

INTRODUCTION TO EECS II
DIGITAL COMMUNICATION SYSTEMS

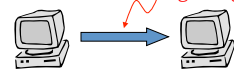
**6.02 Fall 2011
Lecture #19**

- Multi-hop networks
- Packet switching
- Queues and Little's Law


6.02 Fall 2011 Lecture 19, Slide #1

From Links to Networks

- Have: digital point-to-point We've worked on link signaling, reliability, sharing

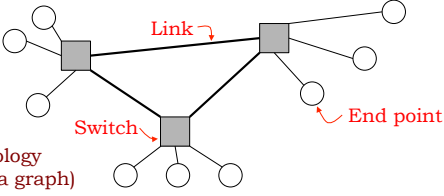


- Want: many interconnected points



6.02 Fall 2011 Lecture 19, Slide #2

Multi-hop Networks



Network topology (modeled as a graph)

- What's wrong with just connecting every pair of computers with dedicated links?
- Switches orchestrate flow of information through the network, often multiplexing many logically-independent flows over a single physical link
- **Packet switching: model for sharing in most current communication networks**

6.02 Fall 2011 Lecture 19, Slide #3

Sharing the Network

We have many application-level communications, which we'll call "connections", that need to be mapped onto a smaller number of links

How should we share the links between all the connections?

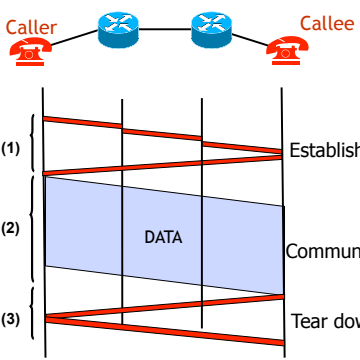
Two approaches possible:

- **Circuit switching** (isochronous)
- **Packet switching** (asynchronous)

6.02 Fall 2011 Lecture 19, Slide #8

Circuit Switching

- First establish a *circuit* between end points
 - E.g., done when you dial a phone number
 - Message propagates from caller toward callee, establishing some state in each switch
- Then, ends send data (“talk”) to each other
- After call, *tear down* (close) circuit
 - Remove state



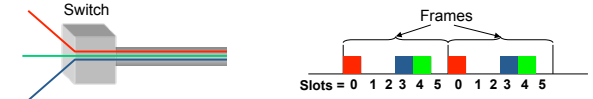
(1) Establish

(2) Communicate

(3) Tear down

6.02 Fall 2011 Lecture 19, Slide #9

Multiplexing/Demultiplexing

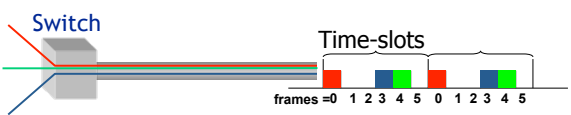


One sharing technique: time-division multiplexing (TDM)

- Time divided into frames and frames divided into slots
 - Number of slots = number of concurrent conversations
- Relative slot position inside a frame determines which conversation the data belongs to
 - E.g., slot 0 belongs to the red conversation
 - Mapping established during setup, removed at tear down
- Forwarding step at switch: consult table

6.02 Fall 2011 Lecture 19, Slide #10

TDM Shares Link Equally, But Has Limitations



- Suppose link capacity is C bits/sec
- Each communication requires R bits/sec
- #frames in one “epoch” (one frame per communication) = C/R
- Maximum number of concurrent communications is C/R
- What happens if we have more than C/R communications?
- What happens if the communication sends less/more than R bits/sec?

→ Design is unsuitable when traffic arrives in *bursts*

6.02 Fall 2011 Lecture 19, Slide #11

Packet-Switched Networks

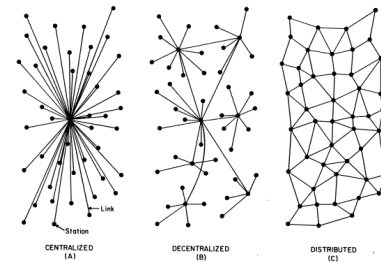


FIG 1 – Centralized, Decentralized and Distributed Networks

The pioneering research of Paul Baran in the 1960s, who envisioned a communications network that would survive a major enemy attack. The sketch shows three different network topologies described in his RAND Memorandum, [“On Distributed Communications: 1. Introduction to Distributed Communications Network”](#) (August 1964). The distributed network structure was judged to offer the best survivability.

6.02 Fall 2011 http://www.cybergeography.org/atlas/historical.html Lecture 19, Slide #12

Packet Switching

- Used in the Internet
- Data is sent in packets (header contains control info, e.g., source and destination addresses)

- Per-packet routing
- At each node the entire packet is received, stored, and then forwarded (store-and-forward networks)
- No capacity is allocated

6.02 Fall 2011 Lecture 19, Slide #13

Packet Switching: Multiplexing/Demultiplexing

- Router has a routing table that contains information about which link to use to reach a destination
- For each link, packets are maintained in a queue
 - If queue is full, packets will be dropped
- Demultiplex using information in packet header
 - Header has destination

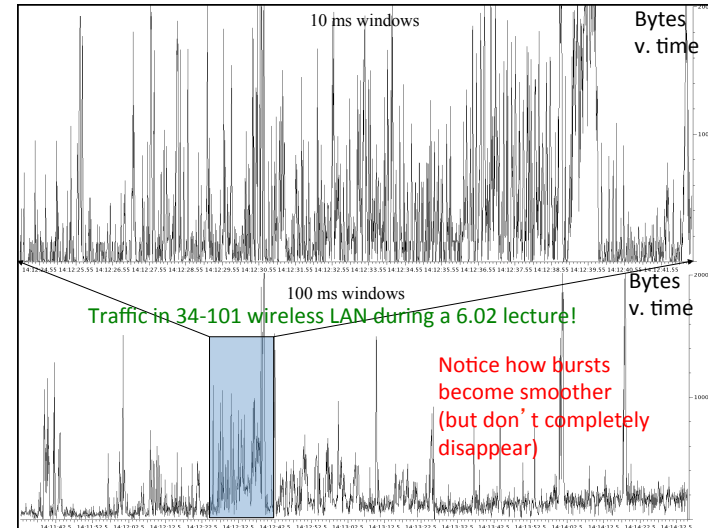
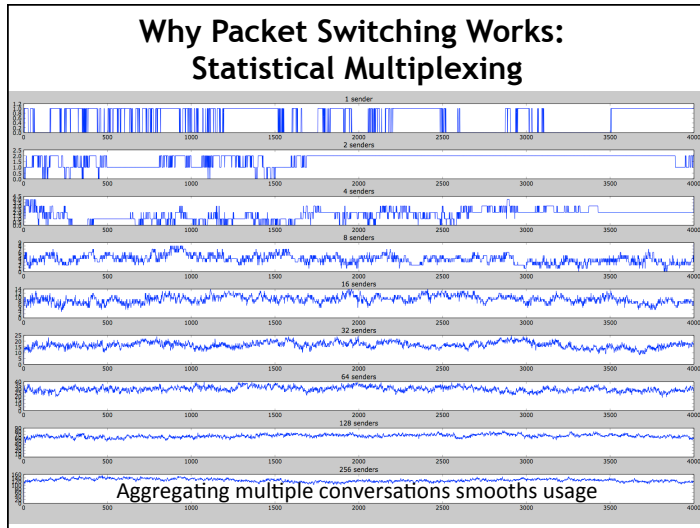
6.02 Fall 2011 Lecture 19, Slide #14

Simple header example

Destination Address
Hop Limit
Source Address
Length

6.02 Fall 2011 Lecture 19, Slide #15

Version	Traffic Class	Flow Label	
Length	Next Header	Hop Limit	
IP Version 6 header Destination Address			
Source Address			



Queues are Essential

- Queues manage packets between arrival and departure
- They are a “necessary evil”
 - Needed to absorb bursts
 - But they add delay by making packets wait until link is available
- So they shouldn't be too big

6.02 Fall 2011 Lecture 19, Slide #21

Little's Law

$n(t) = \# \text{ pkts at time } t \text{ in queue}$

- P packets are forwarded in time T (assume T large)
- Rate = $\lambda = P/T$
- Let A = area under the $n(t)$ curve from 0 to T
- Mean number of packets in queue = $N = A/T$
- A is aggregate delay weighted by each packet's time in queue. So, mean delay D per packet = A/P
- Therefore, **$N = \lambda D$** ← Little's Law
- For a given link rate, increasing queue size increases delay


6.02 Fall 2011 Lecture 19, Slide #22

Best Effort Delivery Model

No Guarantees!

- No guarantee of delivery at all!
 - Packets get dropped (due to corruption or congestion)
 - Use Acknowledgement/Retransmission protocol to recover
 - How to determine when to retransmit? Timeout?
- Each packet is individually routed
 - May arrive at final destination reordered from the transmit order
- No latency guarantee for delivery
 - Delays through the network vary packet-to-packet
- If packet is retransmitted too soon → duplicate

Sounds like the US Mail



6.02 Fall 2011 Lecture 19, Slide #23

Four Sources of Delay (Latency) in Networks

- Propagation delay
 - Speed-of-signal (light) delay: Time to send 1 (first) bit
- Processing delay
 - Time spent by the hosts and switches to process packet (lookup header, compute checksums, etc.)
- Transmission delay
 - Time spent sending packet of size S bits over link(s)
 - On a given link of rate R bits/s, transmission delay = S/R sec
- Queueing delay
 - Time spent waiting in queue
 - Variable
 - Can be calculated from Little's law

$$\#pkts_in_queue / rate$$

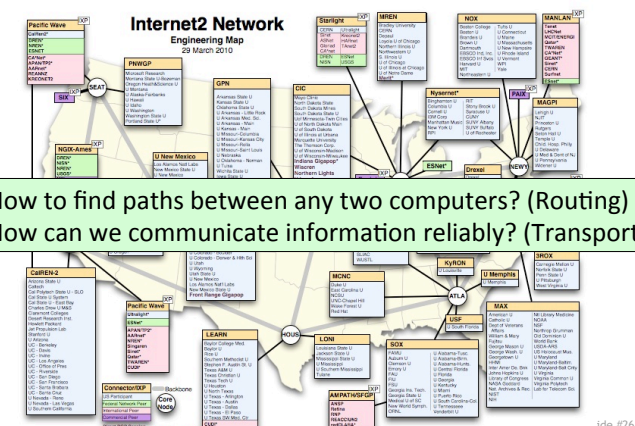
6.02 Fall 2011 Lecture 19, Slide #24

Circuit v. Packet Switching

Circuit switching	Packet Switching
Guaranteed rate	No guarantees (best effort)
Link capacity wasted if data is bursty	More efficient
Before sending data establishes a path	Send data immediately
All data in a single flow follow one path	Different packets might follow different paths
No reordering; constant delay; no dropped packets	Packets may be reordered, delayed, or dropped

6.02 Fall 2011 Lecture 19, Slide #25

Plan for Rest of 6.02



How to find paths between any two computers? (Routing)
How can we communicate information reliably? (Transport)

6.02 Fall 2011 Lecture 19, Slide #26