## Chapter 1: roadmap

## How do loss and delay occur?
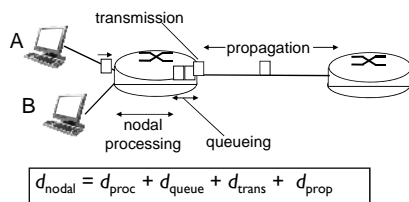
packets *queue* in router buffers

❖ packet arrival rate to link (temporarily) exceeds output link capacity
❖ packets queue, wait for turn



packet being transmitted (delay)

A

B

packets queueing (delay)

free (available) buffers: arriving packets dropped (loss) if no free buffers

## Four sources of packet delay



transmission

A

B

propagation

nodal processing

queueing

$$d_{nodal} = d_{proc} + d_{queue} + d_{trans} + d_{prop}$$
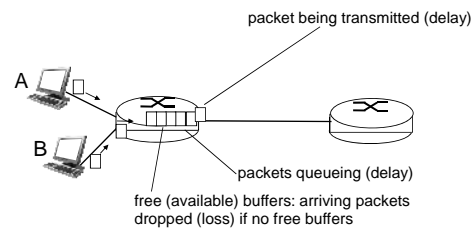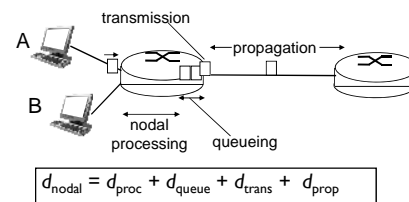
$d_{proc}$: nodal processing
- check bit errors
- determine output link
- typically < msec

$d_{queue}$: queueing delay
- time waiting at output link for transmission
- depends on congestion level of router

## Four sources of packet delay



transmission

A

B

propagation

nodal processing

queueing

$$d_{nodal} = d_{proc} + d_{queue} + d_{trans} + d_{prop}$$

$d_{trans}$: transmission delay:
- $L$: packet length (bits)
- $R$: link *bandwidth (bps)*
- $d_{trans} = L/R$

$d_{prop}$: propagation delay:
- $d$: length of physical link
- $s$: propagation speed in medium ($\sim 2 \times 10^8$ m/sec)
- $d_{prop} = d/s$

$d_{trans}$ and $d_{prop}$ *very* different

* Check out the Java applet for an interactive animation on trans vs. prop delay

## Queueing delay (revisited)

- *R:* link bandwidth (bps)
- *L:* packet length (bits)
- a: average packet arrival rate



traffic intensity
= La/R
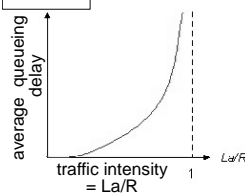
- *La/R ~ 0*: avg. queueing delay small
- *La/R -> 1*: avg. queueing delay large
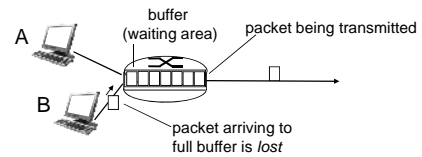- *La/R > 1*: more "work" arriving than can be serviced, average delay infinite!

La/R ~ 0

La/R -> 1

\* Check out the Java applet for an interactive animation on queuing and loss

## Packet loss

- ❖ queue (aka buffer) preceding link in buffer has finite capacity
- ❖ packet arriving to full queue dropped (aka lost)
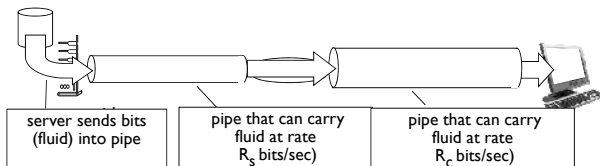- ❖ lost packet may be retransmitted by previous node, by source end system, or not at all



buffer (waiting area)   packet being transmitted

A

B

packet arriving to full buffer is *lost*

\* Check out the Java applet for an interactive animation on queuing and loss

## Throughput

- ❖ *throughput:* rate (bits/time unit) at which bits transferred between sender/receiver
  - *instantaneous:* rate at given point in time
  - *average:* rate over longer period of time



server sends bits (fluid) into pipe

pipe that can carry fluid at rate $R_s$ bits/sec)

pipe that can carry fluid at rate $R_c$ bits/sec)

## Throughput (more)

- ❖ $R_s < R_c$  What is average end-end throughput?



$R_s$ bits/sec        $R_c$ bits/sec

- ❖ $R_s > R_c$  What is average end-end throughput?



$R_s$ bits/sec        $R_c$ bits/sec
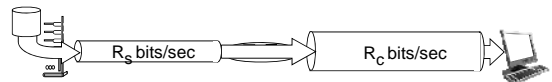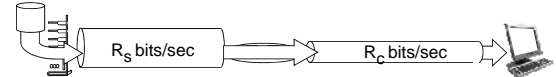
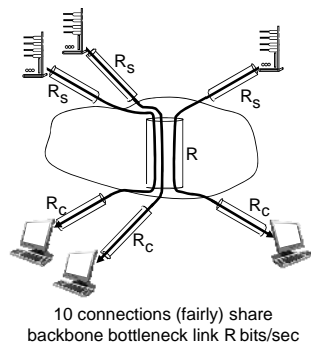*bottleneck link*
link on end-end path that constrains end-end throughput

## Throughput: Internet scenario

❖ per-connection end-end throughput: $min(R_c, R_s, R/10)$

❖ in practice: $R_c$ or $R_s$ is often bottleneck

10 connections (fairly) share backbone bottleneck link R bits/sec

## Chapter 1: roadmap

1.1 what *is* the Internet?

1.2 network edge

  ▪ end systems, access networks, links

1.3 network core

  ▪ packet switching, circuit switching, network structure

1.4 delay, loss, throughput in networks

1.5 protocol layers, service models

1.6 networks under attack: security

1.7 history

## Protocol "layers"

*Networks are complex, with many "pieces":*

  ▪ hosts
  ▪ routers
  ▪ links of various media
  ▪ applications
  ▪ protocols
  ▪ hardware, software

*Question:*

is there any hope of *organizing* structure of network?

.... or at least our discussion of networks?

## Why layering?

dealing with complex systems:

❖ explicit structure allows identification, relationship of complex system's pieces

  ▪ layered *reference model* for discussion

❖ modularization eases maintenance, updating of system

  ▪ change of implementation of layer's service transparent to rest of system

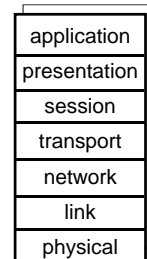❖ layering considered harmful?

3

# Internet protocol stack

❖ *application:* supporting network
   applications
   ▪ FTP, SMTP, HTTP
❖ *transport:* process-process data
   transfer
   ▪ TCP, UDP
❖ *network:* routing of datagrams
   from source to destination
   ▪ IP, routing protocols
❖ *link:* data transfer between
   neighboring  network elements
   ▪ Ethernet, 802.111 (WiFi), PPP
❖ *physical:* bits "on the wire"
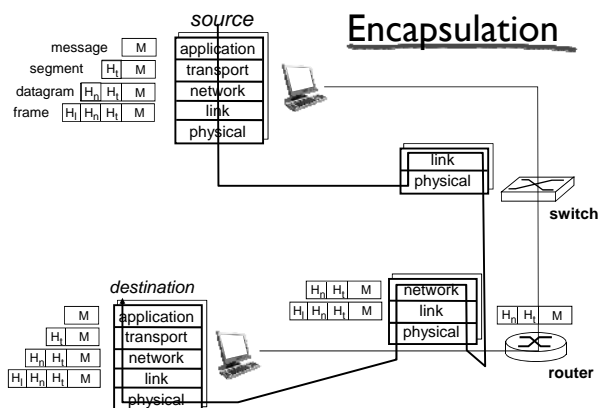
| application |
| transport |
| network |
| link |
| physical |
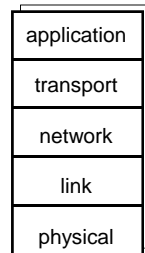
# ISO/OSI reference model

❖ *presentation:* allow applications
   to interpret meaning of data,
   e.g., encryption, compression,
   machine-specific conventions
❖ *session:* synchronization,
   checkpointing, recovery of data
   exchange
❖ Internet stack "missing" these
   layers!
   ▪ these services, *if needed,* must be
     implemented in application
   ▪ needed?

| application |
| presentation |
| session |
| transport |
| network |
| link |
| physical |

# Encapsulation

*source*

message | M
segment | $H_t$ | M
datagram | $H_n$ | $H_t$ | M
frame | $H_l$ | $H_n$ | $H_t$ | M

application
transport
network
link
physical

link
physical

**switch**

*destination*

M
$H_t$ | M
$H_n$ | $H_t$ | M
$H_l$ | $H_n$ | $H_t$ | M

application
transport
network
link
physical

$H_n$ | $H_t$ | M
$H_l$ | $H_n$ | $H_t$ | M

network
link
physical

$H_n$ | $H_t$ | M

**router**

# Chapter 1: roadmap

1.1 what *is* the Internet?
1.2 network edge

   ▪ end systems, access networks, links

1.3 network core

   ▪ packet switching, circuit switching, network structure

1.4 delay, loss, throughput in networks
1.5 protocol layers, service models
1.6 networks under attack: security
1.7 history

## Internet history

*1961-1972: Early packet-switching principles*

- ❖ 1961: Kleinrock - queueing theory shows effectiveness of packet-switching
- ❖ 1964: Baran - packet-switching in military nets
- ❖ 1967: ARPAnet conceived by Advanced Research Projects Agency
- ❖ 1969: first ARPAnet node operational

- ❖ 1972:
  - ■ ARPAnet public demo
  - ■ NCP (Network Control Protocol) first host-host protocol
  - ■ first e-mail program
  - ■ ARPAnet has 15 nodes



THE ARPA NETWORK

## Internet history

*1972-1980: Internetworking, new and proprietary nets*

- ❖ 1970: ALOHAnet satellite network in Hawaii
- ❖ 1974: Cerf and Kahn - architecture for interconnecting networks
- ❖ 1976: Ethernet at Xerox PARC
- ❖ late70's: proprietary architectures: DECnet, SNA, XNA
- ❖ late 70's: switching fixed length packets (ATM precursor)
- ❖ 1979: ARPAnet has 200 nodes

Cerf and Kahn's internetworking principles:
- ■ minimalism, autonomy - no internal changes required to interconnect networks
- ■ best effort service model
- ■ stateless routers
- ■ decentralized control

define today's Internet architecture

## Internet history

*1980-1990: new protocols, a proliferation of networks*

- ❖ 1983: deployment of TCP/IP
- ❖ 1982: smtp e-mail protocol defined
- ❖ 1983: DNS defined for name-to-IP-address translation
- ❖ 1985: ftp protocol defined
- ❖ 1988: TCP congestion control

- ❖ new national networks: Csnet, BITnet, NSFnet, Minitel
- ❖ 100,000 hosts connected to confederation of networks

## Internet history

*1990, 2000's: commercialization, the Web, new apps*

- ❖ early 1990's: ARPAnet decommissioned
- ❖ 1991: NSF lifts restrictions on commercial use of NSFnet (decommissioned, 1995)
- ❖ early 1990s: Web
  - ■ hypertext [Bush 1945, Nelson 1960's]
  - ■ HTML, HTTP: Berners-Lee
  - ■ 1994: Mosaic, later Netscape
  - ■ late 1990's: commercialization of the Web

late 1990's – 2000's:
- ❖ more killer apps: instant messaging, P2P file sharing
- ❖ network security to forefront
- ❖ est. 50 million host, 100 million+ users
- ❖ backbone links running at Gbps

## Internet history

*2005-present*
- ❖ ~750 million hosts
  - ▪ Smartphones and tablets
- ❖ Aggressive deployment of broadband access
- ❖ Increasing ubiquity of high-speed wireless access
- ❖ Emergence of online social networks:
  - ▪ Facebook: soon one billion users
- ❖ Service providers (Google, Microsoft) create their own networks
  - ▪ Bypass Internet, providing "instantaneous" access to search, emai, etc.
- ❖ E-commerce, universities, enterprises running their services in "cloud" (eg, Amazon EC2)

## Introduction: summary

*covered a "ton" of material!*
- ❖ Internet overview
- ❖ what's a protocol?
- ❖ network edge, core, access network
  - ▪ packet-switching versus circuit-switching
  - ▪ Internet structure
- ❖ performance: loss, delay, throughput
- ❖ layering, service models
- ❖ security
- ❖ history

*you now have:*
- ❖ context, overview, "feel" of networking
- ❖ more depth, detail *to follow!*