CS 348
Computer Networks
Lec 8

Spring 2020 IIT Goa

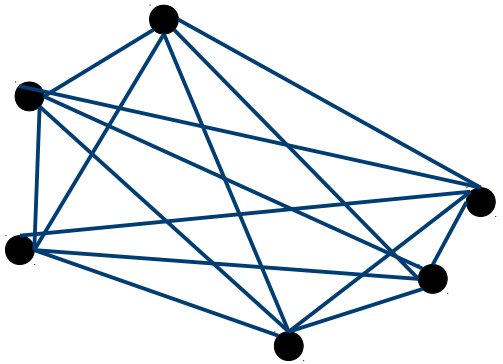
Course Instructor: Dr. Neha Karanjkar

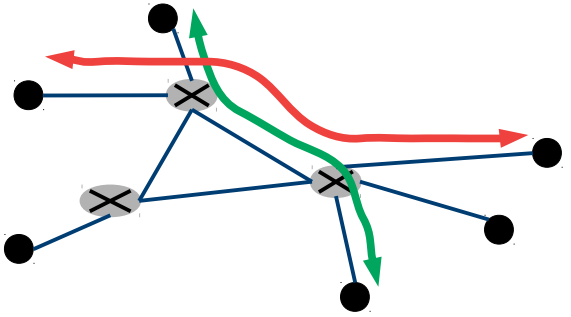
Disclaimer: These slides are adapted from Computer Networking: A Top-down Approach by Kurose & Ross, 7th ed. and lecture slides of cs 168-2020 (http://cs168.io/) by Prof. Sylvia Ratnasamy

Questions

- Why does the Internet use **Packet Switching** ?
- What are the alternatives?
- What is the history of Packet switching?

Connecting together multiple hosts





✗

O(n^2) links

✓

Links and switches need to support multiple "**flows**" at a time.

Network resources (links, switches) must be shared between end-hosts.

Sharing Network Resources

How can network resources (links, switches) be shared between flows?

Two approaches:

- **Reservations:** end-hosts explicitly reserve bandwidth along the entire path at the start of a communication session, and release it when done.

- **Best-effort:** just send data packets across. The packets use up resources (link bandwidth, switch buffer space) as and when needed (“on demand”).

Sharing Network Resources

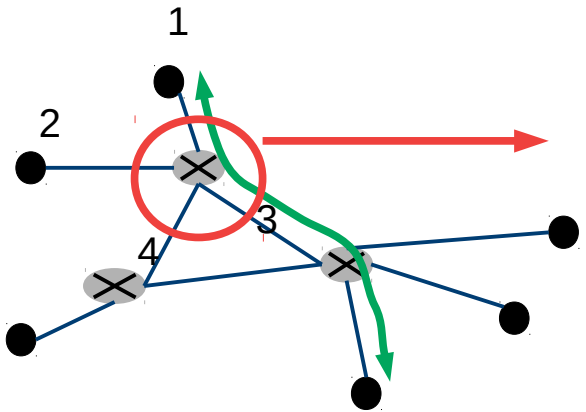
Two approaches:

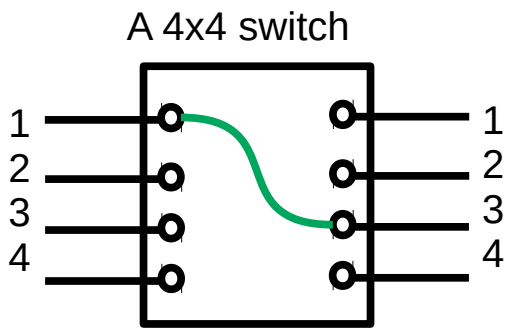
- **Reservations:** this is the approach used by **Circuit switching**

- **Best-effort:** this is the approach used by **Packet switching**

Circuit Switching

- Network resources **along a route** are **reserved** between the end-hosts for the duration of a communications session.
- The terminology comes from early telephone exchanges. A “circuit” was literally formed between the end-hosts by configuring switches in telephone exchanges.



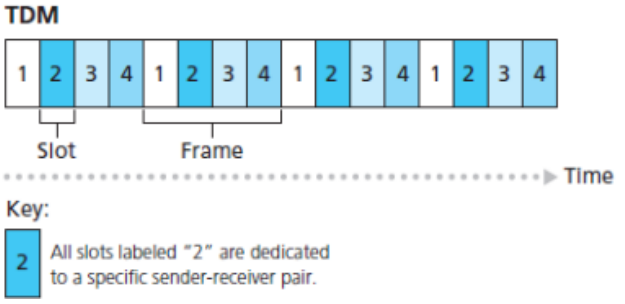


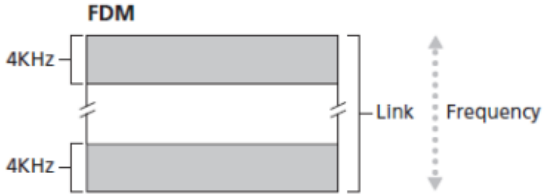
Circuit Switching

- Network resources **along a route** are **reserved** between the end-hosts for the duration of a communications session.
  - source sends a reservation request to the destination
  - switches “establish a circuit”
  - source starts sending data
  - source sends a “teardown circuit” message
- Currently used in the Public Switched Telephone Network (PSTN)

Circuit Switching

- One link doesn't imply one communication channel. A single link can support multiple "channels" at a time:
  - Time division multiplexing (TDM)
  - Frequency division multiplexing (FDM)





- Thus "channels" (not links) along a path are reserved for a communication session.

Circuit Switching vs Packet Switching

How do they compare along the following axes?

- Service model for applications (are there any performance guarantees?)
- Efficiency
- Handling failures
- Implementation complexity

Circuit Switching vs Packet Switching

Service model for Applications:

- **Circuit switching:** guaranteed bandwidth for the duration of the session
- **Packet switching:** no bandwidth/performance guarantees for a flow:
  - no guarantee for allocation of space inside switch buffers
  - no guarantee for link bandwidth

Circuit Switching vs Packet Switching

Efficiency:

- **Circuit switching**: less efficient for “bursty” traffic.
  - Reserved bandwidth cannot be given to other “flows” even when unutilized.
  - Time spent for setting up/tearing down circuit. Inefficient for short flows.
- **Packet switching:** more efficient for “bursty” traffic.

Circuit Switching vs Packet Switching

Handling Failures: What happens if a link goes down?

- **Packet switching**:
  - Network must detect failure
  - Network recalculates routes (Job of the routing protocols)
  - Endhosts and individual flows do nothing special except cope with the temporary loss of service.
- **Circuit switching:**
  - All of the above
  - In addition, Endhosts must detect failure, teardown old reservations, send a new reservation request. All impacted endhosts must do this, for each impacted flow

Circuit Switching vs Packet Switching

Implementation Complexity

- **Packet switching**:
  - Network does not need to maintain per-flow “state” (the network does not need to remember anything about which “flows” are active)
- **Circuit switching:**
  - Setup/teardown
  - Network needs to maintain per-flow state.

History of Packet Switching

- 1961: Leonard Kleinrock proposes the idea of packet switching in his MIT doctoral thesis.
- 1964: Packet switching studied and developed further by Paul Baran at the RAND corp. Motivation was to develop resilient and survivable communication systems.
- 1965: Davies designed a store-and-forward packet switching system and coined the term “packet”.
- 1969: ARPANET is designed to work on packet switching and becomes functional with a few nodes

(Ref: https://en.wikipedia.org/wiki/Packet_switching)

Questions (Going forward)

- How are forwarding tables populated? How/when do routers update the forwarding tables when some link goes down?
- How does TCP provide reliable transmission over an unreliable channel?
- How does TCP perform congestion control?
- What are some Application-layer protocols? How do they work?
- What is the interface between the Application layer and the transport layer?

Reference

- Kurose and Ross, Section 1.3.2