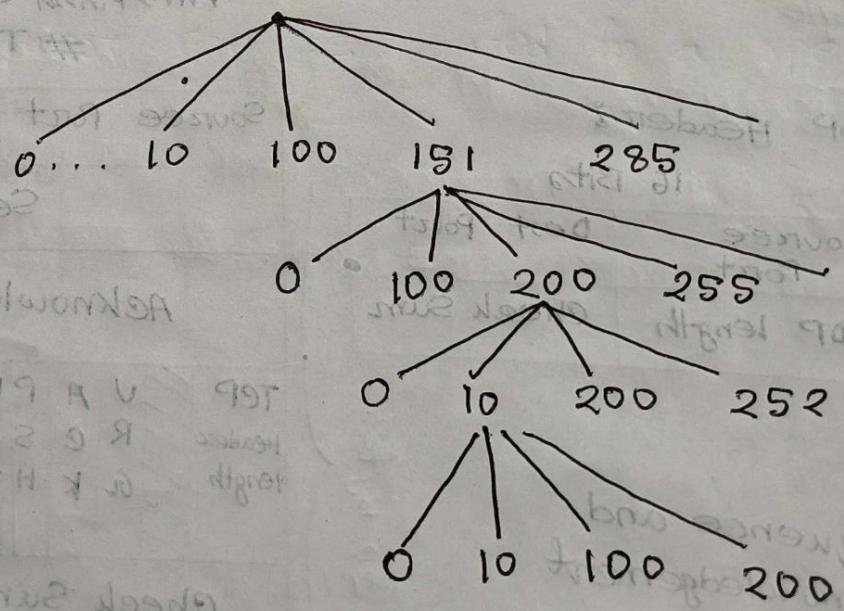
151.200.10.100 → Path to specific medium

www.cnn.com.

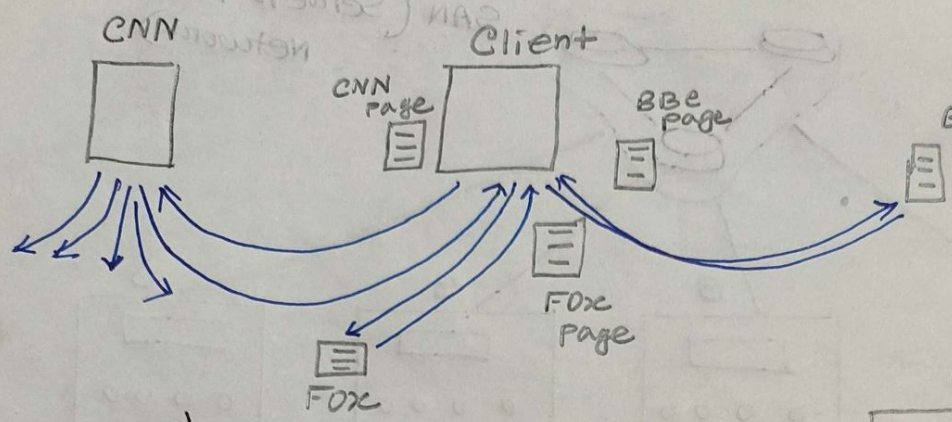
Under Dot server but not necessary

www.cnn.com

Client → http://www.cnn.com/filename

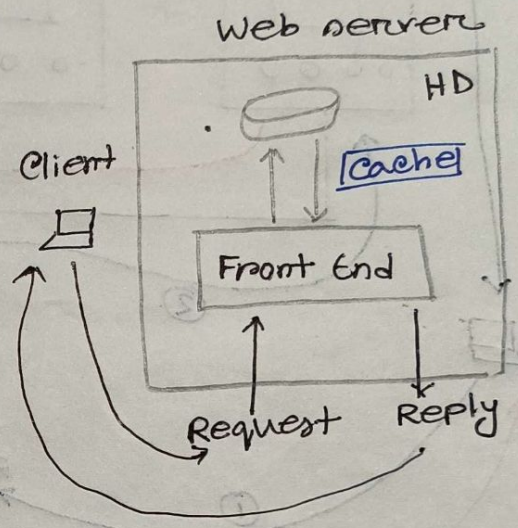


www (world wide web):



Speed

↑ Register (B)
Cache (MB)
RAM (GB)
HD (TB)
Tape Drive (100 TB) ↓ Capacity

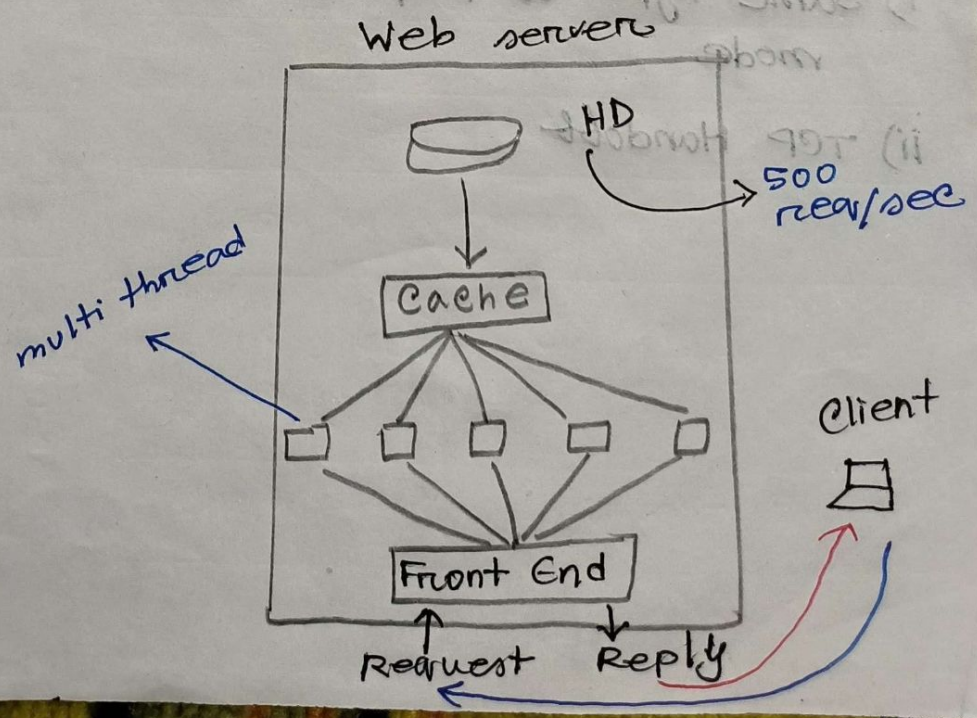


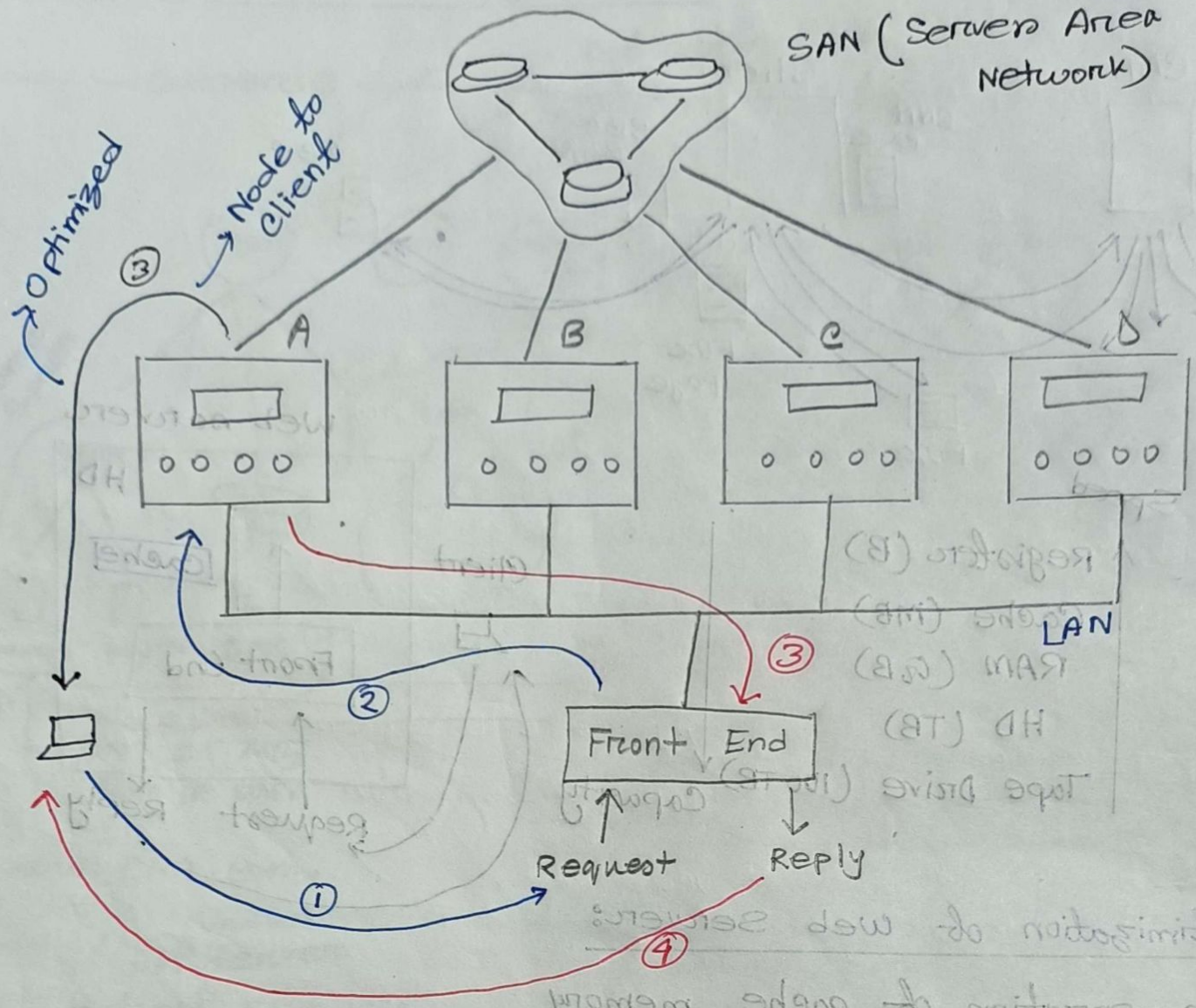
Optimization of web server:

i) Incorporation of cache memory

ii) Incorporation of multiple threads

iii) Server farm





Optimization of server farm:

i) Same type of request can be routed to a specific mode

ii) TCP Handoff.