Chapter 1: Introduction

Our goal:

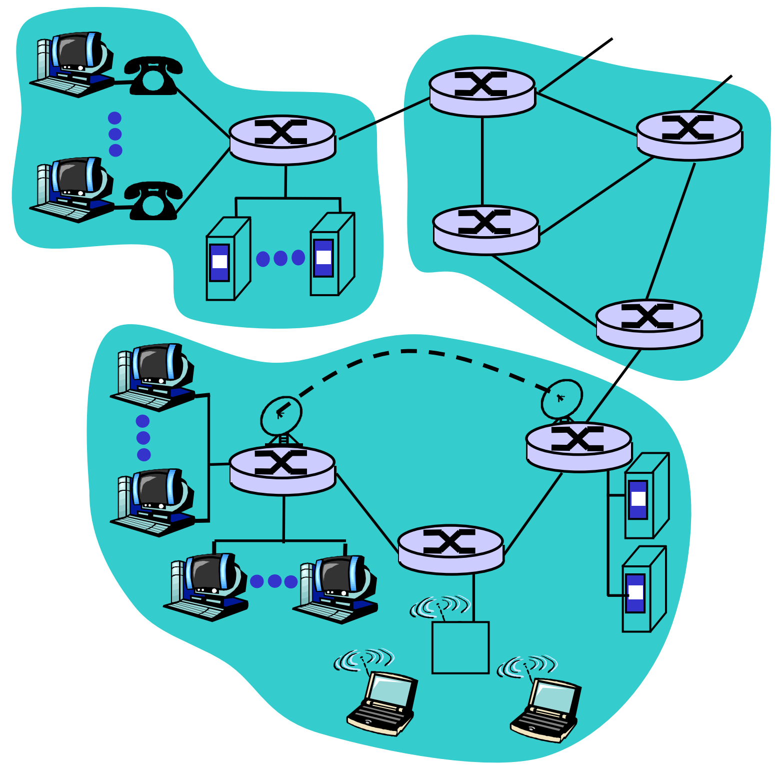
- learn basic network terminologies
- more depth, detail *later* in course
- approach:
  - use Internet as example

Chapter 1: Introduction

1 What *is* the Internet?
2 Network edge
3 Network core
4 Internet structure and ISPs
5 Protocol layers, service models
6 Delay & loss in packet-switched networks

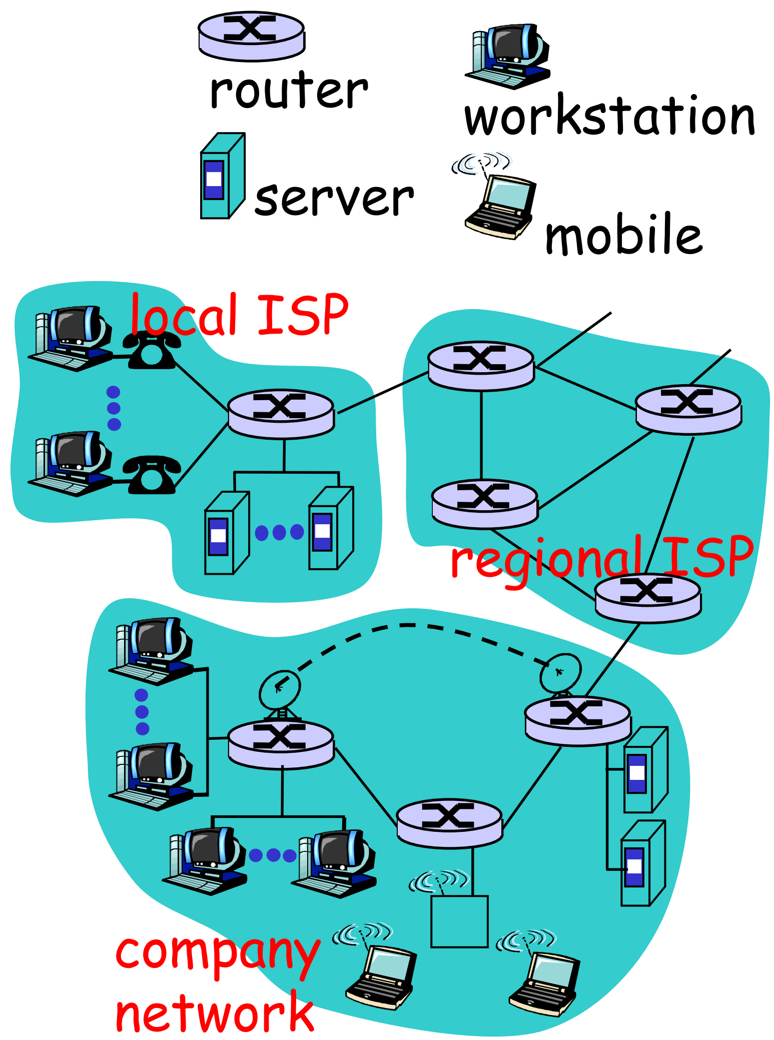
What's the Internet: a "service" view

- communication *infrastructure* enables distributed apps:
  - Enables apps to communicate
  - Web, email, games, e-commerce, file sharing
- communication services provided to apps:
  - Offers services



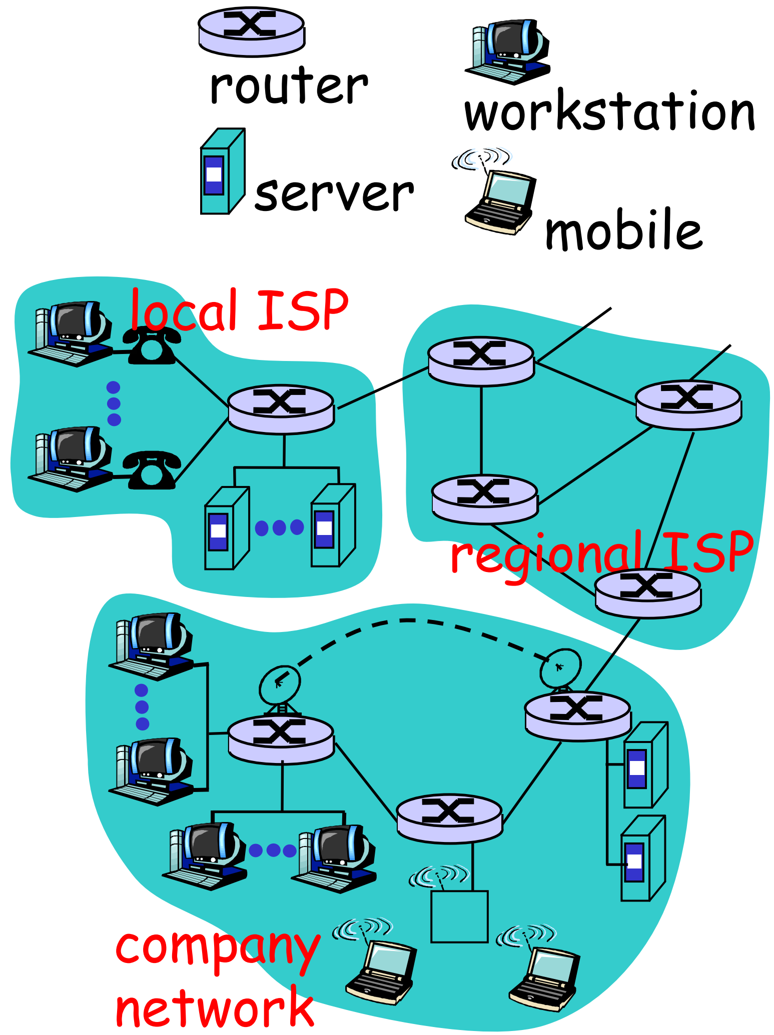
What's the Internet: "nuts and bolts" view

- millions of connected computing devices: called *hosts* or *end systems*
  - e.g., Laptops, workstations
  - running *network apps*
- *routers & switches:*
  - forward packets (chunks of data)
- *communication links*
  - e.g., fiber, copper, radio, satellite



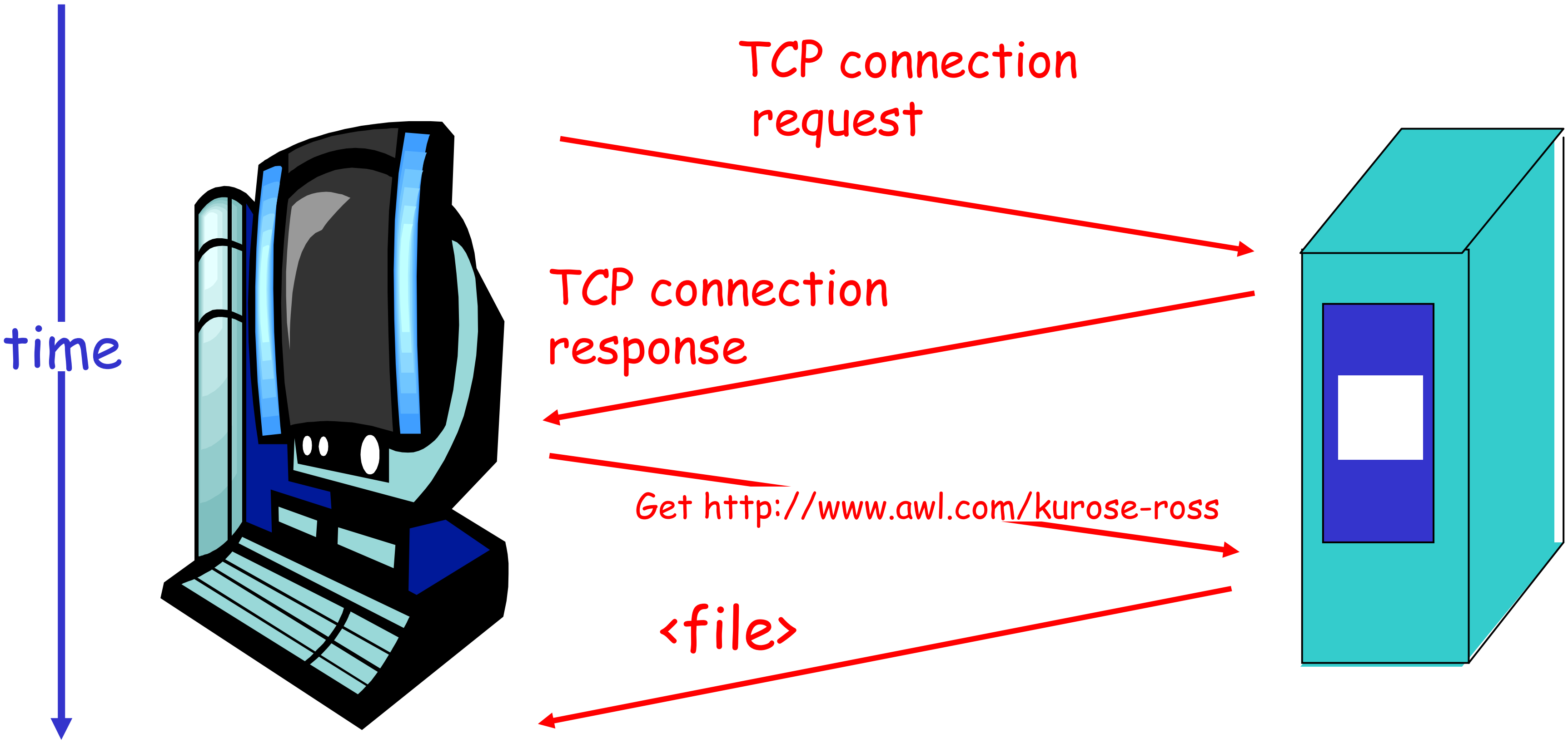
What's the Internet: "nuts and bolts" view

- Internet standards
  - IETF (Internet Eng. Task Force)
    - RFC: Request for comments
  - IEEE: for links/hardware E.g., Ethernet
- *network protocols*
  - control sending/receiving of messages
  - e.g., TCP, IP, HTTP, FTP, PPP



What's a protocol?

a computer network protocol:



What's a protocol?

human protocols:

- "what's the time?"
- "I have a question"
- introductions

… specific msgs sent

… specific actions taken when msgs received, or other events

network protocols:

- machines rather than humans
- all communication activity in Internet governed by protocols

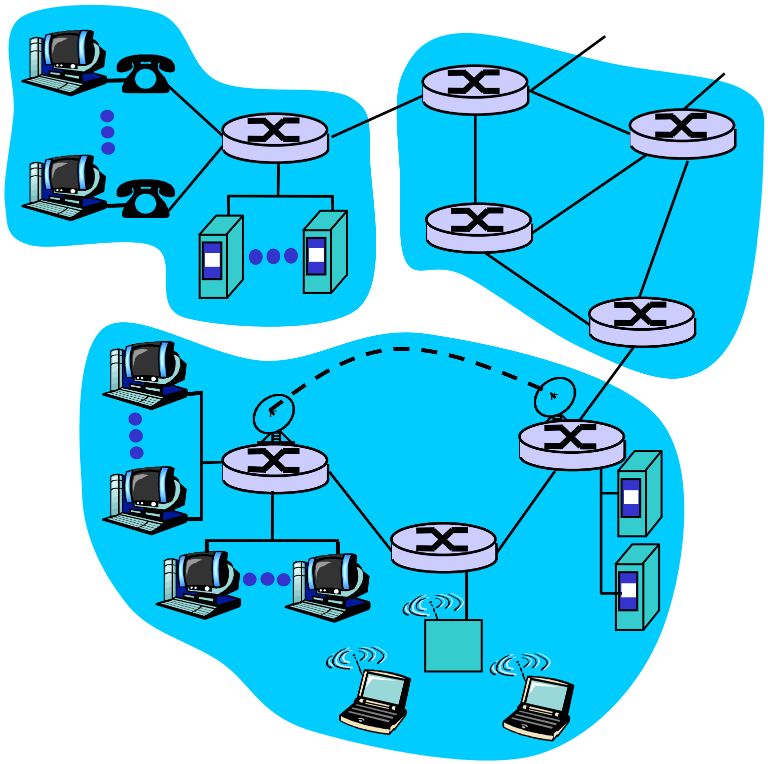
protocols define (1) format, order of msgs sent and received among network entities, and (2) actions taken on msg transmission, receipt

Chapter 1: Introduction

1 What *is* the Internet?
2 Network edge
3 Network core
4 Internet structure and ISPs
5 Protocol layers, service models
6 Delay & loss in packet-switched networks

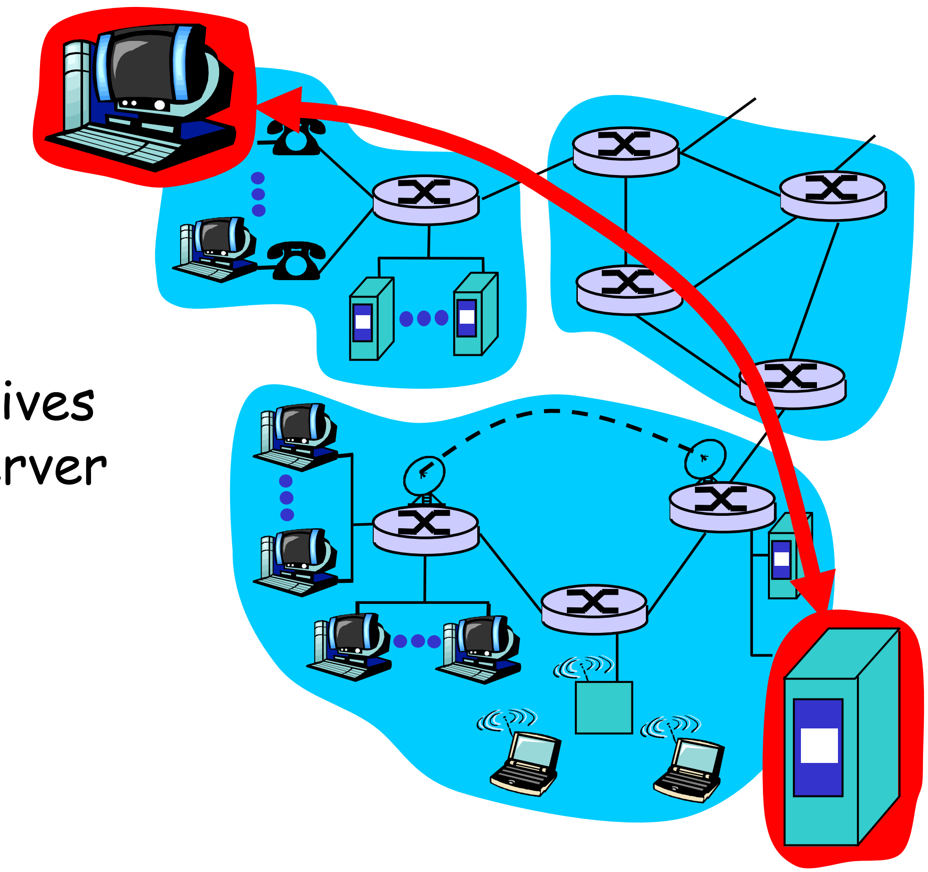
A closer look at network structure:

- network edge: applications and hosts
- network core:
  - routers
  - network of networks
- access networks, physical media: communication links



The network edge: service models

- end systems (hosts):
  - run application programs
  - e.g. Web, email
  - at “edge of network”



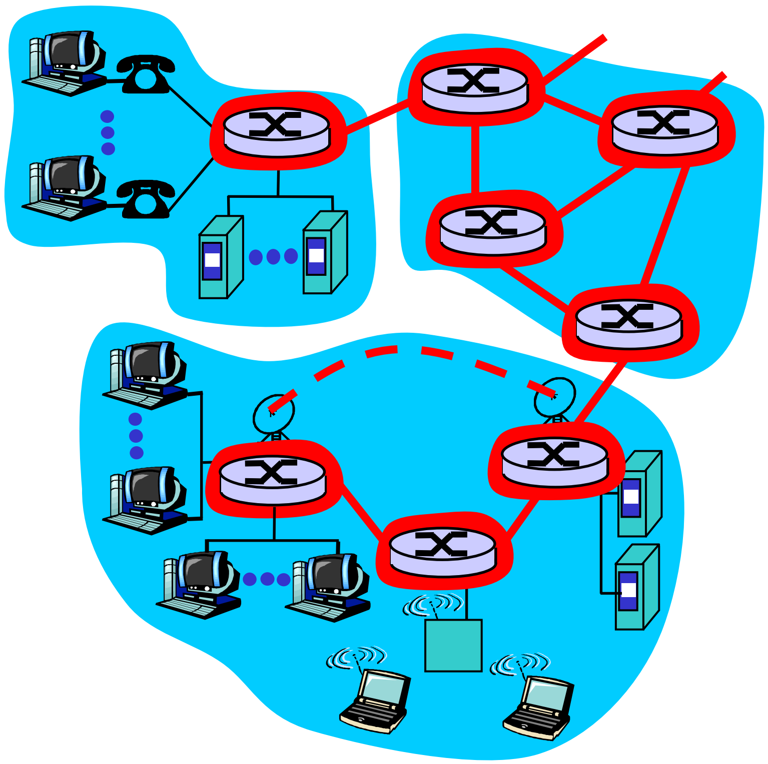
- client/server model
  - client host requests, receives service from always-on server
  - e.g. Web browser/server; email client/server
- peer-to-peer model:
  - minimal (or no) use of dedicated servers
  - e.g. Skype, BitTorrent, KaZaA

Chapter 1: Introduction

1 What *is* the Internet?
2 Network edge
3 Network core
4 Network access and physical media
5 Internet structure and ISPs
6 Protocol layers, service models
7 Delay & loss in packet-switched networks

The Network Core

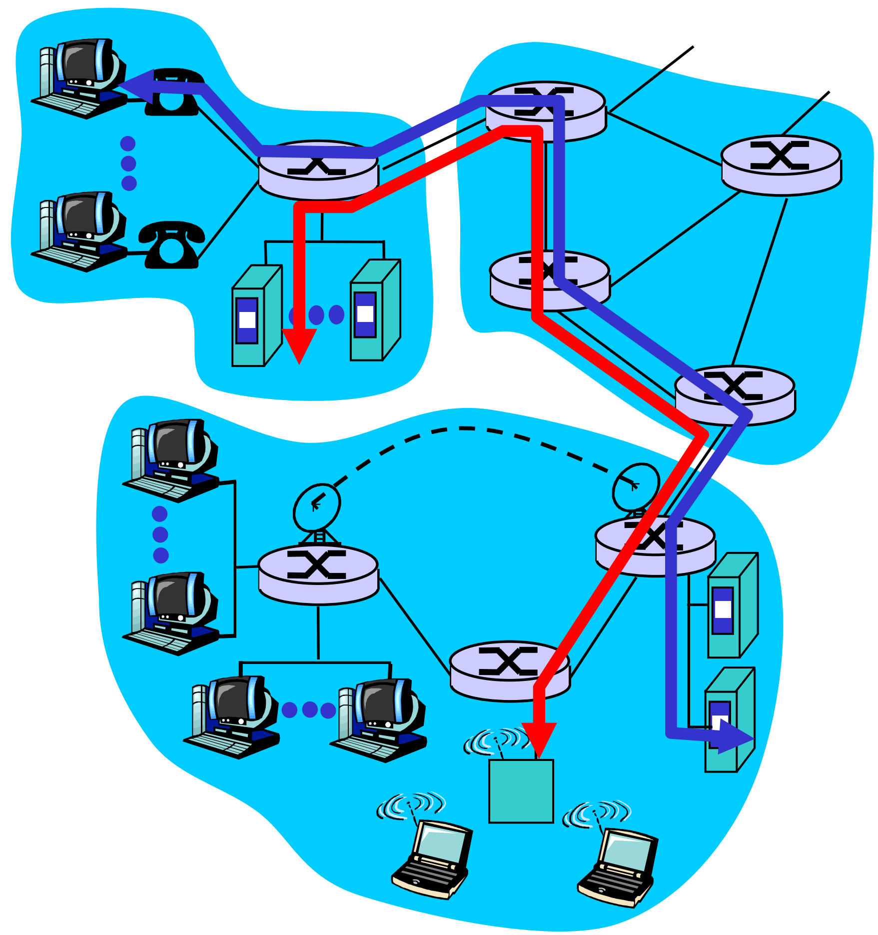
- mesh of interconnected routers
- *the* fundamental question: how is data transferred through net?
  - circuit switching: dedicated circuit per call: telephone net
  - packet-switching: data sent thru net in discrete "chunks"



Network Core: Circuit Switching

End-end resources reserved for "call"

- dedicated resources: no sharing
- call setup required
- circuit-like (guaranteed) performance
- same path for all chunks

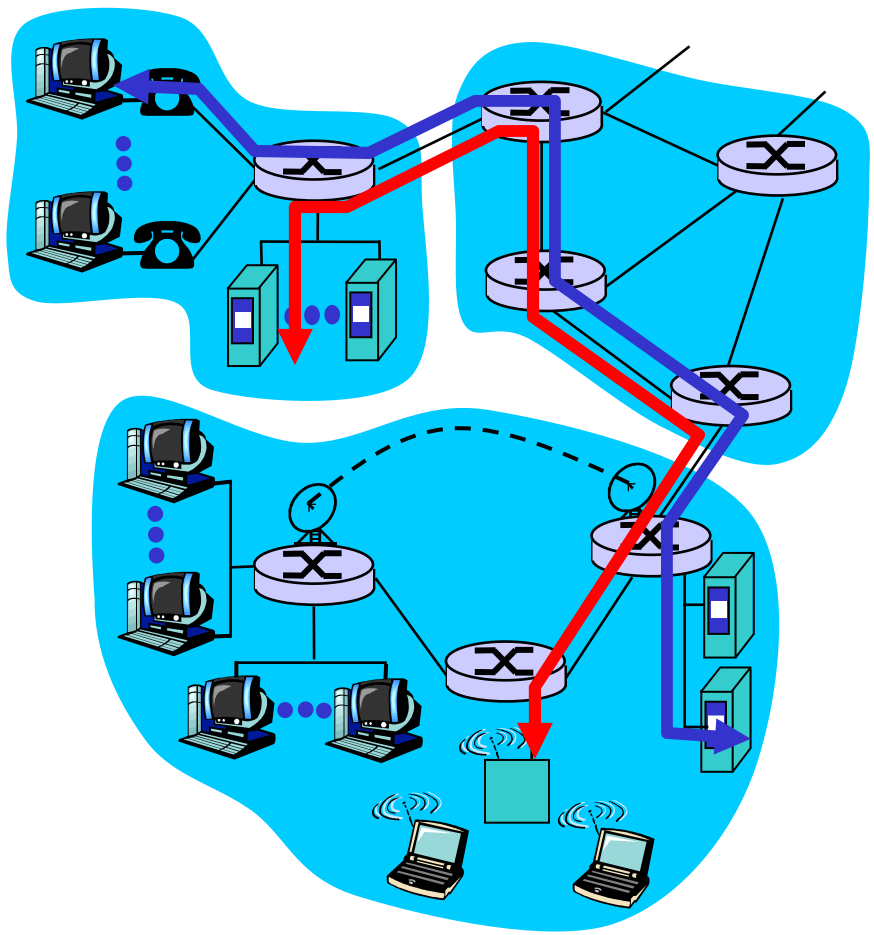


Network Core: Circuit Switching

network resources (e.g., bandwidth) divided into "pieces"

- allocated pieces per call
- no sharing

resource piece *idle* if not used by owning call

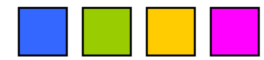


Network Core: Circuit Switching

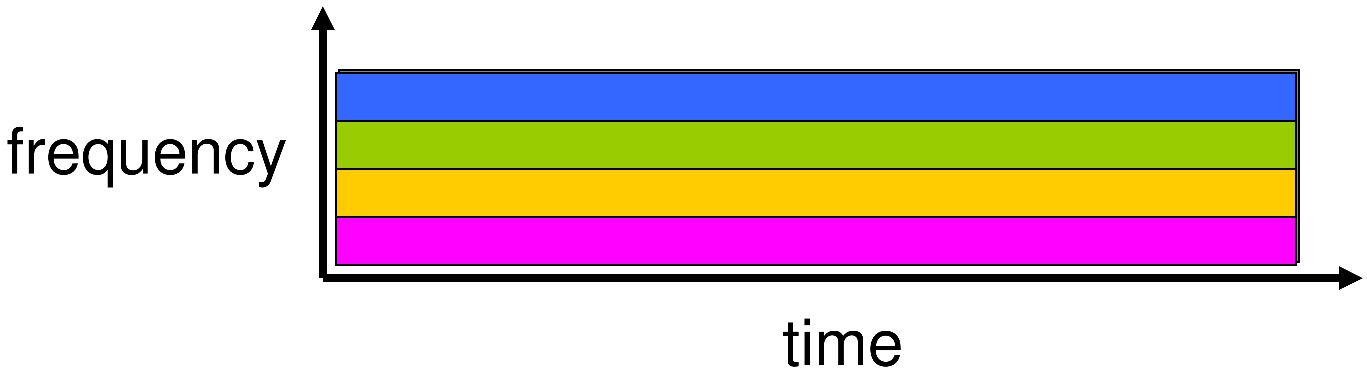
- Two ways of dividing bandwidth into "pieces"
  - frequency division
  - time division

Circuit Switching: FDM and TDM

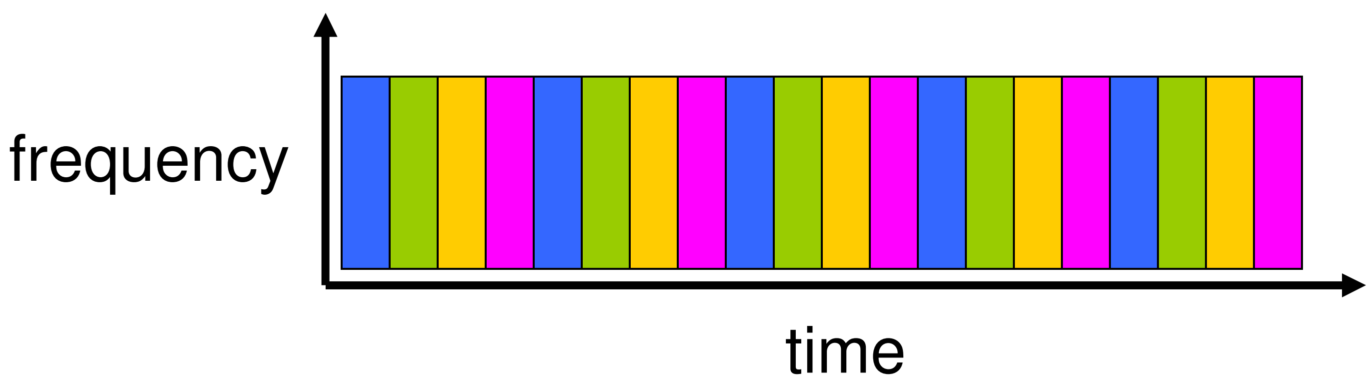
Example:

4 users 

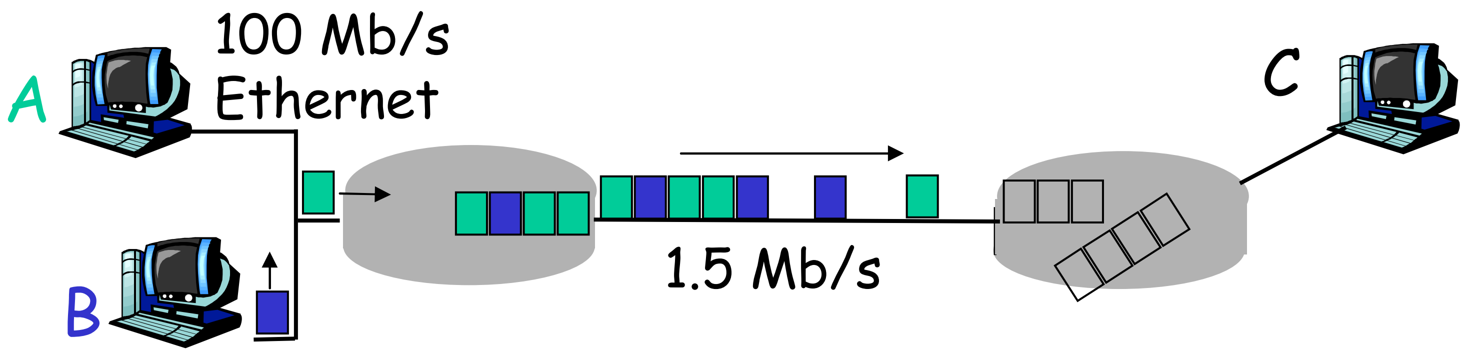
Freq. Division Multiplx. (FDM)



Time Division Multiplx. (TDM)



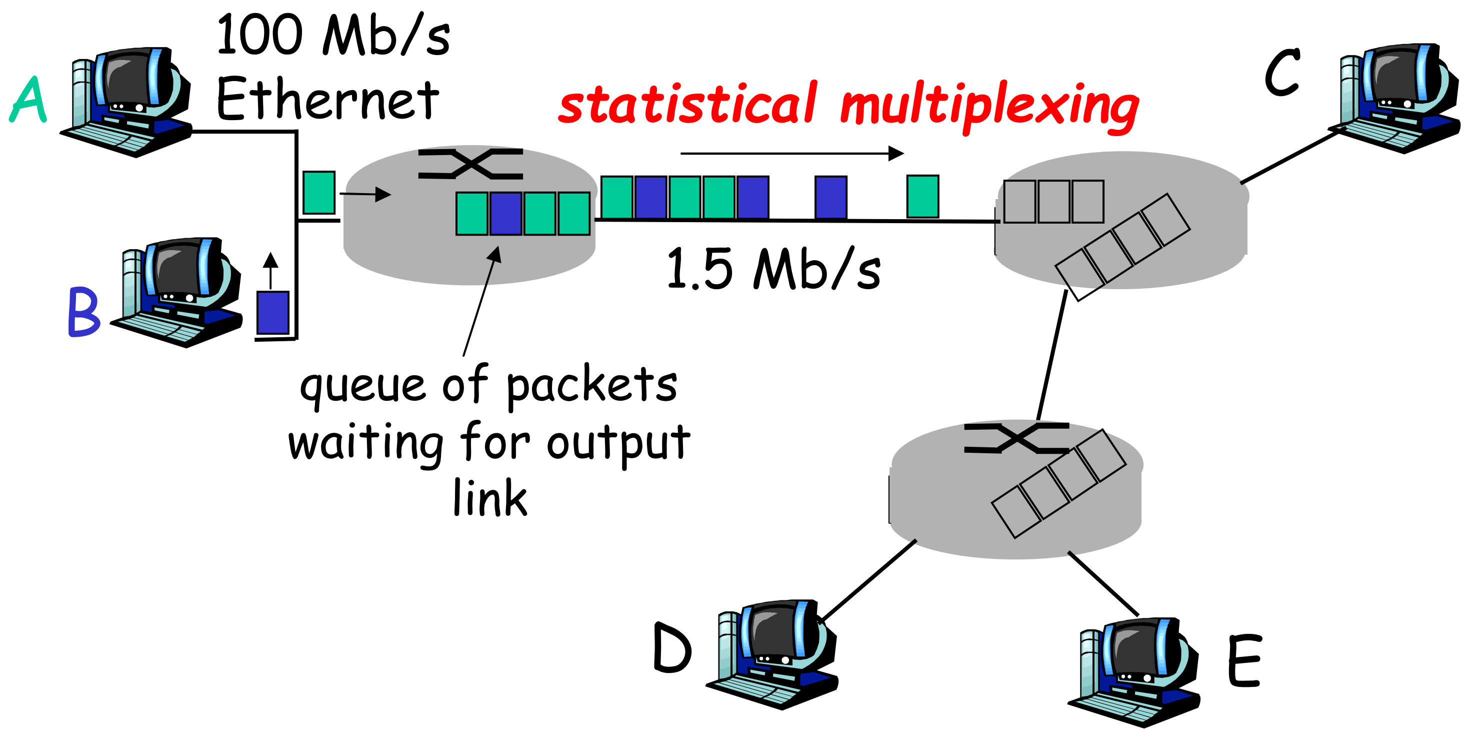
Network Core: Packet Switching



each end-to-end data stream is divided into *packets*

- no dedication/reservation: all streams *share* resources
- no setup is required
- resources used as needed
- each packet uses full link bandwidth
- aggregate resource demand can exceed capacity
- no guarantee

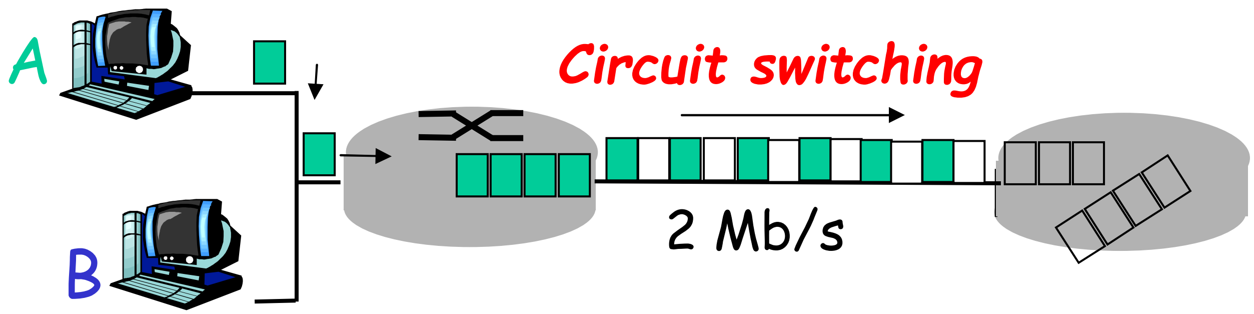
Network Core: statistical multiplexing



Sequence of A & B packets does not have fixed pattern, shared on demand ➔ ***statistical multiplexing***.

Whereas in TDM, each host gets same slot (periodically)

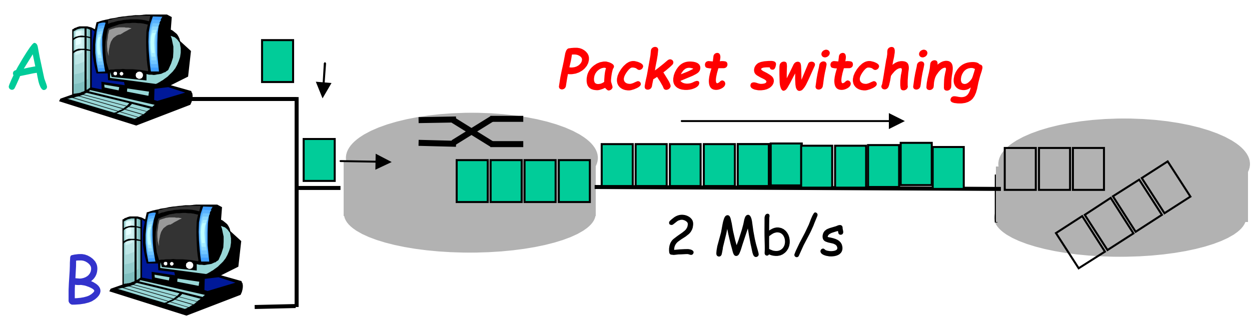
Packet switching versus circuit switching



- 2 circuits (use TDM)
- A reserves 1 circuit
- B reserves 1 circuit

Utilization = 50% only = 1 Mb/s

B: has no packets to send



- statistical multiplex.
- B uses full link since A is not using it

Utilization = 100% = 2 Mb/s

Packet switching versus circuit switching

	Packet-switching	Circuit-switching
Resources	sharing	dedicated
Congestion	may lead to it	admission control
Overhead	less overhead; no connection setup	more overhead; reserve resources 1st
Guarantee	Best-effort no guarantee	provide guarantee good for multimedia

Numerical example

- How long does it take to send a file of 640,000 bits from host A to host B over a circuit-switched network?
  - The link's transmission rate = 0.64 Mbps
  - Each link uses TDM with 10 slots/sec
  - 0.5 sec to establish end-to-end circuit

Let's work it out! You have few minutes!

- Solution:

- Bandwidth of circuit (in kbps)= .64x1000/10 = 64 kbps
- Time to send: 640 kbits/64 kbps + 0.5s = 10.5s

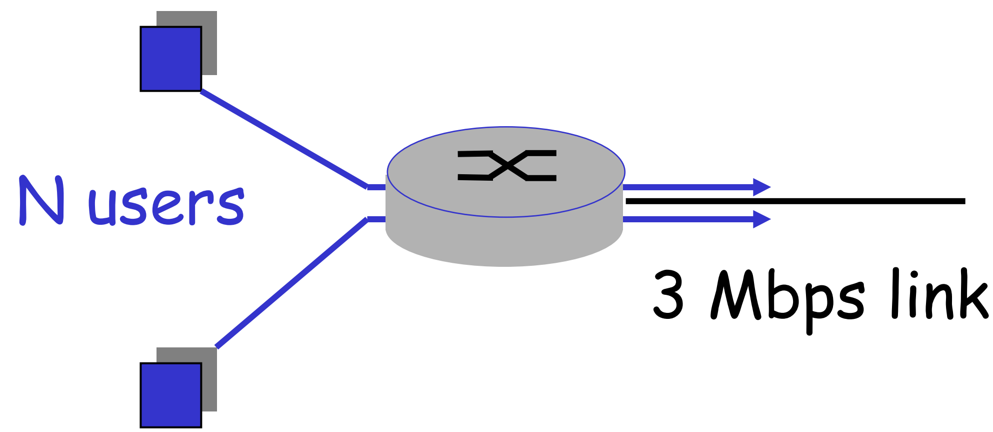
Packet switching versus circuit switching

	Packet-switching	Circuit-switching
Resources	sharing	dedicated
Congestion	may lead to it	admission control
Overhead	less overhead; no connection setup	more overhead; reserve resources 1st
Guarantee	Best-effort no guarantee	provide guarantee good for multimedia

Packet switching versus circuit switching

Packet switching allows more users to use network!

- 3 Mb/s link
- each user:
  - 1 Mb/s when "active"
  - active 1/3 of time
- circuit-switching:
  - 3 users
- packet switching:
  - With N=4 users, what are the chances that a user won't get 1 Mb/s? I.e., what is the prob. that more than 3 (strictly) users are active?
  - With N=5 users, what are the chances that a user won't get 1 Mb/s?
  - With N=6 users, what are the chances that a user won't get 1 Mb/s?



Packet switching versus circuit switching

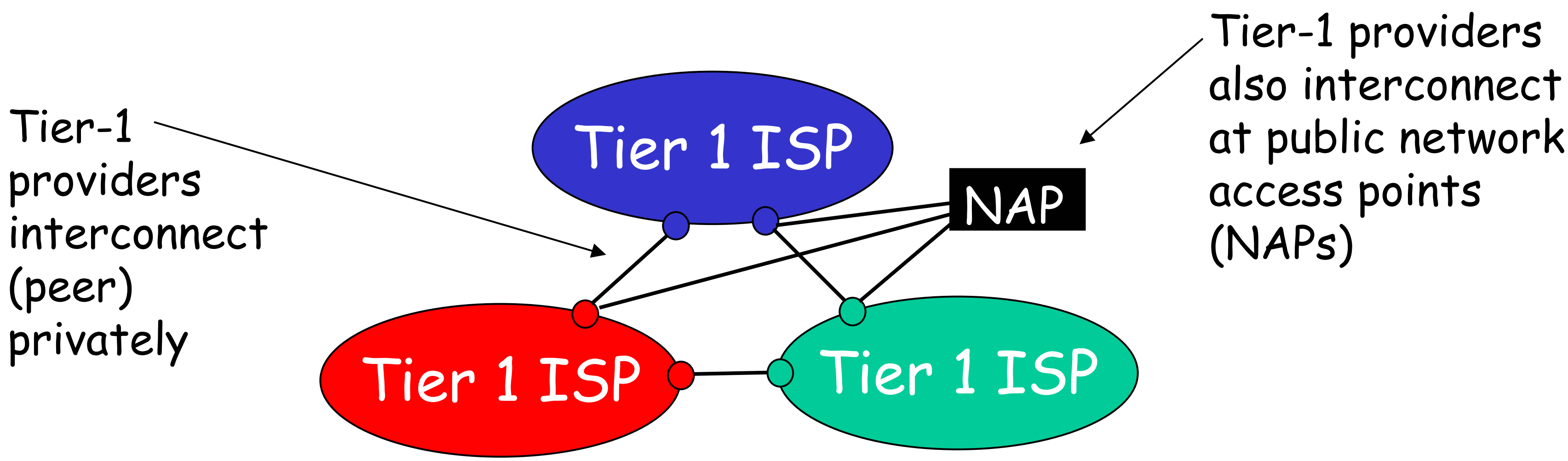
Board ...

Chapter 1: Introduction

1 What *is* the Internet?
2 Network edge
3 Network core
4 Internet structure and ISPs
5 Protocol layers, service models
6 Delay & loss in packet-switched networks

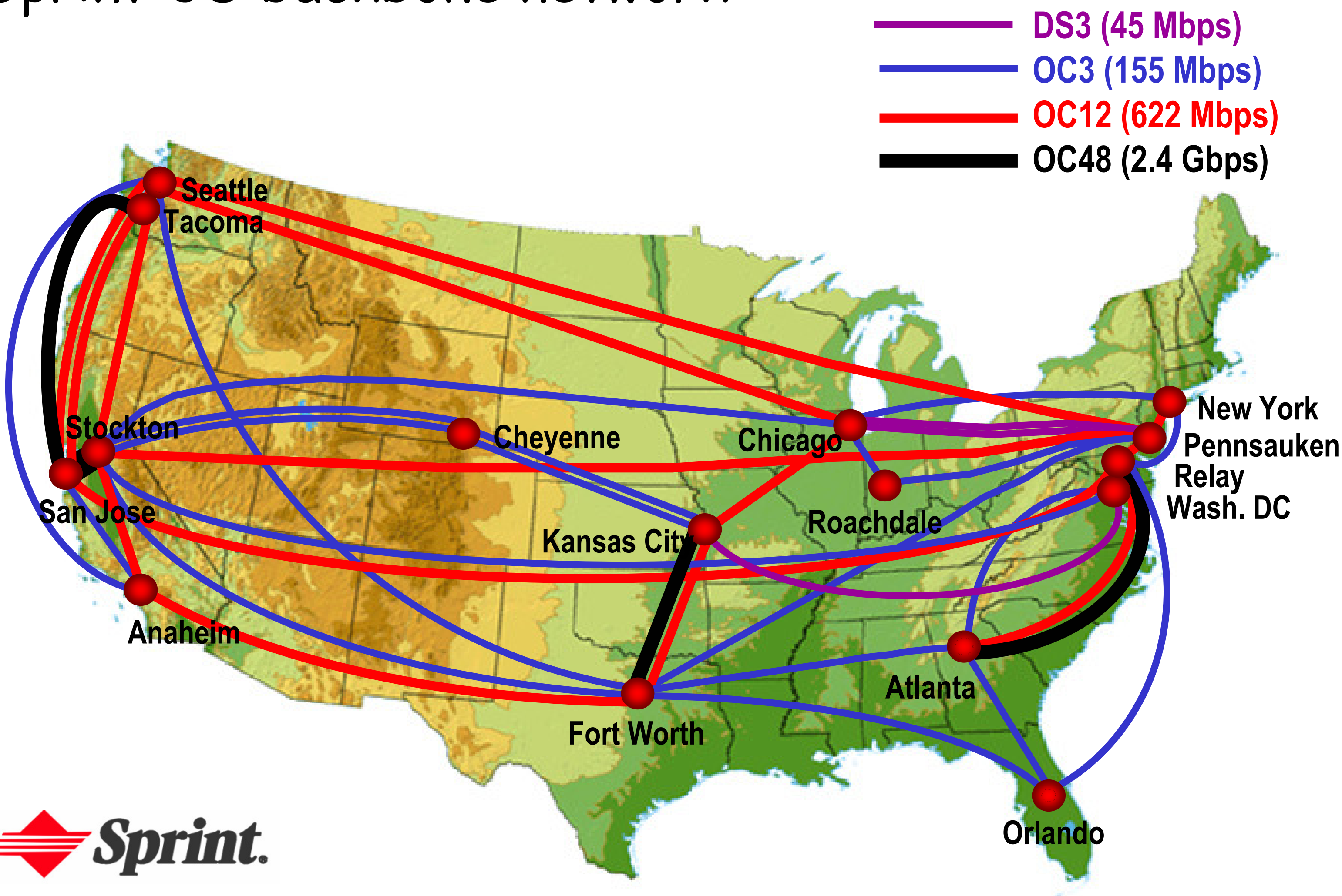
Internet structure: network of networks

- roughly hierarchical: tier 1, tier 2, and tier 3
- at center: "tier-1" ISPs
  - e.g., MCI, Sprint, AT&T, Cable and Wireless,
  - national/international coverage



Tier-1 ISP: e.g., Sprint

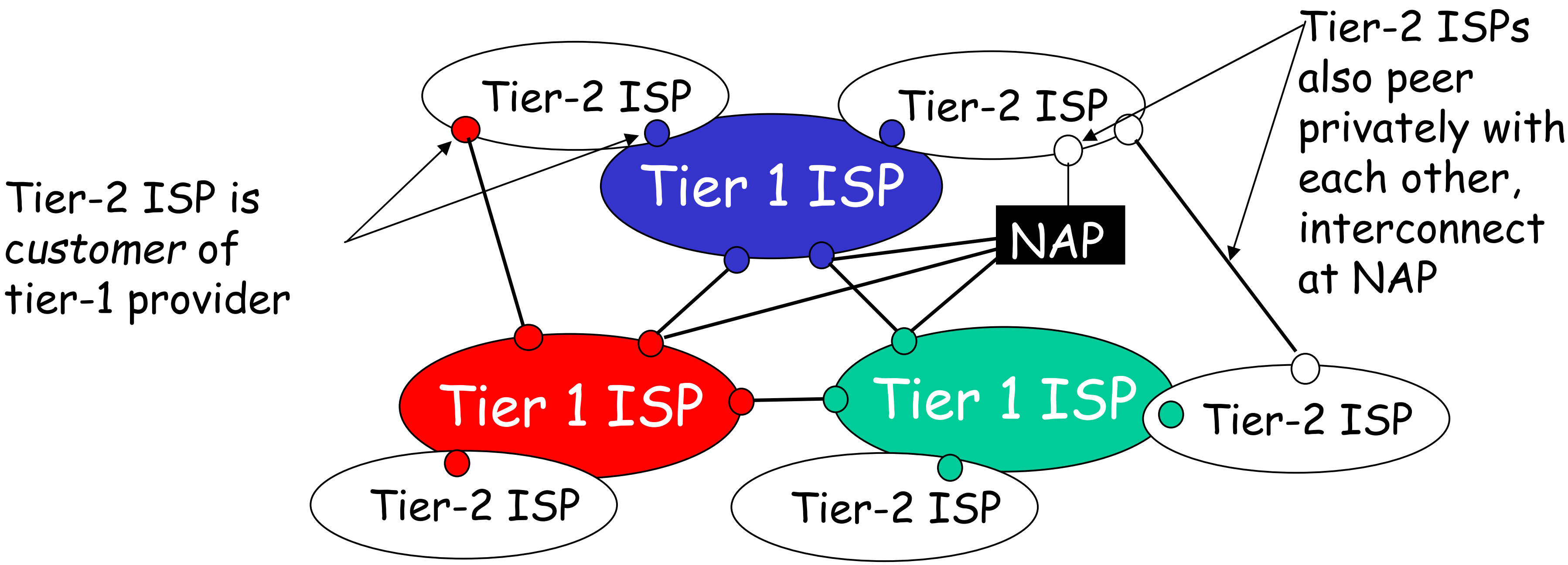
Sprint US backbone network



Sprint.

Internet structure: network of networks

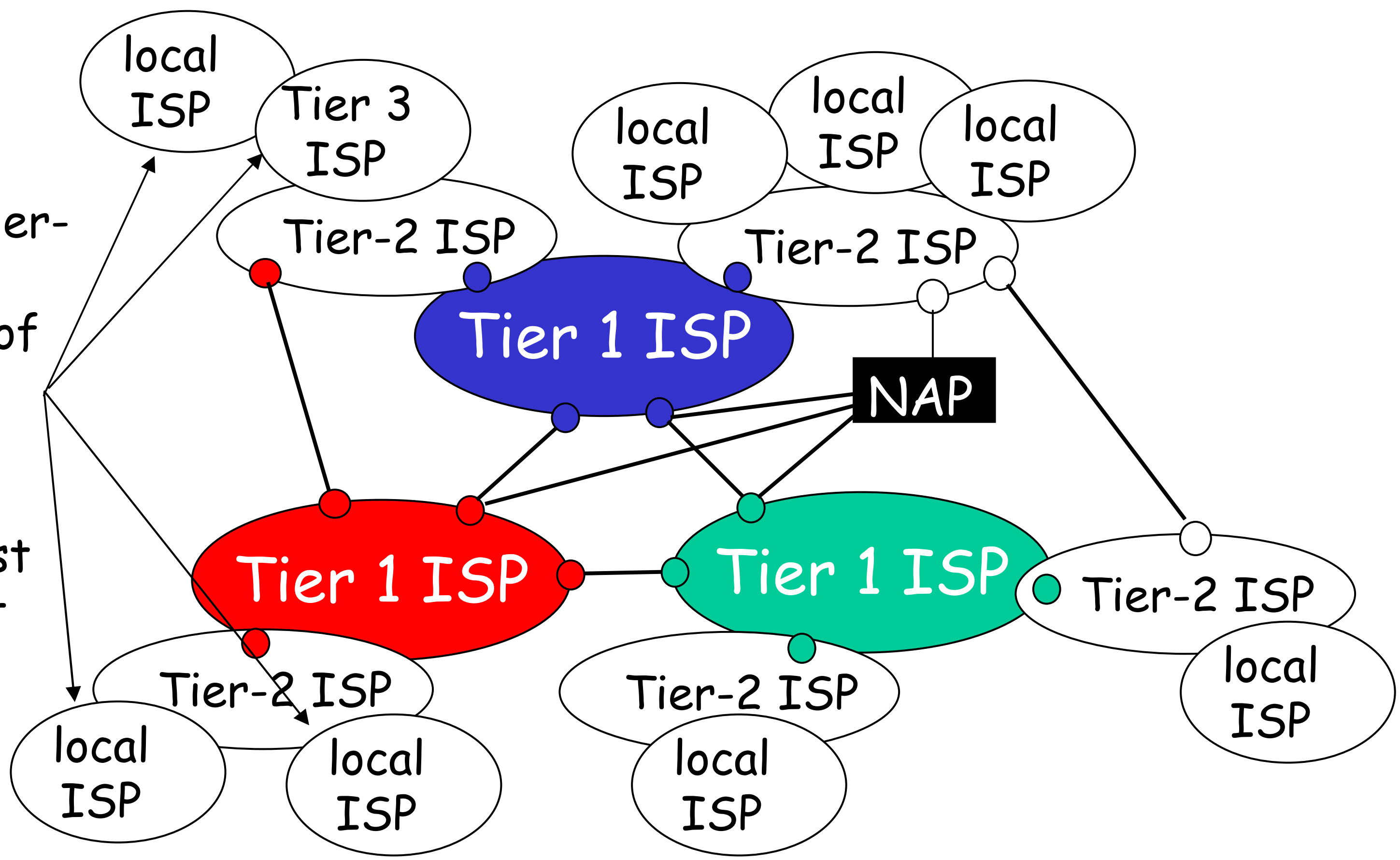
- "Tier-2" ISPs: smaller (often regional) ISPs
  - Connect to one or more tier-1 ISPs, possibly other tier-2 ISPs



Internet structure: network of networks

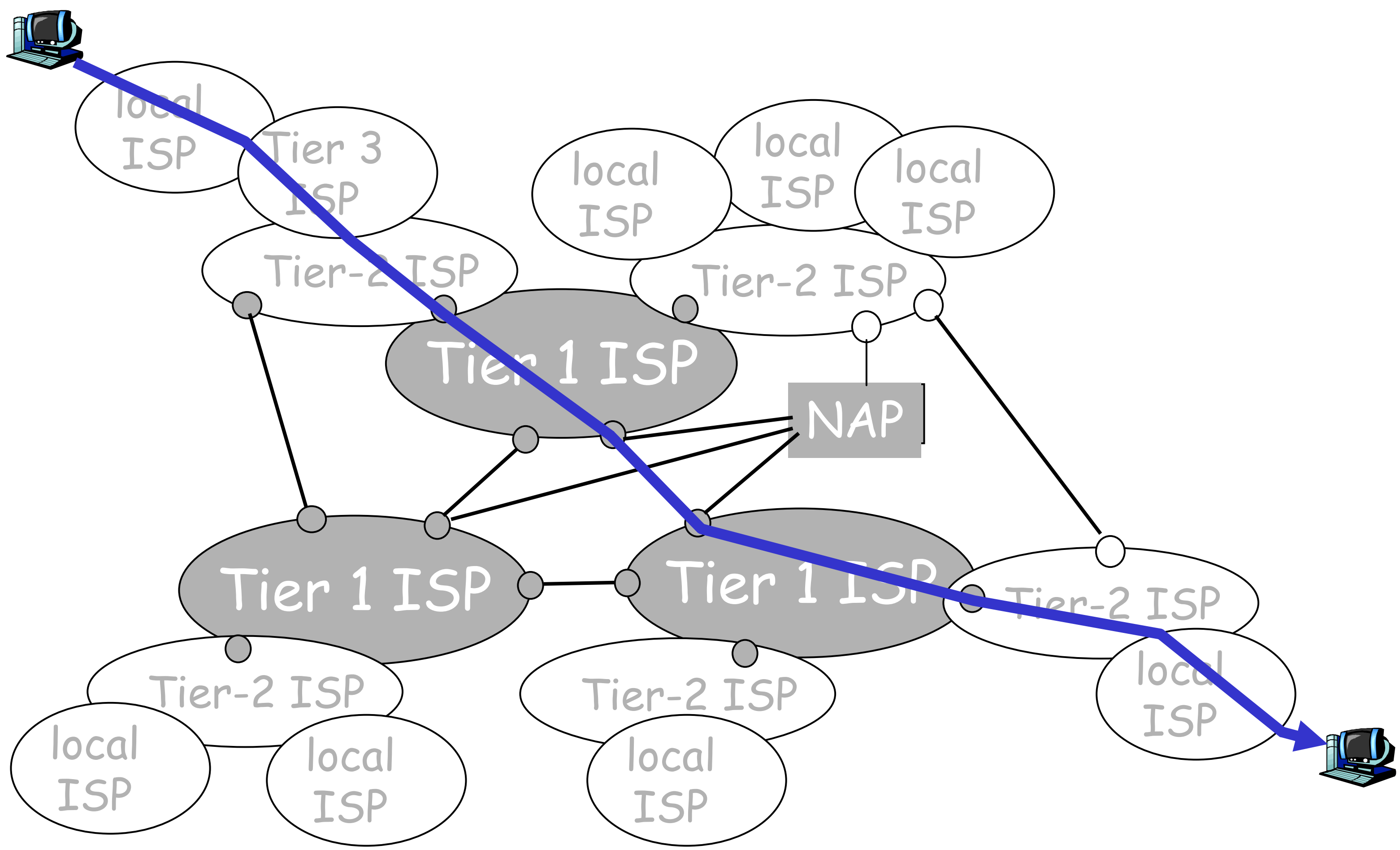
- "Tier-3" ISPs and local ISPs
  - last hop ("access") network (closest to end systems)

Local and tier-3 ISPs are *customers* of higher tier ISPs connecting them to rest of Internet



Internet structure: network of networks

- a packet passes through many networks!



Chapter 1: Introduction

1 What *is* the Internet?

2 Network edge

3 Network core

4 Internet structure and ISPs

5 Protocol layers, service models

6 Delay & loss in packet-switched networks

Protocol "Layers"

Networks are complex!

- many "pieces":
  - hosts
  - routers
  - links of various media
  - applications
  - protocols
  - hardware, software

Question:

Is there any hope of an *organizing* structure of network?

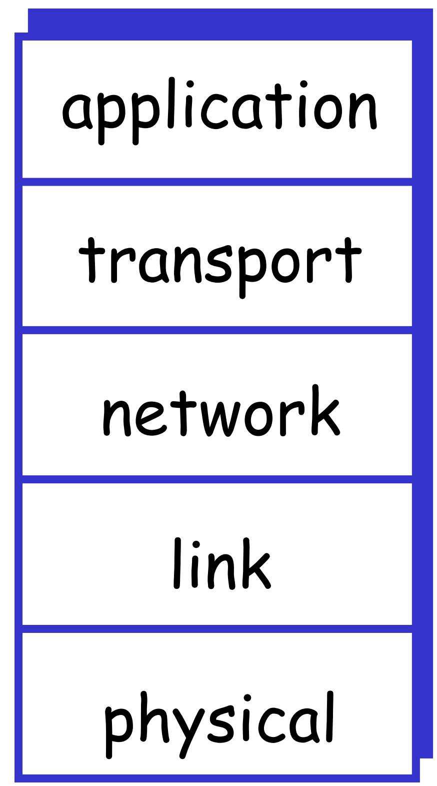
Why layering?

Dealing with complex systems:

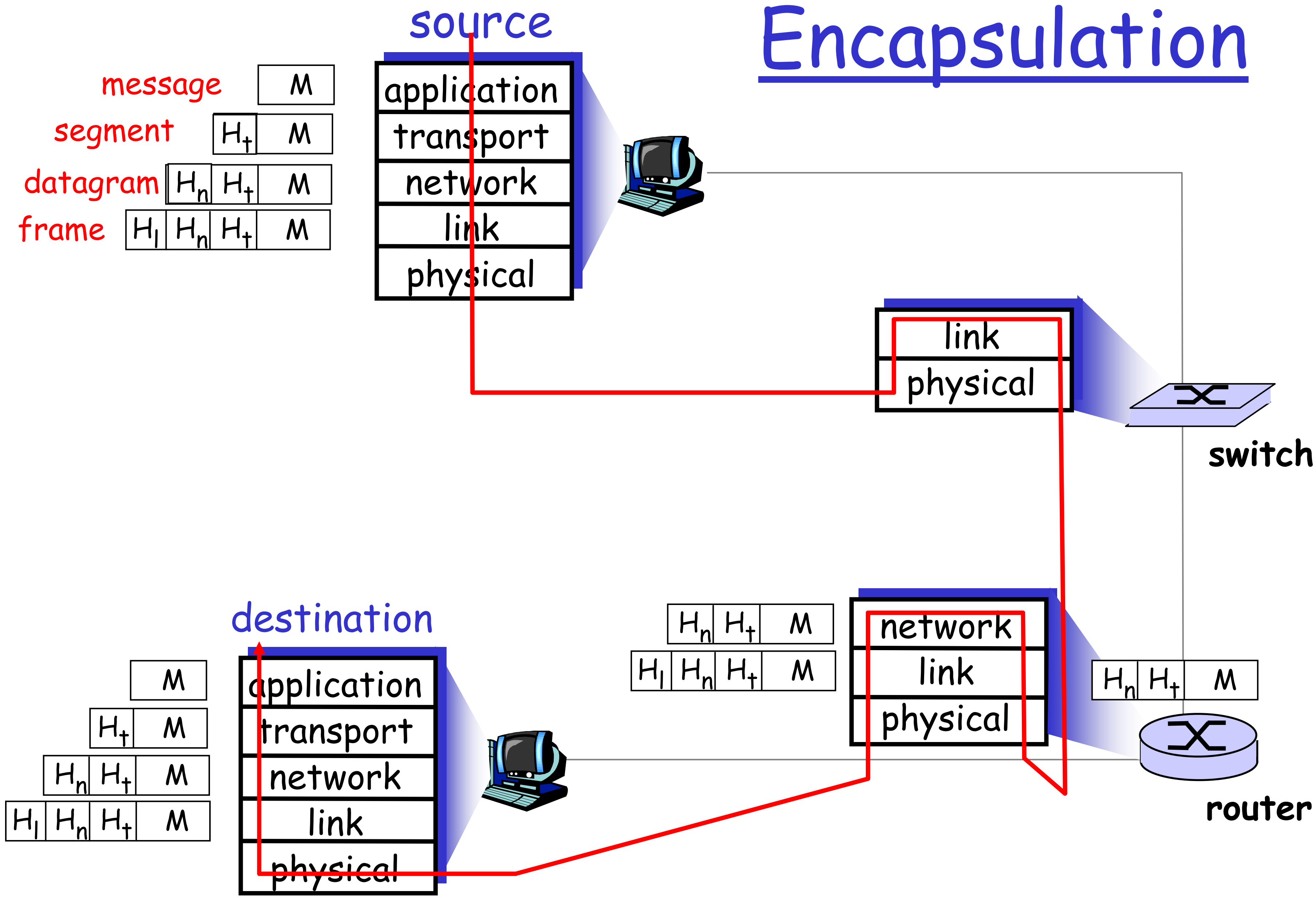
- Easing assignment of tasks
  - identify relationship among pieces of complex systems
- Easing maintenance, updating of system
  - change of implementation of layer's service transparent to rest of system
  - e.g., change in gate procedure doesn't affect rest of system

Internet protocol stack

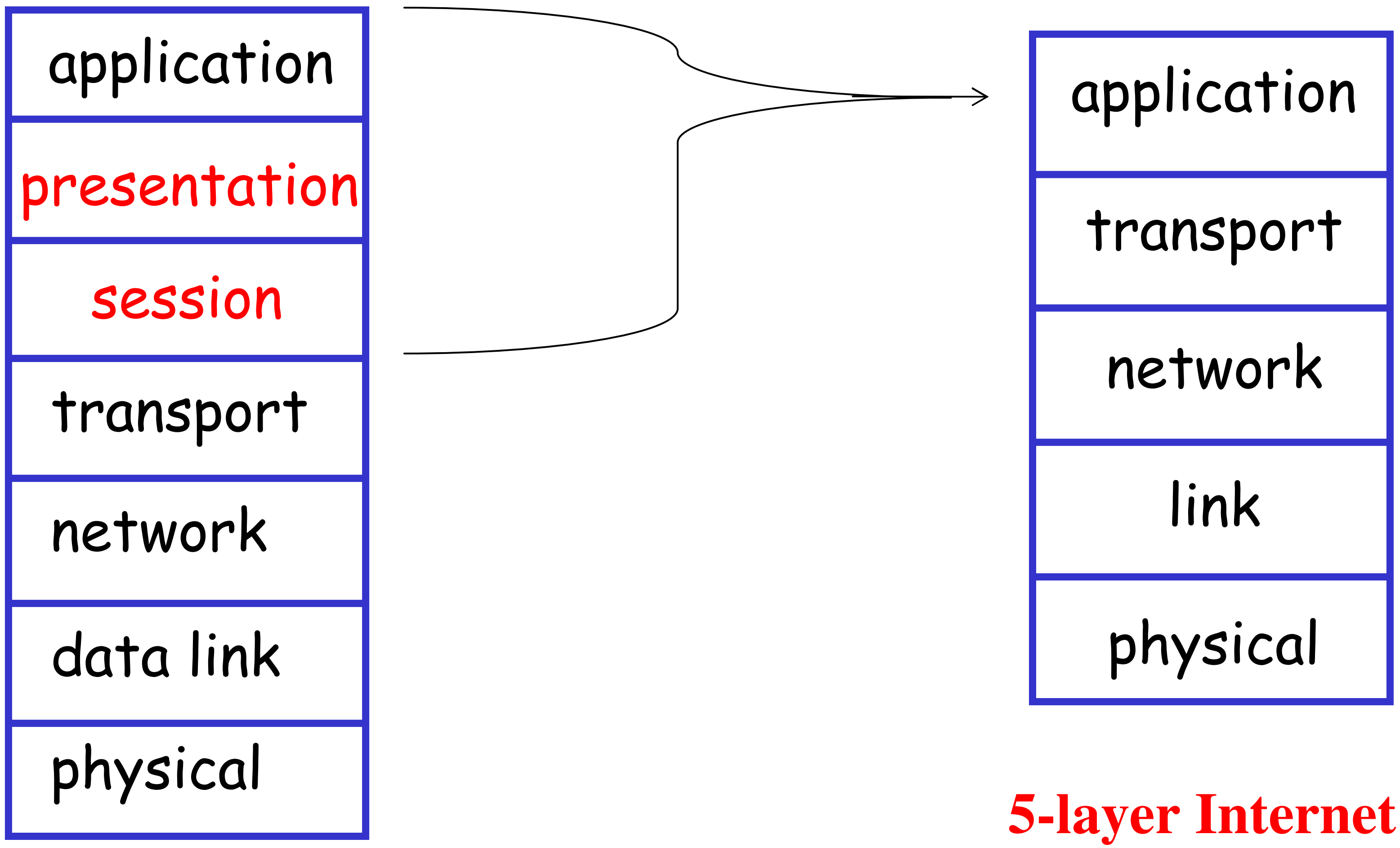
- application: supporting network applications
  - FTP, SMTP, HTTP
- transport: process-process data transfer
  - TCP, UDP
- network: routing of datagrams from source to destination
  - IP, routing protocols
- link: data transfer between neighboring network elements
  - PPP, Ethernet
- physical: bits "on the wire"



Encapsulation



ISO/OSI Model: late 70's



5-layer Internet Protocol Stack

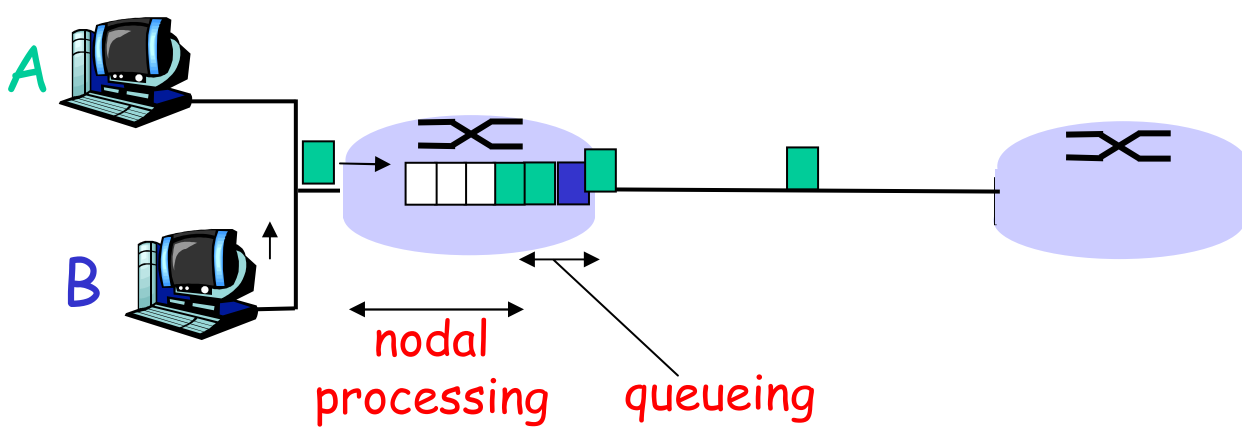
7-layer ISO/OSI model
(OSI: open system interconnections)

Chapter 1: Introduction

1 What *is* the Internet?
2 Network edge
3 Network core
4 Internet structure and ISPs
5 Protocol layers, service models
6 Delay & loss in packet-switched networks

Sources of packet delay

- 1. processing:
  - check bit errors
  - determine output link
- 2. queueing
  - time waiting at output link for transmission
  - depends on congestion level of router

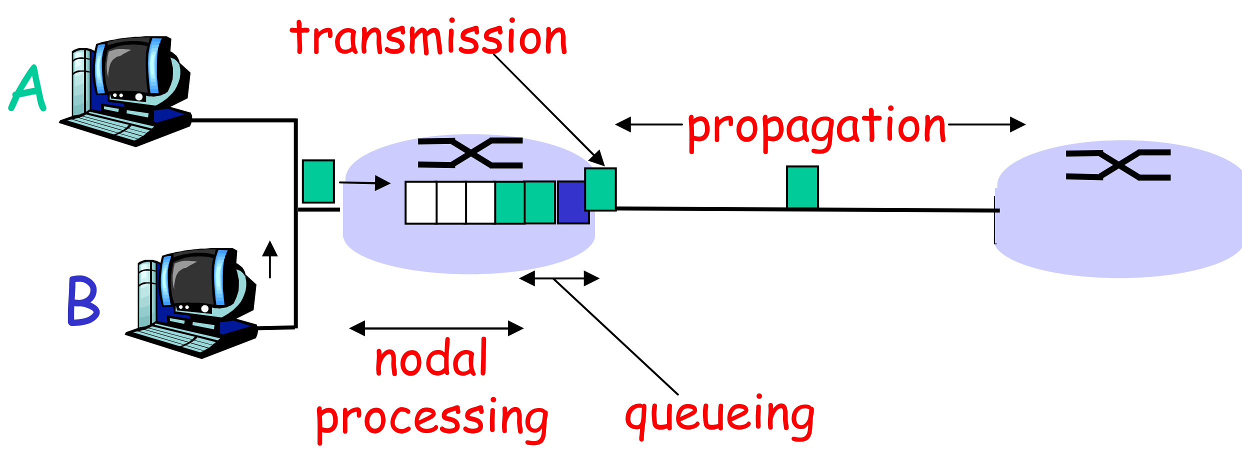


Sources of packet delay

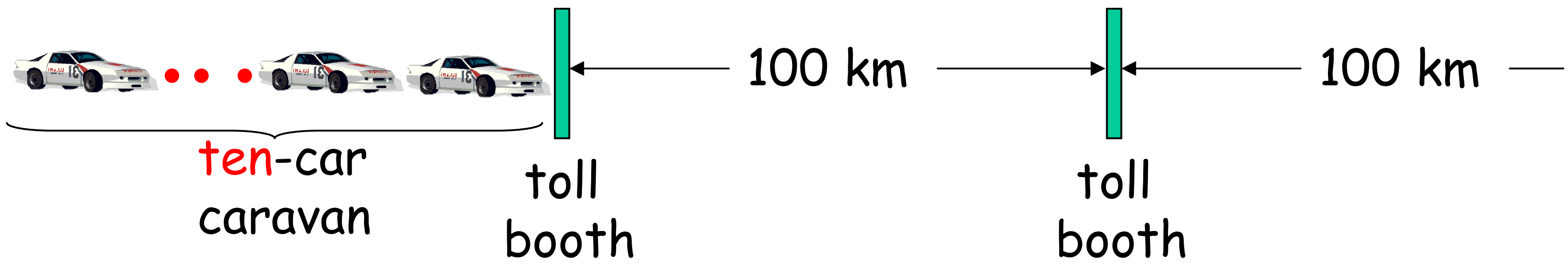
3. Transmission delay:
- R=link bandwidth (bps)
- L=packet length (bits)
- trans. delay = L/R

4. Propagation delay:
- d = length of physical link
- s = propagation speed in medium (~2×10^8 m/sec)
- propagation delay = d/s

Note: s and R are *very* different quantities!

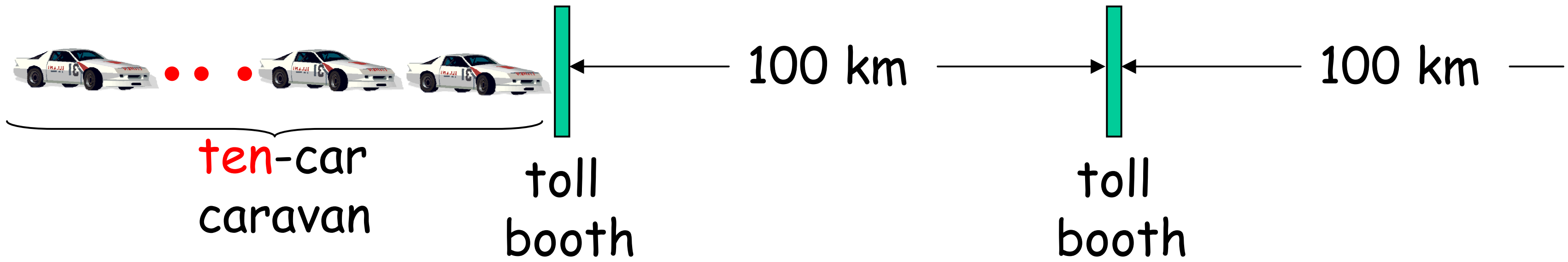


Caravan analogy



- Cars run at 100 km/hr (speed of propagation)
- Booth takes 12 sec to service a car (transmission time)
- Car ~ bit; caravan ~ packet
- Q: How long until caravan is lined up before 2nd toll booth?
- Time to "push" entire caravan through toll booth = 12*10 = 120 sec = 2 mns
- Time for last car to propagate from 1st to 2nd toll both: =100km/(100km/hr)= 1 hr
- A: 1 hr 2 minutes

Caravan analogy (more)



- Cars now "propagate" at 1000 km/hr
- Toll booth now takes 1 min to service a car
- Q: Will cars arrive to 2nd booth before all cars serviced at 1st booth?
- Yes! After 7 min, 1st car at 2nd booth and 8th car still at 1st booth.
- 1st bit of packet can arrive at 2nd router before packet is fully transmitted at 1st router!

Exercise 1

Packet length = L bits

trans. rate R = 1 Mbps

Host A → Host B

distance = 1 km, speed = 2×10^8 m/s

Question:

- Which bit is being transmitted at the time the first bit arrives at Host B for

Answer:

First bit arrives after

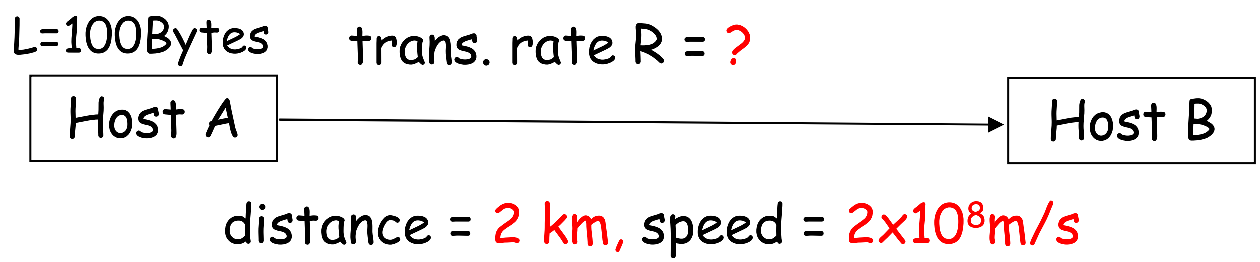
$$1/R + d/s = 1/10^6 + 10^3/(2\times10^8) = 10^{-6} + 5\times10^{-6} = 6\times10^{-6} = 6\ \mu\text{sec}$$

After 6 μsec

6 bits are already transmitted; so 7th bit is being transmitted

Exercise 2

Transmission vs. propagation



Question:

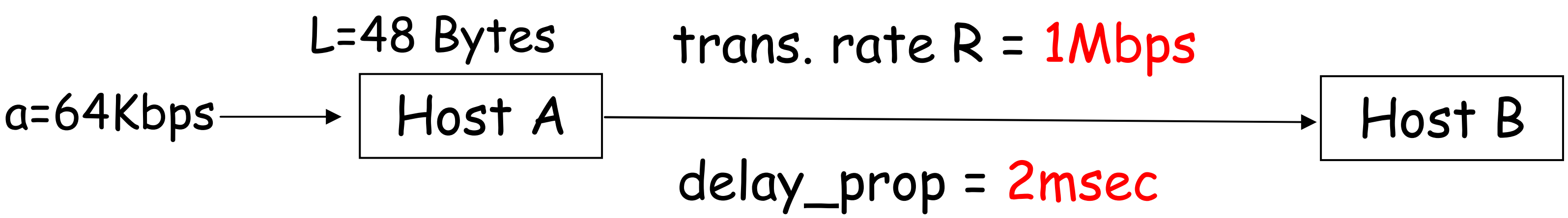
- At what rate (bandwidth) R would the propagation delay equal the transmission delay?

Answer:

- Propagation delay = 2×10^3 (m)/2×10^8 (m/s) = 10^{-5} sec
- Transmission delay = 100x8 (bits)/R
- Prop. delay = trans. delay => R=10^5x100x8 = 80 Mbps

Exercise 3

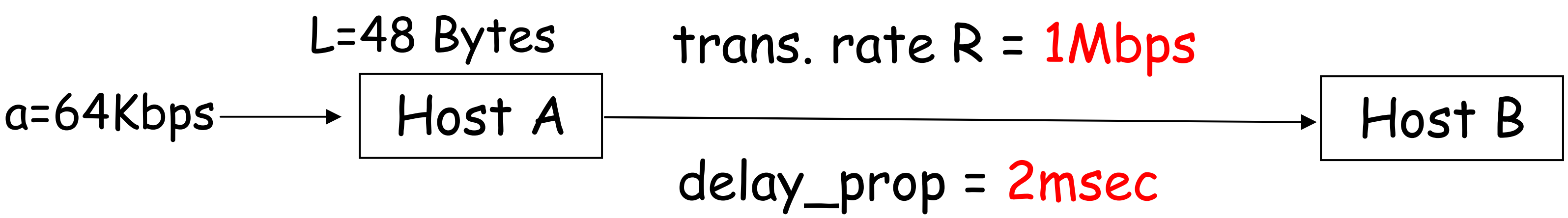
Voice over IP



- Host A
  - converts analog to digital at a=64Kbps
  - groups bits into L=48Byte packets
  - sends packet to Host B as soon it gathers a packet
- Host B
  - As soon as it receives the whole pckt, it converts it to analog
- Question:
  - How much time elapses from the 1st bit of 1st packet is created until the last bit of the 1st packet arrives at Host B?

Exercise 3

Voice over IP



Answer:

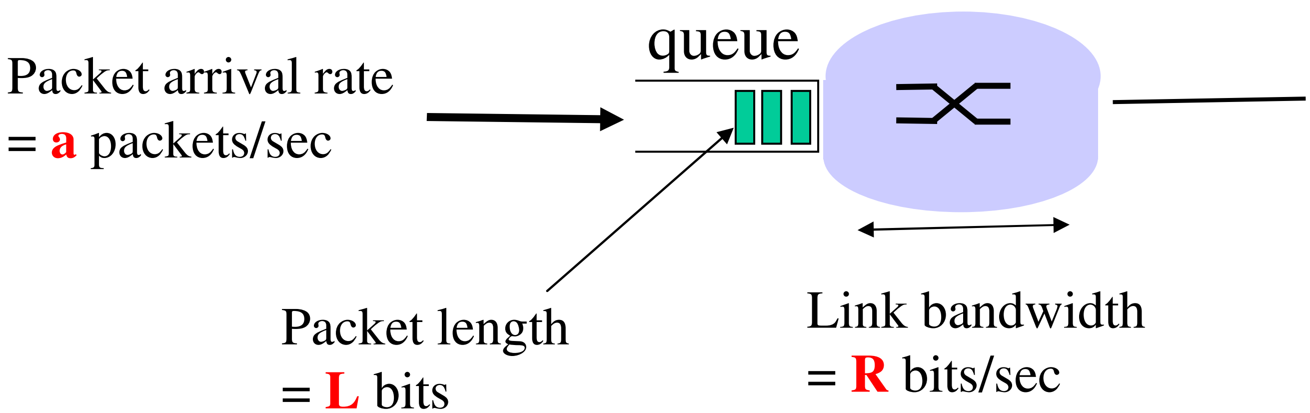
- Time to gather 1st pkt: 48x8 (bits)/64x1000 (b/s) = 6 msec
- Time to push 1st pkt to link: 48x8 (bits)/1x10^6 (b/s) = 0.384 msec
- Time to propagate: 2 msec
- Total delay = 6 + 0.384 + 2 = 8.384 msec

Nodal delay

$$\boxed{d_{\text{nodal}} = d_{\text{proc}} + d_{\text{queue}} + d_{\text{trans}} + d_{\text{prop}}}$$

- d_{proc} = processing delay
  - typically a few microsecs or less
- d_{queue} = queuing delay
  - depends on congestion
- d_{trans} = transmission delay
  - = L/R, significant for low-speed links
- d_{prop} = propagation delay
  - a few microsecs to hundreds of msecs

Queueing delay (more insight)

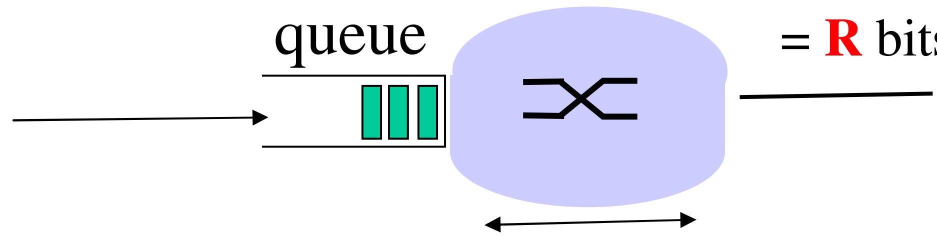


- Every second: aL bits arrive to queue
- Every second: R bits leave the router
- Question: what happens if aL > R ?
- Answer: queue will fill up, and packets will get dropped!!

aL/R is called traffic intensity

Queueing delay: illustration

1 packet arrives every L/R seconds



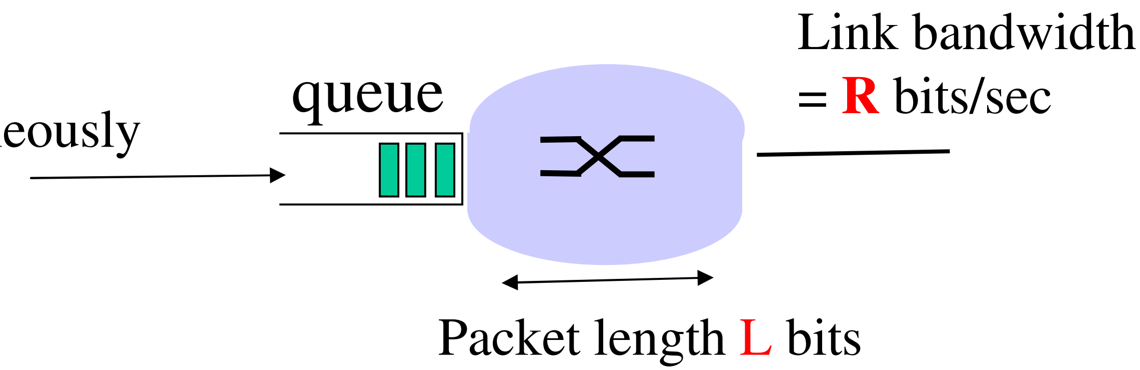
Arrival rate: a = 1/(L/R) = R/L (packet/second)

Traffic intensity = aL/R = (R/L) (L/R) = 1

Average queueing delay = 0
(queue is initially empty)

Queueing delay: illustration

N packet arrive simultaneously
every LN/R seconds



Arrival rate: a = N/(LN/R) = R/L packet/second

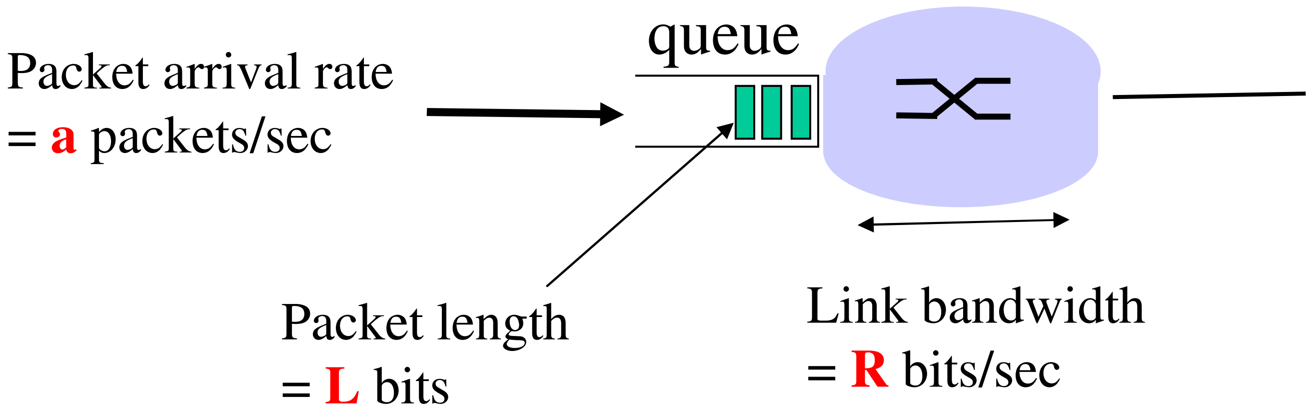
Traffic intensity = aL/R = (R/L) (L/R) = 1

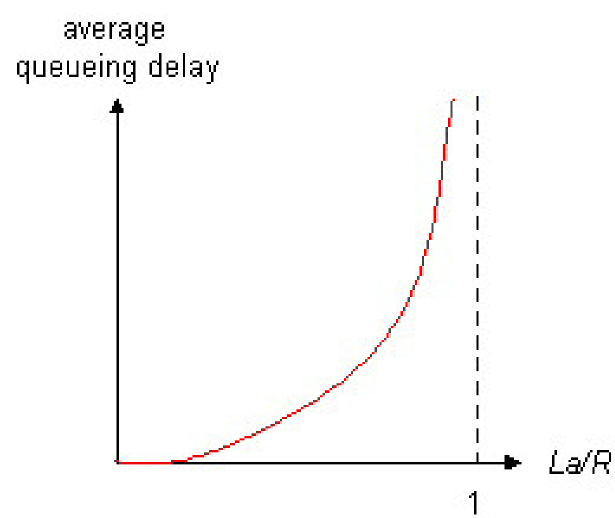
Average queueing delay (queue is empty @ time 0) ?

$\{0 + L/R + 2L/R + \ldots + (N-1)L/R\}/N = L/(RN)\{1+2+\ldots+(N-1)\} = L(N-1)/(2R)$

Note: traffic intensity is same as previous scenario, but queueing delay is different

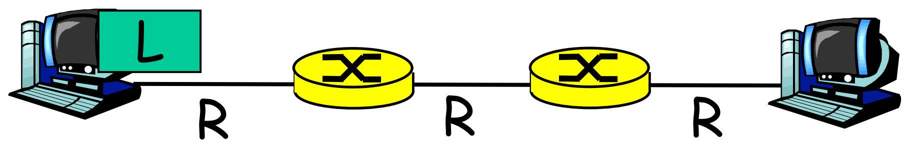
Queueing delay: behavior





- La/R ~ 0: avg. queuing delay small
- La/R -> 1: delays become large
- La/R > 1: more "work" than can be serviced, average delay infinite! (this is when a is random!)

Packet-switching: store-and-forward



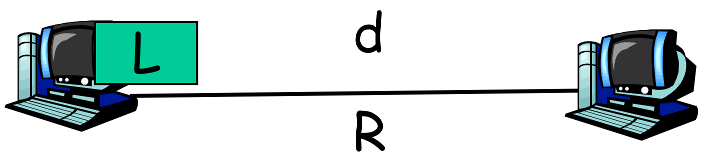
Entire packet must arrive at router before it can be transmitted on next link: *store and forward*

- Takes L/R seconds to transmit (push out) packet of L bits on to link of R bps
- delay = 3L/R (assuming zero propagation delay)

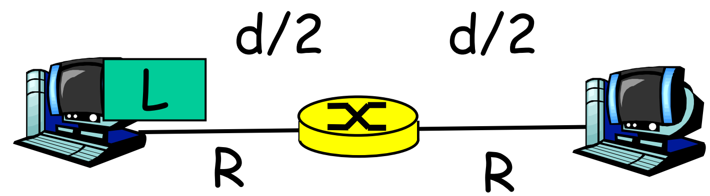
more on this next…

Store-and-forward: illustration

- distance = d meters; speed of propagation = s m/sec
- transmission rate of link = R bits/s



- delay (one packet only)
  = L/R + d/s



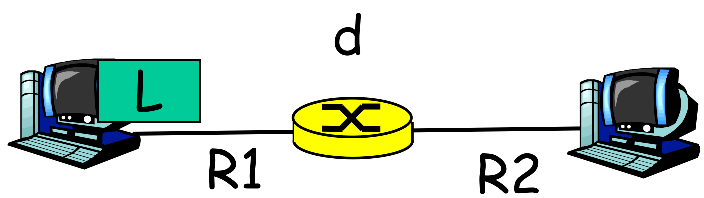
- delay (one packet only)
  $= L/R + \frac{1}{2}d/s + L/R + \frac{1}{2}d/s$
  $= 2L/R + d/s$

Example:
- d/s = 0.5 sec
- L = 10 Mbits
- R = 1 Mbps
- delay = 10.5 sec

Example:
- d/s = 0.5 sec
- L = 10 Mbits
- R = 1 Mbps
- delay = 20.5 sec

Store-and-forward & queuing delay

- distance = d meters; speed of propagation = s m/sec
- transmission rate of link = R1 and R2 bits/s
- Consider sending two packets A and B back to back

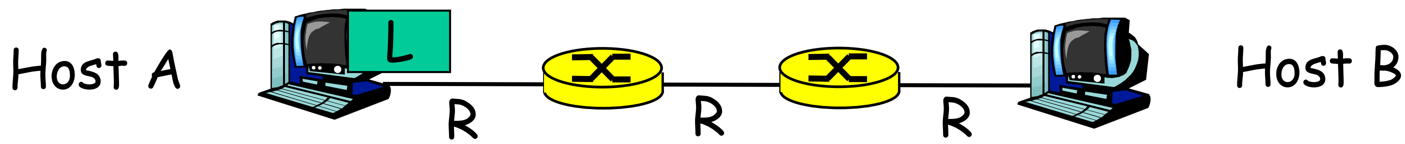


- Case 1: Assume R1 < R2
- Case 2: Assume R1 > R2

Q: is there a queuing delay? how much is this delay?

Answer (queue is empty initially):
Time for last bit of 2nd pkt to arrive at router: $d1 = L/R1 + L/R1 + d/(2s)$
Time for last bit of 1st pkt to leave router: $d2 = L/R1 + d/(2s) + L/R2$
Queueing delay = $d2 - d1 = L/R2 - L/R1$ if positive, otherwise 0. Hence:
when $R1 < R2$, queueing delay = $d2 - d1 = 0$
when $R1 > R2$, queueing delay = $d2 - d1 = L/R2 - L/R1$

Throughput analysis



- Suppose: Host A has huge file of size F bits to send to Host B
- File is split into N packets, each of length L bits (i.e., N=F/L)
- Ignore propagation delay for now
  - Question 1: how long it takes to send the file?

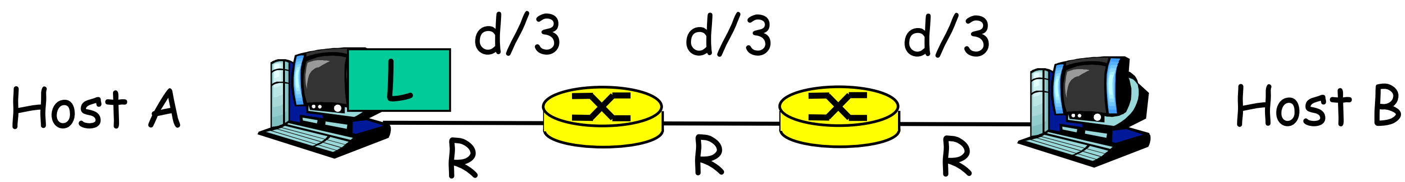
    A: (N+2)L/R = (F+2L)/R

  - Question 2: what is the average throughput achieved when sending the file?

    A: NL/[(N+2)L/R]=NR/(N+2) = FR/(F+2L) = R/(1+2L/F)

Note: throughput = number of total bits sent / total time taken

Throughput analysis



- Suppose: Host A has huge file of size F bits to send to Host B
- File is split into N packets, each of length L bits (i.e., N=F/L)
- Do NOT ignore propagation delay (assume prop. speed = s m/s)

- Question 1: how long it takes to send the file?

  A: $(N+2)L/R + d/s = (F+2L)/R + d/s$

- Question 2: what is the average throughput achieved when sending the file?

  A: $NL/[(N+2)L/R + d/s] = FR/[(N+2)L + dR/s] = FR/[F+2L+dR/s]$

Introduction: Summary

Covered a "ton" of material!

- Internet overview
- Network protocol
- Network edge, core, access network
- Packet-switching versus circuit-switching
- Internet/ISP structure
- layering and service models
- performance: delay and throughput analysis