Computer N/W's Introduction :

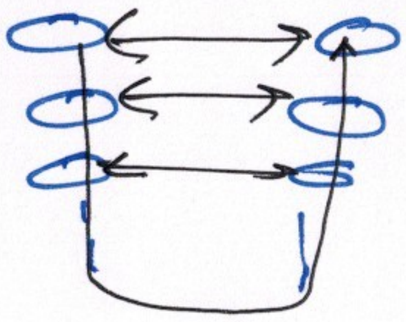
(*) five components of data communication
- Message
- Sender
- Receiver
- Transmission medium
- Protocol

(*) Protocols Can be
- Simple
- Complex

(*) In Complex scenarios think about issues and group them logically and design algo/protocols/devices to realize that.
⟹ Leads to layering ⟹ Modular programing
(useful for reliability/scalability/debugging ---)

(*) Logical Connection ⟹ Layer to layer Communication.

(*) TCP/IP Suit ⟹ 5 layers :
- Application layer
- Transport Layer
- Network Layer
- Data link layer
- Physical layer



⊛ End to End and hop to hop

A → Message
T → TCP/IP packet (user datagram)
N → Datagram
D → frame
P → Bits

Data units

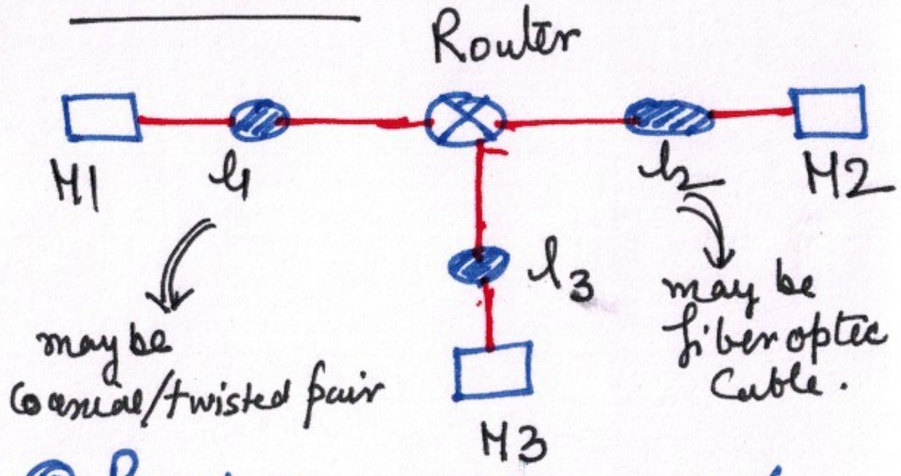
Data unit cannot be changed by Router or lesser layer Switch.

Data unit created by host can be changed by Router NOT Switch.

A, T, N ⇒ End to End

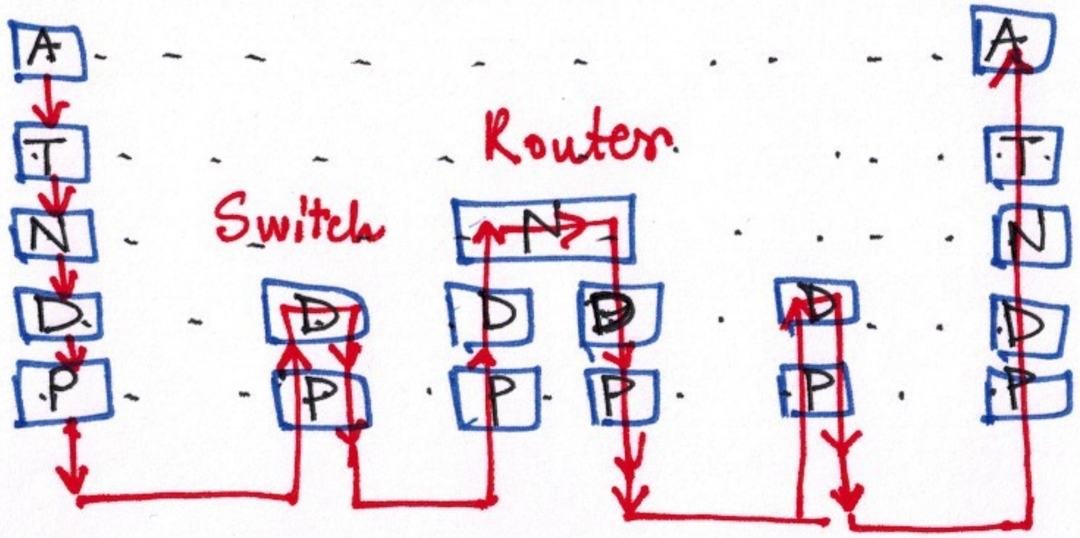
D, P ⇒ hop to hop

Example:



⊛ Router works on N/L and Switch works on D/L.

⊛ Router is involved in only one N/w layer but may have (n) combinations of link + P.L.



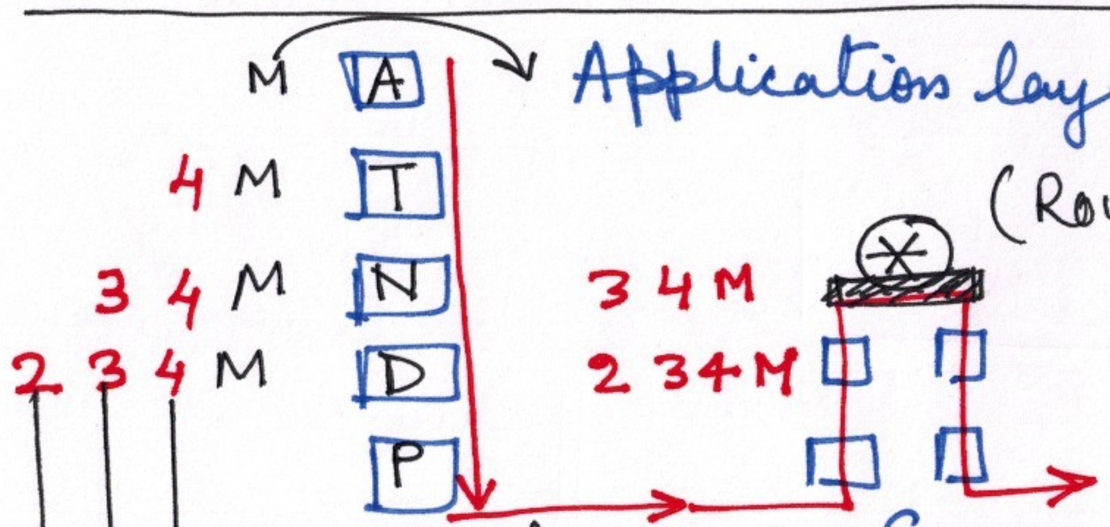
⊛ Router receives a packet from link 1 based on one pair of protocol and deliver it to link 2 on another protocol pair.

⊛ Link Switch can be involved in only one set of protocol.

⇒ Why Network data unit remains unchanged; having Router in between.

⊛ Encapsulation and Decapsulation :

- Source ⇒ Encapsulate.
- Desti ⇒ Decapsulate.



T.L header :
- Source and destination applications program.
- End to end delivery info (flow/error/congg)

N.L. header : Source and destination addresses. (Congest. control).

D.L. header : link layer address of host / next hop. (⊛ why multiple addresses.)

At a Router ÷

→ Bits from P.L ; packaged by D.L.L ; decapsulated and passed to N/L.

→ N/L checks Source/Dest address in datagram ; Consults its forwarding table for next hop.

→ D.L.L again encapsulate the datagram into frame and pass to P.L for transmission.

(A) Physical layer : ⊛ Get frame from D.L.L ; extract bits and send them over link. (via transmission Medium)

⊛ Encoding and Signalling

⊛ Each bit transfer with high probability ⇒ (Not 100% Reliable)

⊛ Topology and N/w design ⇒ Bus, Star, ring, hybrid.

⊛ Transmission Modes ← Simplex, full/half duplex.

(B) Data link layer ⊛ Data link layer is responcible to carry/transfer datagram to next hop determined by Router. It does framing

⊛ Acknowledgemed and Sequence no. management.

⊛ Error Detection and Correction: → Detection (CRC, LRC, VRC, Checksum)
→ Correction (Haming distance)

⊛ Flow Control (Sliding window protocols)

⊛ Access Control: Req. as media is shared and accessed by all. (CSMA/CD, CA)

⊛ Physical addressing

(C) Network layer : Routing and Congestion Control

(how packets are transffued from source to dest.)

Static
- fixed routes
- Static table wired into Router
- Rarely changed

Dynamic
- packet of an application can follow different route
- Depends on
  - topology
  - Shortest Path
  - load

Semi Dynamic
- At connection initiation route is fixed for an application.
- All packets of that application follow it

(*) It donot gurantees that packet will reach to its destination.

Connectionless Service
- Each packet is independently routed.
- Datagram approach

Connection oriented Service
- Connection is required to be established first.
- All datagrams follow same path (tube)
- Virtual ckt approach

(*) what are pros & Cons of CL/CO Services.

(*) Congestion Control : Packet may be dropped but managed properly.

(*) Internetworking : Support multiple n/w.

(*) Logical / IP addressing.

(D) Transpoot layer : - Provide services to A.L.
(End to End) - Got message from Source AL deliver to destination. Multiple protocols are there.

⊙ TCP : Transmission Control protocol, Connection oriented.
↳ Provides ⊙ flow Control
⊙ Error Control
⊙ Congestion Control.

⊙ UDP : User datagram protocol, Connection less.
↳ Nothing like above controls are provided.
Simple ; low overhead;
It also does multiplexing and demultiplexing. Protocol need to specify (in some field) in its header to identify which protocol encapsulated packet belongs. (Source: Multiplexing Dest: Demul.)

(E) Application layer : Program/Process to process Communication.
HTTP, WWW, SMTP, FTP, IMAP, SSH, DNS ...

Message — A — Application N/w.
Segment/User datagram — T — Port No
Datagram — N — Logical (IP add)
Frame — D — Physical (MAC add)
Bits — P

Internetworking devices:

(A) Repeater :-

- Dumb electronic device.
- Regenerate the Signal, to overcome the Signal attenuation (help to move long)

(Operates on P.L)

4 Repeater LAN extension Rule.

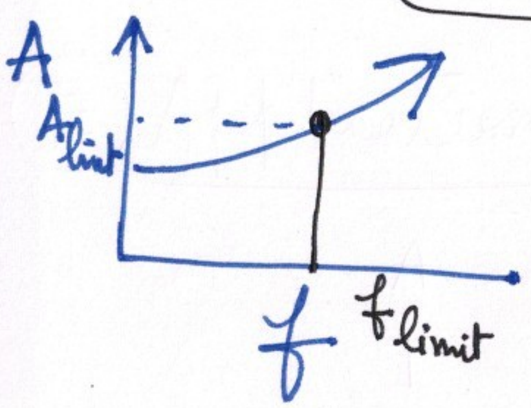
① Why Signal attenuates :

① $A_{db} = \alpha \left[\frac{db}{MHz.cm}\right] l\ [c.m]\ f\ [MHz]$

α → Medium Const
Air – 1·64
Water – 0·002

(directly dependent on frequency & length)

② High frequency → High attenuation
→ High datarate



③ There is a limit for attenuation over which Signal & noise will be indistinguishable (A_{limit})

$A = \alpha l \cdot f = A_{limit}$

$l \cdot f = Constt$

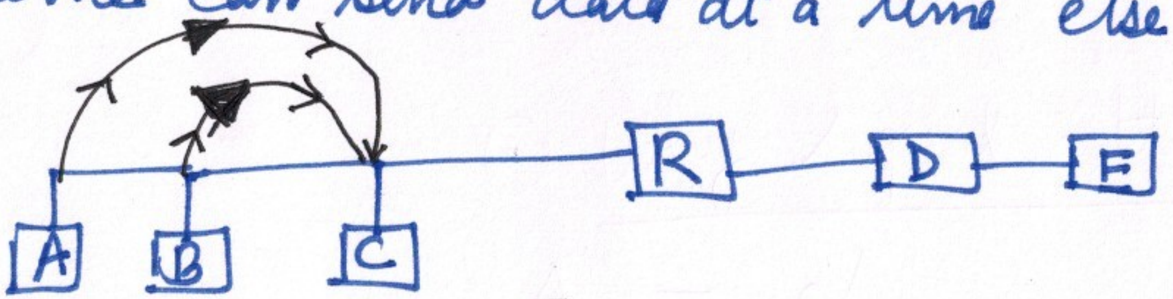
∴ l and f are inversely proportional

④ Therefore for practical lengths f_{max} also got limited.

Higher b/w → Lower Cable length → More repeaters.

(B) HUB :- Multiport repeater ; N/W topology changed from bus to STAR.
(Physical layer)

* Arrived data is repeated on all ports except on the arrival port.
* Only one device can send data at a time else collision occur. (CSMA/CD)
* Problem :- with both (R) & (H)



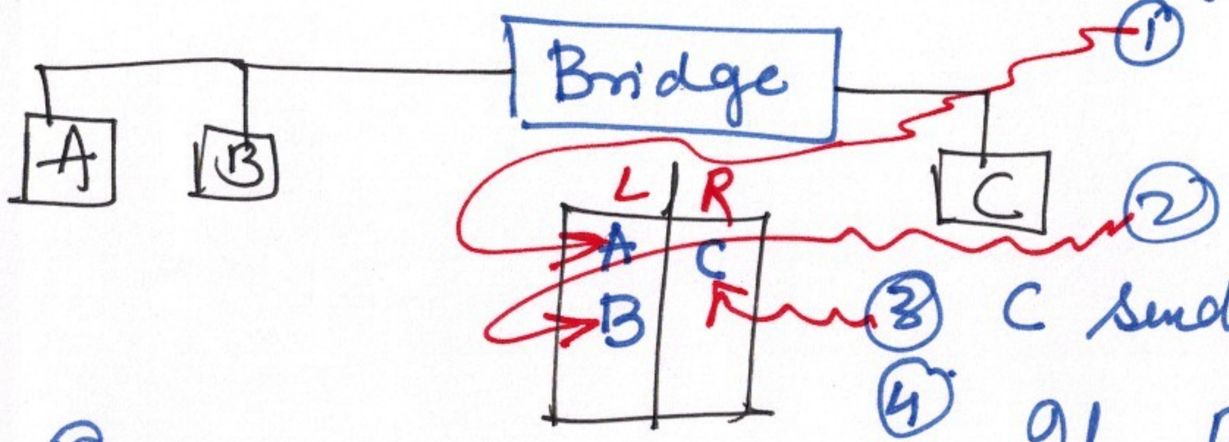
* A & B both want to send data to C.
* Collision occur and this information is forwarded to whole N/W.
* (R) & (H) both broadcast blindly and treat all ethernet segments as part of single big ethernet.

Solution :-
- Divide the collision domain ; ethernet segments are
- treated as individual ethernet.
- Intelligent Broadcasting is required. (Bridge)

(c) **Bridge** :- Used to separate N/W Segments logically within Same N/W (Dividing the Collision domain)

(*) Works on D.L.L and can broadcast intelligently.
(Have/manage a Bridge table with MAC addresses.) ⟹ Take intelligent decision about whether or not to pass signal on to the next segment of N/W.

→ Receive a frame ; lookup the destination address in Bridge table
→ Decides wheather to filter, flood or Copy frame to next Segment.
→ Self managed bridge tables refreshed in (30 sec) = (Right | left) Neighbour discovery learnt from transmission.
→ Until nodes communicate L/R entries are not generated.



1 A sends B (Bridge forwards Until B Communicate)
2 B replays to A (Bridge do not forward)
3 C sends to B (Bridge ferward)
4

(*) No Collision info got forwarded. If B moves from Left to Right.