

---

# DTS205TC High Performance Computing

## Lecture 3 Network 1

Di Zhang, Spring 2024

# Chapter 1: introduction

## Chapter goal:

- Get “feel,” “big picture,” introduction to terminology
  - more depth, detail *later* in course



## Overview/roadmap:

- What *is* the Internet? What *is* a protocol?
- **Network edge:** hosts, access network, physical media
- **Network core:** packet/circuit switching
- **Performance:** loss, delay, throughput

# The Internet: a “nuts and bolts” view



Billions of connected computing *devices*:

- *hosts* = end systems
- running *network apps* at Internet's “edge”

*Packet switches*: forward packets (chunks of data)

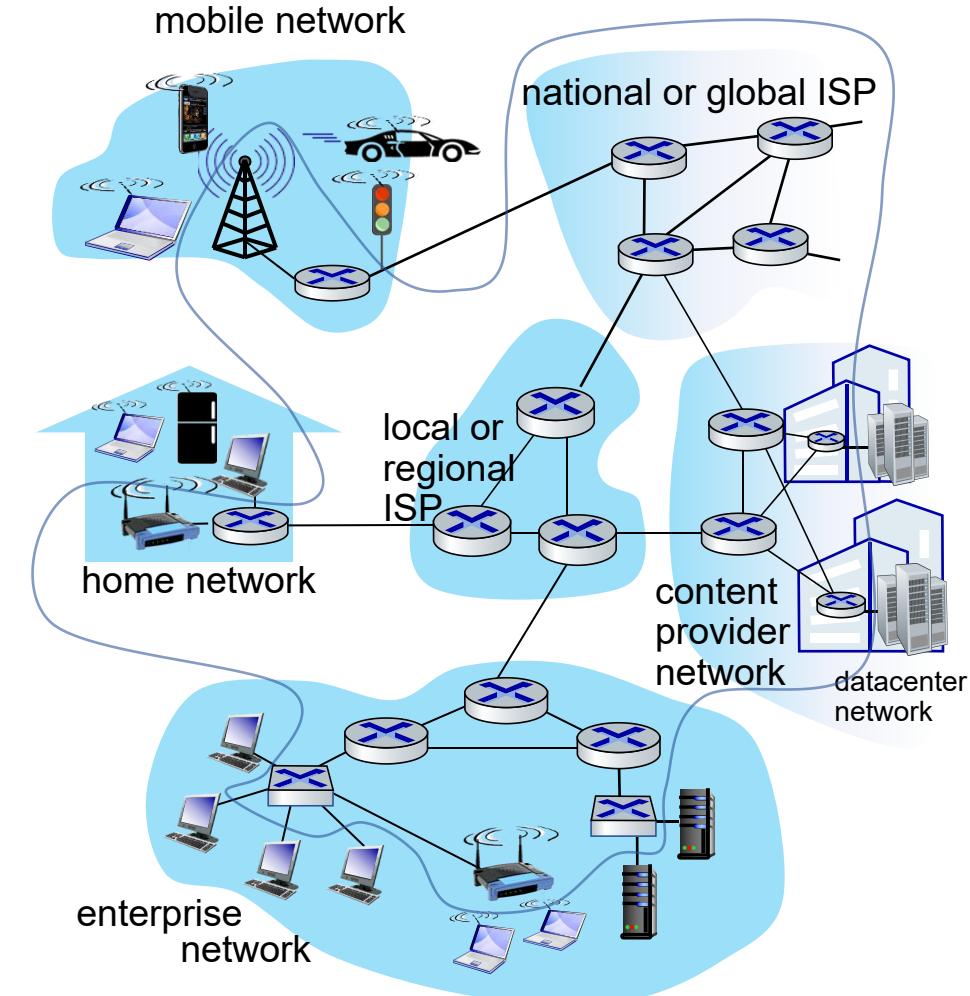
- routers, switches

*Communication links*

- fiber, copper, radio, satellite
- transmission rate: *bandwidth*

*Networks*

- collection of devices, routers, links: managed by an organization



# “Fun” Internet-connected devices



Amazon Echo



Internet refrigerator



Security Camera



Internet phones



IP picture frame



Slingbox: remote control cable TV



Gaming devices



Pacemaker & Monitor



Web-enabled toaster + weather forecaster



sensorized bed mattress



Fitbit



diapers



Tweet-a-watt:  
monitor energy use

bikes



cars

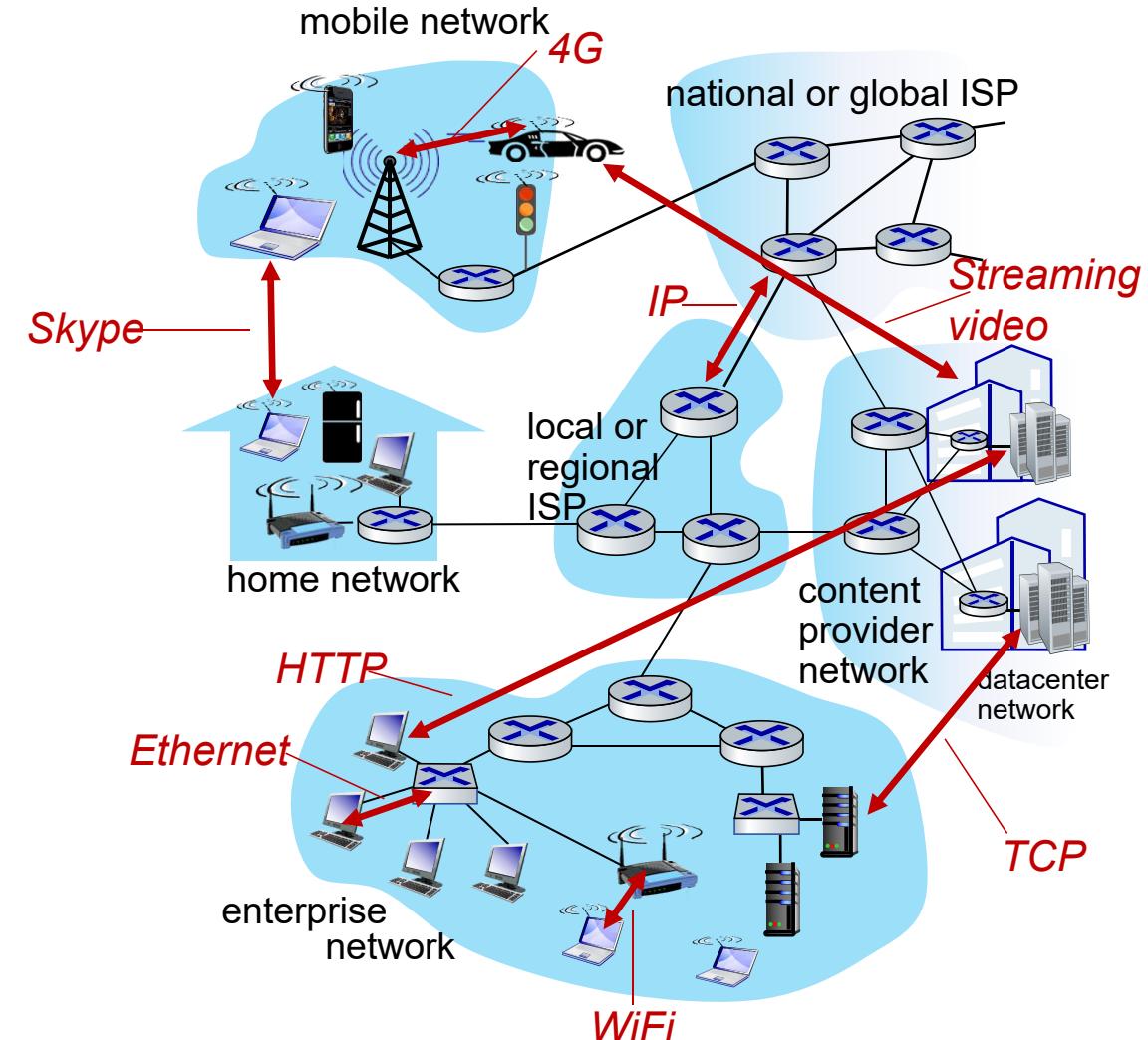


scooters

Others?

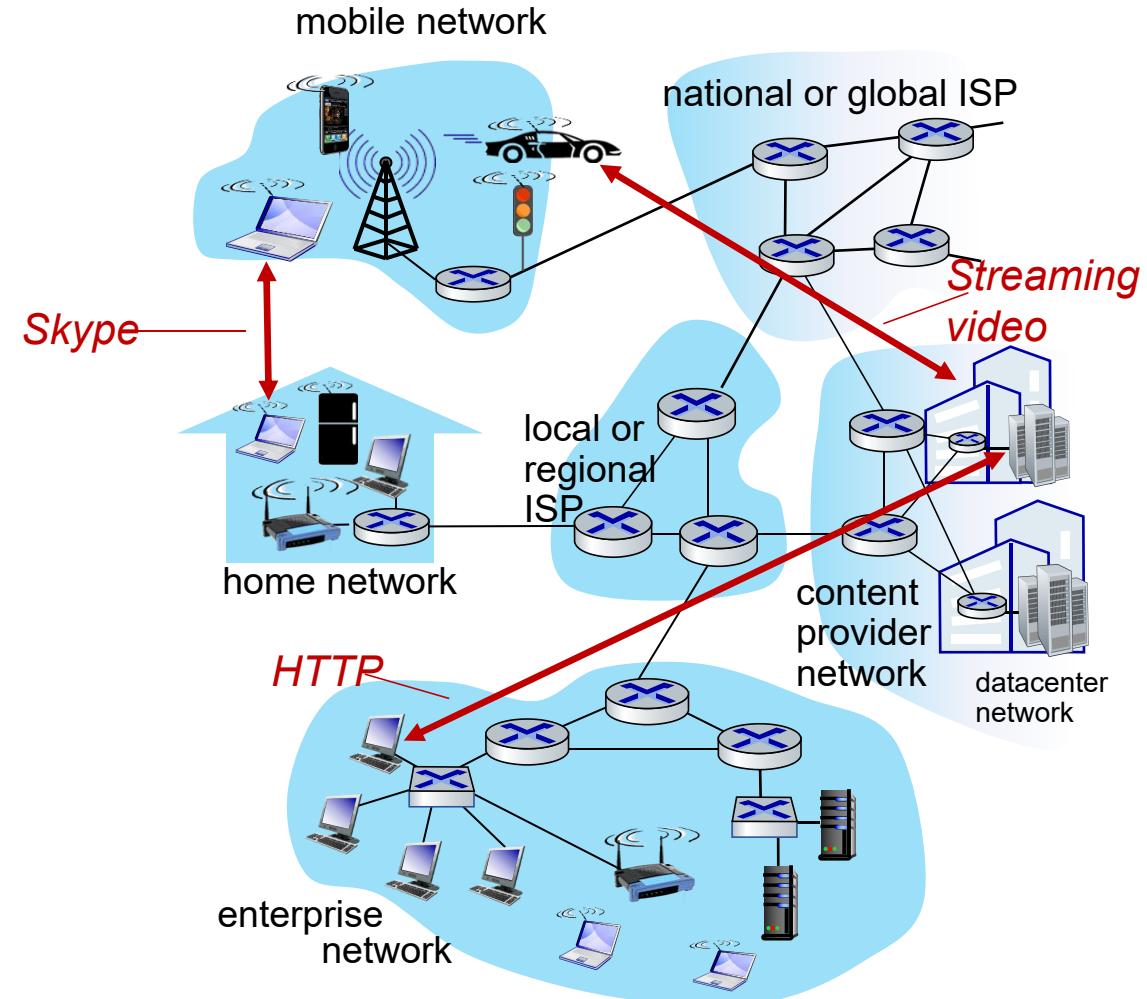
# The Internet: a “nuts and bolts” view

- *Internet: “network of networks”*
  - Interconnected ISPs
- *protocols are everywhere*
  - control sending, receiving of messages
  - e.g., HTTP (Web), streaming video, Skype, TCP, IP, WiFi, 4/5G, Ethernet
- *Internet standards*
  - RFC: Request for Comments
  - IETF: Internet Engineering Task Force



# The Internet: a “services” view

- *Infrastructure* that provides services to applications:
  - Web, streaming video, multimedia teleconferencing, email, games, e-commerce, social media, inter-connected appliances, ...
  - provides *programming interface* to distributed applications:
    - “hooks” allowing sending/receiving apps to “connect” to, use Internet transport service
    - provides service options, analogous to postal service



# What's a protocol?

## *Human protocols:*

- “what’s the time?”
- “I have a question”
- introductions

Rules for:

- ... specific messages sent
- ... specific actions taken when message received, or other events

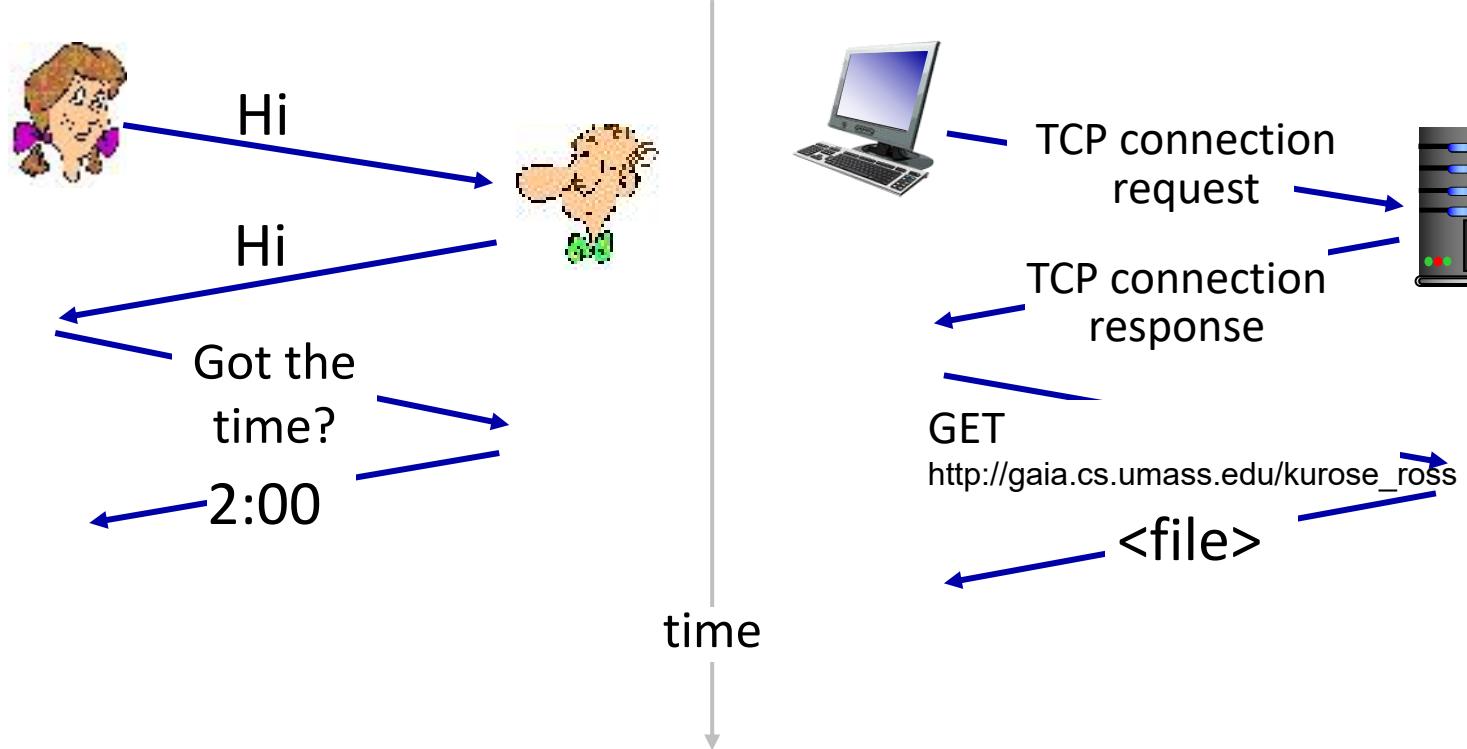
## *Network protocols:*

- computers (devices) rather than humans
- all communication activity in Internet governed by protocols

*Protocols define the format, order of messages sent and received among network entities, and actions taken on message transmission, receipt*

# What's a protocol?

A human protocol and a computer network protocol:



*Q:* other human protocols?

# Chapter 1: roadmap

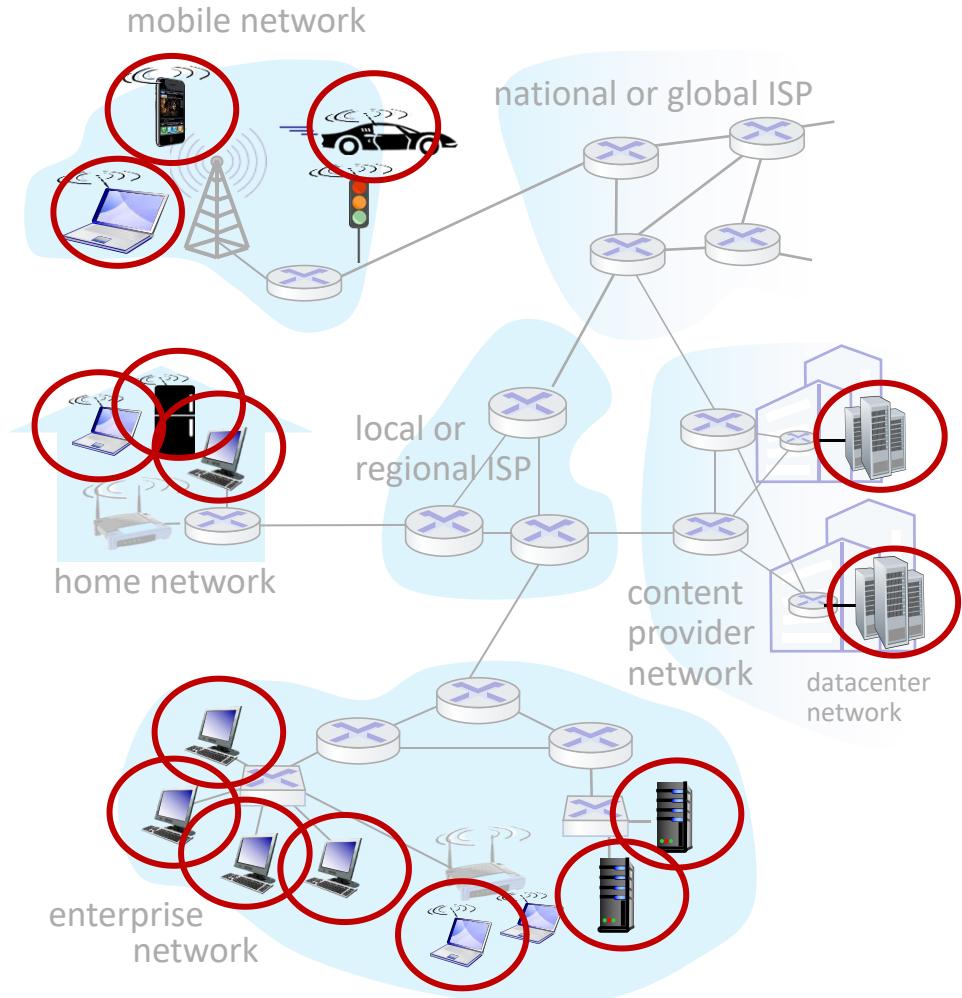
- What *is* the Internet?
- What *is* a protocol?
- **Network edge:** hosts, access network, physical media
- Network core: packet/circuit switching
- Performance: loss, delay, throughput



# A closer look at Internet structure

## Network edge:

- hosts: clients and servers
- servers often in data centers

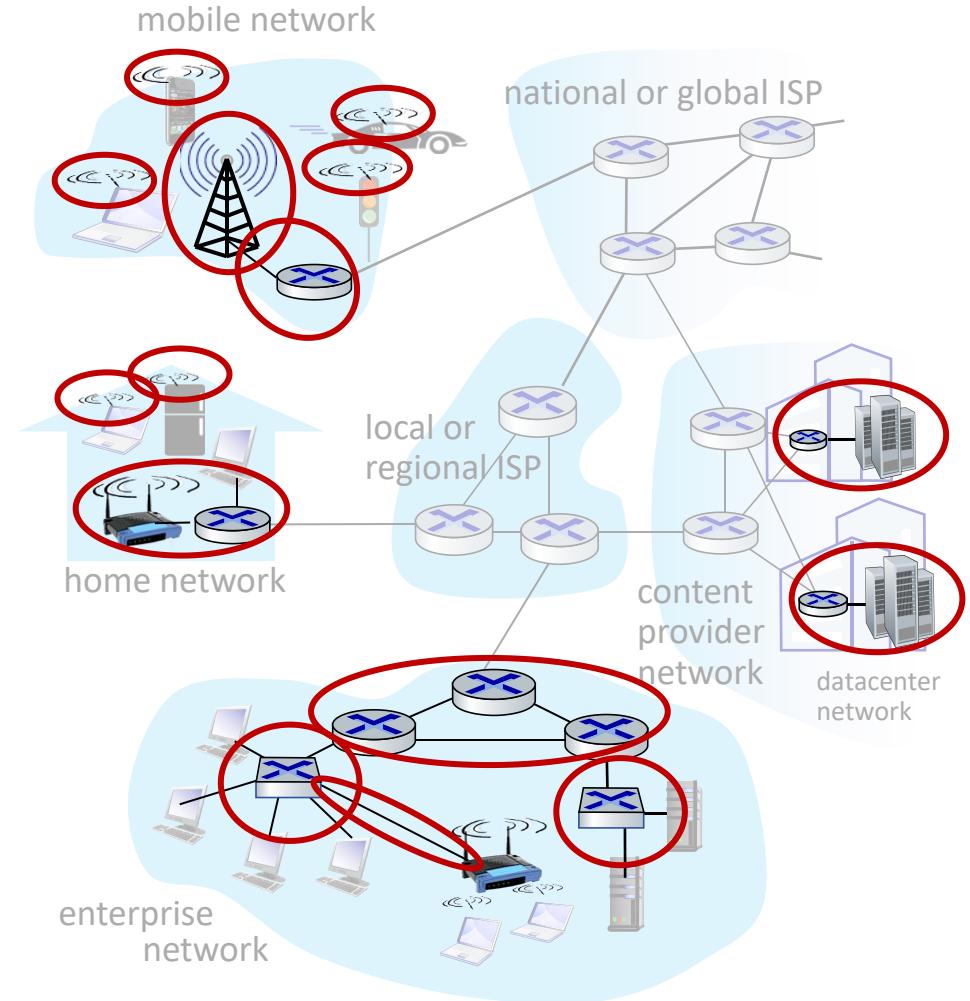


# A closer look at Internet structure

## Network edge:

- hosts: clients and servers
- servers often in data centers

## Access networks, physical media:

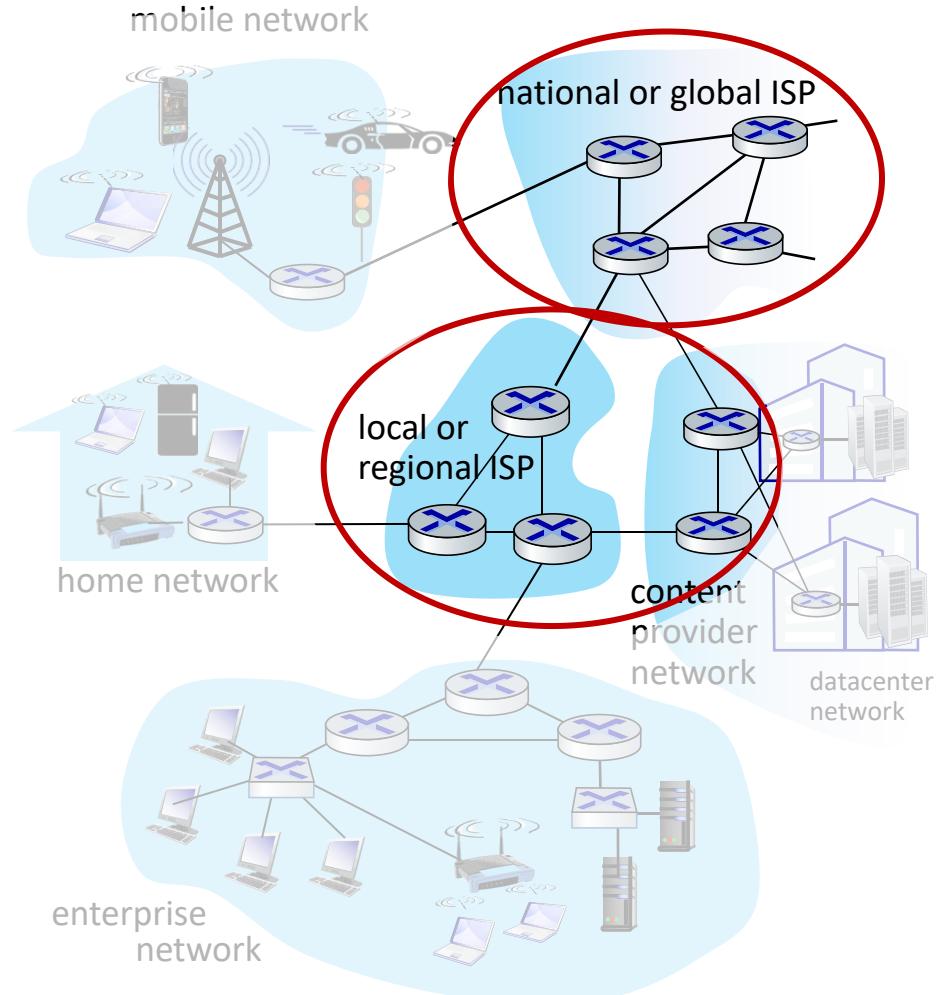


# A closer look at Internet structure

## Network edge:

- hosts: clients and servers
- servers often in data centers

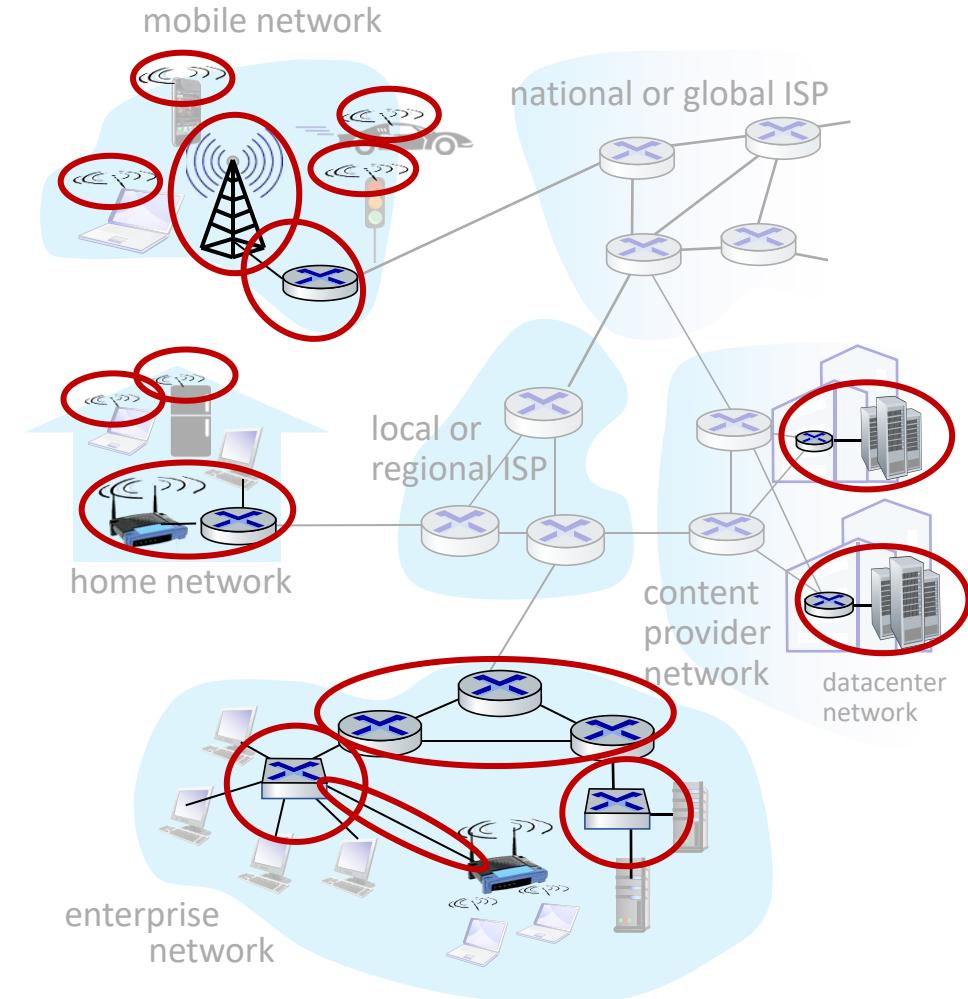
## Access networks, physical media:



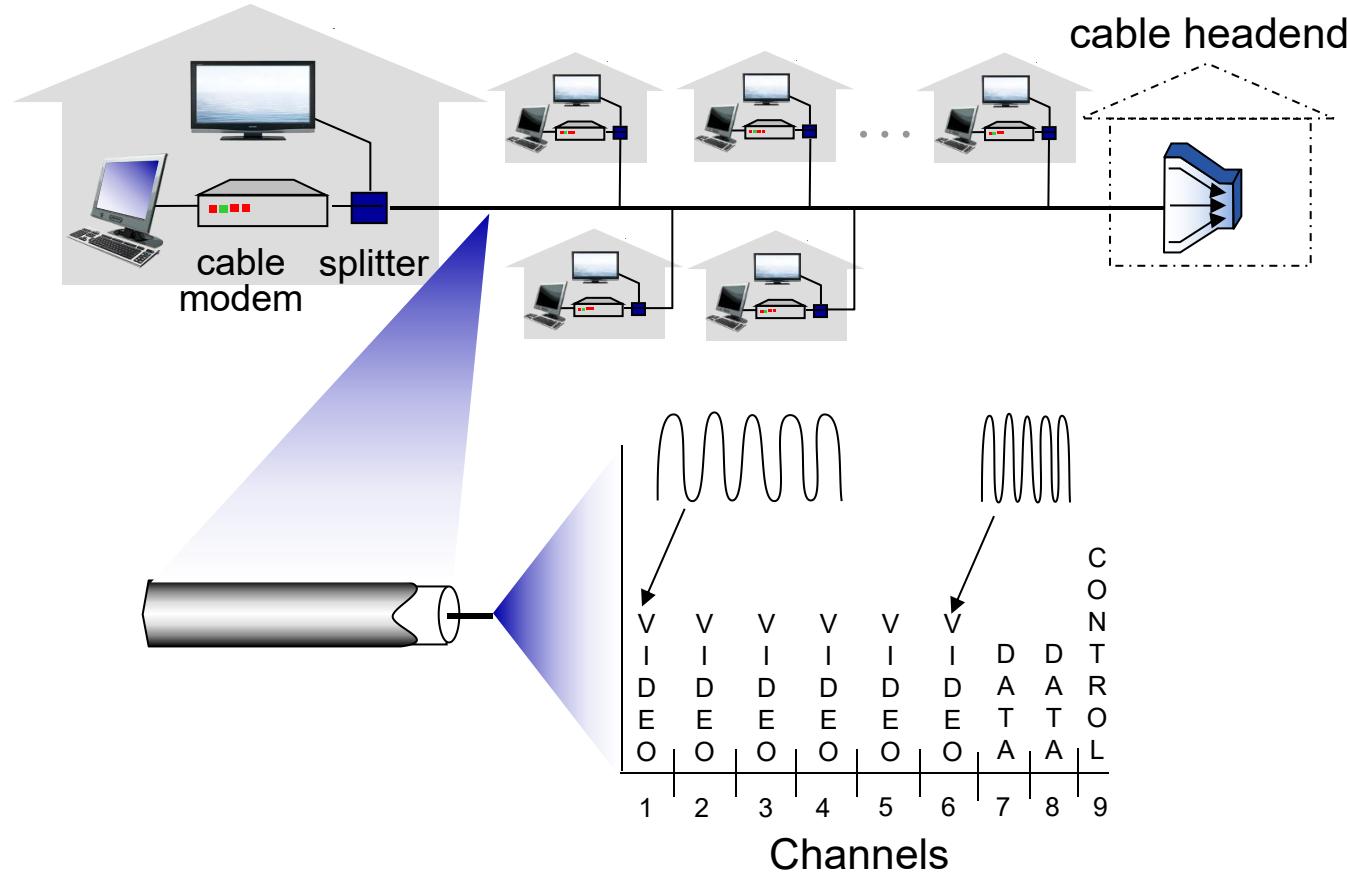
# Access networks and physical media

*Q: How to connect end systems to edge router?*

- residential access nets
- institutional access networks (school, company)
- mobile access networks (WiFi, 4G/5G)

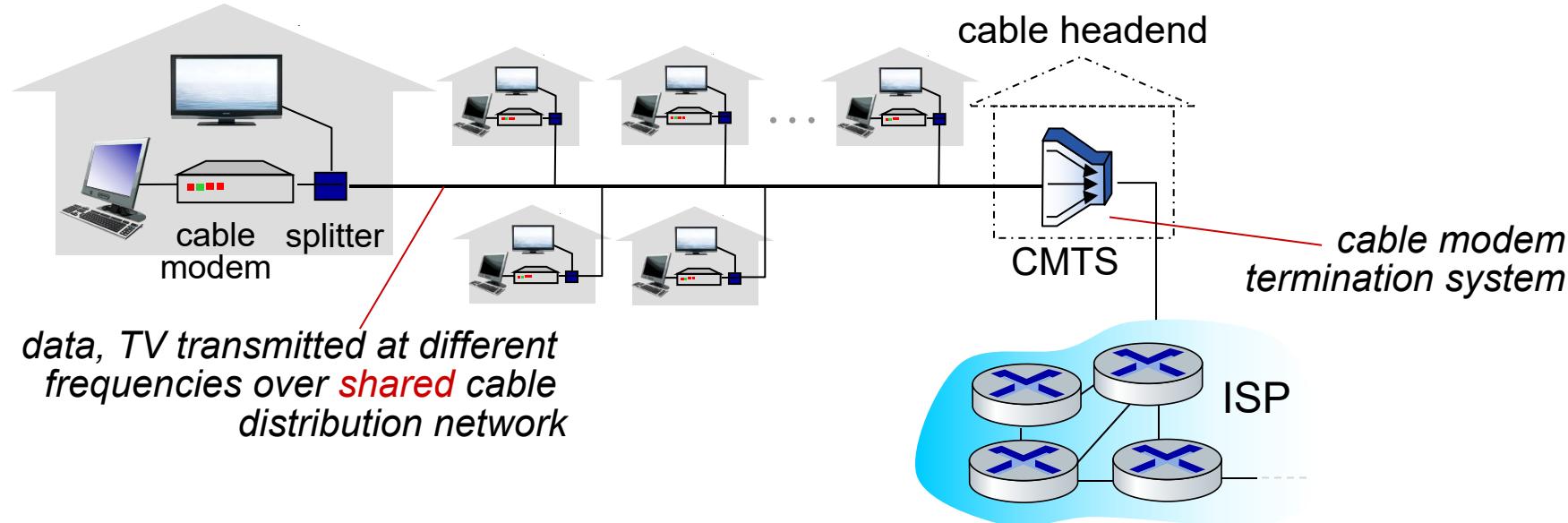


# Access networks: cable-based access



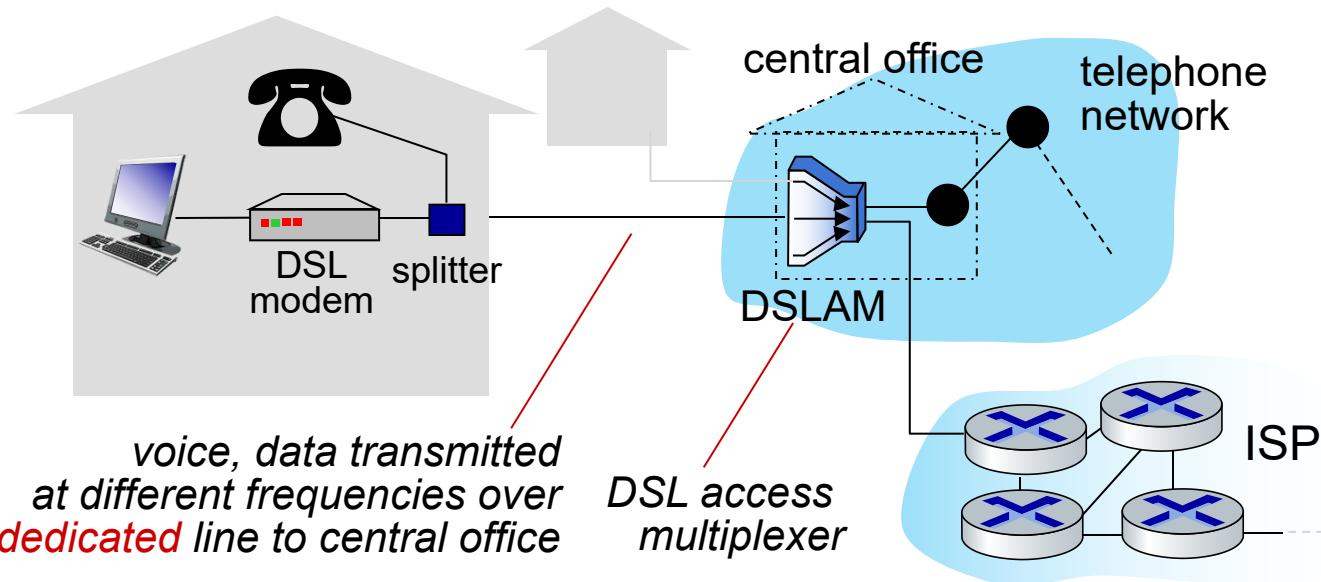
*frequency division multiplexing (FDM)*: different channels transmitted in different frequency bands

# Access networks: cable-based access



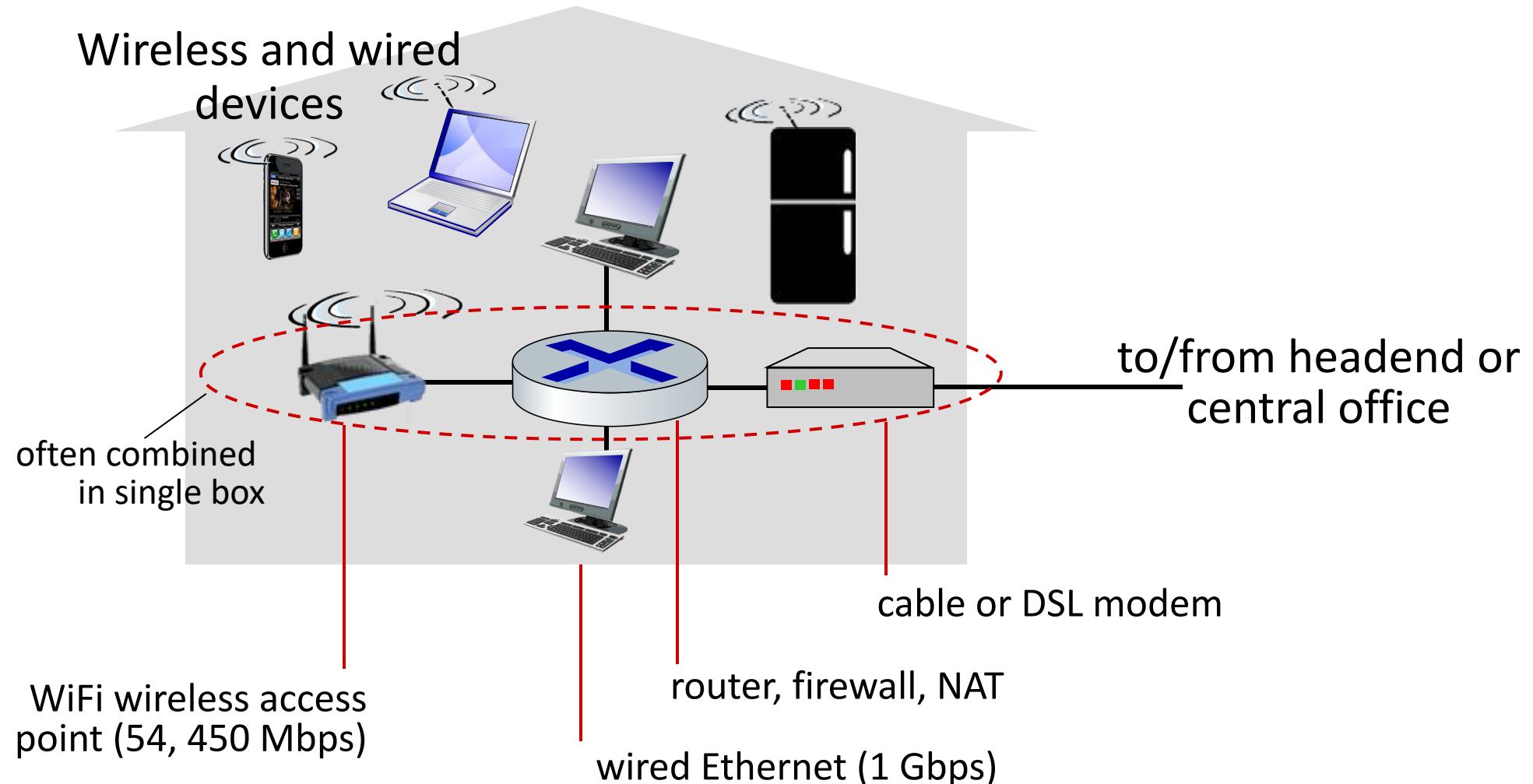
- HFC: hybrid fiber coax
  - asymmetric: up to 40 Mbps – 1.2 Gbps downstream transmission rate, 30-100 Mbps upstream transmission rate
- network of cable, fiber attaches homes to ISP router
  - homes **share access network** to cable headend

# Access networks: digital subscriber line (DSL)



- use *existing* telephone line to central office DSLAM
  - data over DSL phone line goes to Internet
  - voice over DSL phone line goes to telephone net
- 24-52 Mbps dedicated downstream transmission rate
- 3.5-16 Mbps dedicated upstream transmission rate

# Access networks: home networks



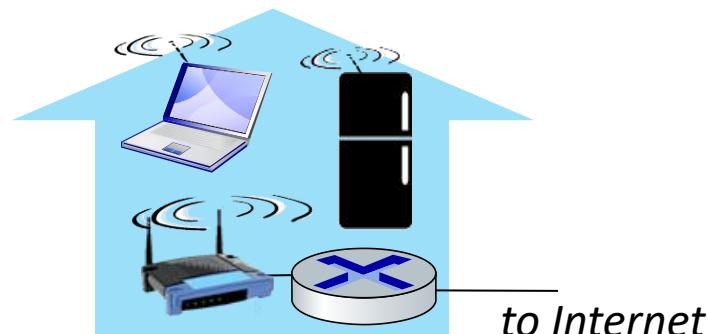
# Wireless access networks

Shared *wireless* access network connects end system to router

- via base station aka “access point”

## Wireless local area networks (WLANs)

- typically within or around building (~100 ft)
- 802.11b/g/n (WiFi): 11, 54, 450 Mbps transmission rate

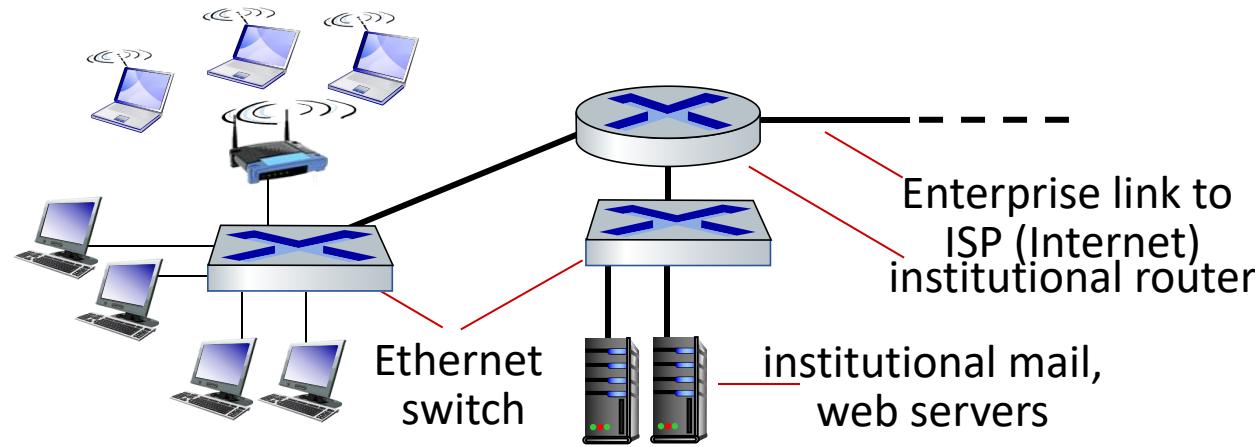


## Wide-area cellular access networks

- provided by mobile, cellular network operator (10's km)
- 10's Mbps
- 4G/5G cellular networks



# Access networks: enterprise networks



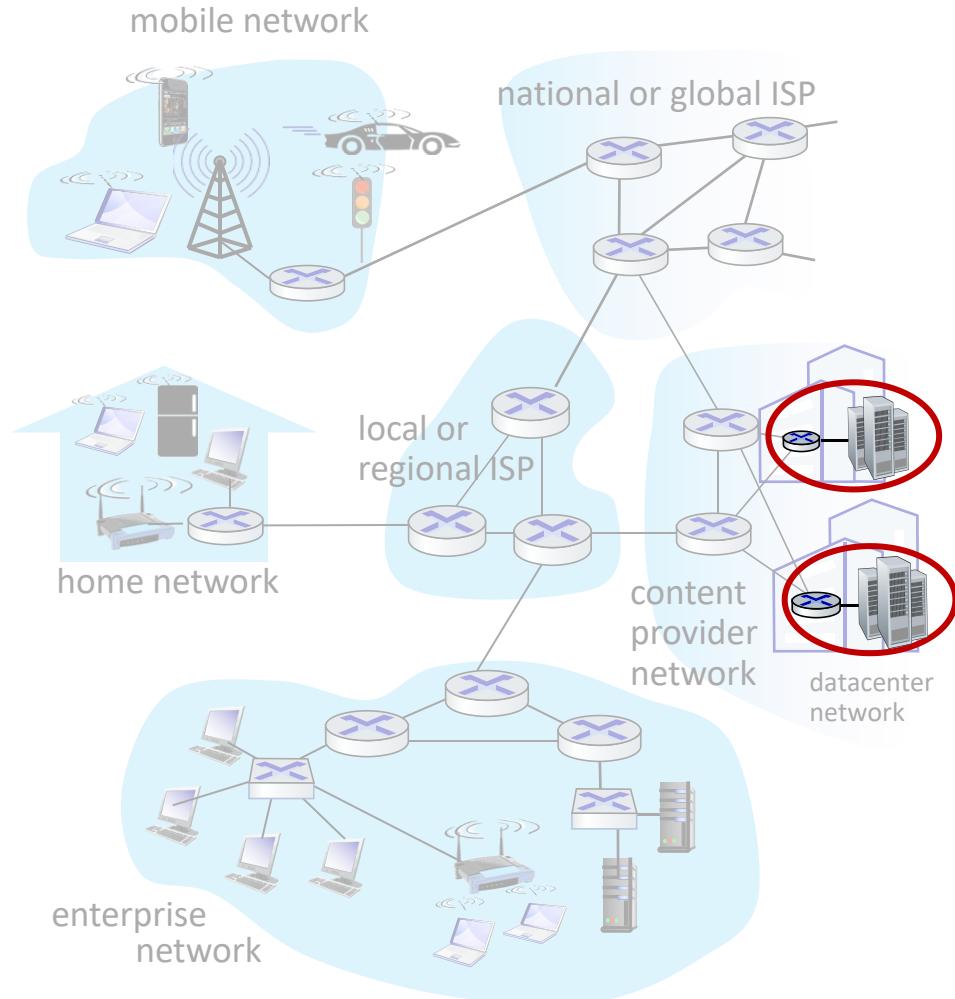
- companies, universities, etc.
- mix of wired, wireless link technologies, connecting a mix of switches and routers (we'll cover differences shortly)
  - Ethernet: wired access at 100Mbps, 1Gbps, 10Gbps
  - WiFi: wireless access points at 11, 54, 450 Mbps

# Access networks: data center networks

- high-bandwidth links (10s to 100s Gbps) connect hundreds to thousands of servers together, and to Internet



Courtesy: Massachusetts Green High Performance Computing Center ([mghpcc.org](http://mghpcc.org))



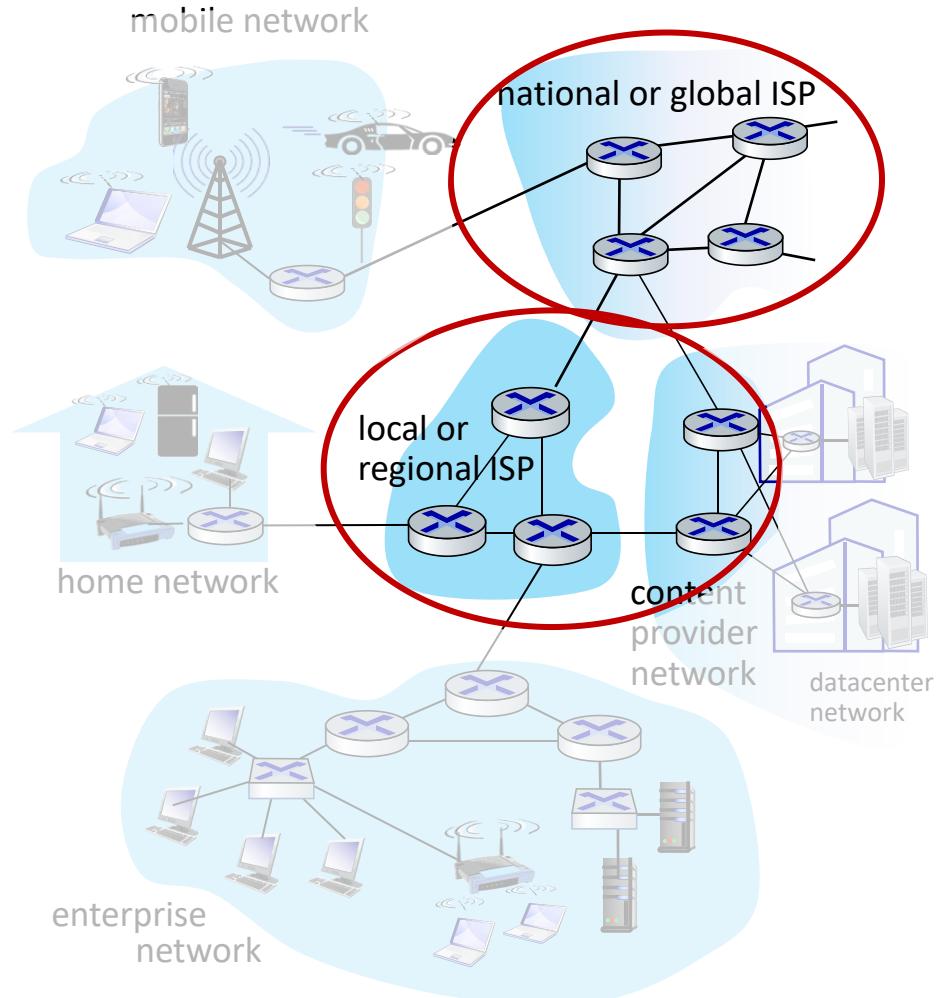
# Chapter 1: roadmap

- What *is* the Internet?
- What *is* a protocol?
- Network edge: hosts, access network, physical media
- **Network core:** packet/circuit switching
- Performance: loss, delay, throughput



# The network core

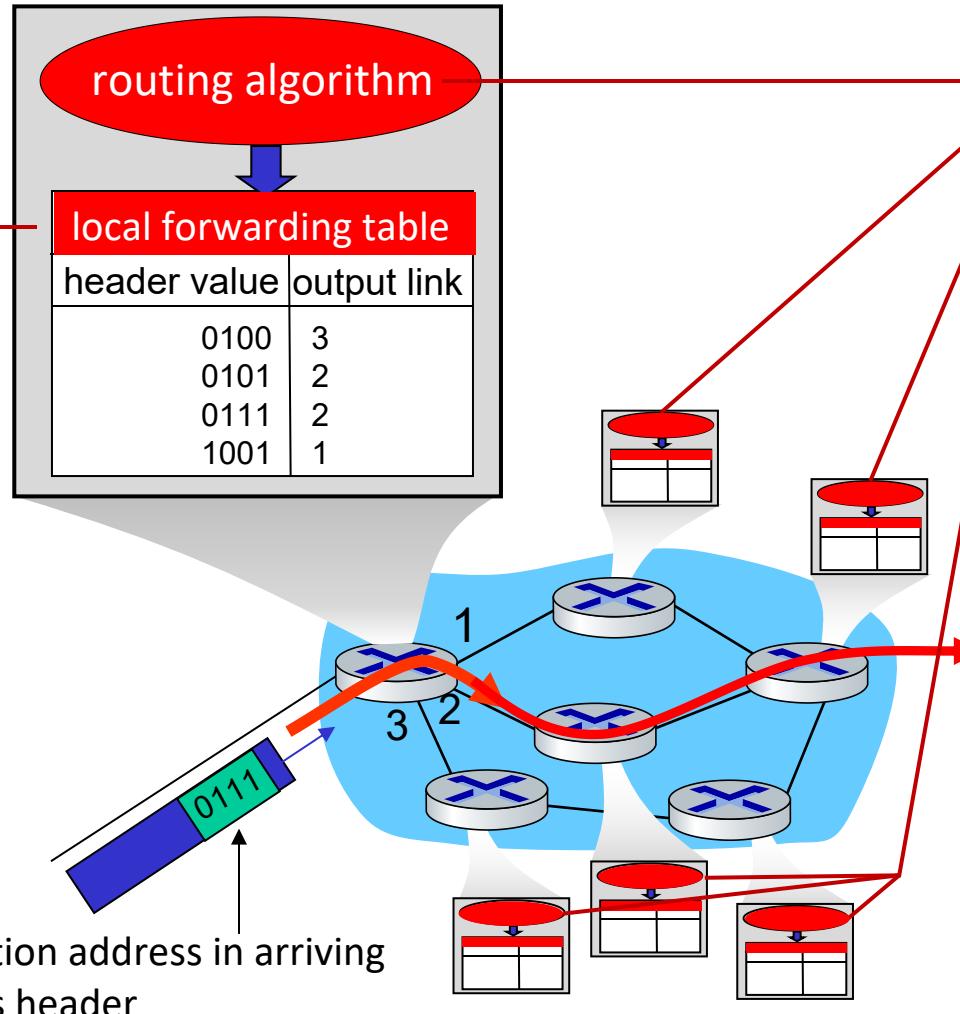
- mesh of interconnected routers
- **packet-switching**: hosts break application-layer messages into *packets*
- **network forwards** packets from one router to the next, across links on path from **source to destination**



# Two key network-core functions

*Forwarding:*

- aka “switching”
- *local* action: move arriving packets from router's input link to appropriate router output link



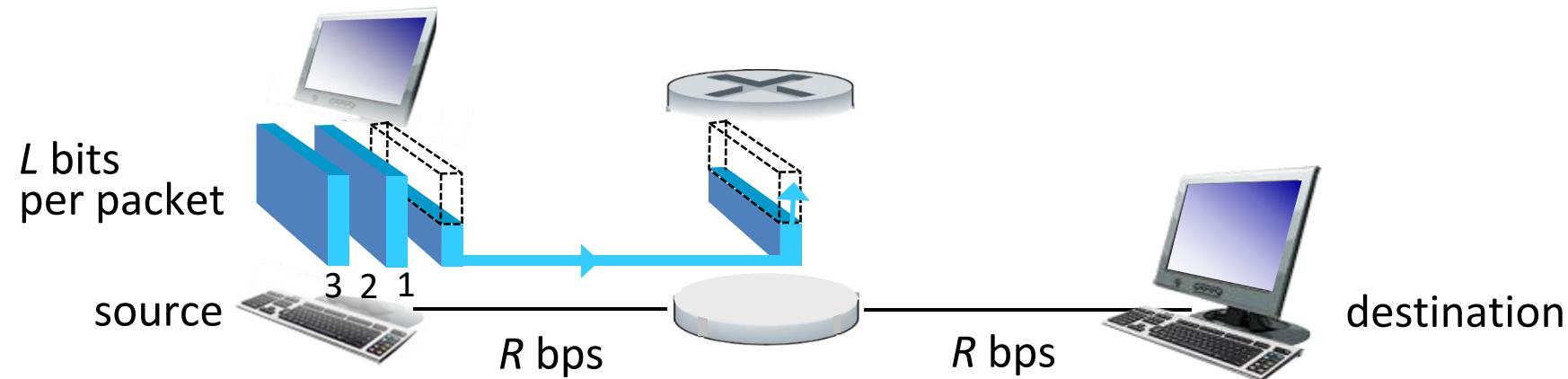
*Routing:*

- *global* action: determine source-destination paths taken by packets
- routing algorithms





# Packet-switching: store-and-forward



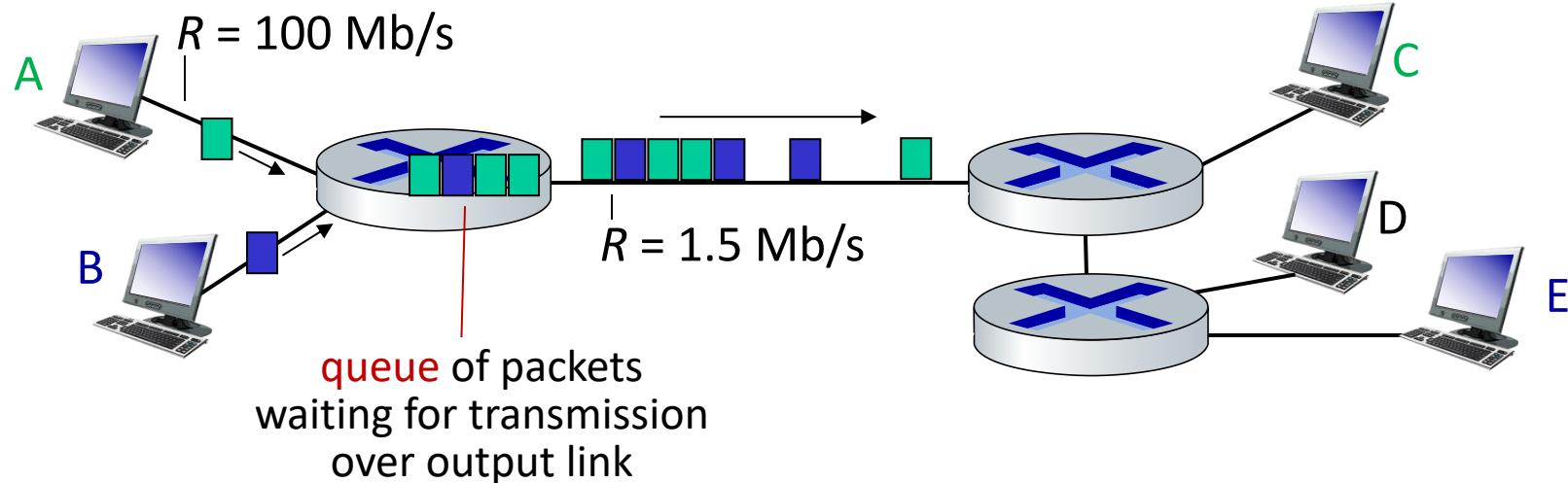
- **packet transmission delay:** takes  $L/R$  seconds to transmit (push out)  $L$ -bit packet into link at  $R$  bps

*One-hop numerical example:*

- $L = 10$  Kbits
- $R = 100$  Mbps
- one-hop transmission delay = 0.1 msec

- **store and forward:** entire packet must arrive

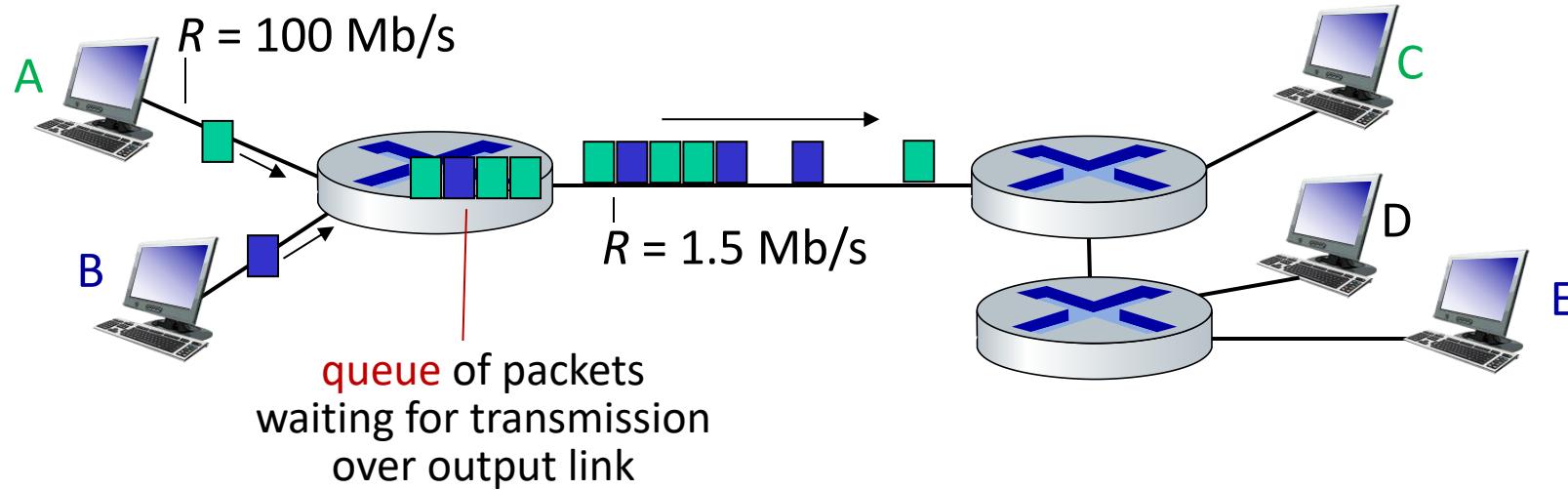
# Packet-switching: queueing



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



# Packet-switching: queueing



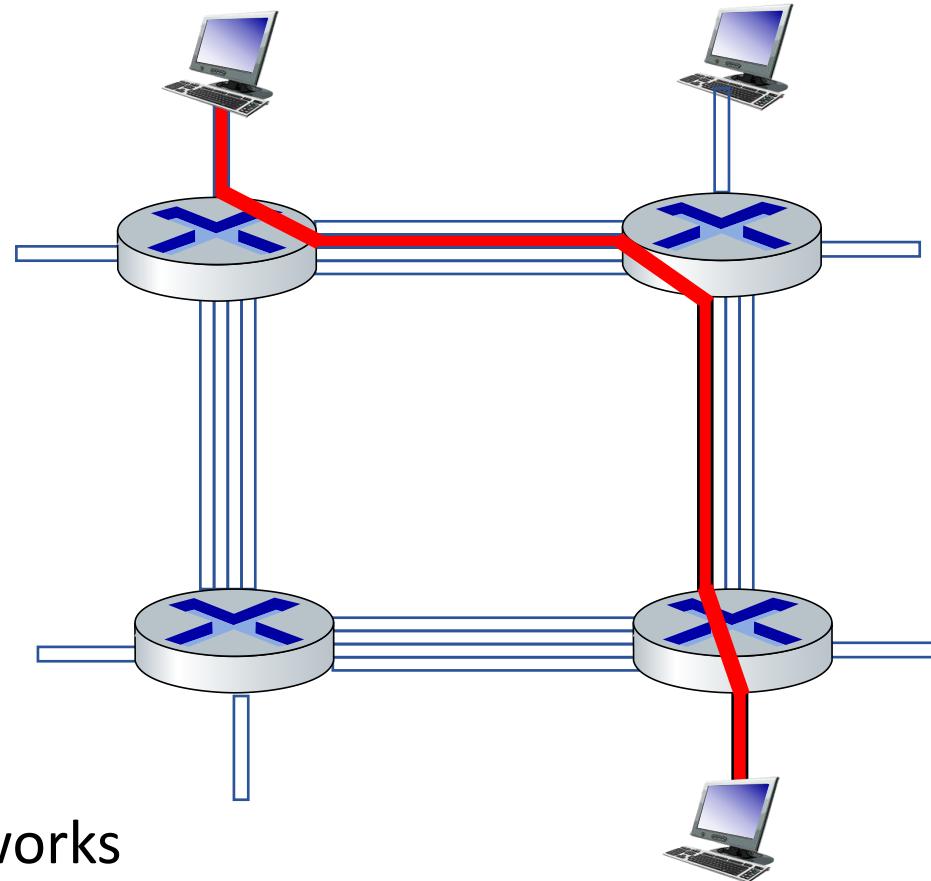
*Packet queuing and loss:* if arrival rate (in bps) to link exceeds transmission rate (bps) of link for some period of time:

- packets will queue, waiting to be transmitted on output link

# Alternative to packet switching: circuit switching

end-end resources allocated to, reserved for “call” between source and destination

- in diagram, each link has four circuits.
  - call gets 2<sup>nd</sup> circuit in top link and 1<sup>st</sup> circuit in right link.
- dedicated resources: no sharing
  - circuit-like (guaranteed) performance
- circuit segment idle if not used by call (no sharing)
  - commonly used in traditional telephone networks



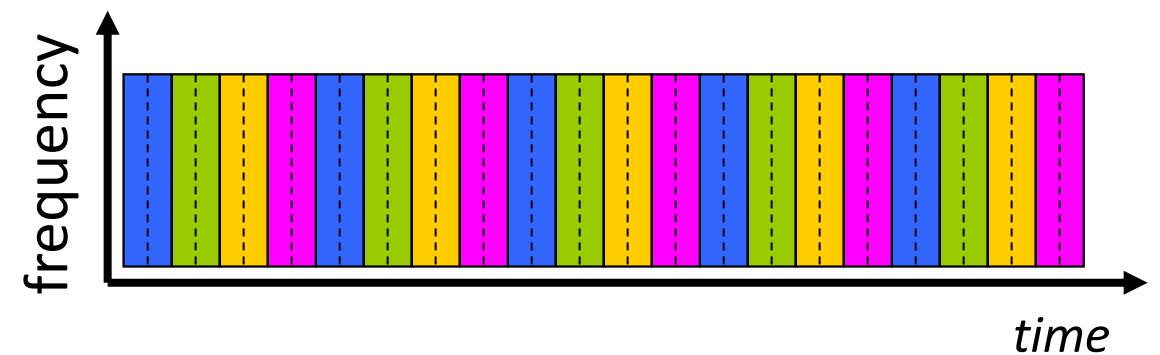
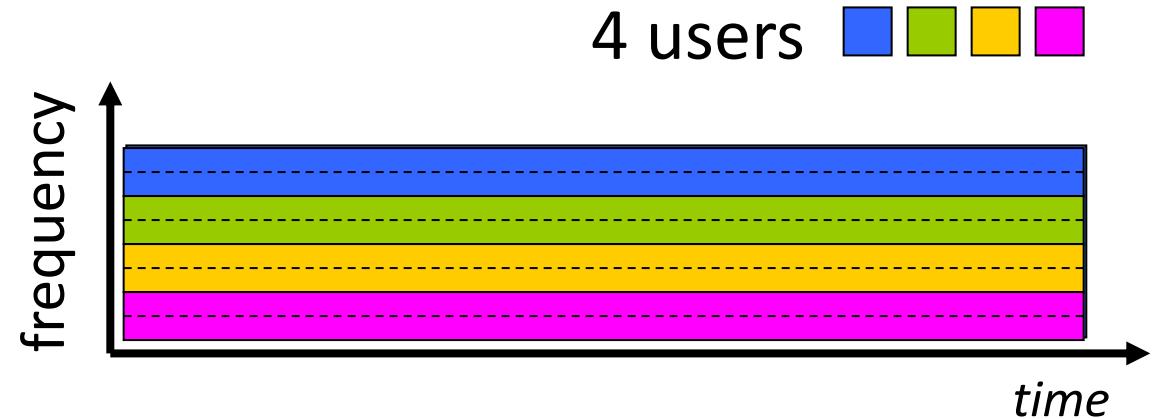
# Circuit switching: FDM and TDM

## Frequency Division Multiplexing (FDM)

- optical, electromagnetic frequencies divided into (narrow) frequency bands
- each call allocated its own band, can transmit at max rate of that narrow band

## Time Division Multiplexing (TDM)

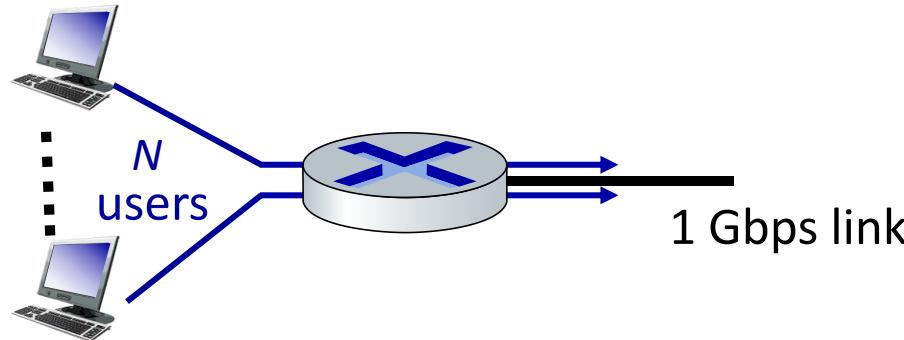
- time divided into slots
- each call allocated periodic slot(s), can transmit at maximum rate of (wider) frequency band (only) during its time slot(s)



# Packet switching versus circuit switching

example:

- 1 Gb/s link
- each user:
  - 100 Mb/s when “active”
  - active 10% of time



*Q:* how many users can use this network under circuit-switching and packet switching?

- *circuit-switching:* 10 users
- *packet switching:* with 35 users,  
probability > 10 active at same time  
is less than .0004 \*

*Q:* how did we get value 0.0004?  
*A:* HW problem (for those with  
course in probability only)

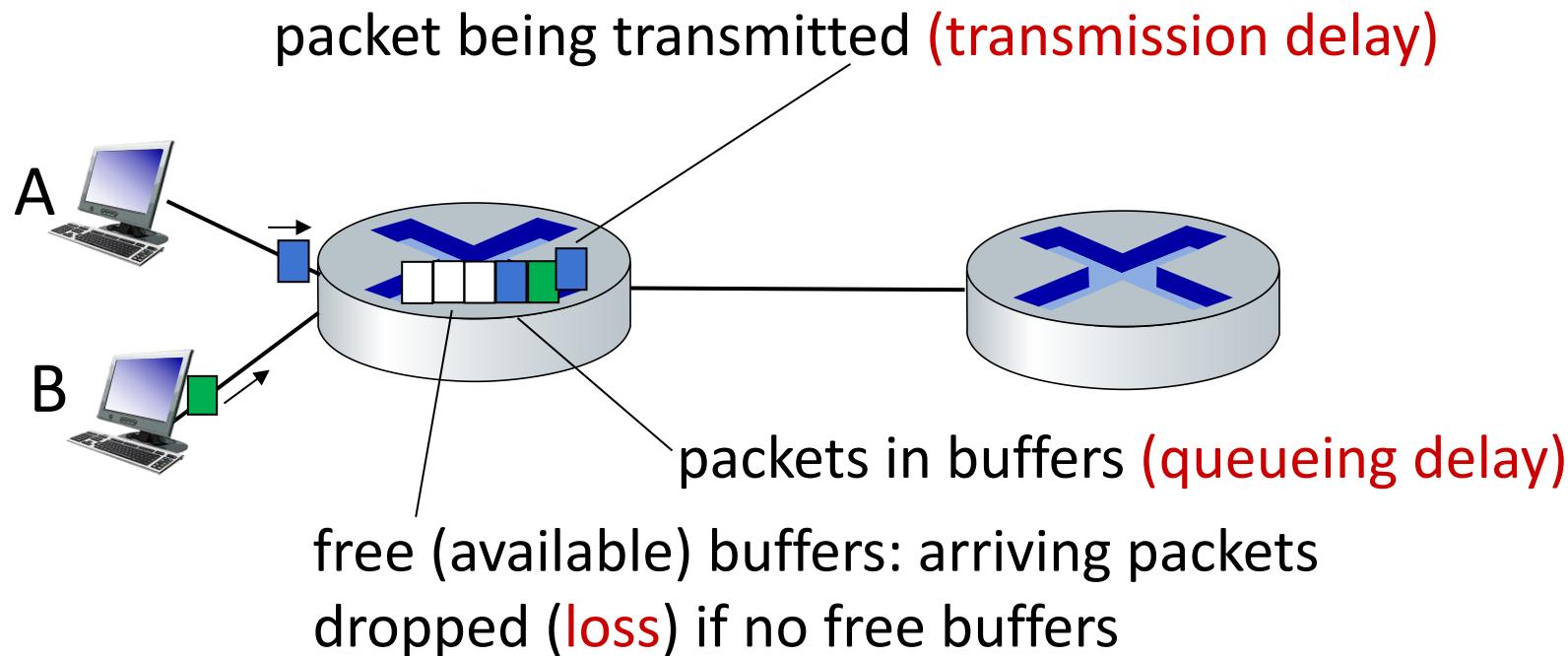
# Chapter 1: roadmap

- What *is* the Internet?
- What *is* a protocol?
- Network edge: hosts, access network, physical media
- Network core: packet/circuit switching
- Performance: loss, delay, throughput

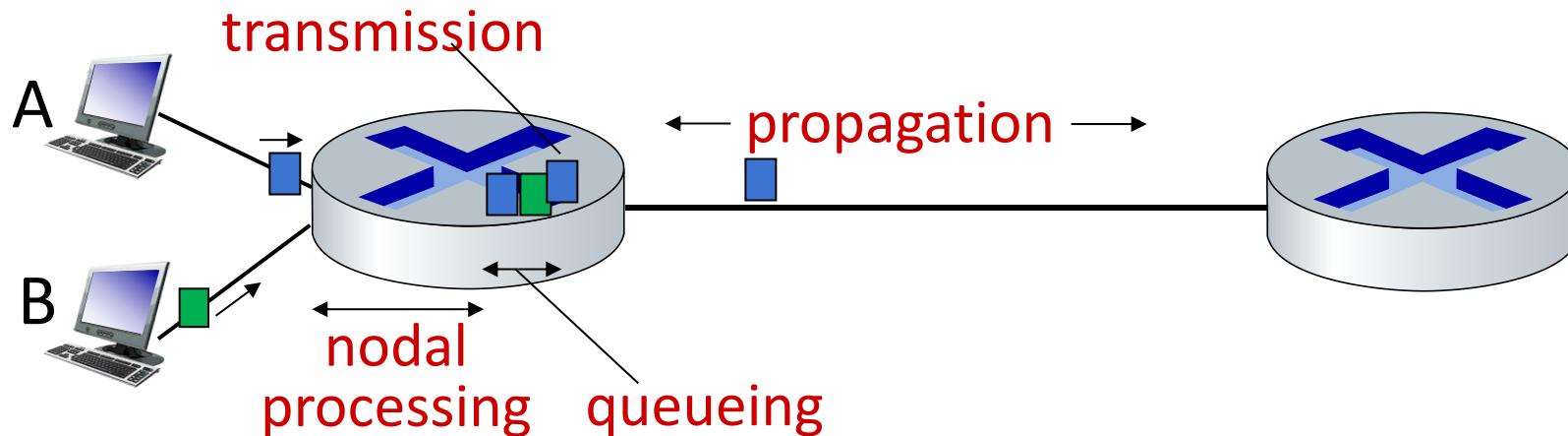


# How do packet delay and loss occur?

- packets **queue** in router buffers, waiting for turn for transmission
  - queue length grows when arrival rate to link (temporarily) exceeds output link capacity
- packet **loss** occurs when memory to hold queued packets fills up



# Packet delay: four sources



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

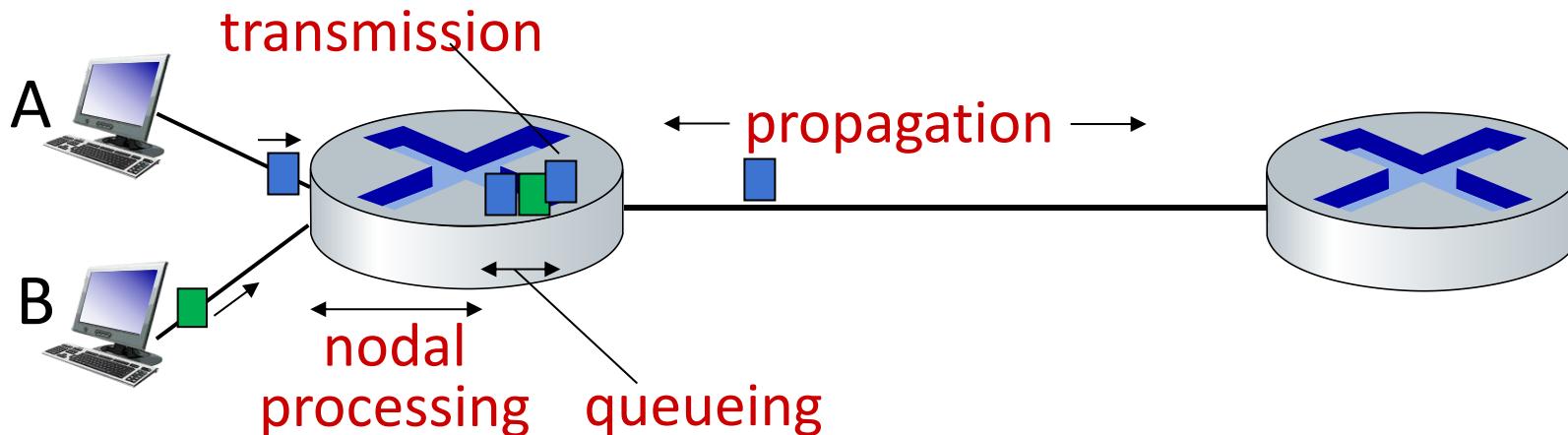
$d_{\text{proc}}$ : nodal processing

- check bit errors
- determine output link
- typically < microsecs

$d_{\text{queue}}$ : queueing delay

- time waiting at output link for transmission
- depends on congestion level of router

# Packet delay: four sources



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

$d_{\text{trans}}$ : transmission delay:

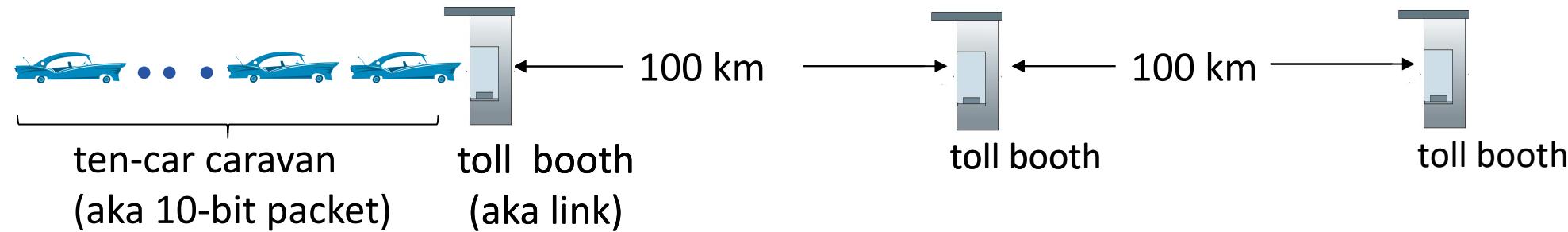
- $L$ : packet length (bits)
- $R$ : link *transmission rate (bps)*
- $d_{\text{trans}} = L/R$

$d_{\text{trans}}$  and  $d_{\text{prop}}$   
very different

$d_{\text{prop}}$ : propagation delay:

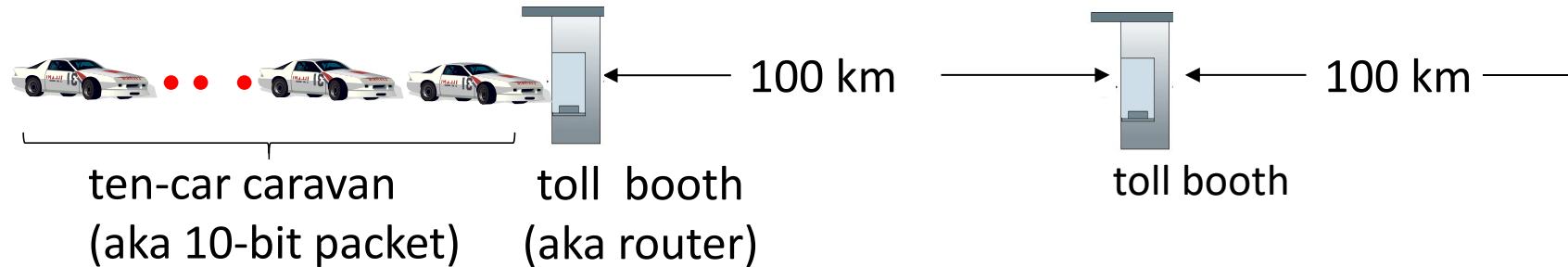
- $d$ : length of physical link
- $s$ : propagation speed ( $\sim 2 \times 10^8$  m/sec)
- $d_{\text{prop}} = d/s$

# Caravan analogy



- car ~ bit; caravan ~ packet; toll service ~ link transmission
- toll booth takes 12 sec to service car (bit transmission time)
- “propagate” at 100 km/hr
- ***Q: How long until caravan is lined up before 2nd toll booth?***
- time to “push” entire caravan through toll booth onto highway =  $12 * 10 = 120$  sec
- time for last car to propagate from 1st to 2nd toll both:  $100\text{km}/(100\text{km/hr}) = 1$  hr
- ***A: 62 minutes***

# Caravan analogy



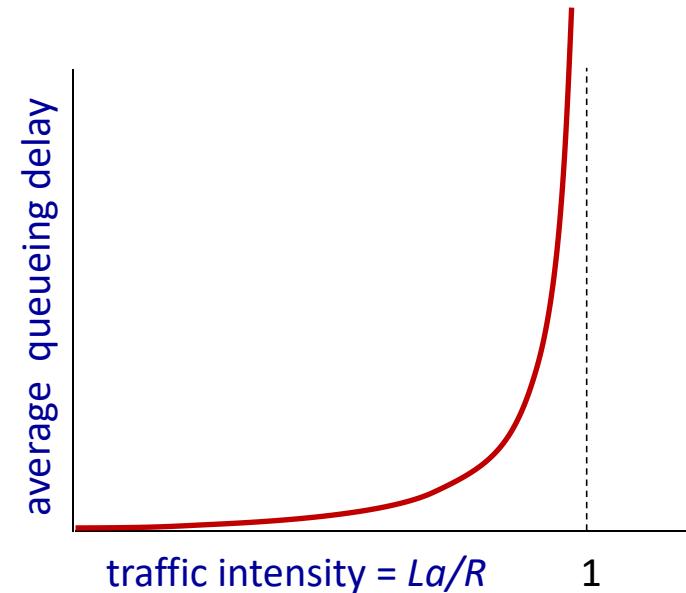
- suppose cars now “propagate” at 1000 km/hr
  - and suppose toll booth now takes one min to service a car
  - ***Q: Will cars arrive to 2nd booth before all cars serviced at first booth?***
- A: Yes!** after 7 min, first car arrives at second booth; three cars still at first booth

# Packet queueing delay (revisited)

- $a$ : average packet arrival rate
- $L$ : packet length (bits)
- $R$ : link bandwidth (bit transmission rate)

$$\frac{L \cdot a}{R} : \frac{\text{arrival rate of bits}}{\text{service rate of bits}}$$

*“traffic intensity”*



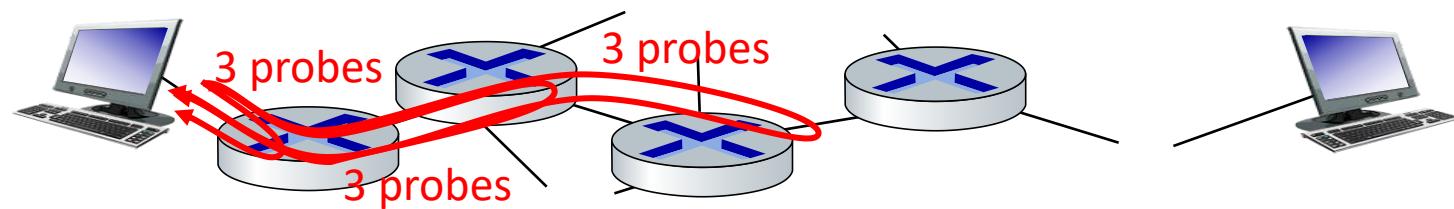
- $La/R \sim 0$ : avg. queueing delay small
- $La/R \rightarrow 1$ : avg. queueing delay large
- $La/R > 1$ : more “work” arriving is more than can be serviced - average delay infinite!



$La/R \sim 0$   
 $La/R > 1$   
 Introduction: 1-38

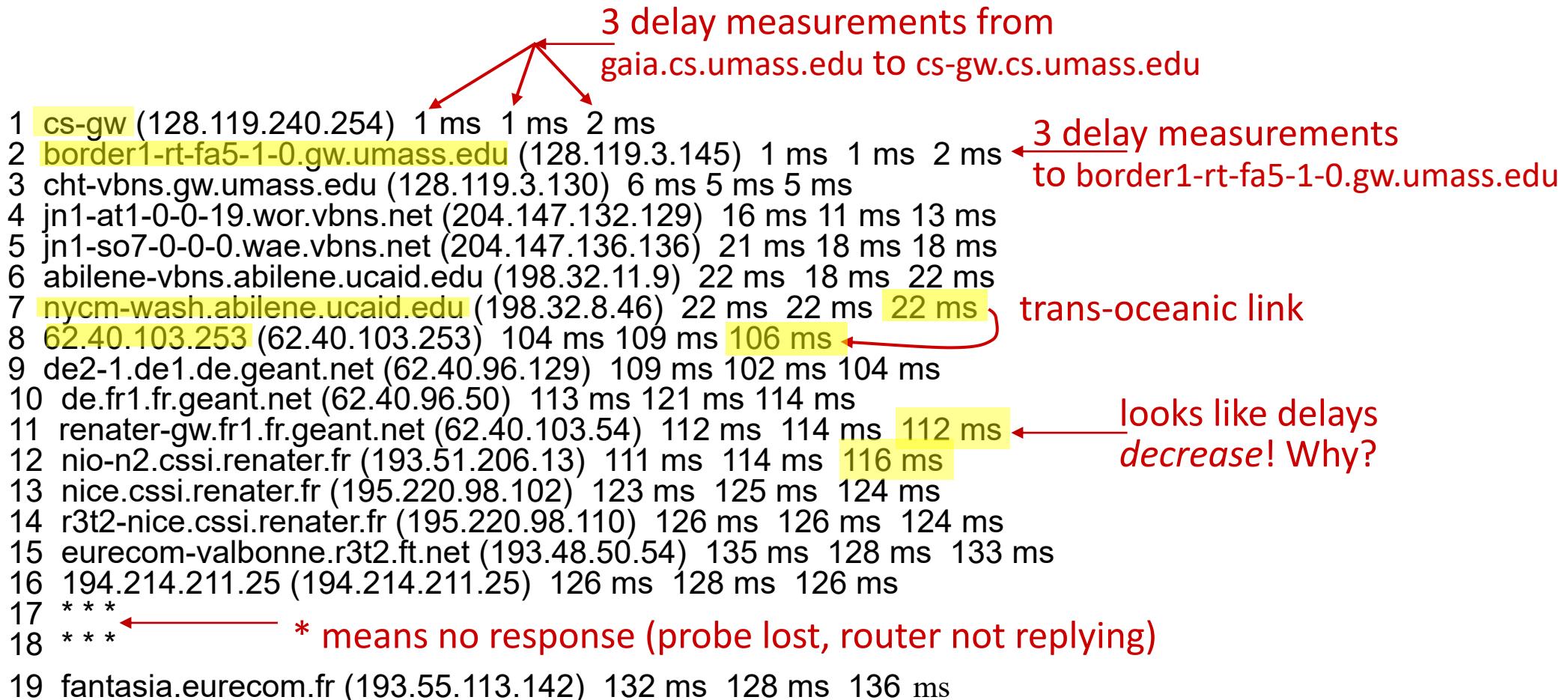
# “Real” Internet delays and routes

- what do “real” Internet delay & loss look like?
- **traceroute** program: provides delay measurement from source to router along end-end Internet path towards destination. For all  $i$ :
  - sends three packets that will reach router  $i$  on path towards destination (with time-to-live field value of  $i$ )
  - router  $i$  will return packets to sender
  - sender measures time interval between transmission and reply



# Real Internet delays and routes

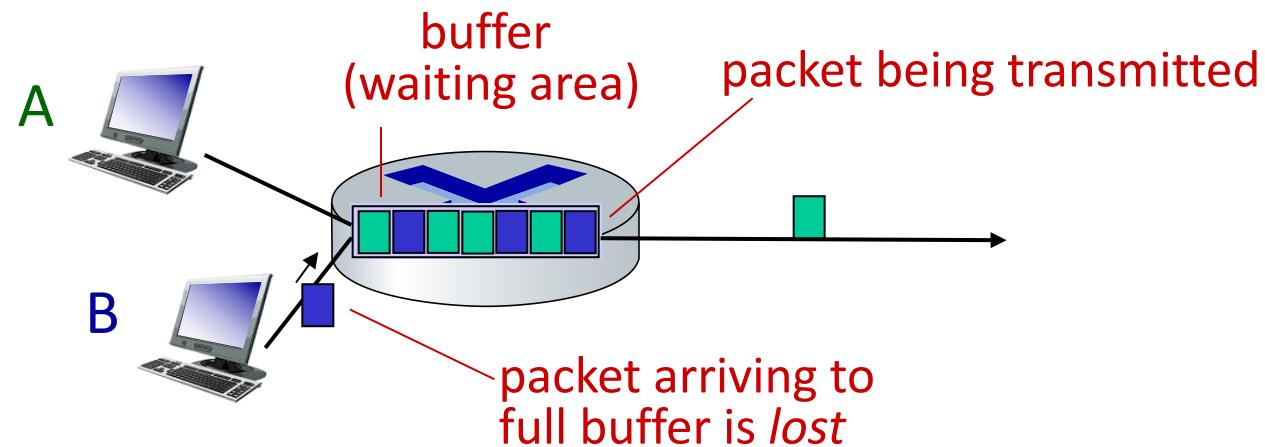
traceroute: gaia.cs.umass.edu to www.eurecom.fr



\* Do some traceroutes from exotic countries at [www.traceroute.org](http://www.traceroute.org)

# Packet loss

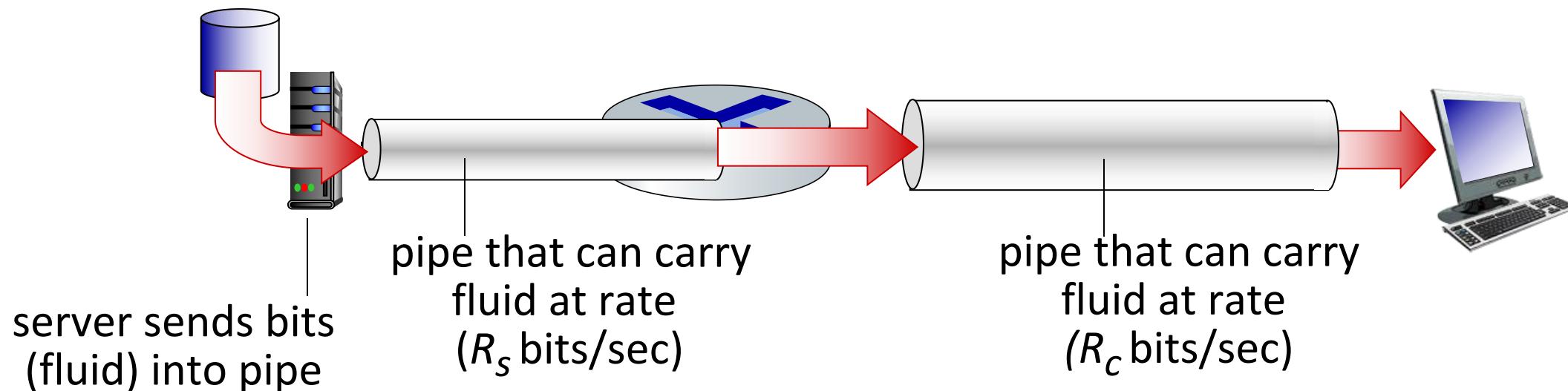
- queue (aka buffer) preceding link in buffer has finite capacity
- packet arriving to full queue dropped (aka lost)
- lost packet may be retransmitted by previous node, by source end system, or not at all



\* Check out the Java applet for an interactive animation (on publisher's website) of queuing and loss

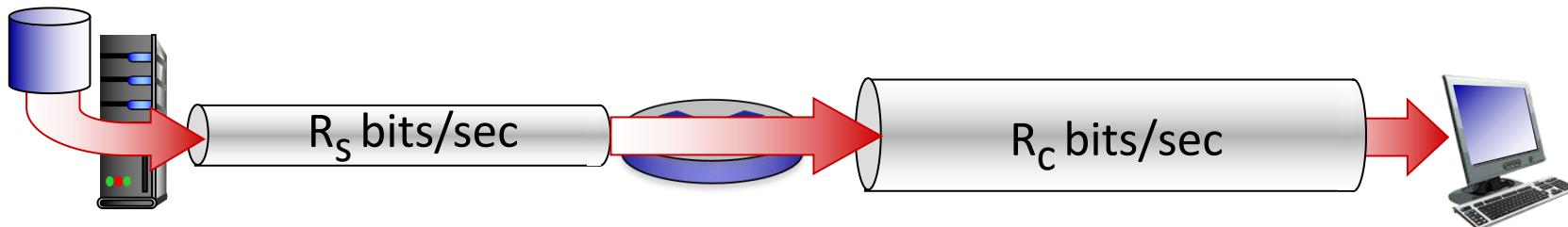
# Throughput

- **throughput:** rate (bits/time unit) at which bits are being sent from sender to receiver
  - *instantaneous:* rate at given point in time
  - *average:* rate over longer period of time

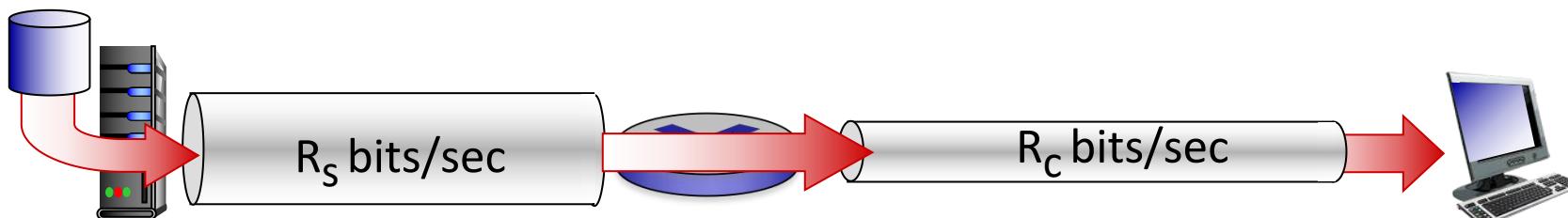


# Throughput

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



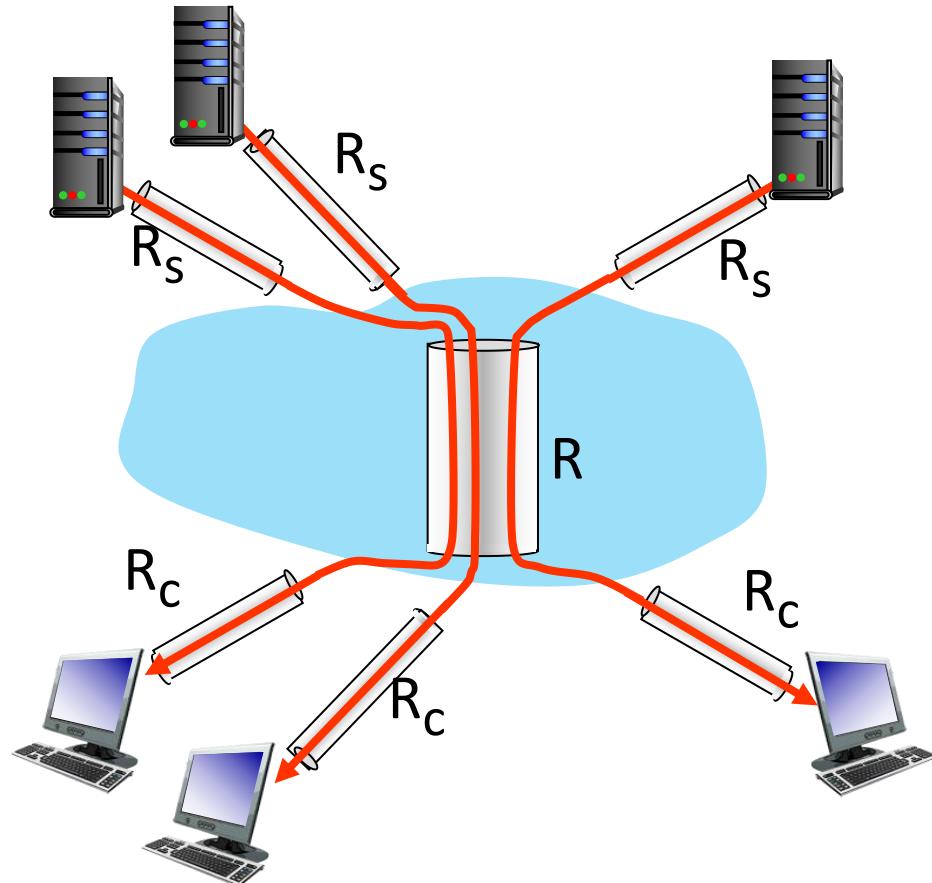
$R_s > R_c$  What is average end-end throughput?



*bottleneck link*

link on end-end path that constrains end-end throughput

# Throughput: network scenario



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

- per-connection end-end throughput:  $\min(R_c, R_s, R/10)$
- in practice:  $R_c$  or  $R_s$  is often bottleneck

\* Check out the online interactive exercises for more examples: [http://gaia.cs.umass.edu/kurose\\_ross/](http://gaia.cs.umass.edu/kurose_ross/)