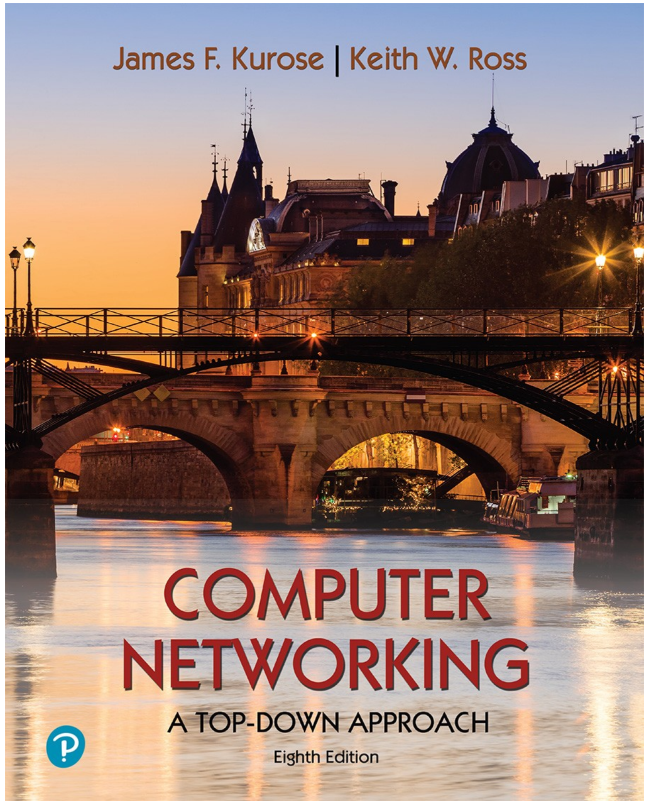
Chapter 1
Introduction



CS 455
Parth Pathak

Computer Networking: A Top-Down Approach
8th edition
Jim Kurose, Keith Ross
Pearson, 2020

Acknowledgement:

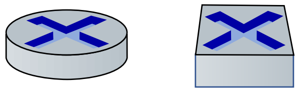
All slides in this course are adapted from the textbook (shown on the right)

Chapter 1: roadmap

- What is the Internet?
- What is a protocol?
- Network edge: hosts, access network, physical media
- Network core: internet structure
- Performance: packet/circuit switching; loss, delay, throughput
- Protocol layers, service models
- History

The Internet: a “nuts and bolts” view

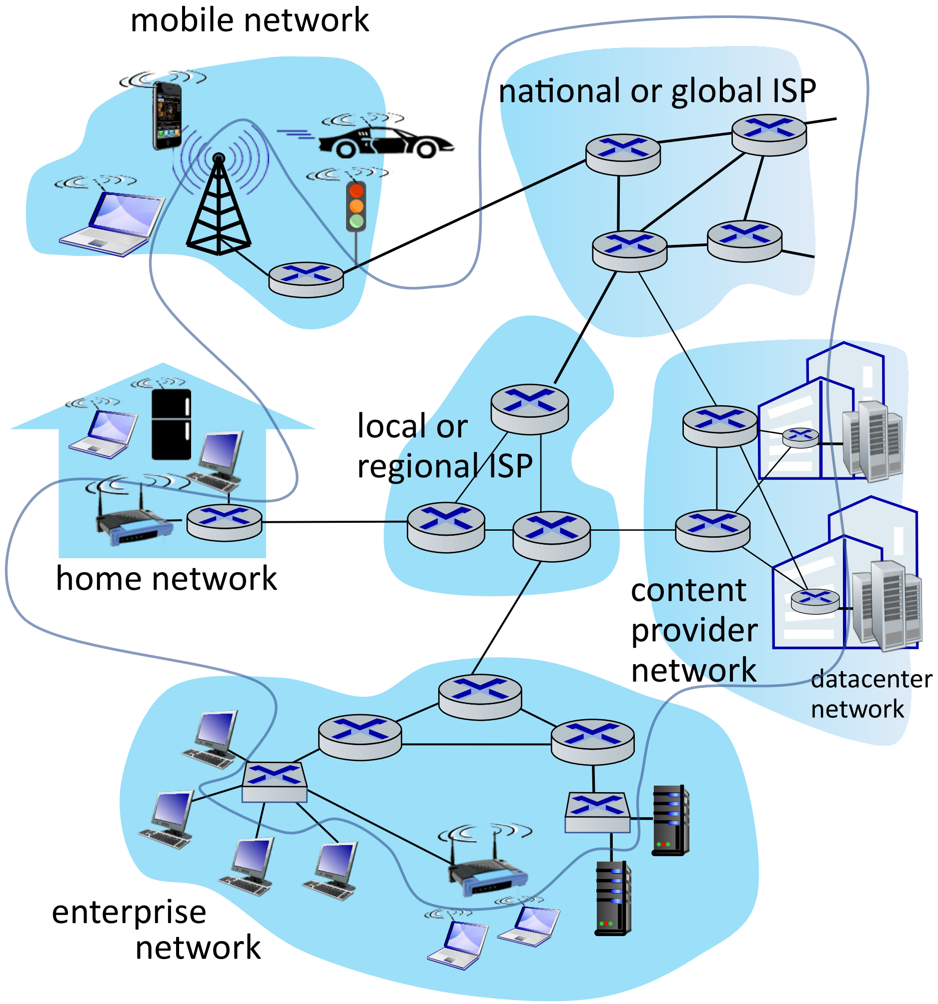
Billions of connected computing *devices*:
- *hosts = end systems*
- running *network apps* at Internet’s “edge”



Packet switches: forward packets (chunks of data)
- *routers, switches*

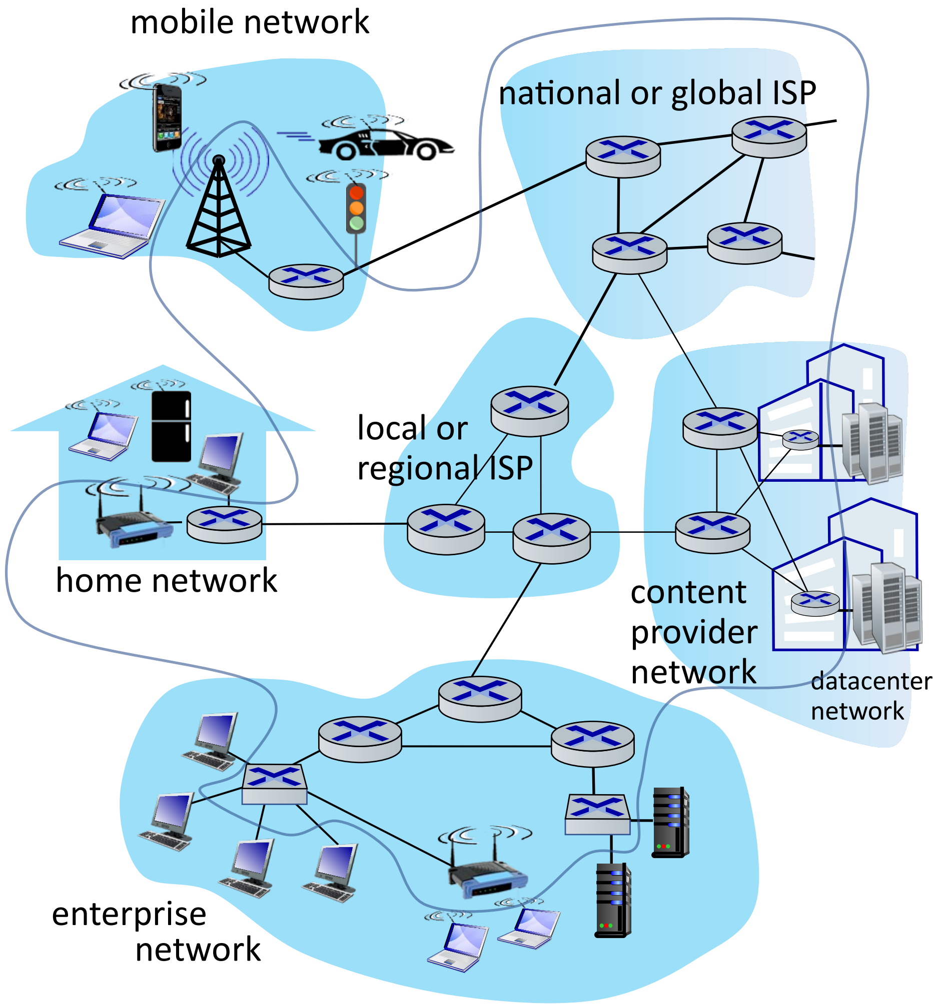
Communication links
- fiber, copper, radio, satellite
- transmission rate: *bandwidth*

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



The Internet: a "nuts and bolts" view

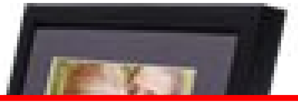
- *Internet:* "network of networks"
  - Interconnected ISPs



Introduction: 1-4

“Fun” Internet-connected devices

Amazon Echo



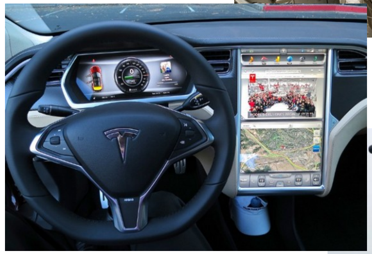
Tweet-a-watt:
monitor energy use

bikes

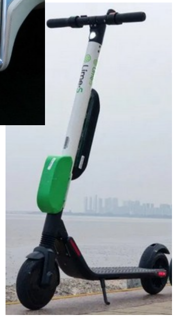
How many devices are connected to Internet?

5 Billion in 2015
2019 said it will be 25 Billion by 2020
It was 50 billion by 2020

-enabled toaster +
ther forecaster



cars



Security Camera

AR devices

scooters

Internet phones

Gaming devices

sensorized,
bed
mattress

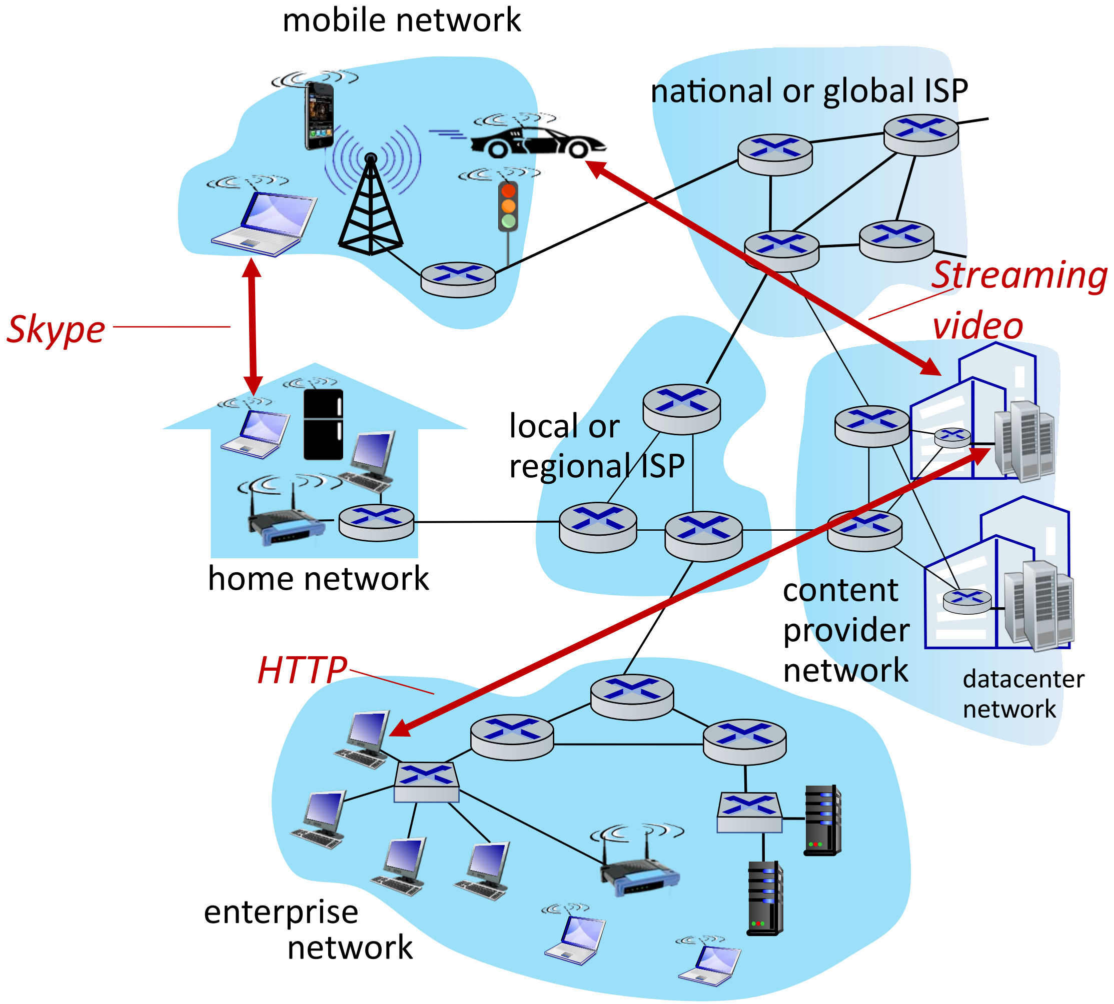
Fitbit

Others?

The Internet: a “services” view

- *Infrastructure* that provides services to applications:
  - Web, streaming video, emails etc.

- provides *programming interface* to distributed applications:
  - “hooks” allowing sending/receiving apps to “connect” to, use Internet transport service



Chapter 1: roadmap

- What is the Internet?
- What is a protocol?
- Network edge: hosts, access network, physical media
- Network core: internet structure
- Performance: packet/circuit switching; loss, delay, throughput
- Protocol layers, service models
- History

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

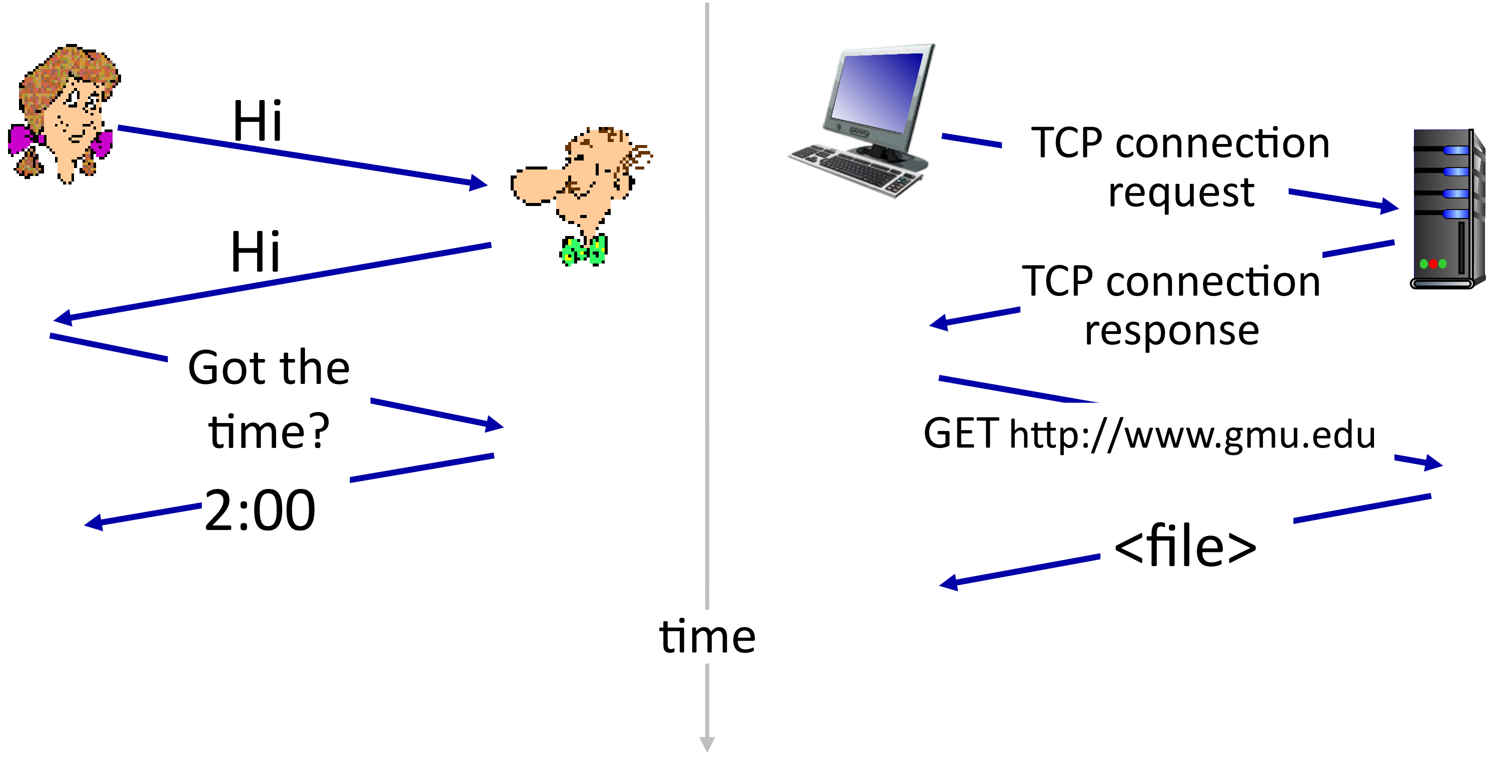
Network protocols:

- 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:



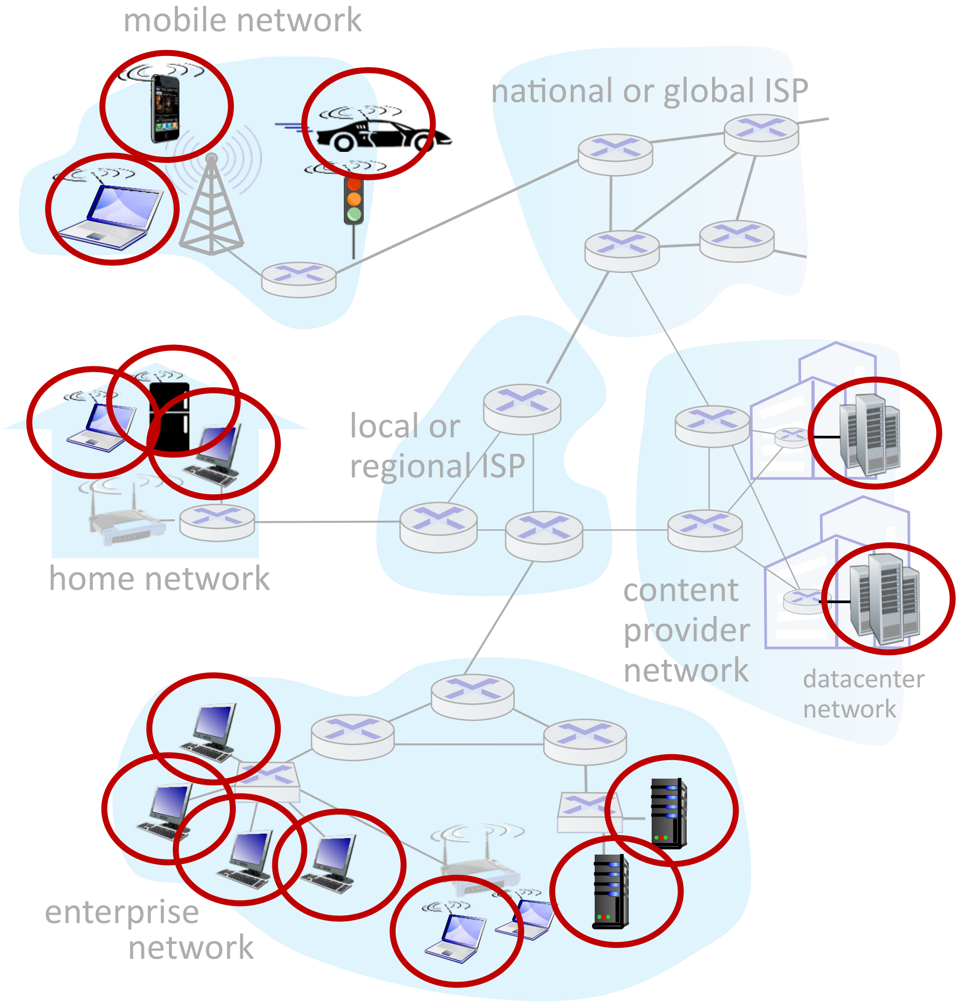
Chapter 1: roadmap

- What is the Internet?
- What is a protocol?
- Network edge: hosts, access network, physical media
- Network core: internet structure
- Performance: packet/circuit switching; loss, delay, throughput
- Protocol layers, service models
- History

A closer look at Internet structure

Network edge:

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



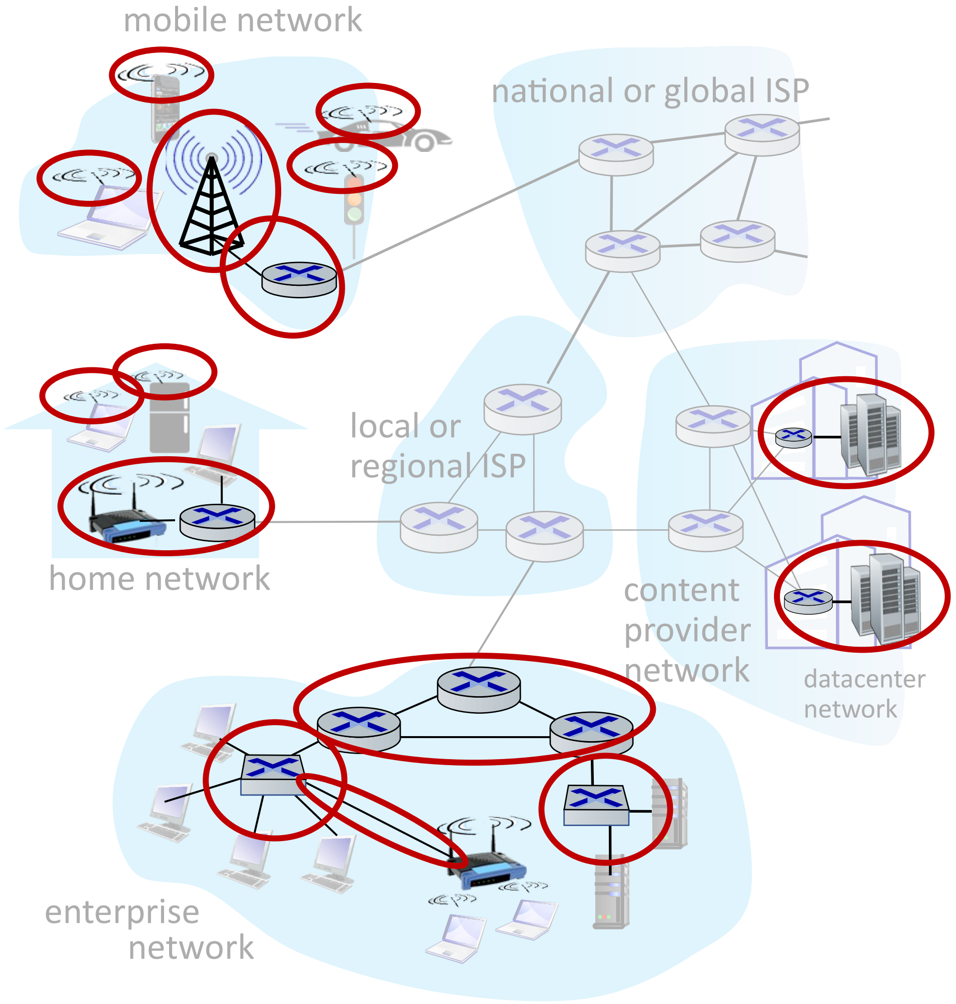
A closer look at Internet structure

Network edge:

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

Access networks, physical media:

- wired, wireless communication links

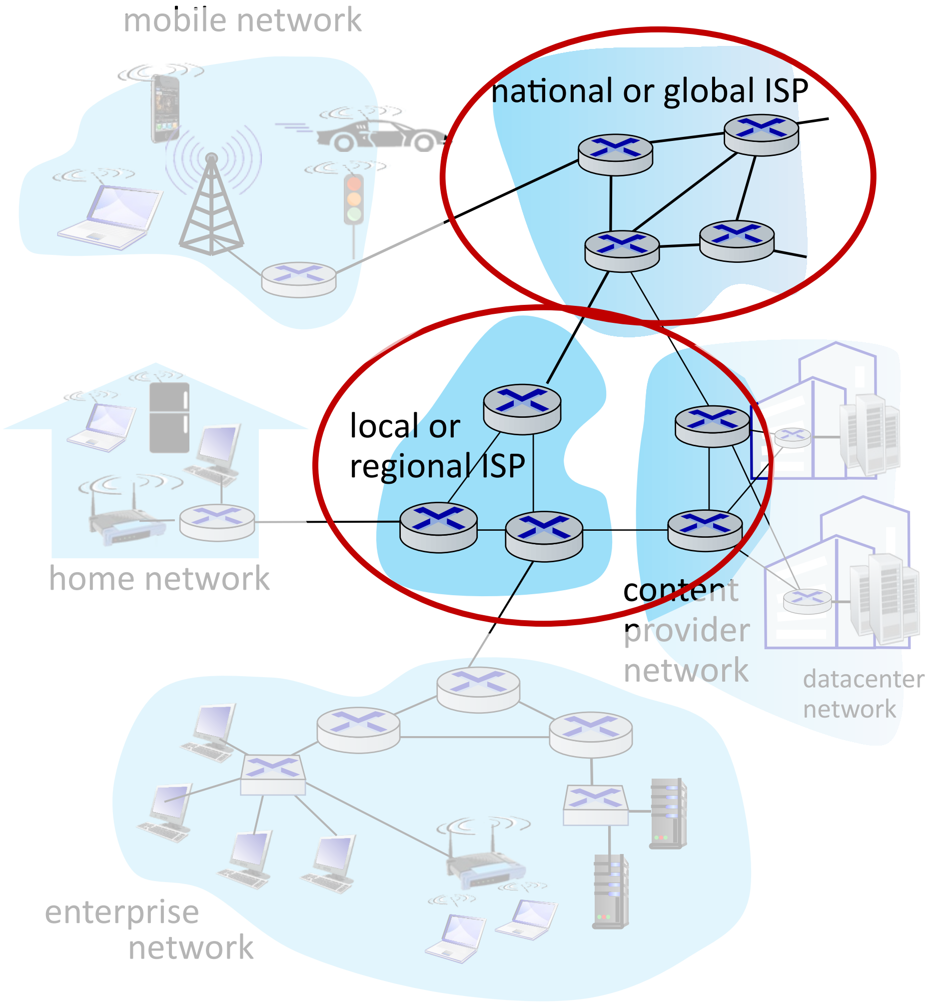


A closer look at Internet structure

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

Access networks, physical media:
- wired, wireless communication links

Network core:
- interconnected routers
- network of networks

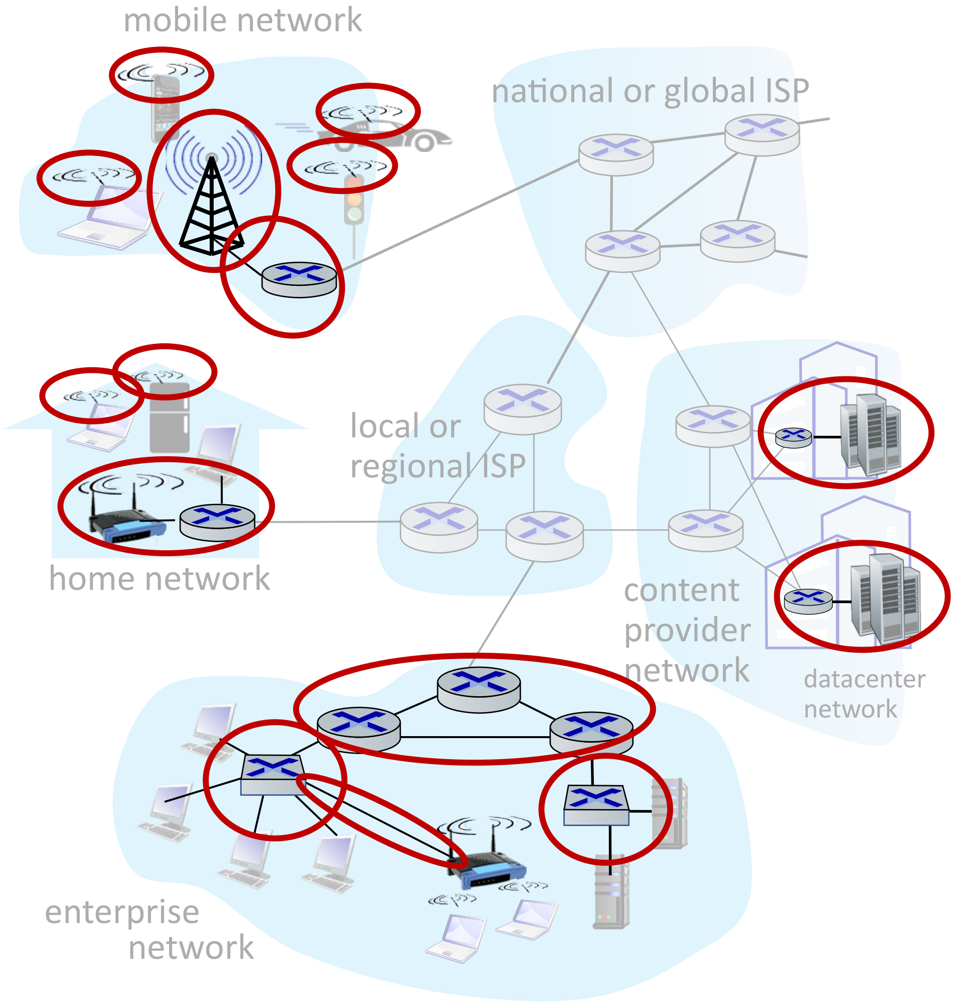


Interstate vs. state highways vs. streets

Access networks and physical media

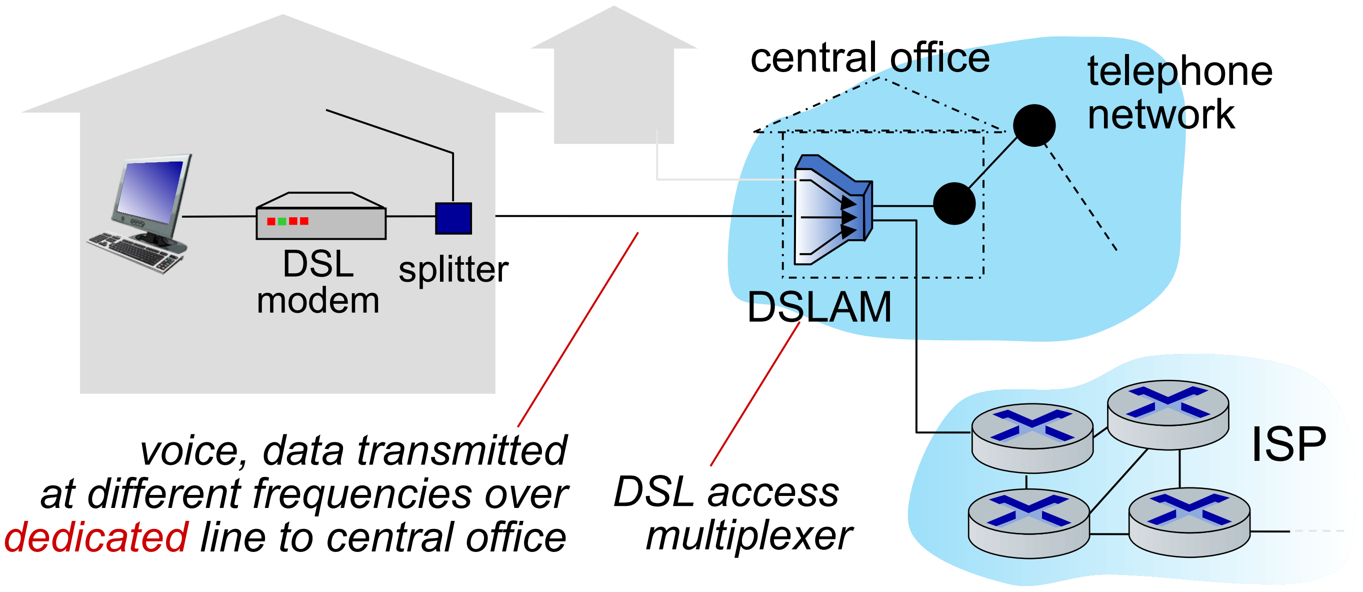
Q: How to connect end systems to edge router?

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



Q: How do you connect your devices to Internet at your home?

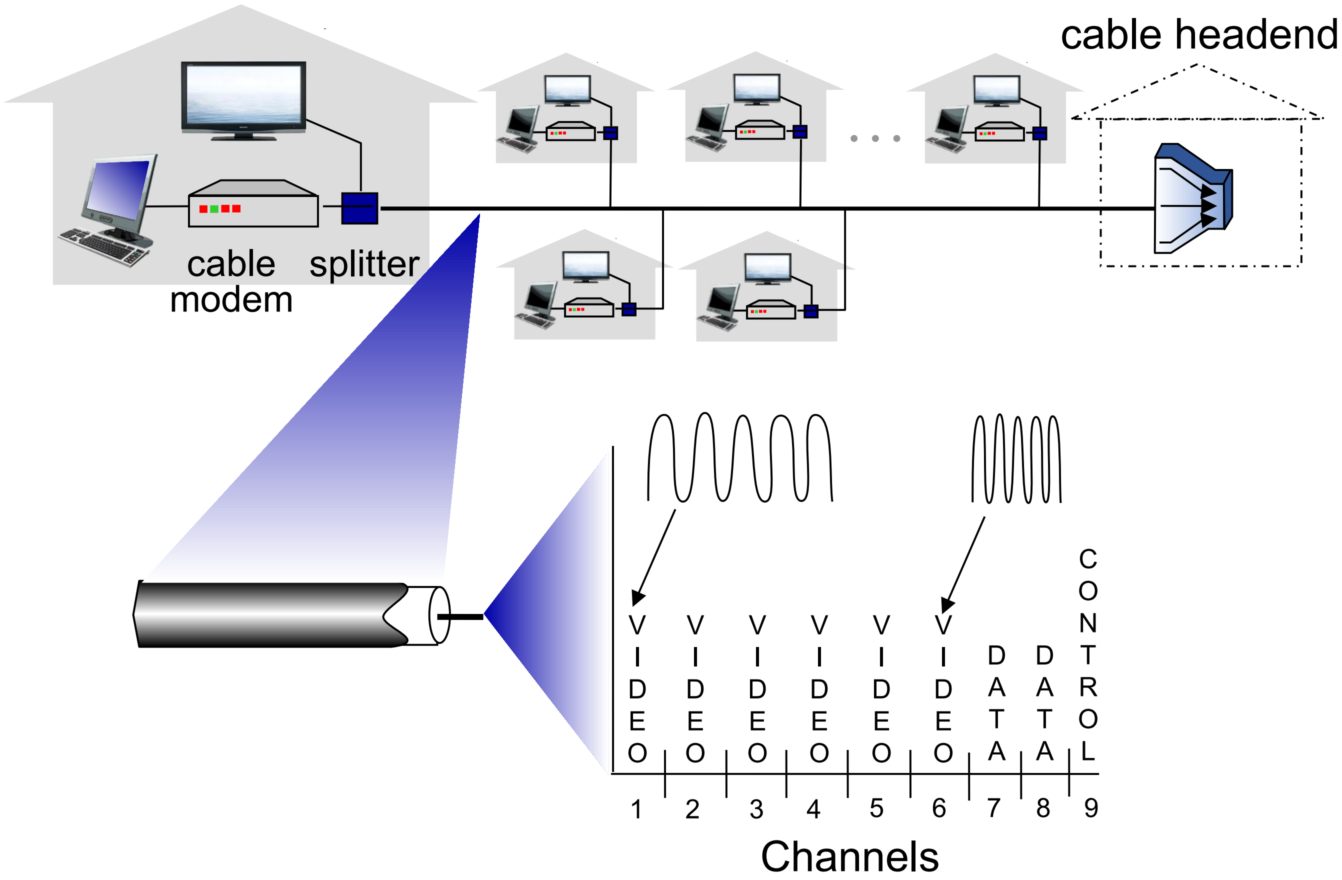
Access networks: digital subscriber line (DSL)



- 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

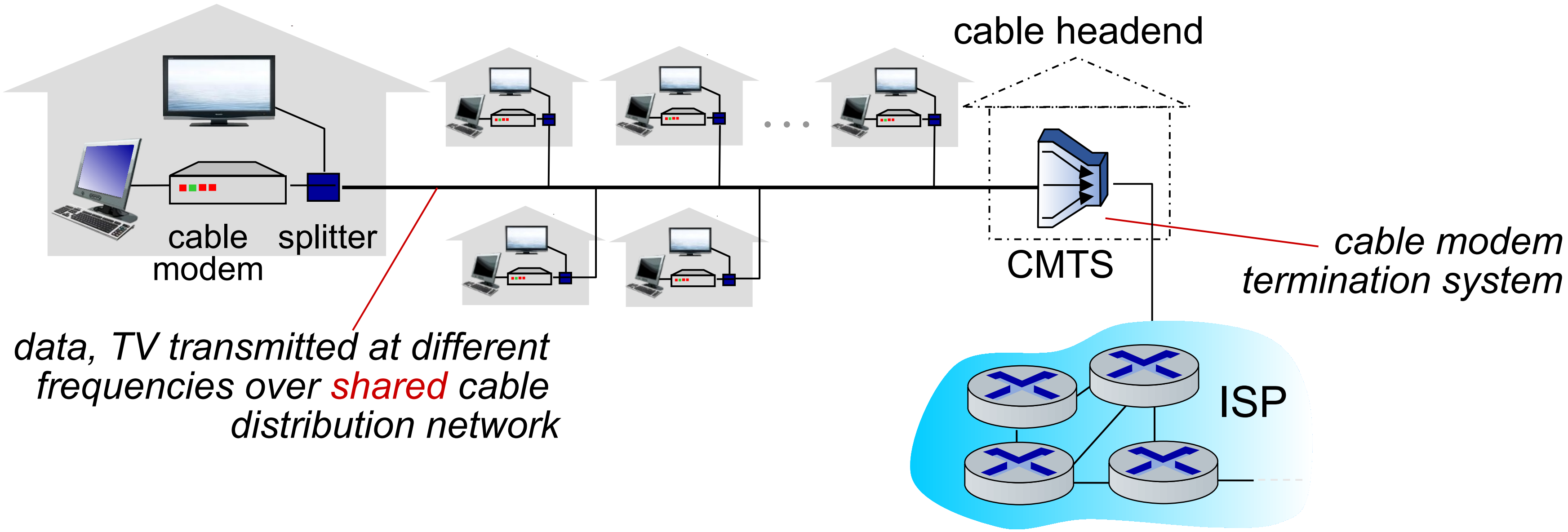
Bps, Kbps, Mbps, Gbps?

Access networks: cable-based access



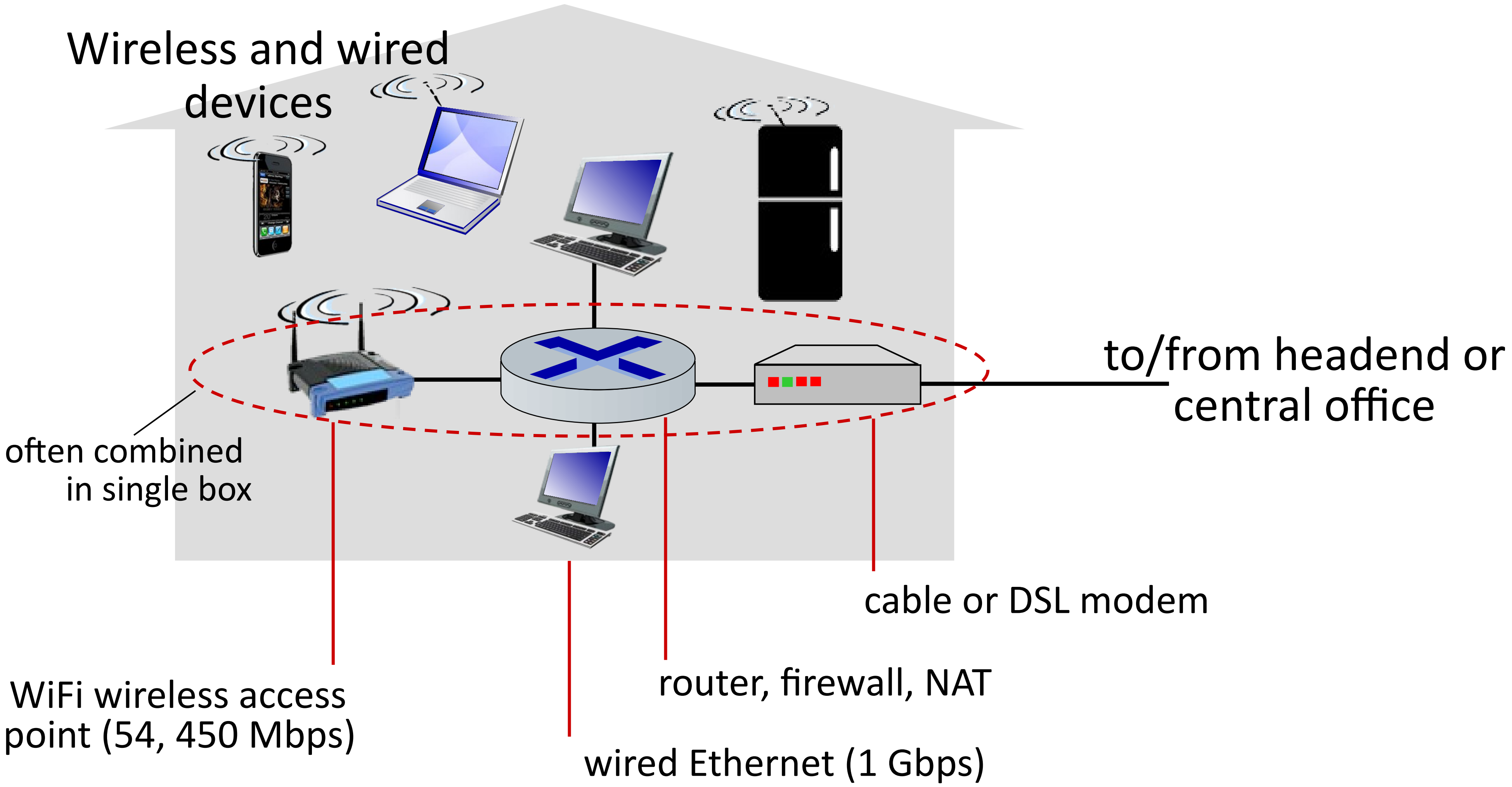
frequency division multiplexing (FDM): different channels transmitted in different frequency bands

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: home networks



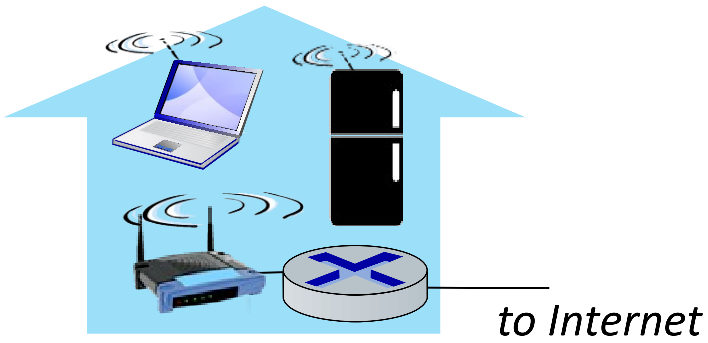
Wireless access networks

Shared *wireless* access network connects end system to router

- via base station aka "access point"

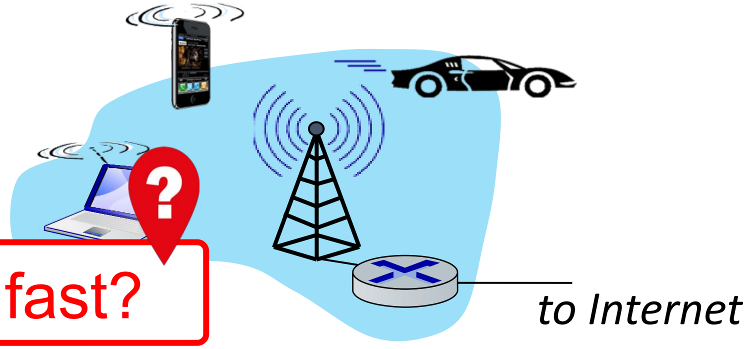
Wireless local area networks (WLANs)

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



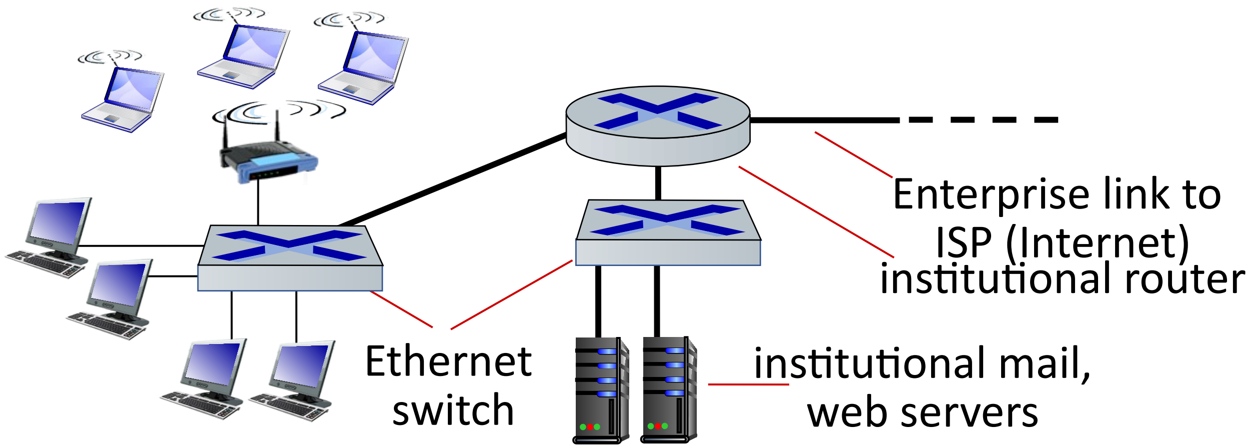
Wide-area cellular access networks

- provided by mobile, cellular network operator (10's km)
- 10's Mbps over 4G
- Over 300 Mbps over 5G



Why is 5G so fast?

Access networks: enterprise networks

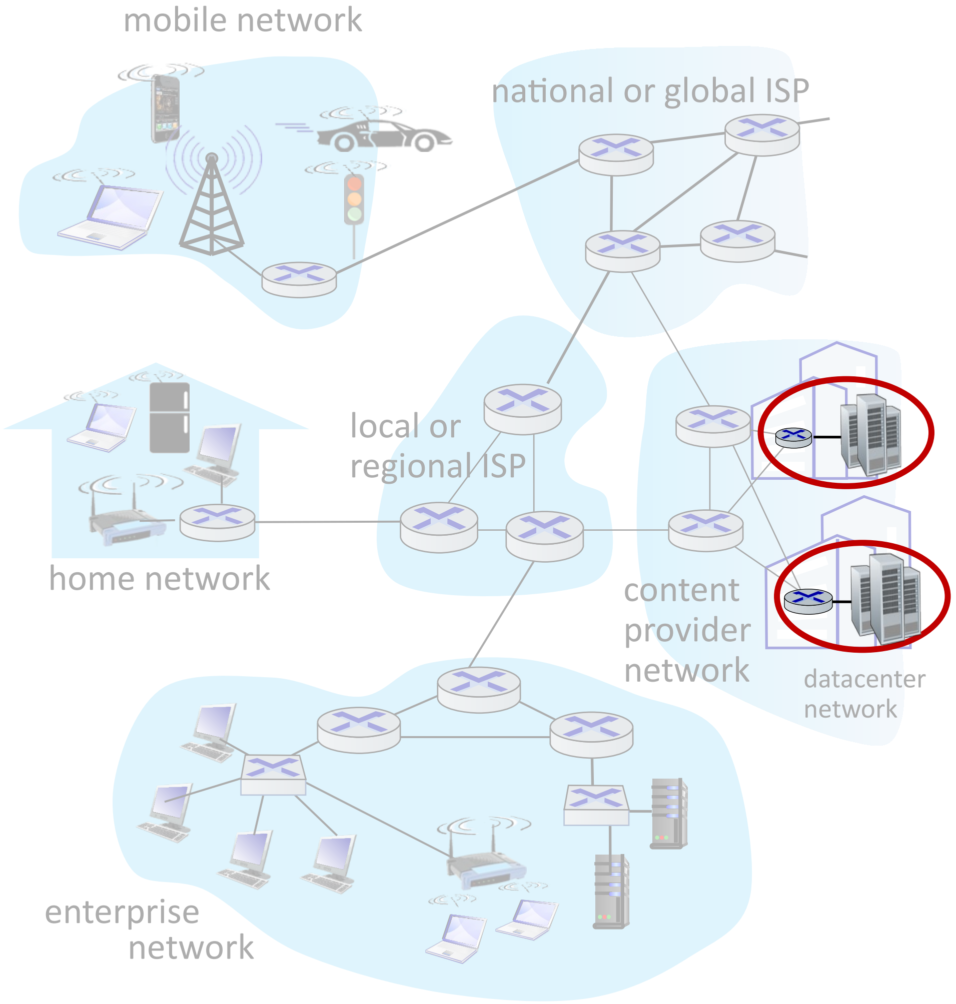


- companies, universities, etc.
- mix of wired, wireless link technologies, connecting a mix of switches and routers (we'll cover differences shortly)
  - Ethernet: wired access at 100Mbps, 1Gbps, 10Gbps
  - WiFi: wireless access points at 11, 54, 450 Mbps

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)

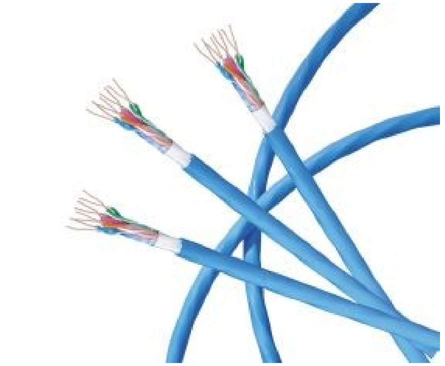


Links: physical media

- physical link: what lies between transmitter & receiver
- Wired or wireless

Twisted pair (TP)

- two insulated copper wires
  - Category 5: 100 Mbps, 1 Gbps Ethernet
  - Category 6: 10Gbps Ethernet



Links: physical media

Coaxial cable:

- two concentric copper conductors
- broadband:
  - multiple frequency channels on cable
  - 100's Mbps per channel

Fiber optic cable:

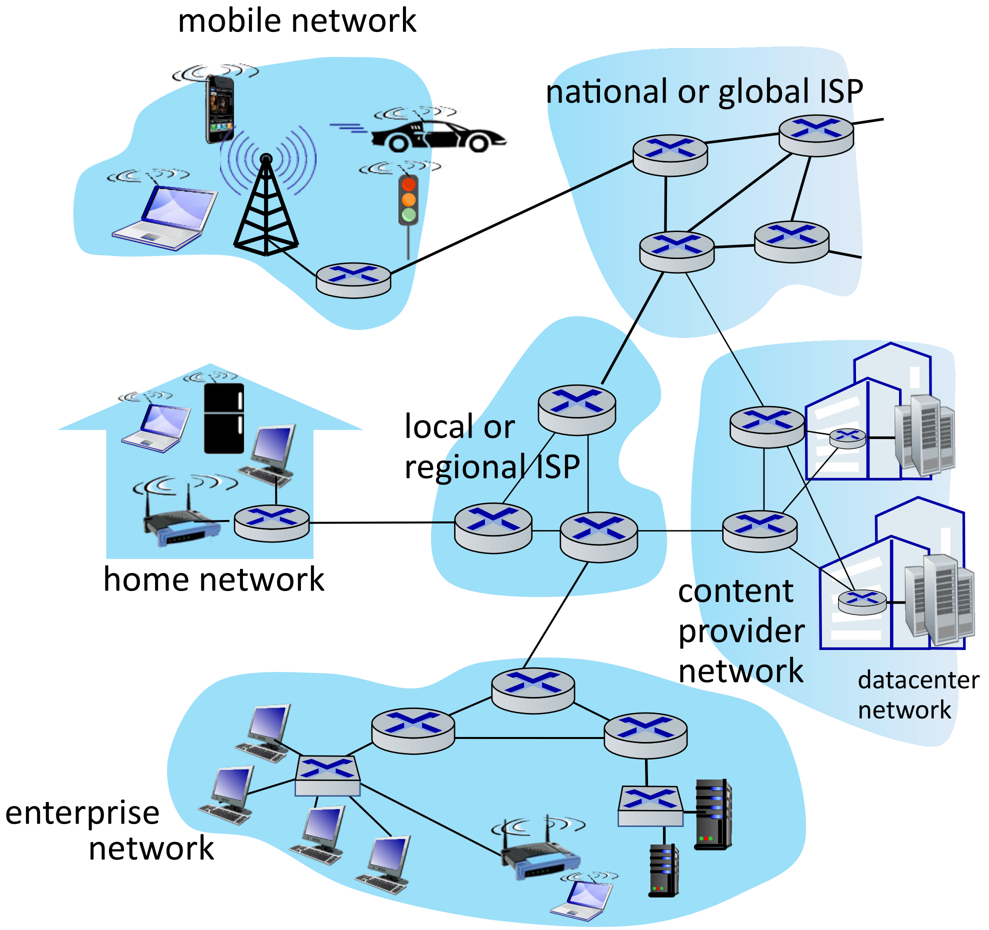
- glass fiber carrying light pulses, each pulse a bit
- high-speed operation:
  - high-speed point-to-point transmission (10's-100's Gbps)

Chapter 1: roadmap

- What is the Internet?
- What is a protocol?
- Network edge: hosts, access network, physical media
- Network core: internet structure
- Performance: packet/circuit switching; loss, delay, throughput
- Protocol layers, service models
- History

Internet structure: a "network of networks"

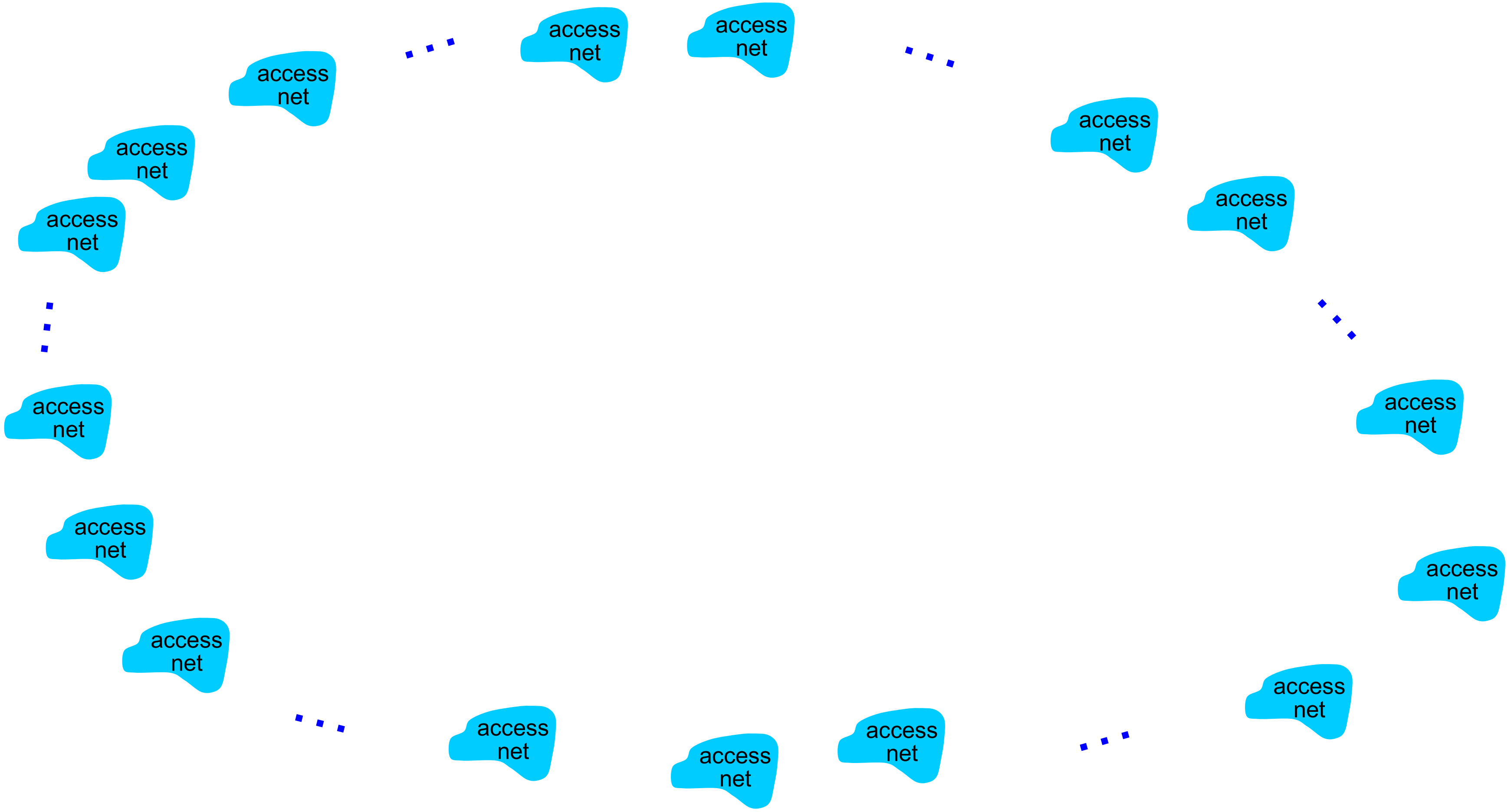
- hosts connect to Internet via access Internet Service Providers (ISPs)
- access ISPs in turn must be interconnected
  - so that *any* two hosts *(anywhere!)* can send packets to each other
- resulting network of networks is very complex
  - evolution driven by economics, national policies



Let's take a stepwise approach to describe current Internet structure

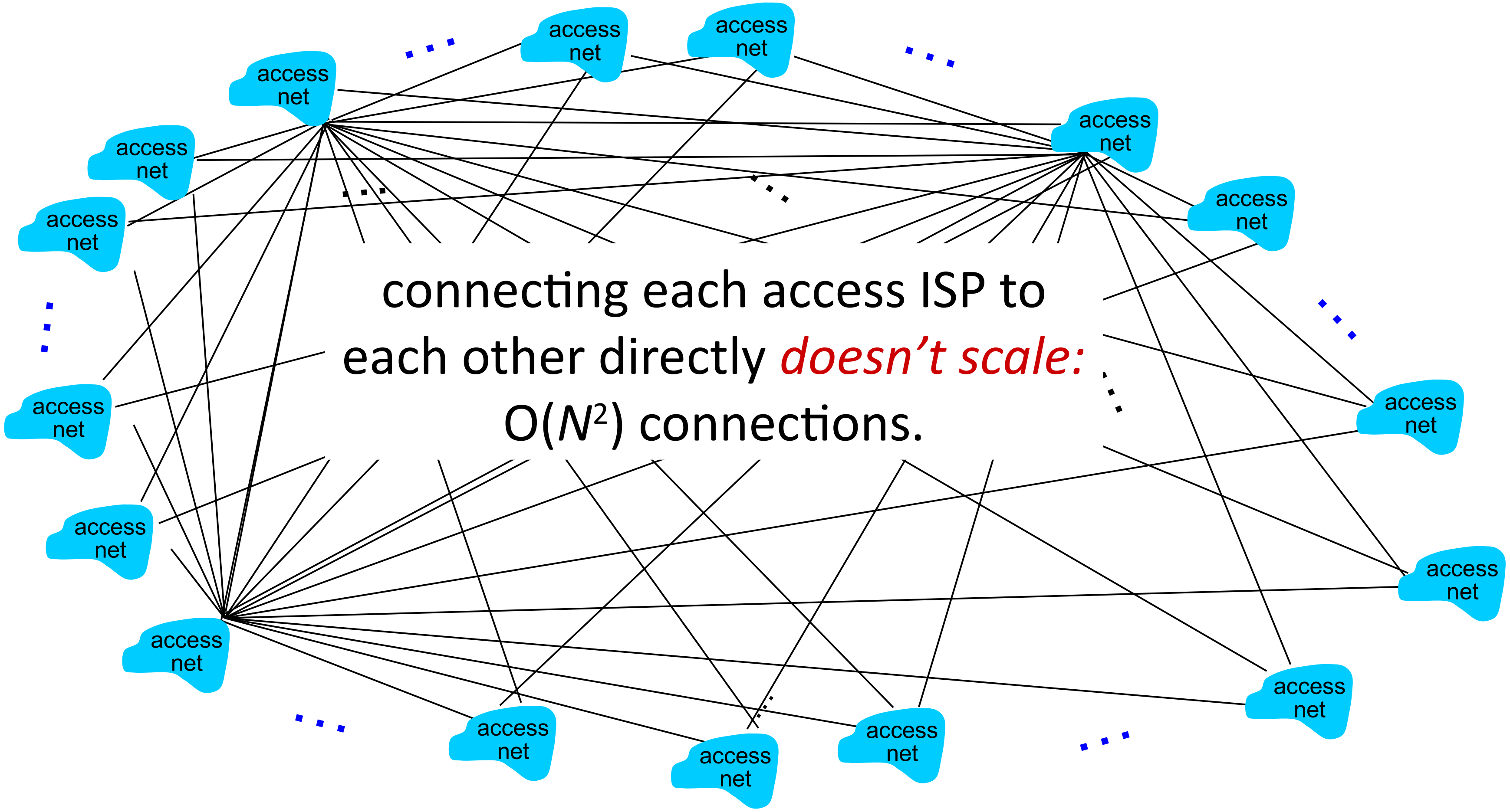
Internet structure: a “network of networks”

Question: given *millions* of access ISPs, how to connect them together?



Internet structure: a "network of networks"

Question: given *millions* of access ISPs, how to connect them together?

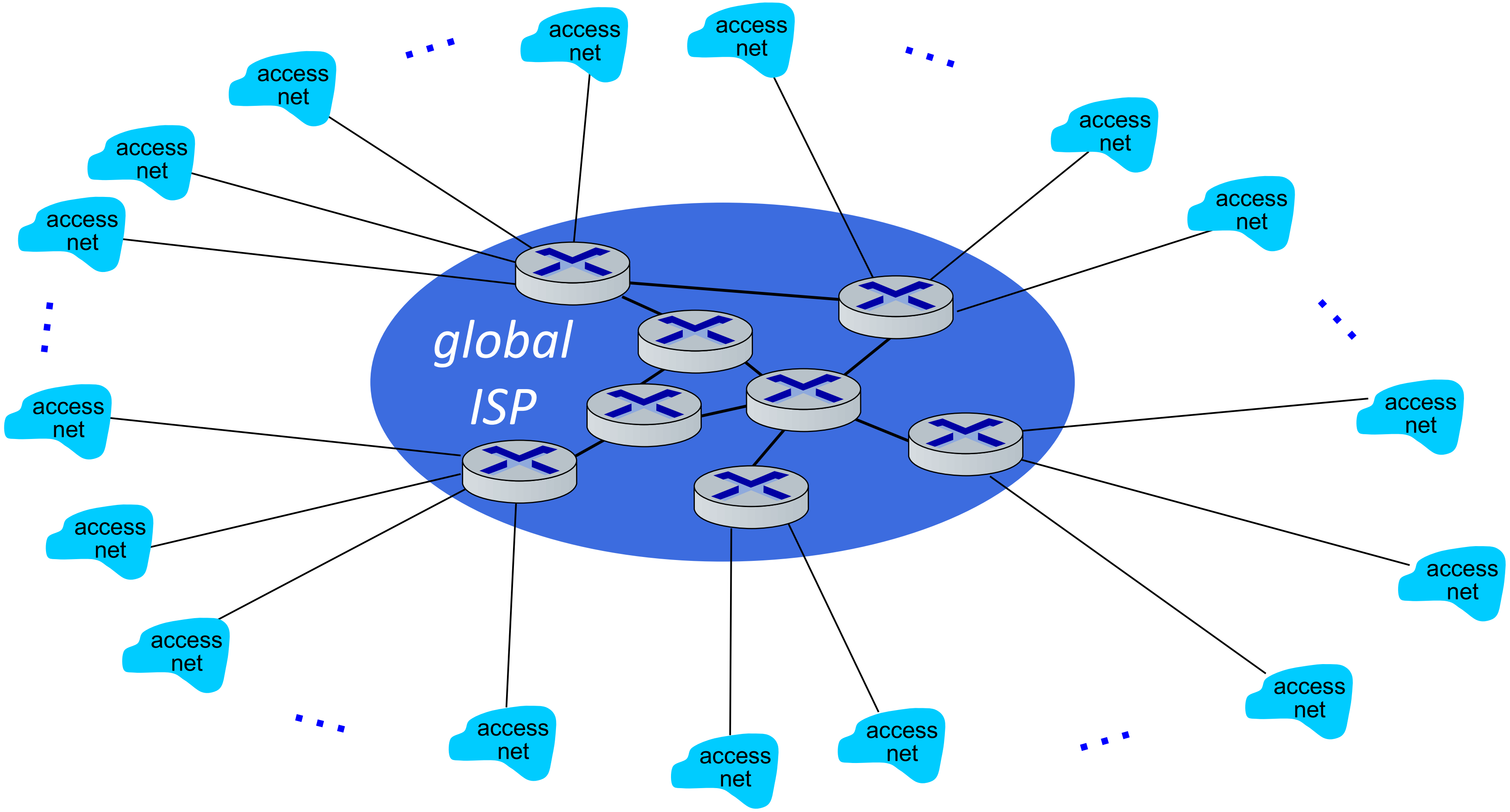


connecting each access ISP to each other directly *doesn't scale:* O(N^2) connections.

Internet structure: a "network of networks"

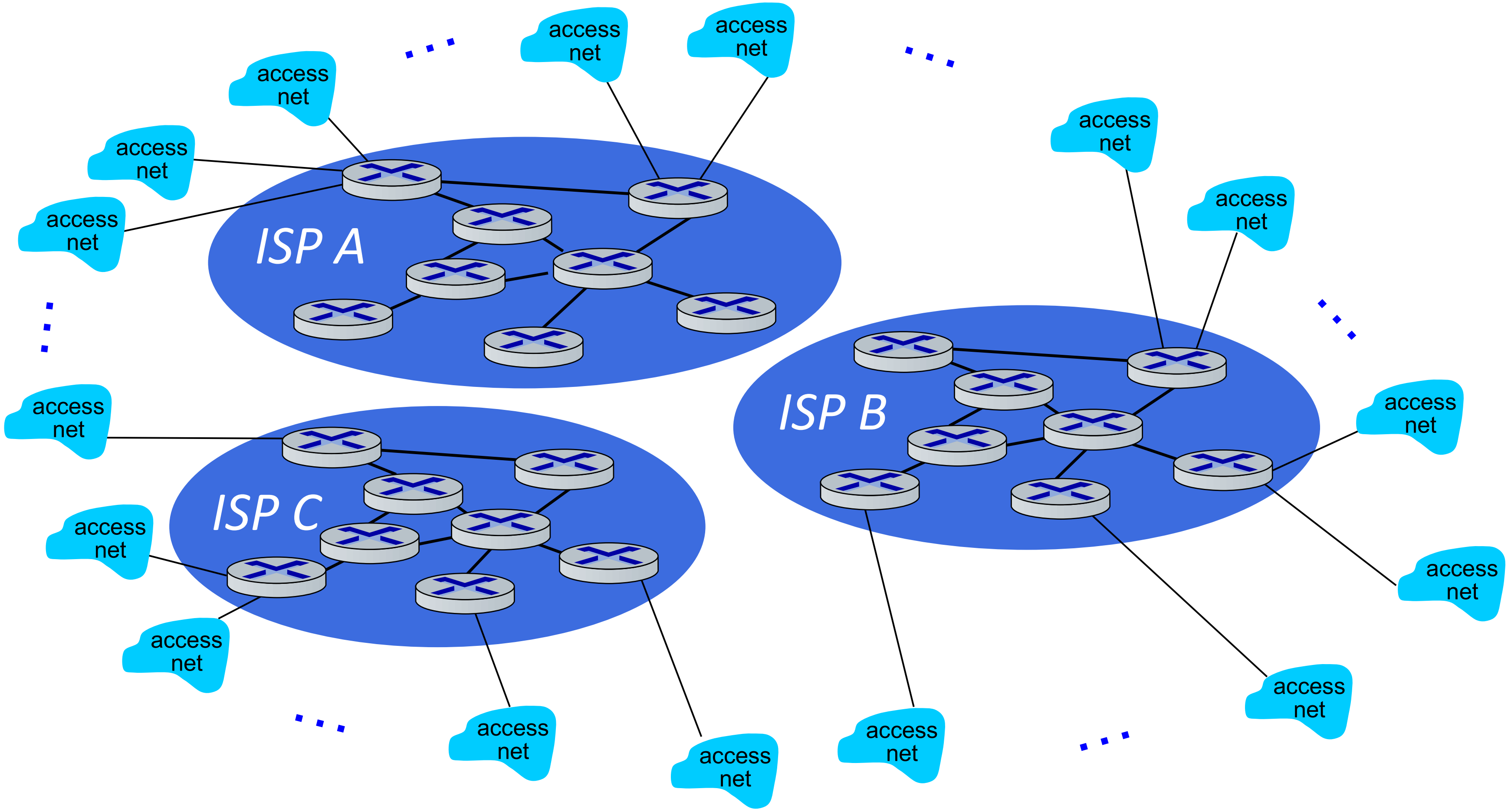
Option: connect each access ISP to one global transit ISP?

Customer and provider ISPs have economic agreement.



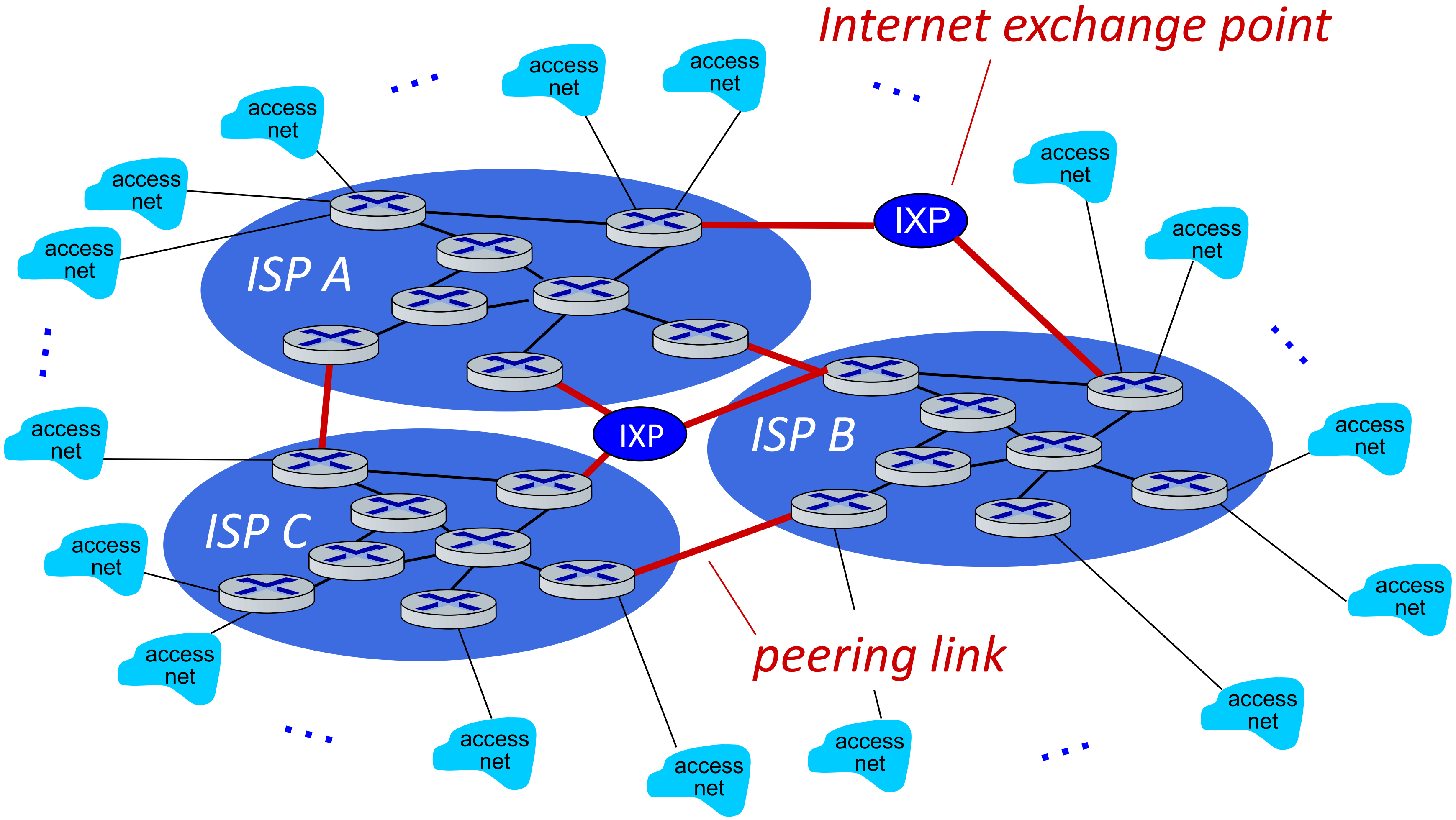
Internet structure: a “network of networks”

But if one global ISP is viable business, there will be competitors ….



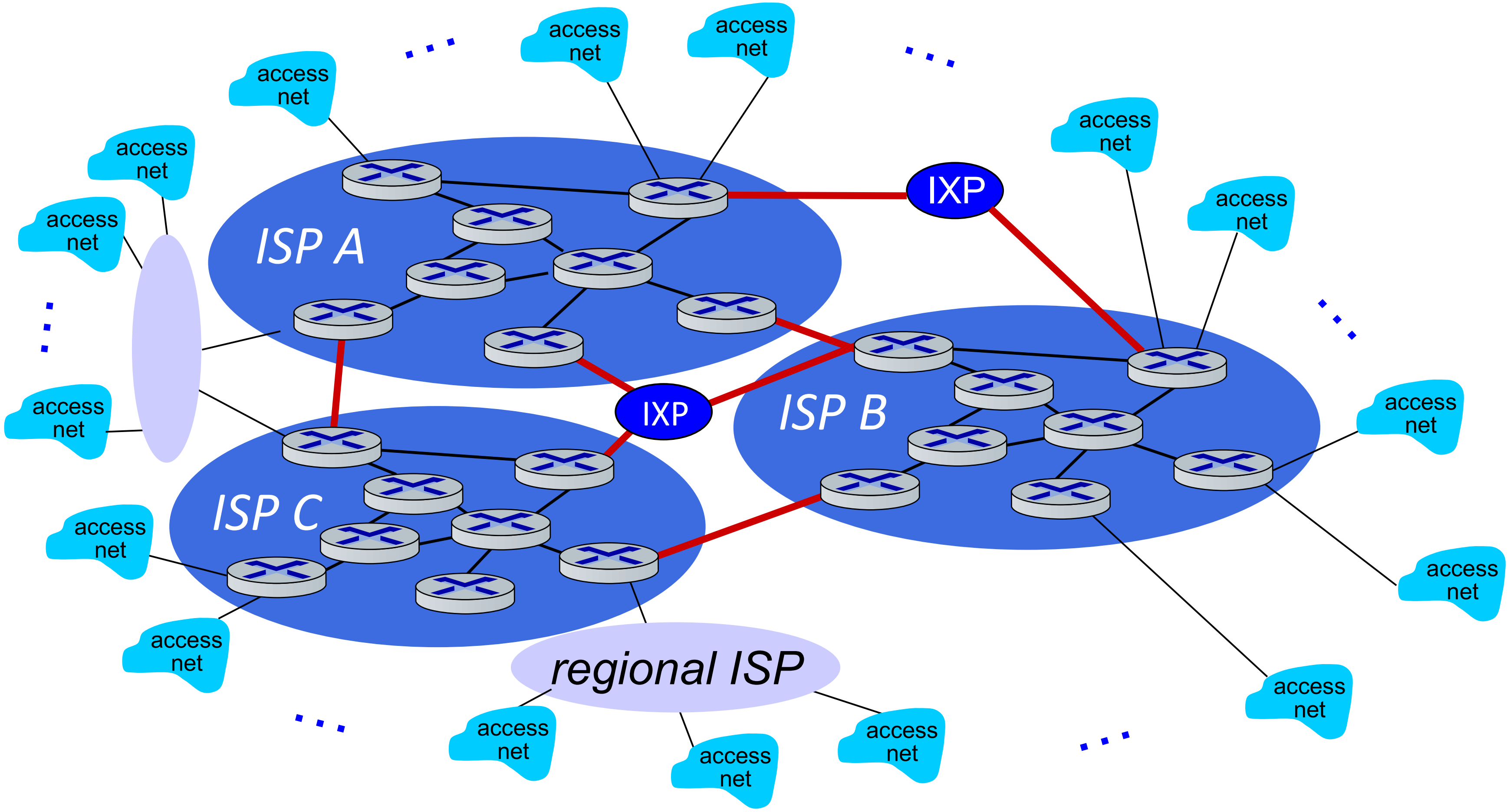
Internet structure: a “network of networks”

But if one global ISP is viable business, there will be competitors …. who will want to be connected



Internet structure: a "network of networks"

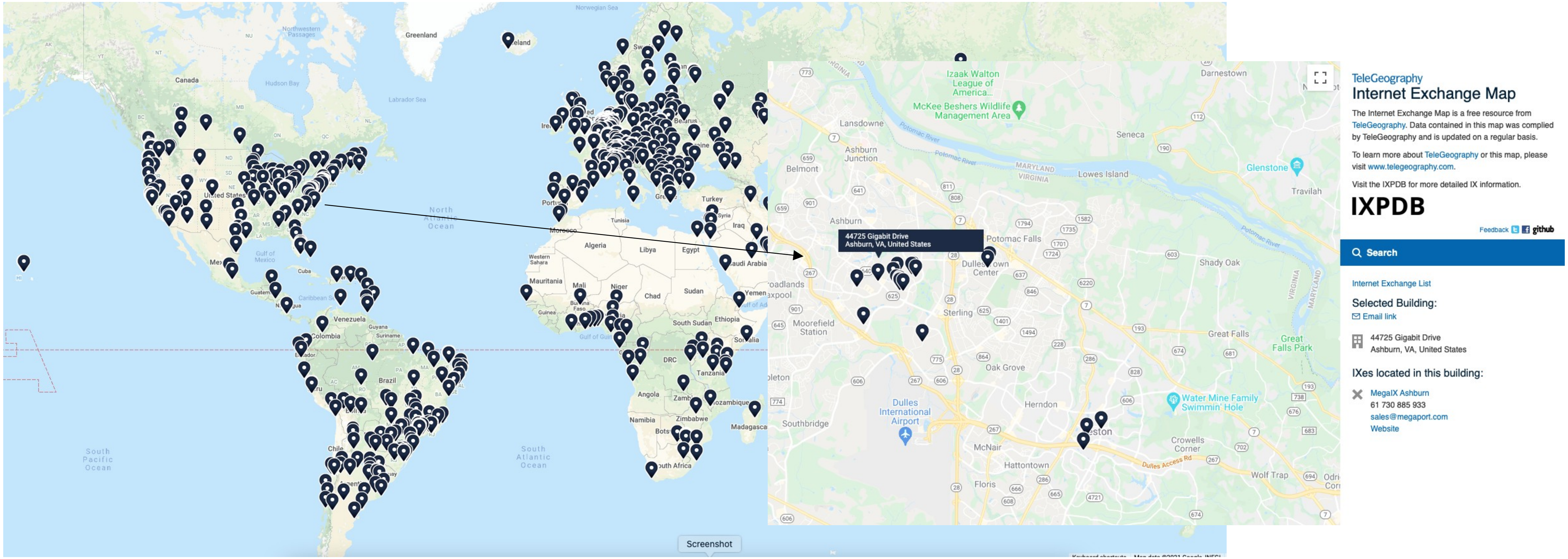
… and regional networks may arise to connect access nets to ISPs



Internet Exchange Points (IXPs)

Critically important component of the Internet

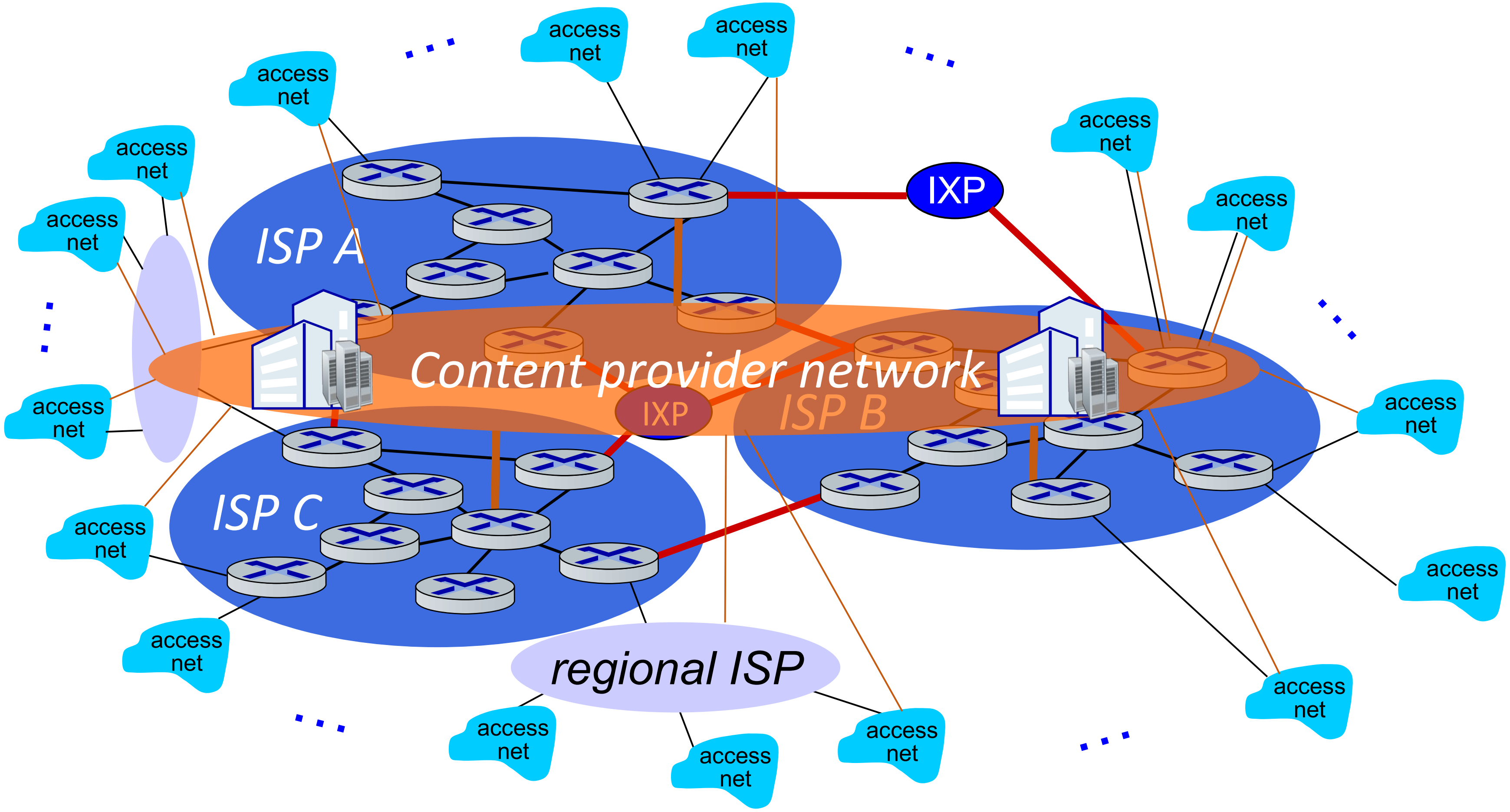
Can be run by non-profits, governments or private companies



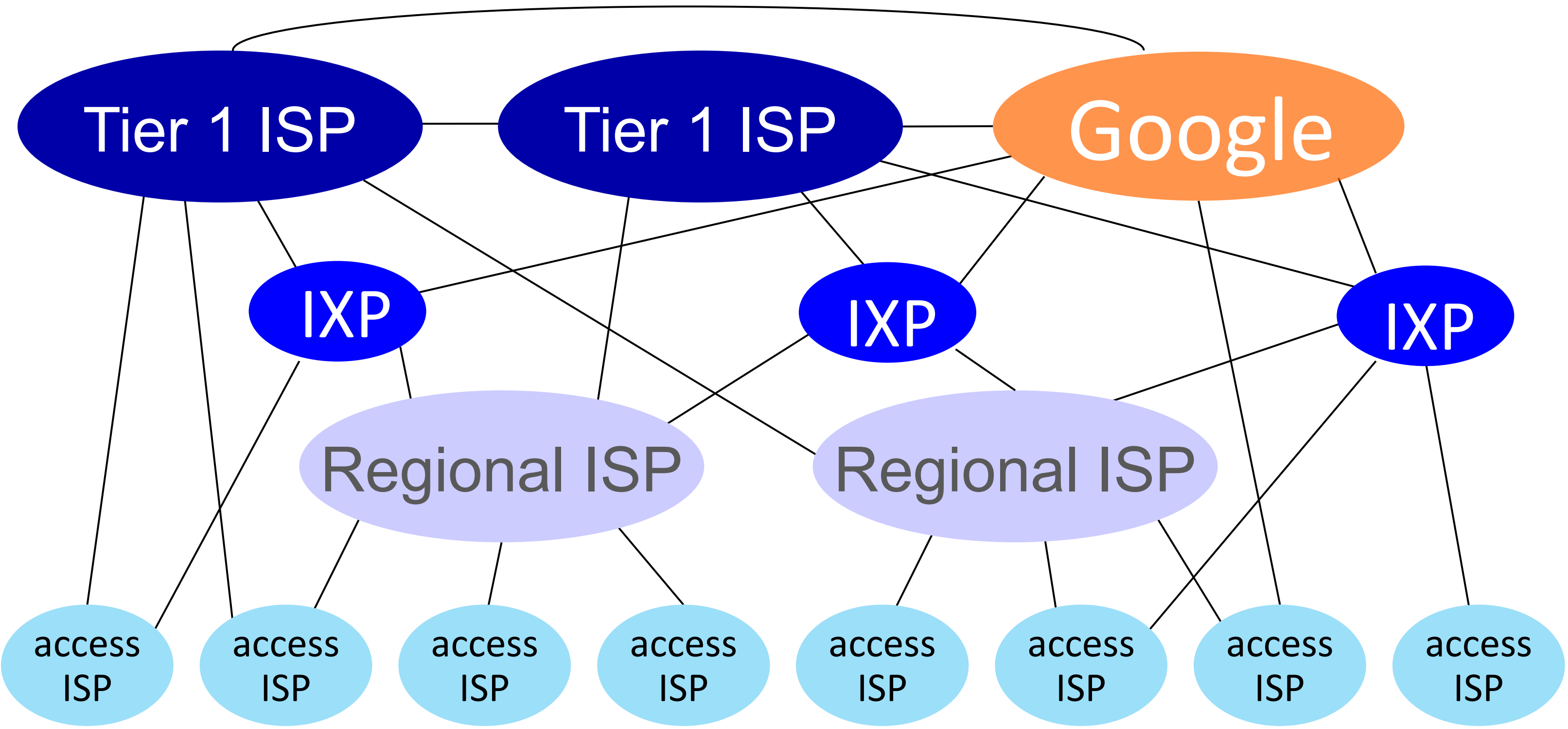
Checkout https://www.internetexchangemap.com/

Internet structure: a "network of networks"

… and content provider networks (e.g., Google, Microsoft, Akamai) may run their own network, to bring services, content close to end users



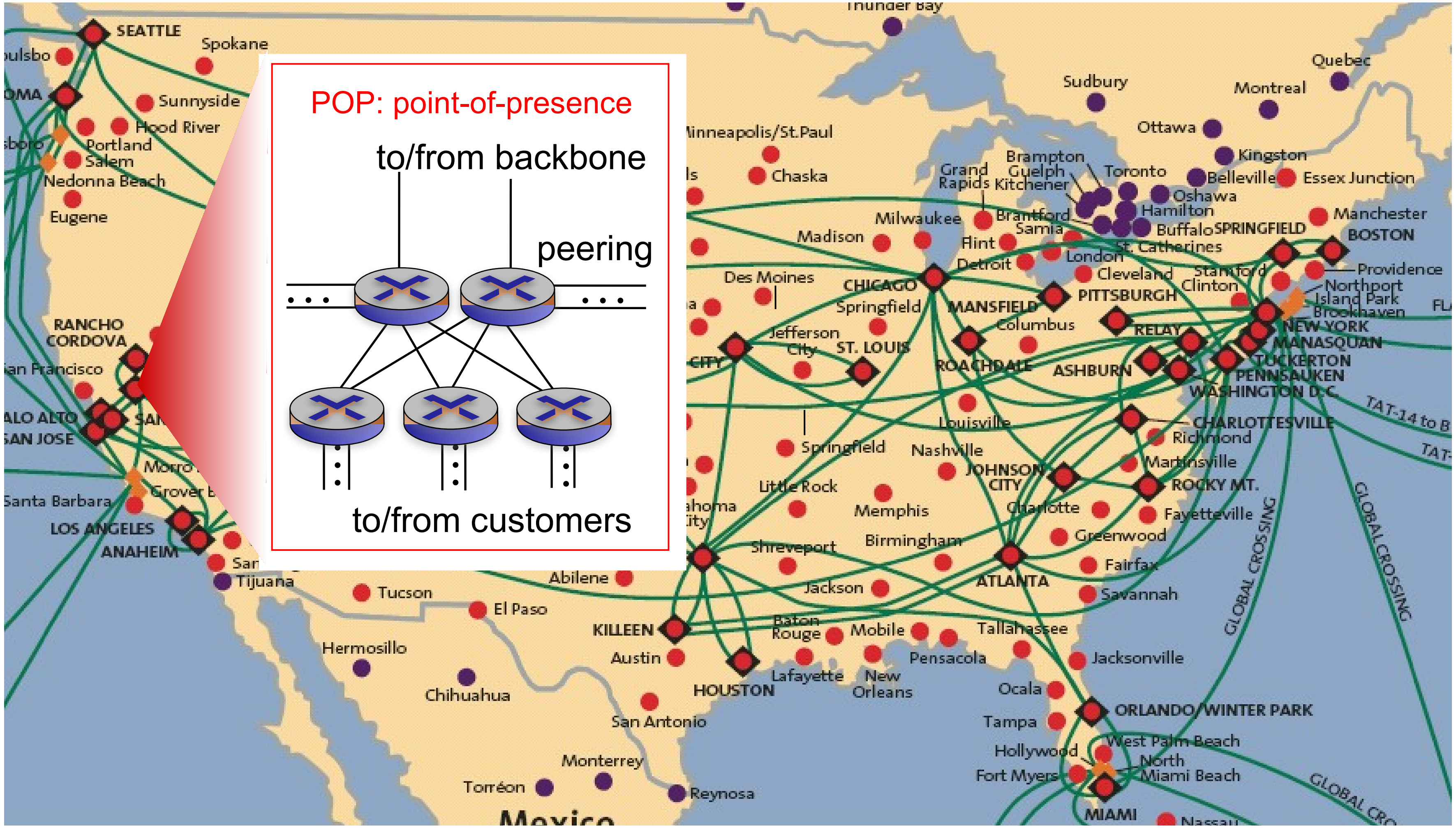
Internet structure: a "network of networks"



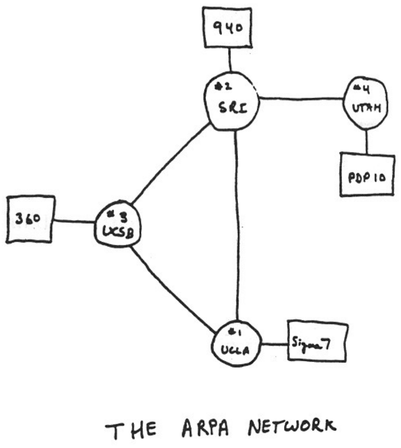
At "center": small # of well-connected large networks

- "tier-1" commercial ISPs (e.g., Level 3, Sprint, AT&T, NTT), national & international coverage
- content provider networks (e.g., Google, Facebook): private network that connects its data centers to Internet, often bypassing tier-1, regional ISPs

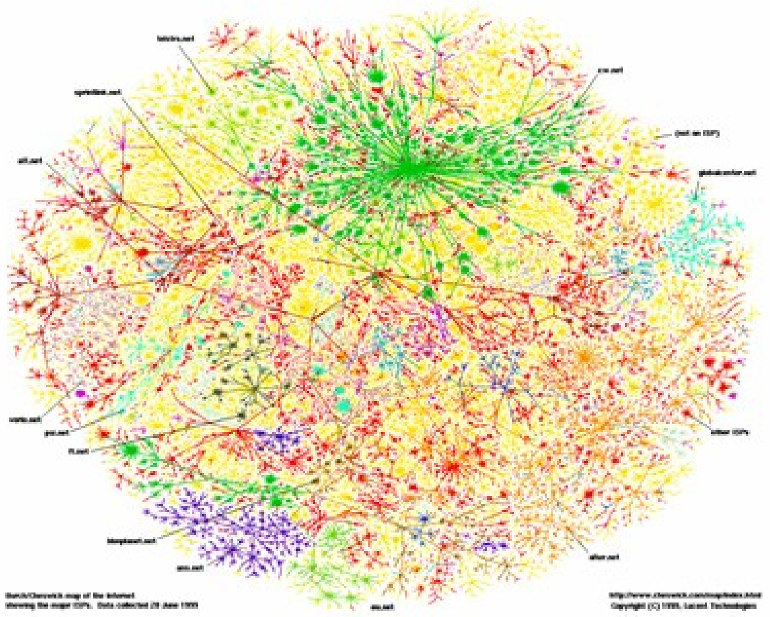
Tier-1 ISP: e.g., Sprint



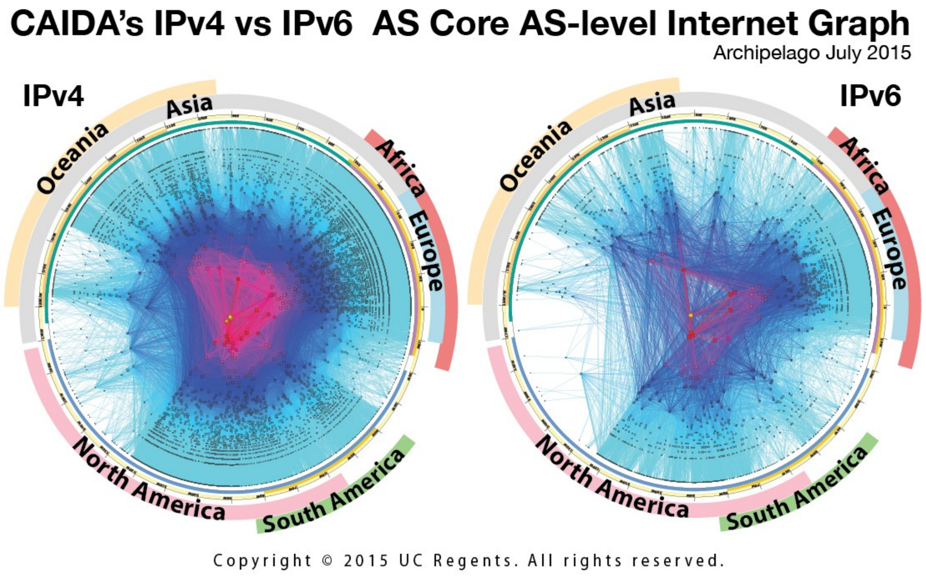
Can we map the Internet?



1972



1998 (Bell labs)



https://www.caida.org/projects/as-core/2015/

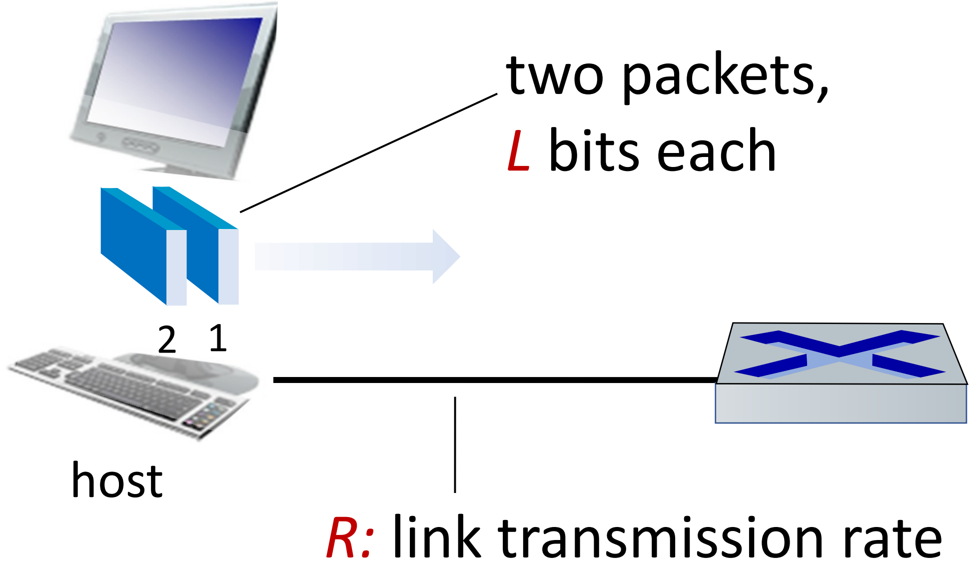
Chapter 1: roadmap

- What is the Internet?
- What is a protocol?
- Network edge: hosts, access network, physical media
- Network core: internet structure
- Performance: packet/circuit switching; loss, delay, throughput
- Protocol layers, service models
- History

Host: sends *packets* of data

host sending function:

- takes application message
- breaks into smaller chunks, known as *packets*, of length *L* bits
- transmits packet into access network at *transmission rate R*
  - link transmission rate, aka link *capacity, aka link bandwidth*



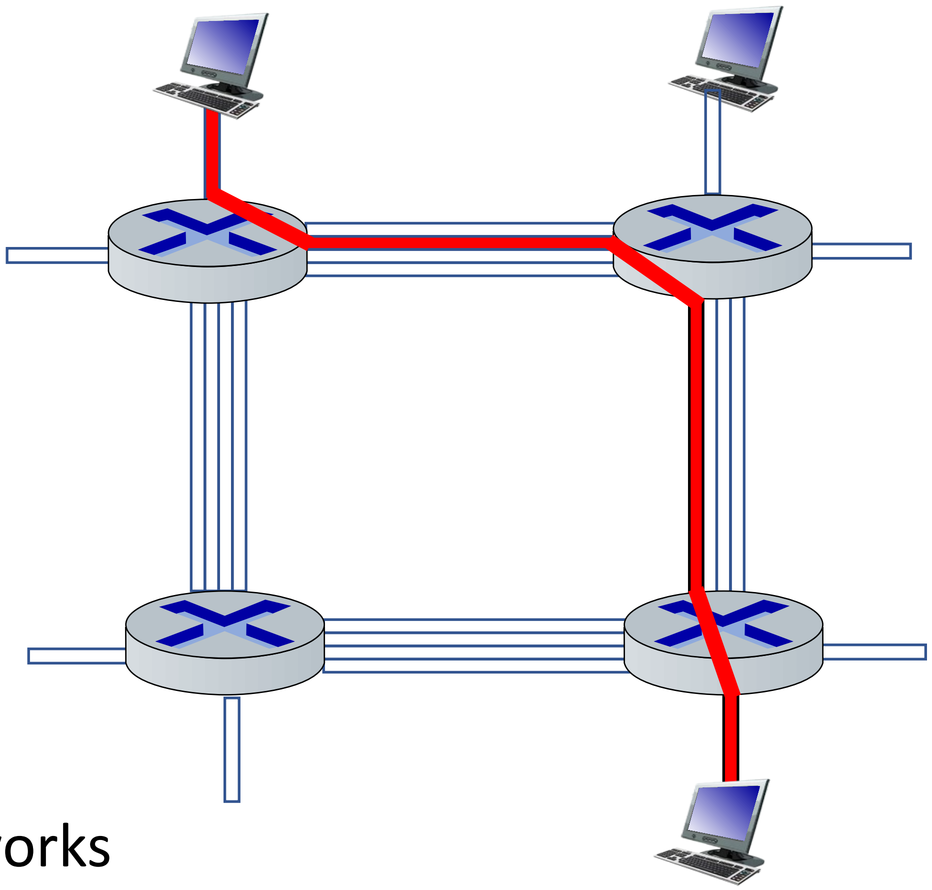
$$\text{packet transmission delay} = \text{time needed to transmit } L\text{-bit packet into link} = \frac{L \text{ (bits)}}{R \text{ (bits/sec)}}$$

Why packet switching?

Alternative to packet switching: circuit switching

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

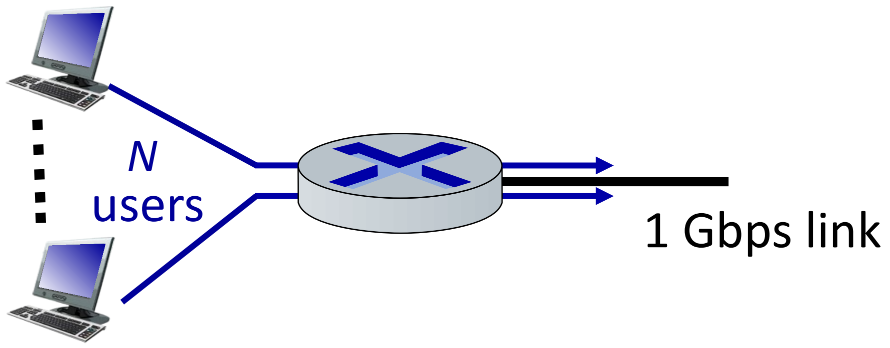
- in diagram, each link has four circuits.
  - call gets 2nd circuit in top link and 1st circuit in right link.
- dedicated resources: no sharing
  - circuit-like (guaranteed) performance
- circuit segment idle if not used by call (no sharing)
- commonly used in traditional telephone networks



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?

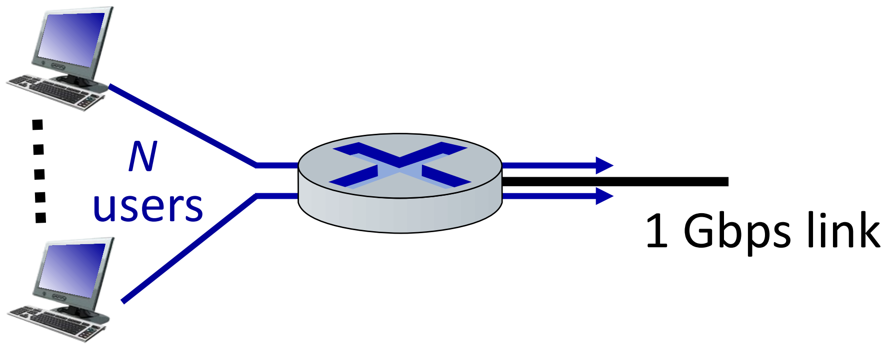
- *circuit-switching:* 10 users
- *packet switching:* with 35 users, probability that any 10 users are active at a time is 0.001

Q: human analogies of reserved resources (circuit switching) versus on-demand allocation (packet switching)?

Packet switching versus circuit switching

example:

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



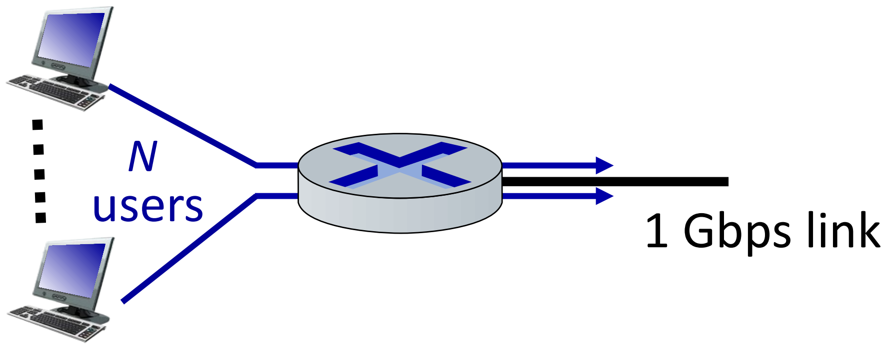
Q: What if there are more than 10 users active at a time?

- *circuit-switching:* well, only 10 users will be served
- *packet switching:* More users can send data but there will be queueing

Packet switching versus circuit switching

example:

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



Q: What if a user sends data at 500 Mb/s and everyone else is inactive? How much throughput it will get?

- *circuit-switching:* 100 Mbps
- *packet switching:* 500 Mbps

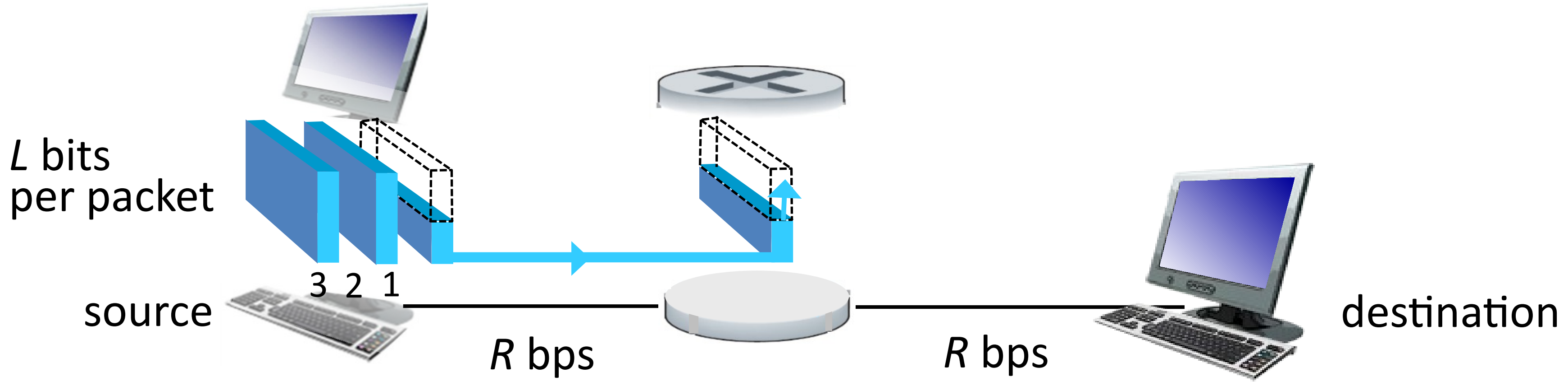
Packet switching versus circuit switching

Is packet switching a "slam dunk winner"?

- **great for "bursty" data – sometimes has data to send, but at other times not**
  - **Better resource sharing (scales better)**
  - **simpler, no call setup**

- excessive congestion possible: packet delay and loss due to buffer overflow
  - protocols needed for reliable data transfer, congestion control

Packet-switching: store-and-forward



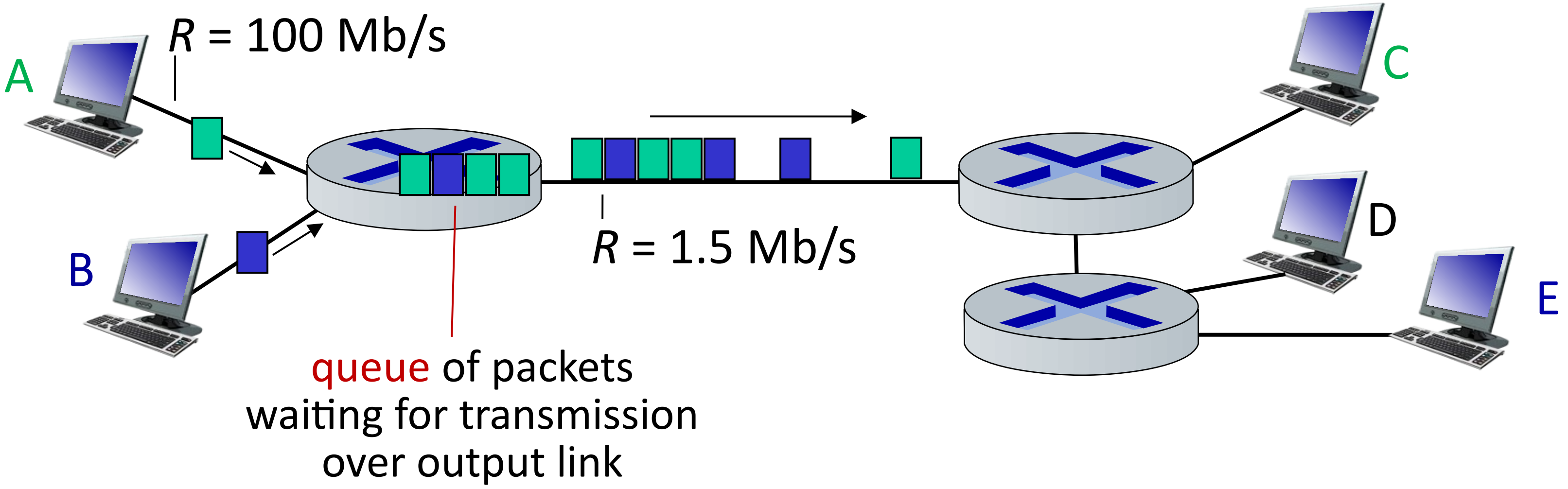
- packet transmission delay: takes L/R seconds to transmit (push out) L-bit packet into link at R bps
- *store and forward: entire* packet must arrive at router before it can be transmitted on next link

Why don't we forward every bit as we receive it?

One-hop numerical example:

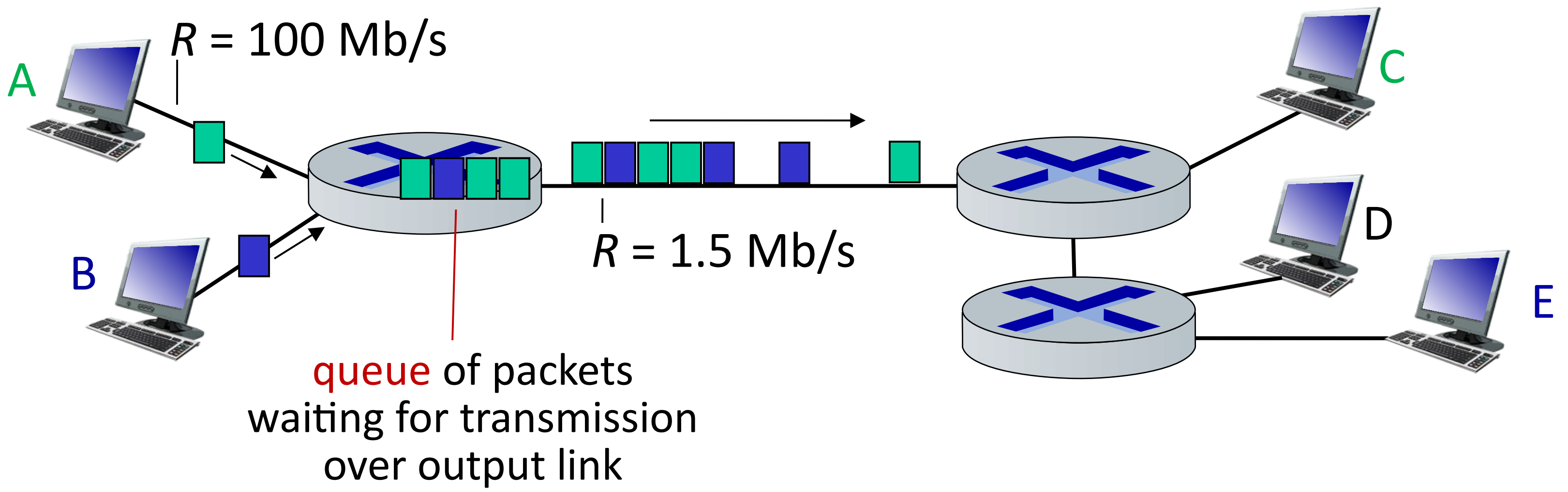
- L = 10 Kbits
- R = 100 Mbps
- one-hop transmission delay = 0.1 msec

Packet-switching: queueing



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

Packet-switching: queueing

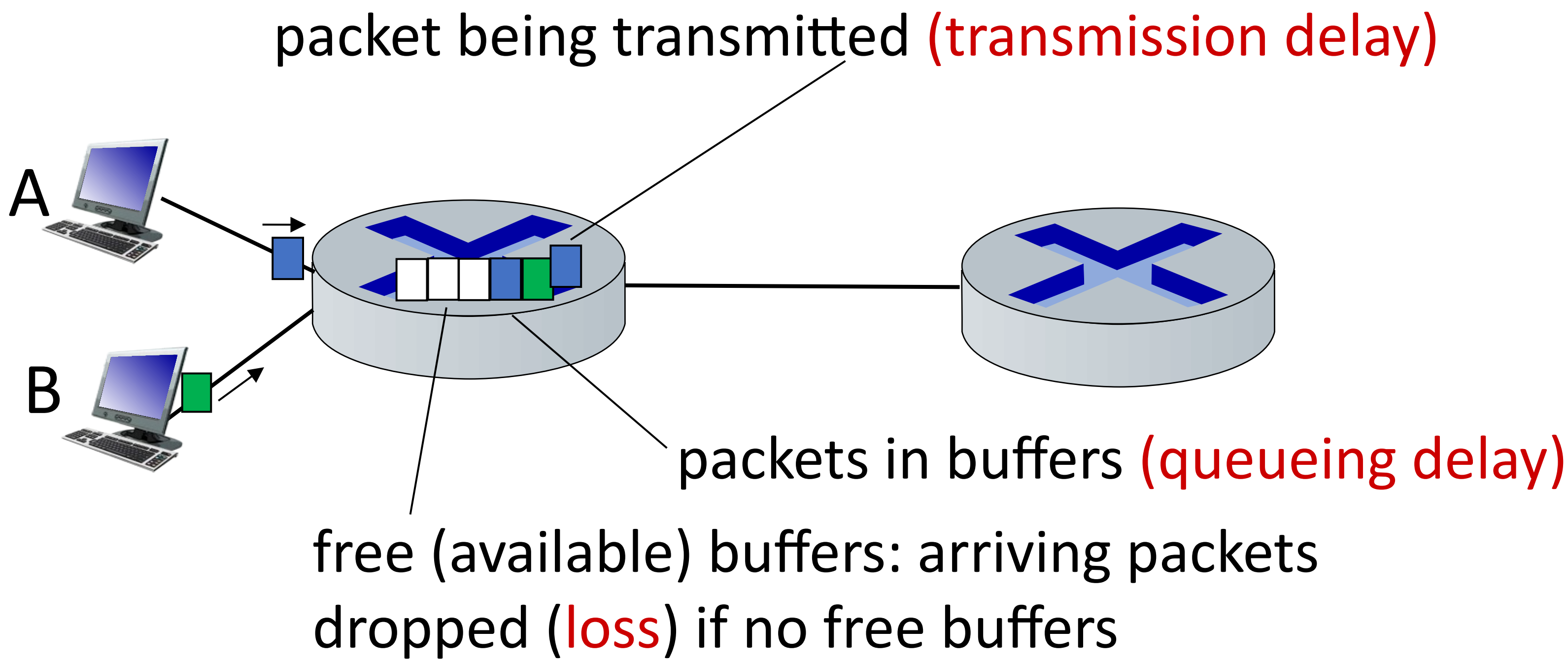


Packet queuing and loss: if arrival rate (in bps) to link exceeds transmission rate (bps) of link for some period of time:

- packets will queue, waiting to be transmitted on output link
- packets can be dropped (lost) if memory (buffer) in router fills up

How do packet delay and loss occur?

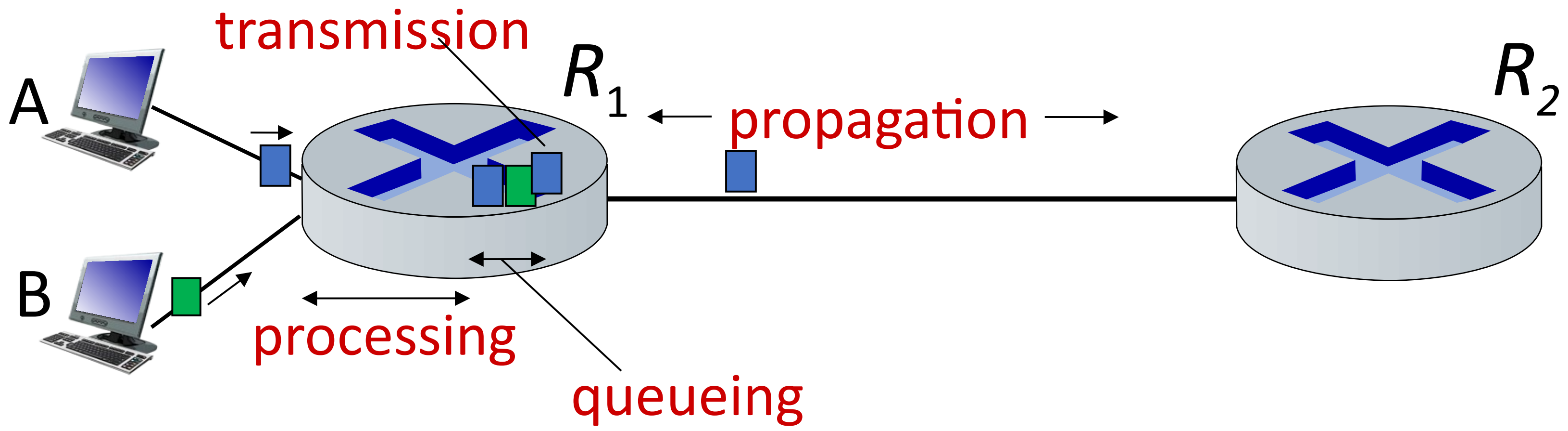
- packets *queue* in router buffers, waiting for turn for transmission
  - queue length grows when arrival rate to link (temporarily) exceeds output link capacity
- packet *loss* occurs when memory to hold queued packets fills up



Chapter 1: roadmap

- What is the Internet?
- What is a protocol?
- Network edge: hosts, access network, physical media
- Network core: internet structure
- Performance: packet/circuit switching; loss, delay, throughput
- Protocol layers, service models
- History

Packet delay: four sources

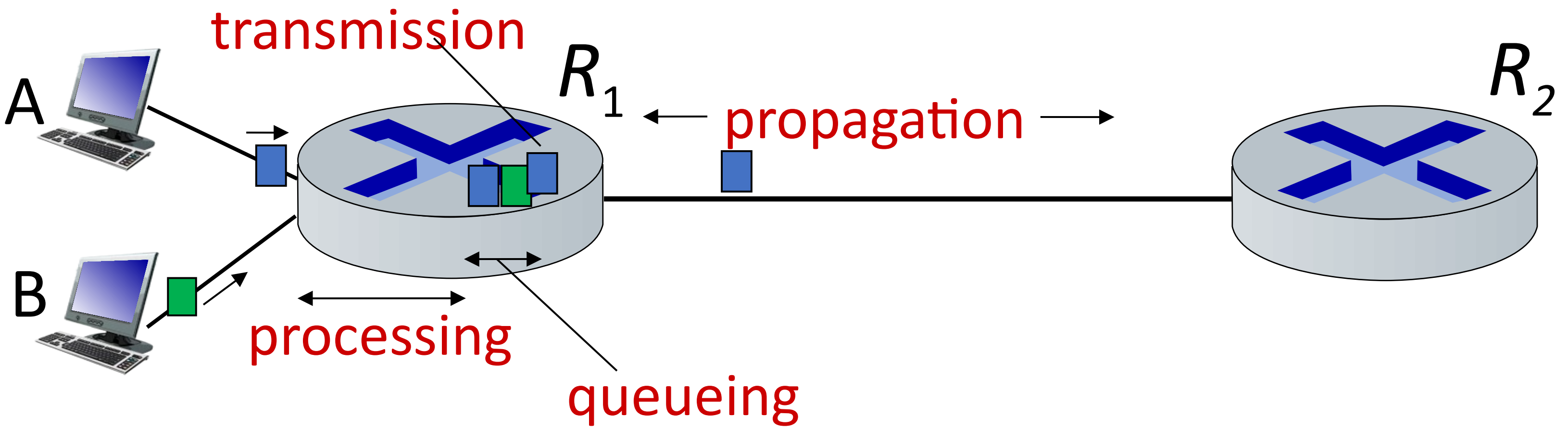


$$d_{nodal} = d_{proc} + d_{queue} + d_{trans} + d_{prop}$$

d_{proc}: processing delay
- check bit errors
- determine output link
- typically < microsecs

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

Packet delay: four sources



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

d_{trans}: transmission delay:

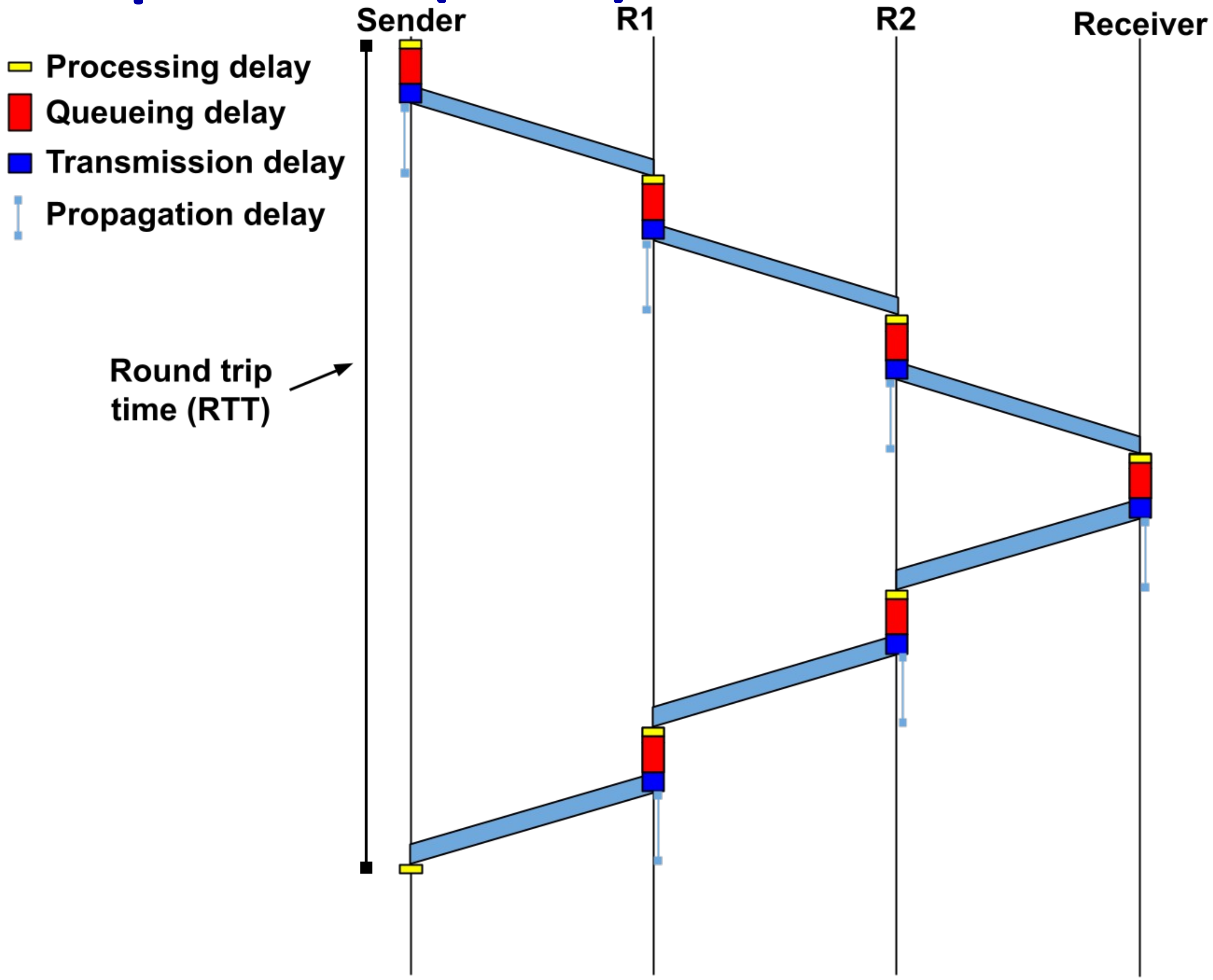
- L: packet length (bits)
- R: link *transmission rate (bps)*
- $d_{trans} = L/R$

d_{prop}: propagation delay:

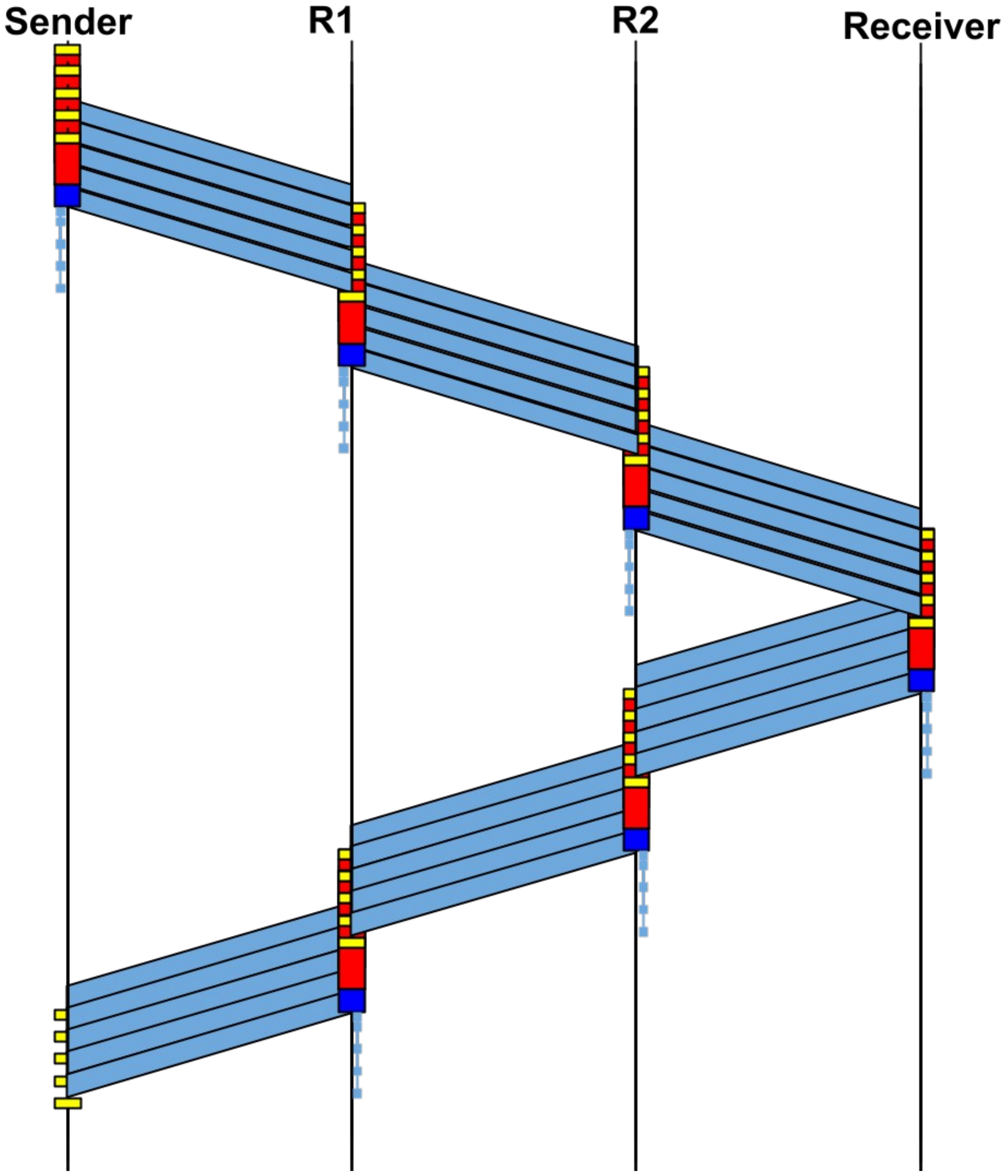
- d: length of physical link
- s: propagation speed (~$2\text{x}10^8$ m/sec)
- $d_{prop} = d/s$

d_{trans} and d_{prop}
very different

Round trip time (RTT)



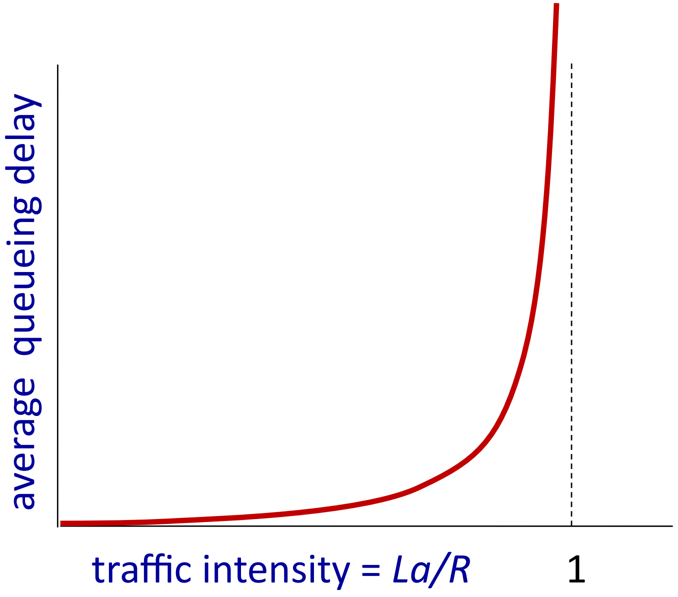
Don’t forget the parallelism



Packet queueing delay (revisited)

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

$$\frac{L \cdot a}{R} : \quad \frac{\text{arrival rate of bits}}{\text{service rate of bits}}$$ *"traffic intensity"*



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


La/R ~ 0


La/R -> 1

Packet queueing delay (revisited)

- *Arrival rate, transmission rate and queueing*

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

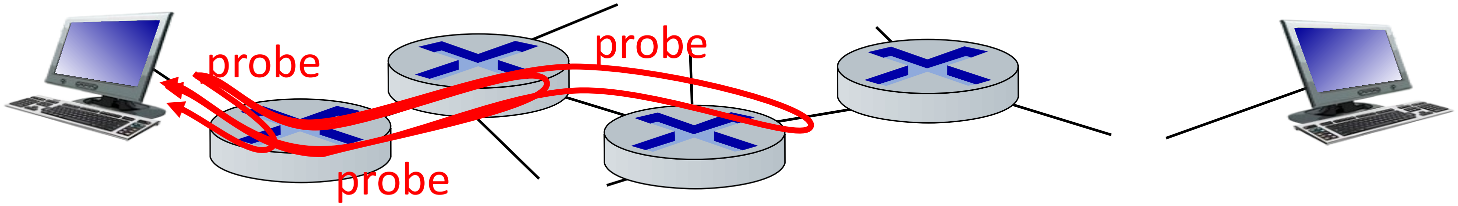
"traffic intensity"

https://www2.tkn.tu-berlin.de/teaching/rn/animations/queue/

Why can't we just use larger buffers?

“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, routes

traceroute: cs.gmu.edu to www.eurecom.fr

3 delay measurements

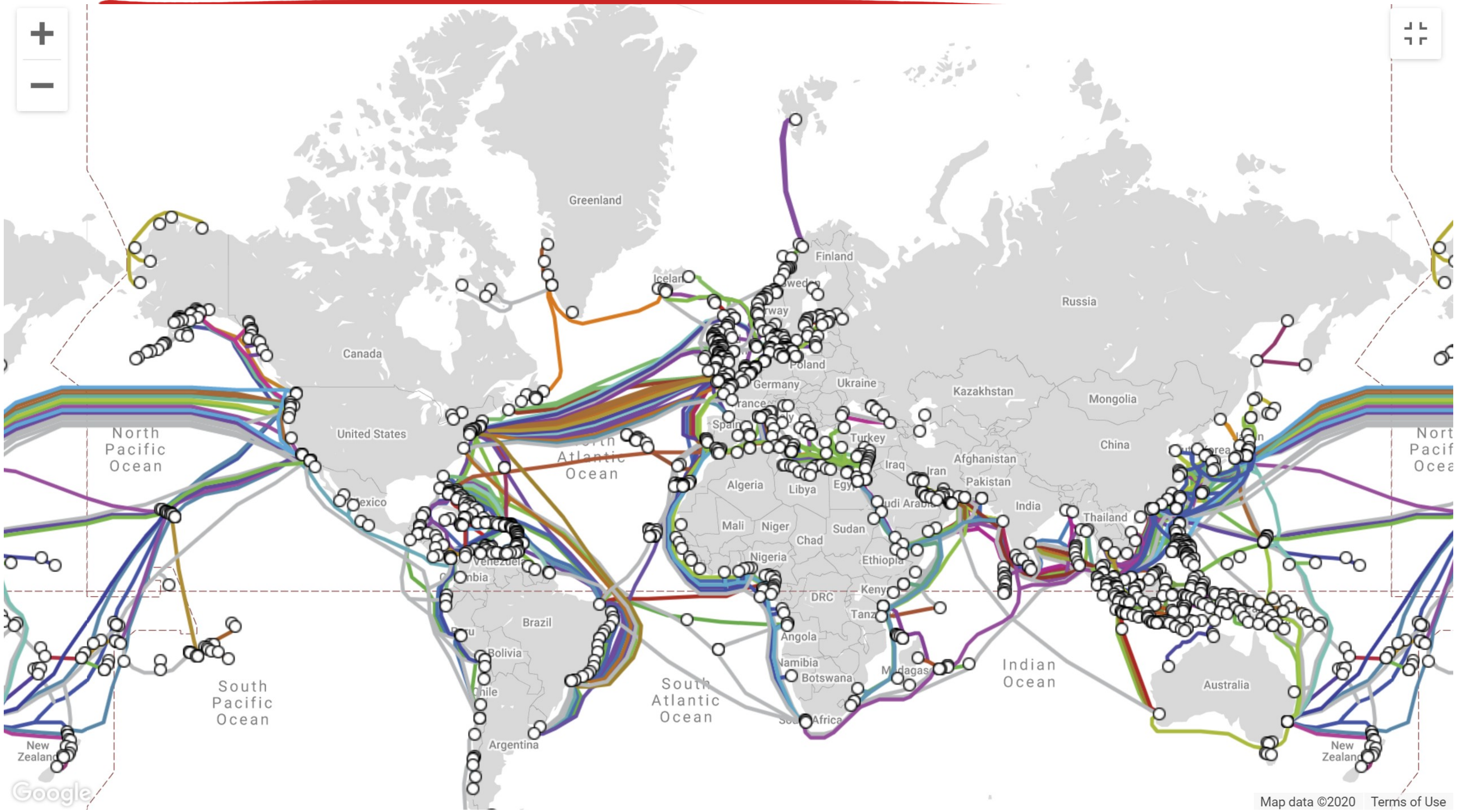
```
1  gateway (129.174.120.1)  2.219 ms  2.141 ms  2.068 ms
2  129.174.255.205 (129.174.255.205)  0.898 ms  0.851 ms  0.795 ms
3  129.174.255.206 (129.174.255.206)  1.287 ms  1.249 ms  1.157 ms
4  10.255.254.253 (10.255.254.253)  2.371 ms  2.479 ms  2.390 ms
5  10.255.252.57 (10.255.252.57)  2.751 ms  2.838 ms  2.926 ms
6  10.255.254.237 (10.255.254.237)  2.587 ms  2.477 ms  2.879 ms
7  192.122.175.80 (192.122.175.80)  2.791 ms  2.654 ms  2.873 ms
8  192.122.175.15 (192.122.175.15)  2.453 ms  2.557 ms  2.541 ms
9  ae-2.4079.rtsw.wash.net.internet2.edu (162.252.70.136)  2.974 ms  3.071 ms  2.962 ms
10  internet2.mx1.lon.uk.geant.net (62.40.124.44)  78.330 ms  78.355 ms  78.329 ms
11  ae6.mx1.lon2.uk.geant.net (62.40.98.37)  79.054 ms  79.041 ms  79.000 ms
12  ae0.mx1.par.fr.geant.net (62.40.98.77)  85.588 ms  85.561 ms  85.501 ms
13  renater-lb1-gw.mx1.par.fr.geant.net (62.40.124.70)  85.808 ms  85.826 ms  86.114 ms
14  193.51.180.63 (193.51.180.63)  99.436 ms te0-3-1-0-lyon1-rtr-001.noc.renater.fr (193.51.177.65)  99.497 ms  99.627 ms
15  193.51.180.11 (193.51.180.11)  97.378 ms xe1-0-1-marseille1-rtr-131.noc.renater.fr (193.51.177.222)  124.059 ms  124.012 ms
16  te1-2-sophia-rtr-021.noc.renater.fr (193.51.177.21)  99.327 ms  99.462 ms  99.350 ms
17  eurocom-valbonne-gi9-7-sophia-rtr-021.noc.renater.fr (193.51.187.17)  99.235 ms  99.293 ms  99.240 ms
18  * * *
19  www.eurecom.fr (193.55.113.240)  100.884 ms  100.184 ms  100.324 ms
```

At which hop, did we cross the Atlantic?

* means no response (probe lost, router not replying)

*Do some traceroutes from exotic countries at www.traceroute.org

Submarine Cable Map

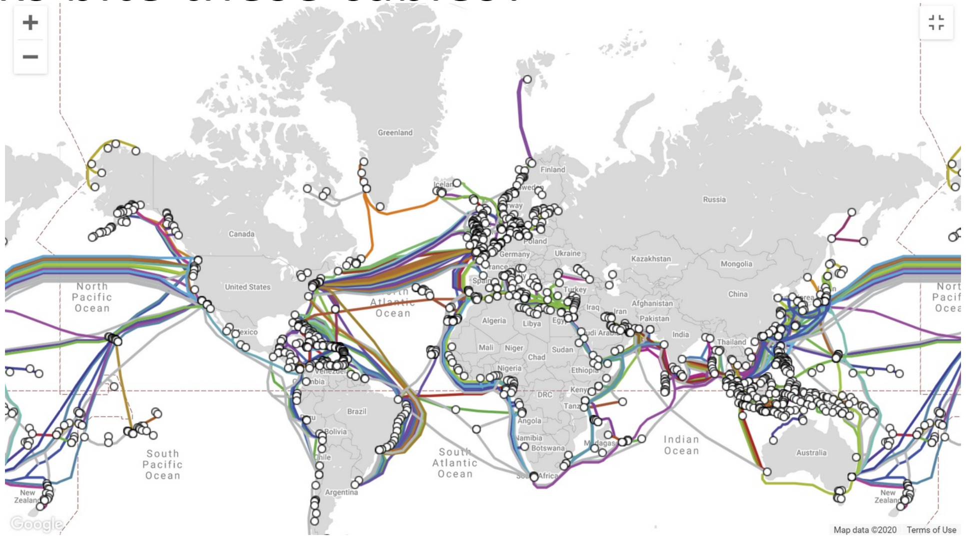


https://www.submarinecablemap.com/

Wait, I have questions..

- How long are these undersea cables?
- Are they actually at the bottom of the sea?
- How thick are the undersea cables?
- Why don't we use satellites instead?
- Most importantly..
  - Can sharks bite these cables?

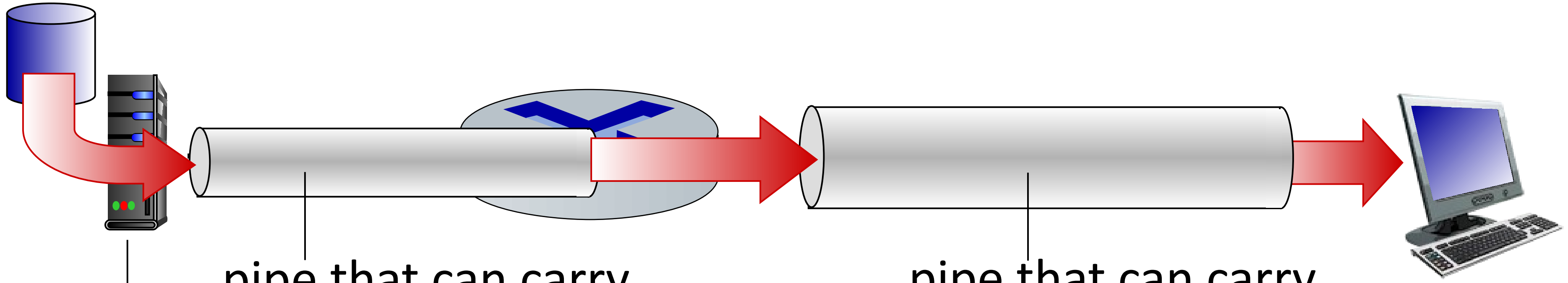
https://www2.telegeography.com/submarine-cable-faqs-frequently-asked-questions



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

Transmission rate vs. throughput?



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)

In-class exercise

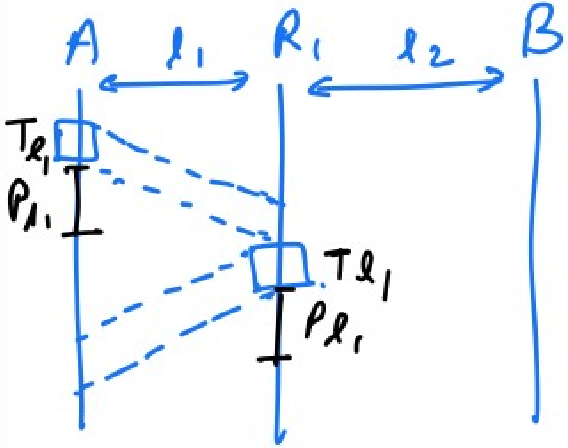
Take a look at the following (fake) traceroute data. For simplicity, let's assume that queueing delay and processing delay at all nodes is 0. The network looks like

```
A------------------R1----------------B
       1000km             2000km
```

```
1    R1   12ms 12ms 12ms
2    B    36ms 36ms 36ms
```

Assuming propagation speed on both links is 2.5x10^8 m/s and the length of the traceroute packets (requests and responses) is 50 bytes, what is the transmission rate of both links (A-R1, R1-B)? Remember the delay values reported by traceroute for each hop are round-trip times.

In-class exercise

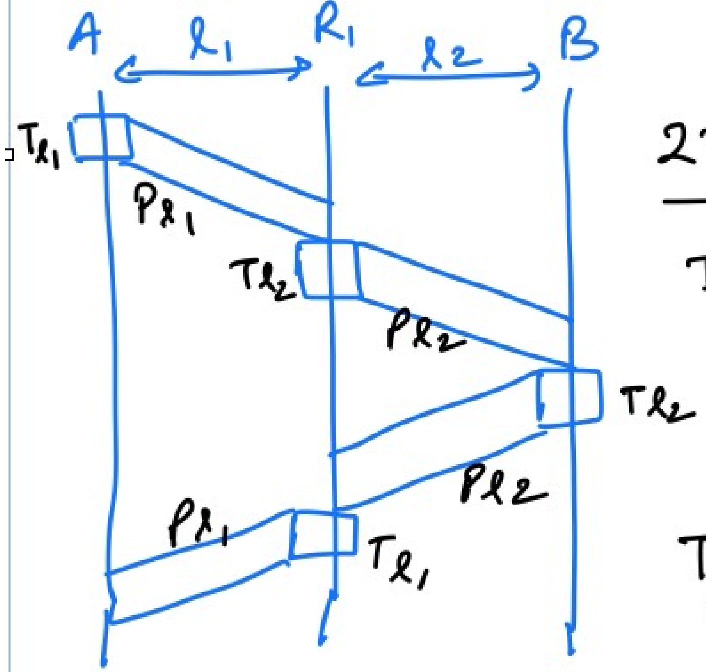


$2T_{l_1} + 2P_{l_1} = 12\text{ ms}$

$P_{l_1} = \frac{1000\text{ km}}{2.5\times10^8\text{ m/s}} = 4\text{ ms}$

$T_{l_1} = 2\text{ ms} = \frac{L}{R_{l_1}} = \frac{50\times8}{R_{l_1}}$

$\boxed{R_{l_1} = 200\text{ kbps}}$



$2T_{l_1} + 2P_{l_1} + 2T_{l_2} + 2P_{l_2} = 36\text{ ms}$

$T_{l_2} + P_{l_2} = 12\text{ ms}$

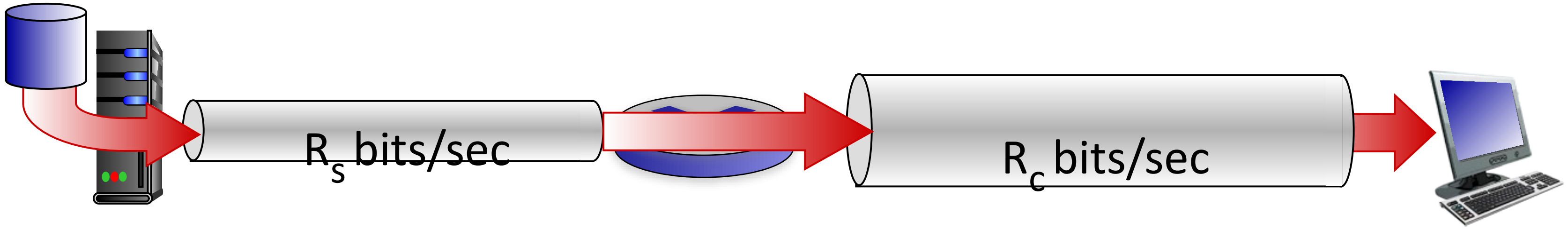
$P_{l_2} = \frac{2000\text{ km}}{2.5\times10^8\text{ m/s}} = 8\text{ ms}$

$T_{l_2} = 4\text{ ms} = \frac{L}{R_{l_2}} = \frac{50\times8}{R_{l_2}}$

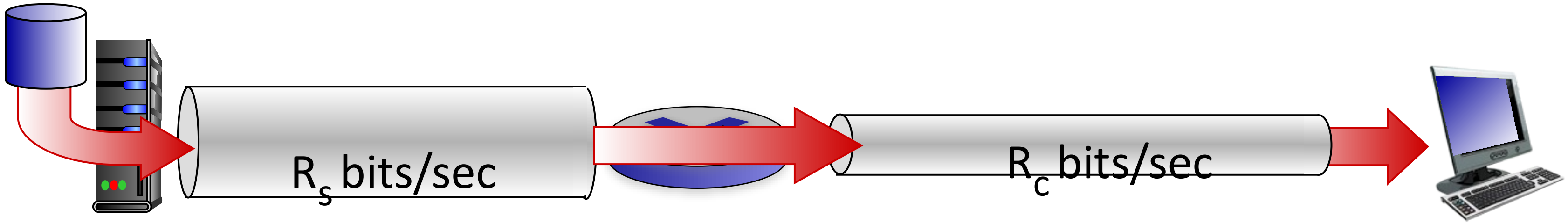
$\boxed{R_{l_2} = 100\text{ kbps}}$

Throughput

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



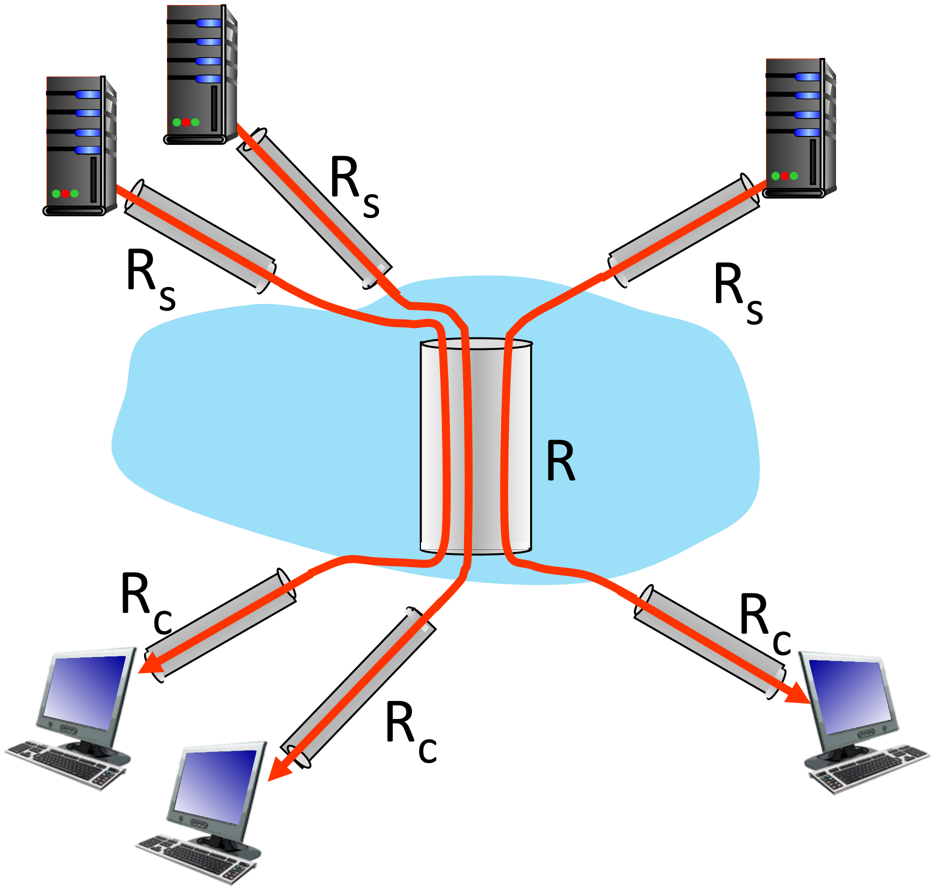
$R_s > R_c$ What is average end-end throughput?



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

Chapter 1: roadmap

- What is the Internet?
- What is a protocol?
- Network edge: hosts, access network, physical media
- Network core: internet structure
- Performance: packet/circuit switching; loss, delay, throughput
- Protocol layers, service models
- History

Protocol "layers" and reference models

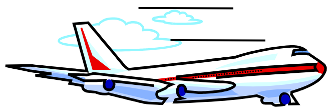
Networks are complex, with many "pieces":

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

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

- and/or our *discussion* of networks?

Example: organization of air travel



end-to-end transfer of person plus baggage

ticket (purchase)	ticket (complain)
baggage (check)	baggage (claim)
gates (load)	gates (unload)
runway takeoff	runway landing
airplane routing	airplane routing

airplane routing

How would you *define/discuss* the *system* of airline travel?

- a series of steps, involving many services

Example: organization of air travel

layers: each layer implements a service

- via its own internal-layer actions
- relying on services provided by layer below

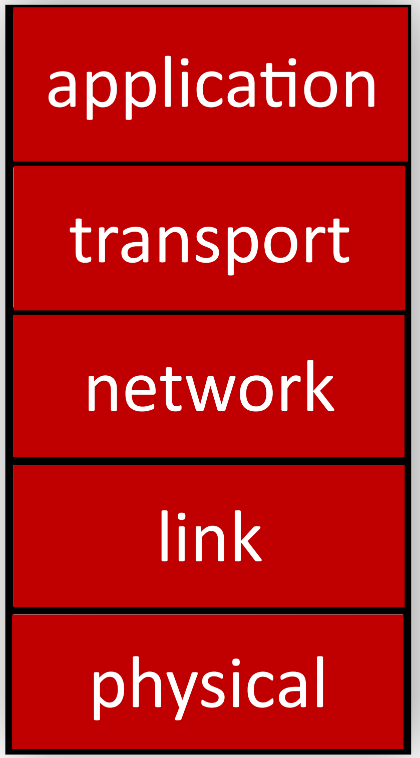
Why layering?

Approach to designing/discussing complex systems:

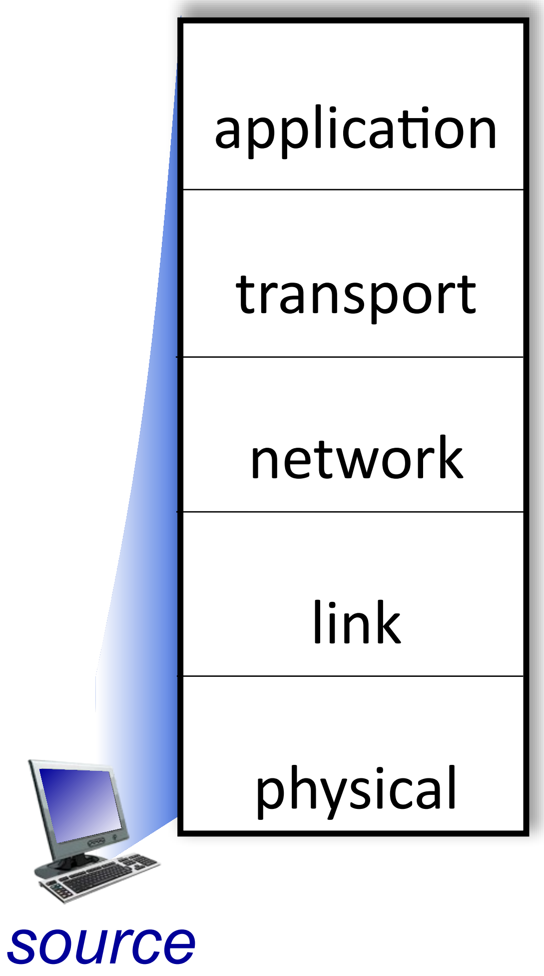
- explicit structure allows identification, relationship of system's pieces
  - layered *reference model* for discussion
- modularization eases maintenance, updating of system
  - change in layer's service *implementation*: transparent to rest of system
  - e.g., change in gate procedure doesn't affect rest of system

Layered Internet protocol stack

- *application:* supporting network applications
  - HTTP, IMAP, SMTP, DNS
- *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, 802.11 (WiFi), PPP
- *physical:* bits “on the wire”



Services, Layering and Encapsulation



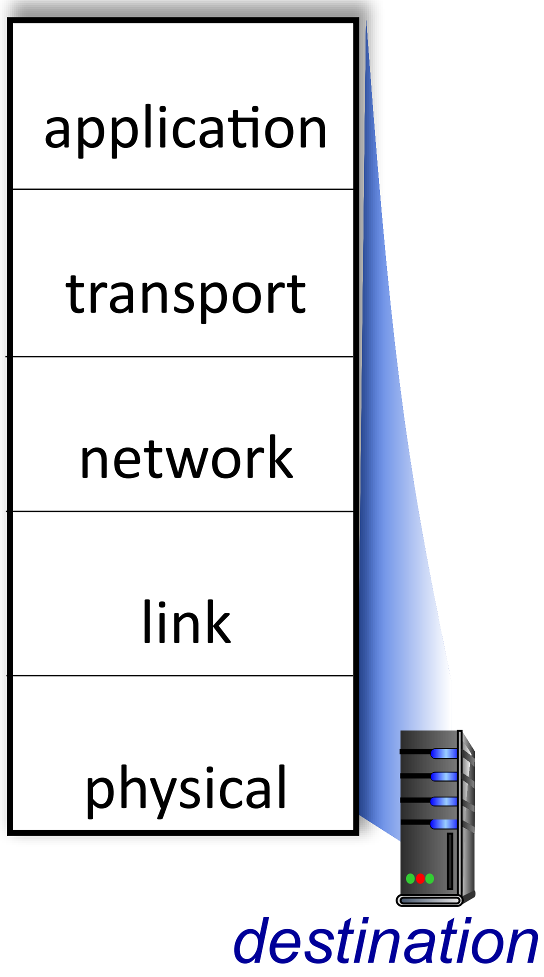
M

Application exchanges messages to implement some application service using *services* of transport layer

H_t M

Transport-layer protocol transfers M (e.g., reliably) from one *process* to another, using services of network layer

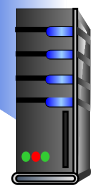
- transport-layer protocol encapsulates application-layer message, M, with *transport* layer-layer header H_t to create a transport-layer segment
  - H_t used by transport layer protocol to implement its service



Services, Layering and Encapsulation

source		destination
application	M	application
transport	H_t | M	transport
network	H_n | H_t | M	network
link		link
physical		physical

source



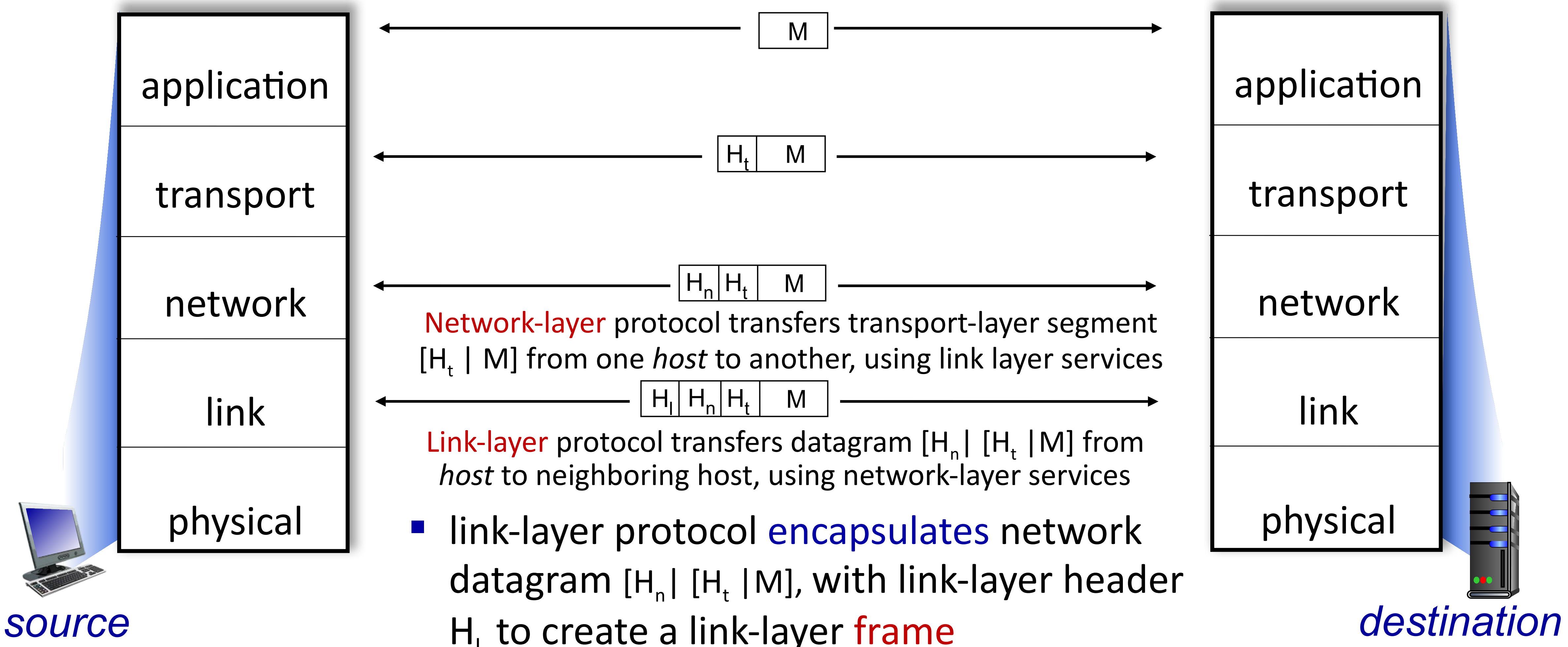
destination

Transport-layer protocol transfers M (e.g., reliably) from one *process* to another, using services of network layer

Network-layer protocol transfers transport-layer segment [H_t | M] from one *host* to another, using link layer services

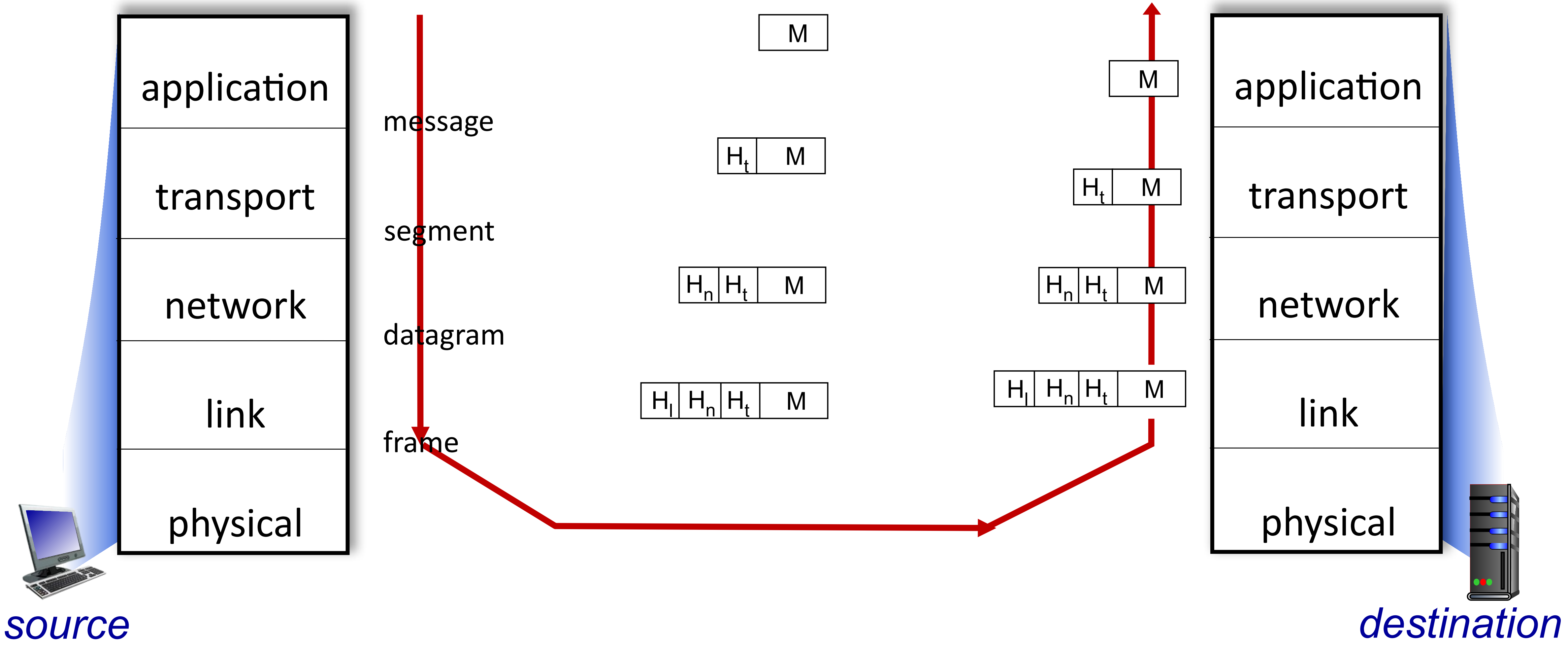
- network-layer protocol encapsulates transport-layer segment [H_t | M] with network layer-layer header H_n to create a network-layer datagram
  - H_n used by network layer protocol to implement its service

Services, Layering and Encapsulation

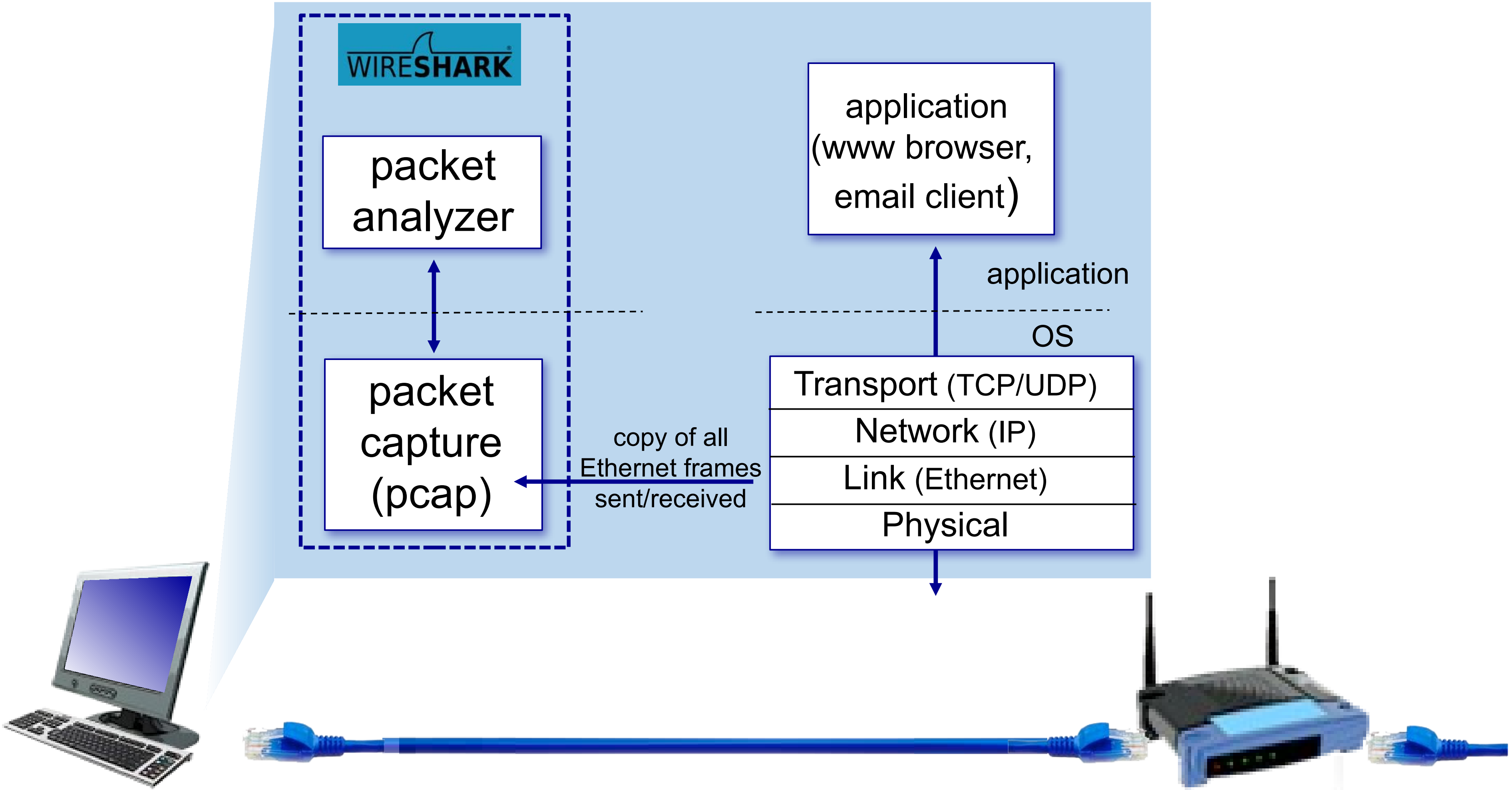


- link-layer protocol encapsulates network datagram [H_n| [H_t |M], with link-layer header H_l to create a link-layer frame

Services, Layering and Encapsulation

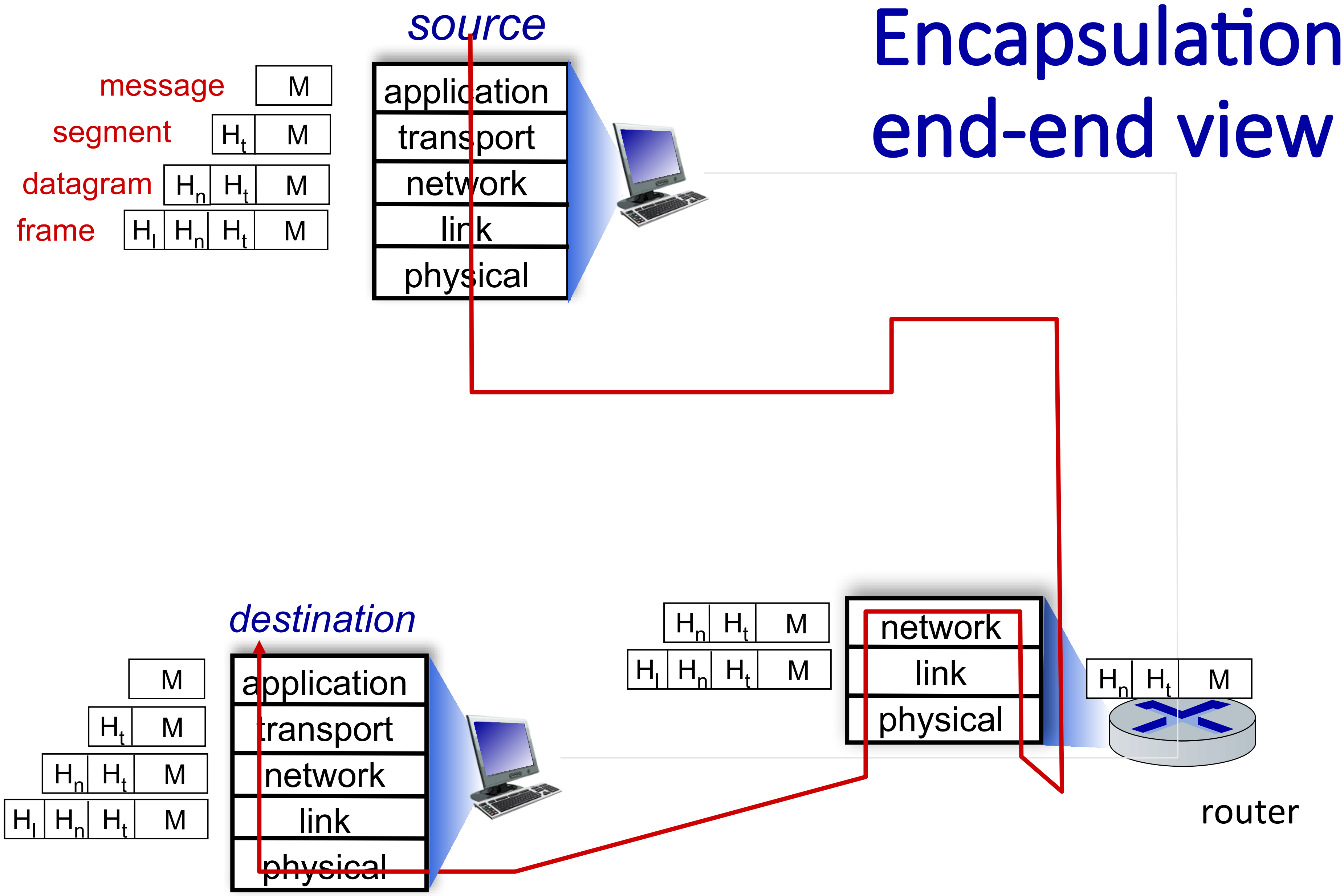


Wireshark

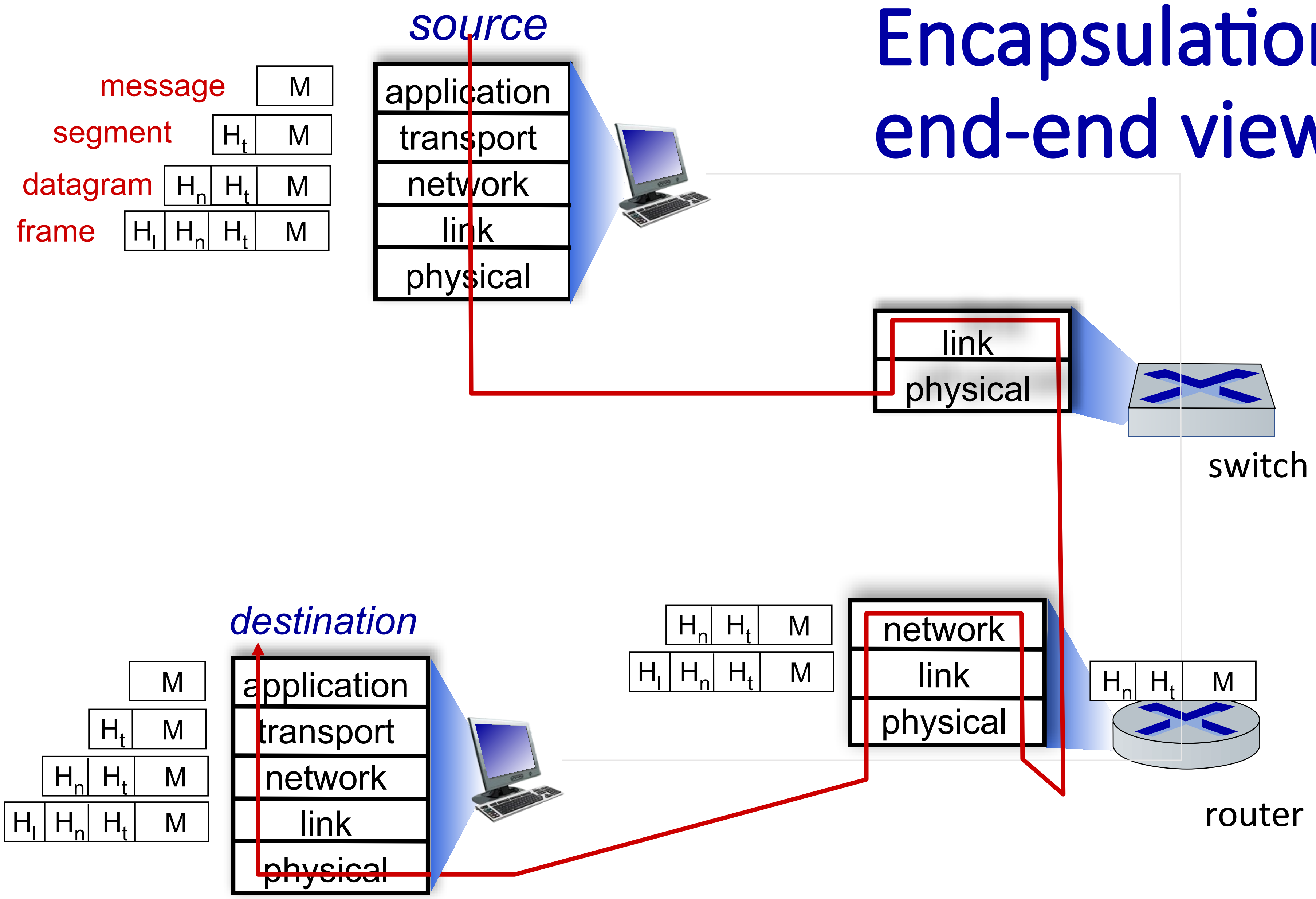


WIRESHARK
packet analyzer
packet capture (pcap)
copy of all Ethernet frames sent/received
application (www browser, email client)
application
OS
Transport (TCP/UDP)
Network (IP)
Link (Ethernet)
Physical

Encapsulation: an end-end view



Encapsulation: an end-end view



Chapter 1: roadmap

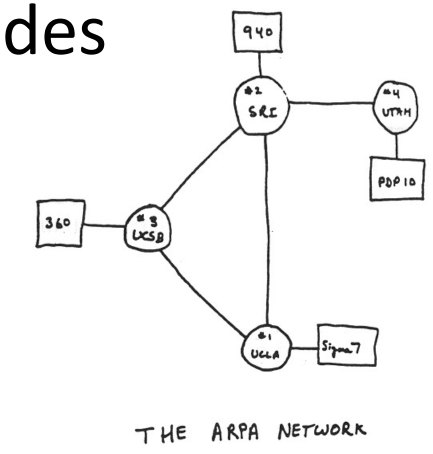
- What *is* the Internet?
- What *is* a protocol?
- Network edge: hosts, access network, physical media
- Network core: packet/circuit switching, internet structure
- Performance: loss, delay, throughput
- Security
- Protocol layers, service models
- History

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



Internet history

1972-1980: Internetworking, new and proprietary networks

- 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
- 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 routing
- decentralized control

define today's Internet architecture

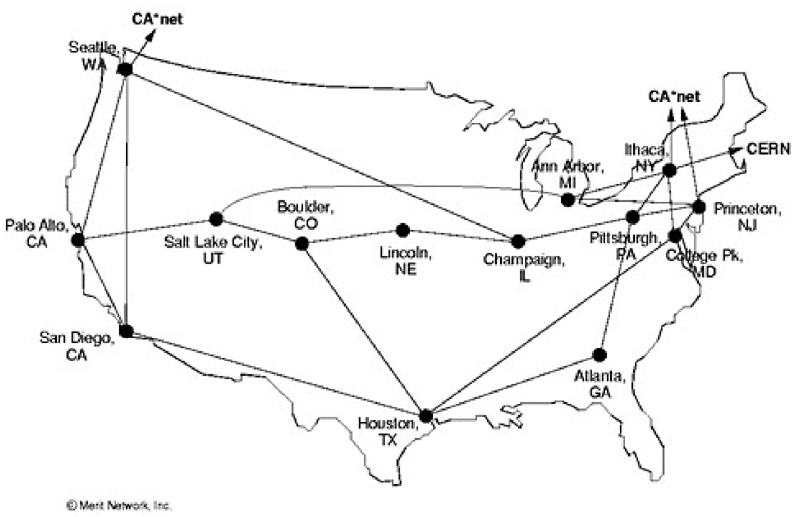
Internet history

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

NSFNET T1 Network 1991



Internet history

1990, 2000s: commercialization, the Web, new applications

- early 1990s: 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 1990s: commercialization of the Web

late 1990s – 2000s:

- 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: scale, SDN, mobility, cloud

- aggressive deployment of broadband home access (10-100's Mbps)
- 2008: software-defined networking (SDN)
- increasing ubiquity of high-speed wireless access: 4G/5G, WiFi
- service providers (Google, FB, Microsoft) create their own networks
  - bypass commercial Internet to connect "close" to end user, providing "instantaneous" access to social media, search, video content, …
- enterprises run their services in "cloud" (e.g., Amazon Web Services, Microsoft Azure)
- rise of smartphones: more mobile than fixed devices on Internet (2017)
- ~18B devices attached to Internet (2017)

Chapter 1: summary

We've covered a "ton" of material!

- Internet overview
- what's a protocol?
- network edge, access network, core
  - packet-switching versus circuit-switching
  - Internet structure
- performance: loss, delay, throughput
- layering, service models
- security
- history

You now have:

- context, overview, vocabulary, "feel" of networking
- more depth, detail, *and fun* to follow*!*