Computer Networks
COL 334/672

To Packet Switch or Not

Slides adapted from K&R book

Tarun Mangla

Sem 1, 2025-26

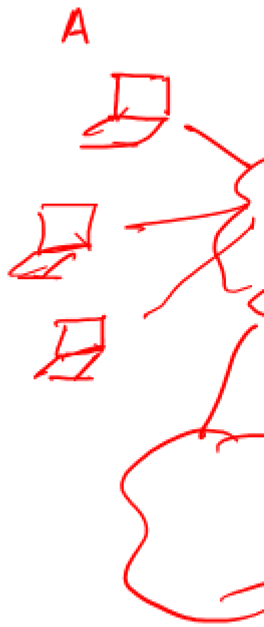
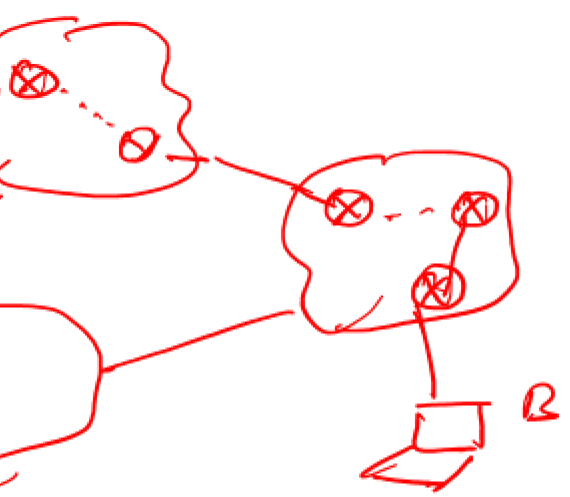
Recap

- Structure of the Internet
  - 3-tier structure
  - Driven by economics and national policies
- Protocols are central to Internet
  - Distributed autonomous networks
  - Need to agree on some ground rules
- Resource sharing
  - Circuit switching vs packet switching

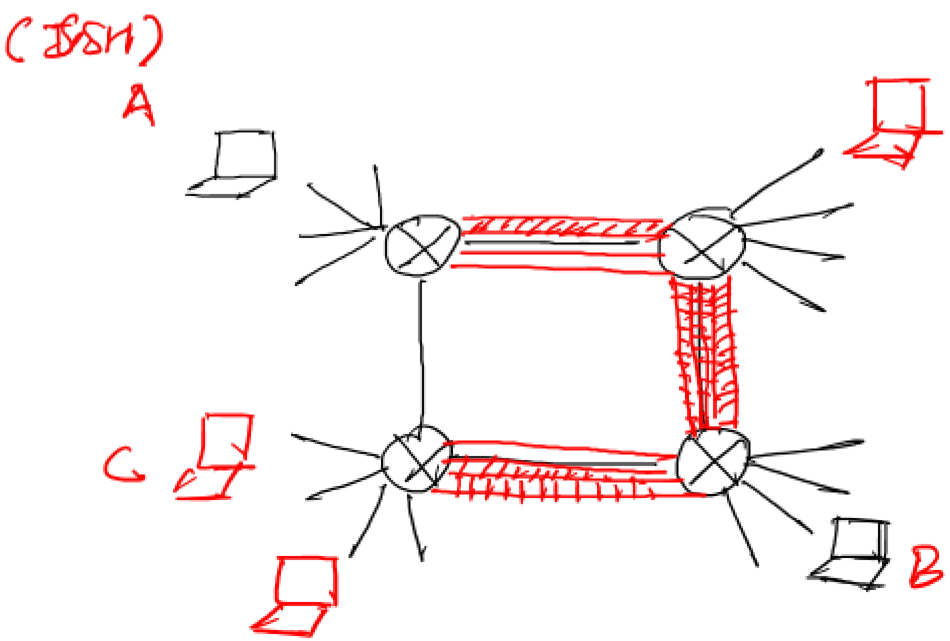
① - Routing, Addressing

② Encoding → bits to signals

③ Resource sharing

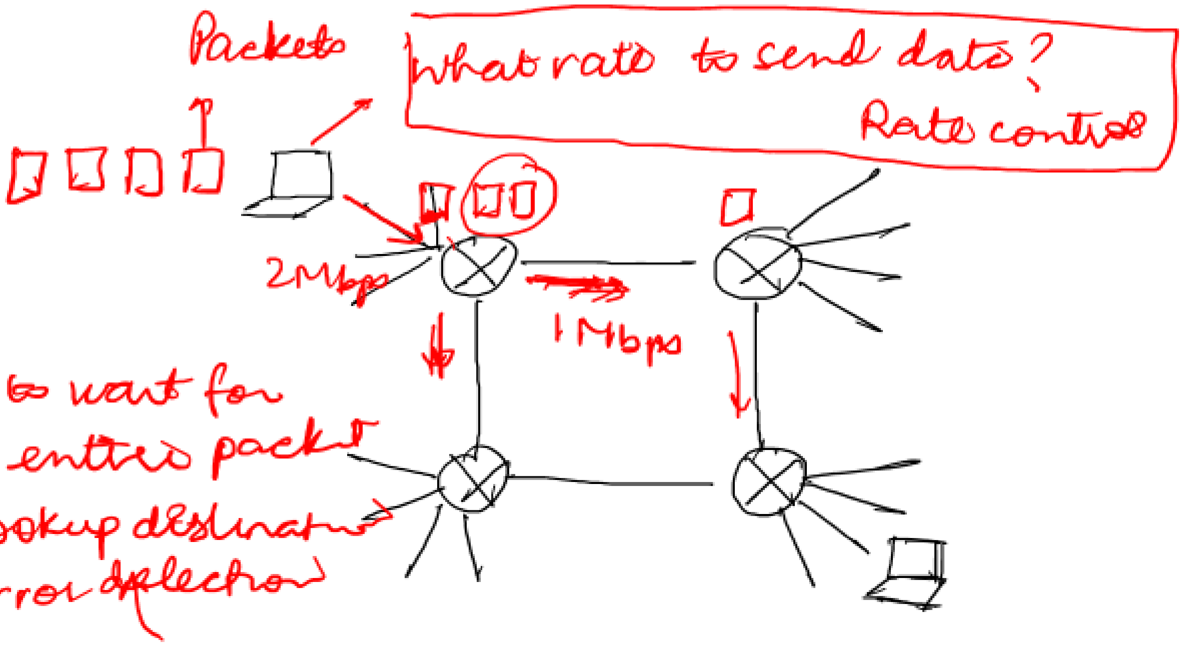



Circuit switching vs Packet switching



Admission control

↳ guaranteed bandwidth



What rate to send data?
Rate control

Need to wait for the entire packet
↳ lookup destination
↳ Error detection

Store & Forward approach

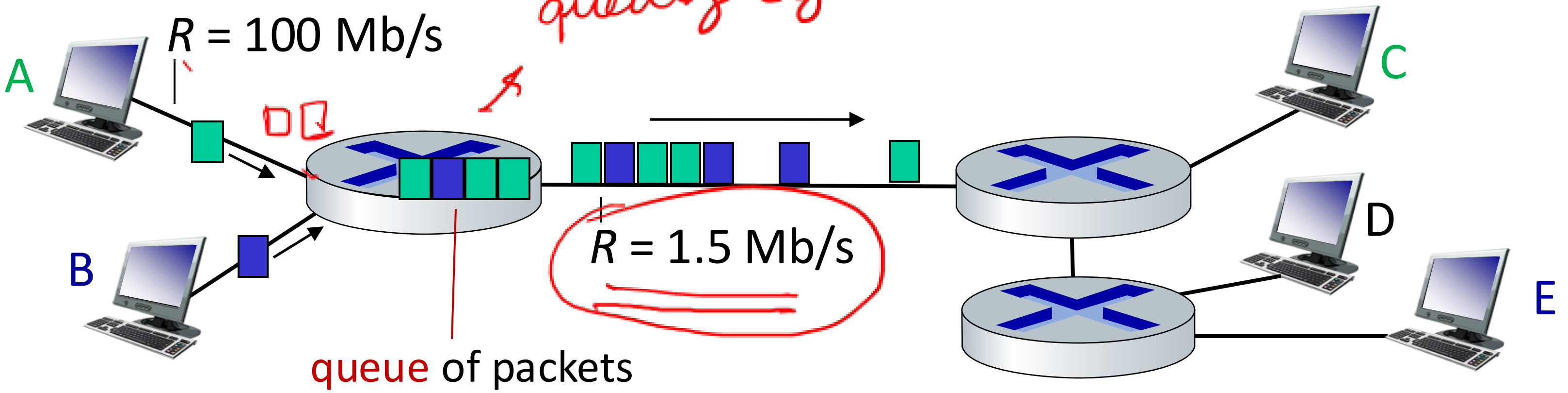
↳ "Best-effort" delivery
(No guarantees)

Per-packet processing

Packet-switching: queueing

Rate control

queueing system



queue of packets waiting for transmission over output link

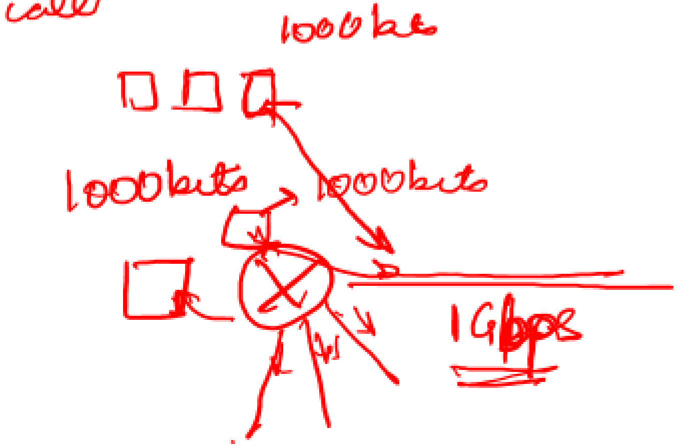
Queueing occurs when work arrives faster than it can be serviced:

Which option did the Internet choose? (

NO FREE LUNCH THEOREM

Packet-switching versus circuit switching

Audio call



$$\frac{1000}{10^9} s$$

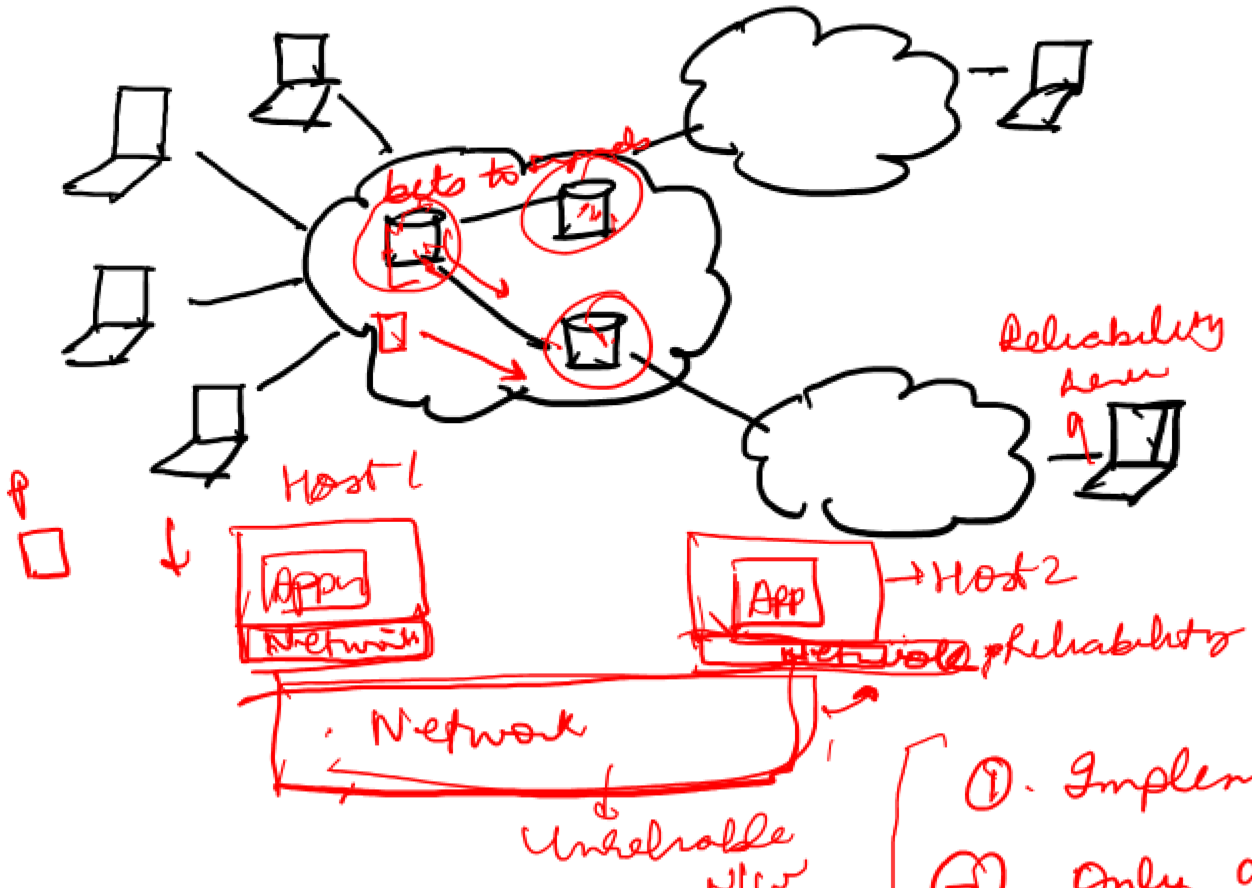
$$= 10^{-6} s$$

Internet uses packet switching

- Great for “bursty” data – sometimes has data to send, but at other times not
  - Efficient resource sharing (why?)
  - Simpler*, no call setup unlike circuit switching
- however, does not provide any performance guarantee, best-effort delivery
- excessive congestion possible: packet delay and loss due to buffer overflow
  - protocols needed for reliable data transfer, in-order delivery, congestion control/Rate control
  - Implication on router design

↳ Per-packet processing

How To Send Data over Distributed Network?



- Data transmission techniques over a link (Encoding)
- Addressing, routing
- Resource sharing
  - Packet switching →
- Other network services
- Reliability
- Rate control to avoid congestion
→ In-order delivery

① Implement everything inside the network

② Only at the end host

How to implement them?

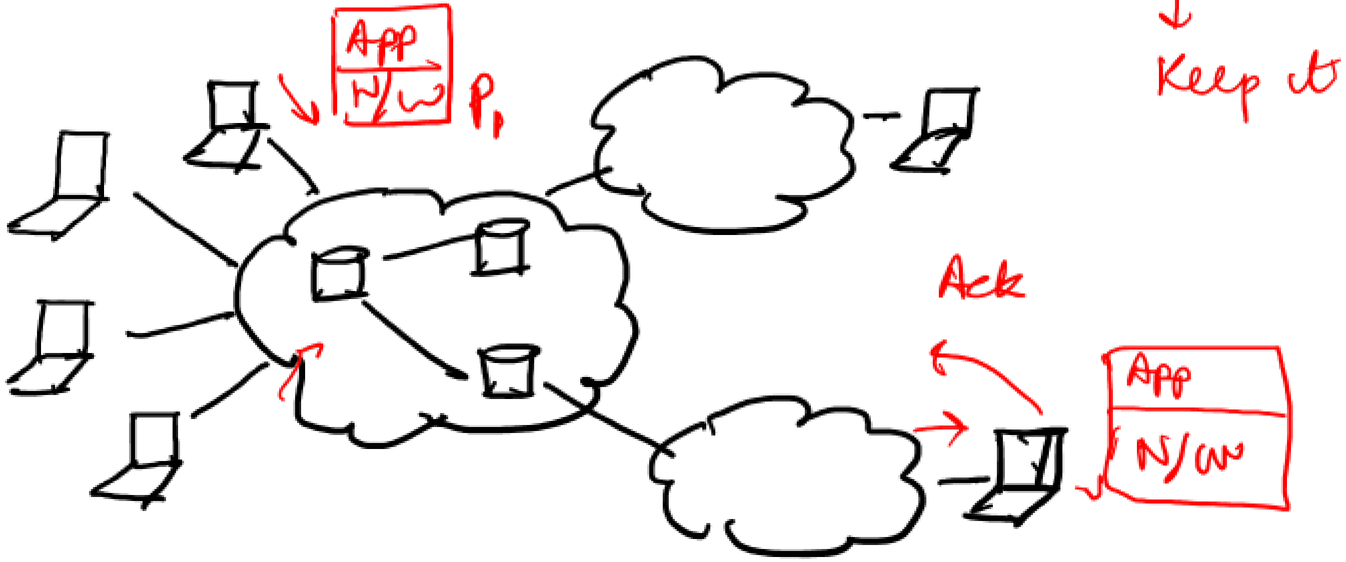
Where to implement them?

Where to implement reliability?

KISS
↓
Keep it simple silly

Why do we need to implement reliability?

Best-effort delivery



In-network support

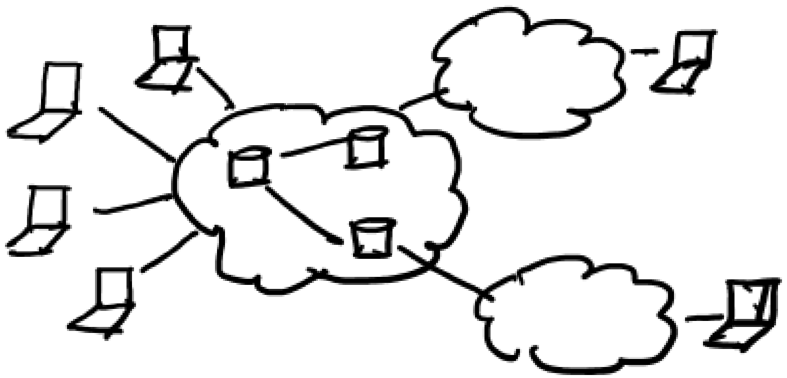
- Routers cache the packet and resend if it is not acknowledged
- Needs extra memory and compute in routers

End-host support only

- End-host send acknowledgement
- Reduced performance*

Where to Implement Network Services?

Two different paradigms



In-network support

- Network with rich functionality that covers most requirements
- Network with multiple “lanes”?
  - CISC-like

End-system support only

- As little functionality as possible in the network
- Most functionality at the end points
- Also called end-to-end principle
  - Dumb network, intelligent end-points
  - Saltzer, Reed, Clark (1981)

Which option did the Internet designers chose?

End-to-end principle

Why End-to-End Principle?

- Need end-to-end correctness anyways
- Not everyone needs it
  - Diminishing returns from in-network functionality
  - Cost-effective
- Not everyone has it
  - All networks are not capable of providing functionalities
- Flexibility of implementation

Are there exceptions?

Any limitations?