

Chapter 1

Introduction

A note on the use of these PowerPoint slides:

We're making these slides freely available to all (faculty, students, readers). They're in PowerPoint form so you see the animations; and can add, modify, and delete slides (including this one) and slide content to suit your needs. They obviously represent a *lot* of work on our part.

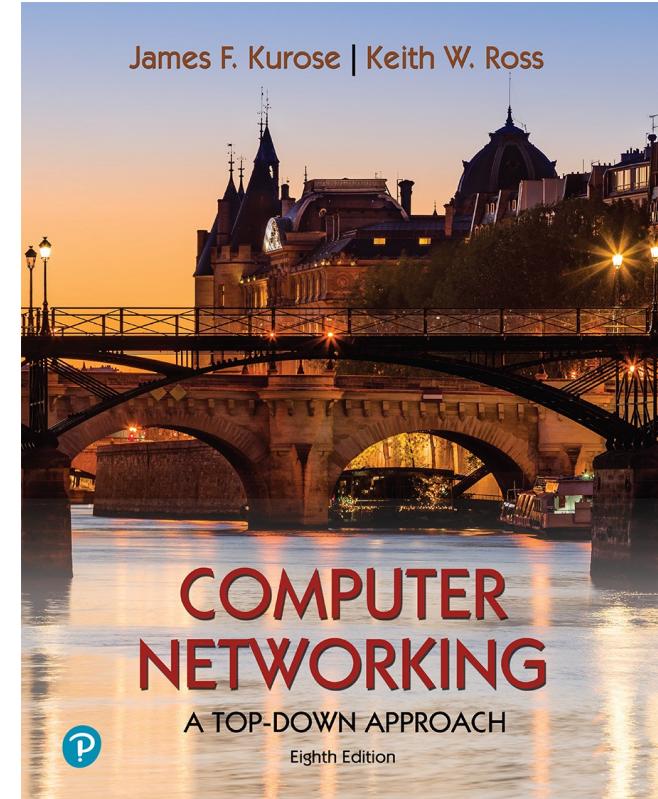
In return for use, we only ask the following:

- If you use these slides (e.g., in a class) that you mention their source (after all, we'd like people to use our book!)
- If you post any slides on a www site, that you note that they are adapted from (or perhaps identical to) our slides, and note our copyright of this material.

For a revision history, see the slide note for this page.

Thanks and enjoy! JFK/KWR

All material copyright 1996-2020
J.F Kurose and K.W. Ross, All Rights Reserved



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

Chapter 1: introduction

Chapter goal:

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



Overview/roadmap:

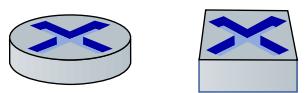
- What is the Internet? What is a protocol?
- Network edge: hosts, access network, physical media
- Network core: packet/circuit switching, internet structure
- Performance: loss, delay, throughput
- Protocol layers, service models
- Security
- History

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

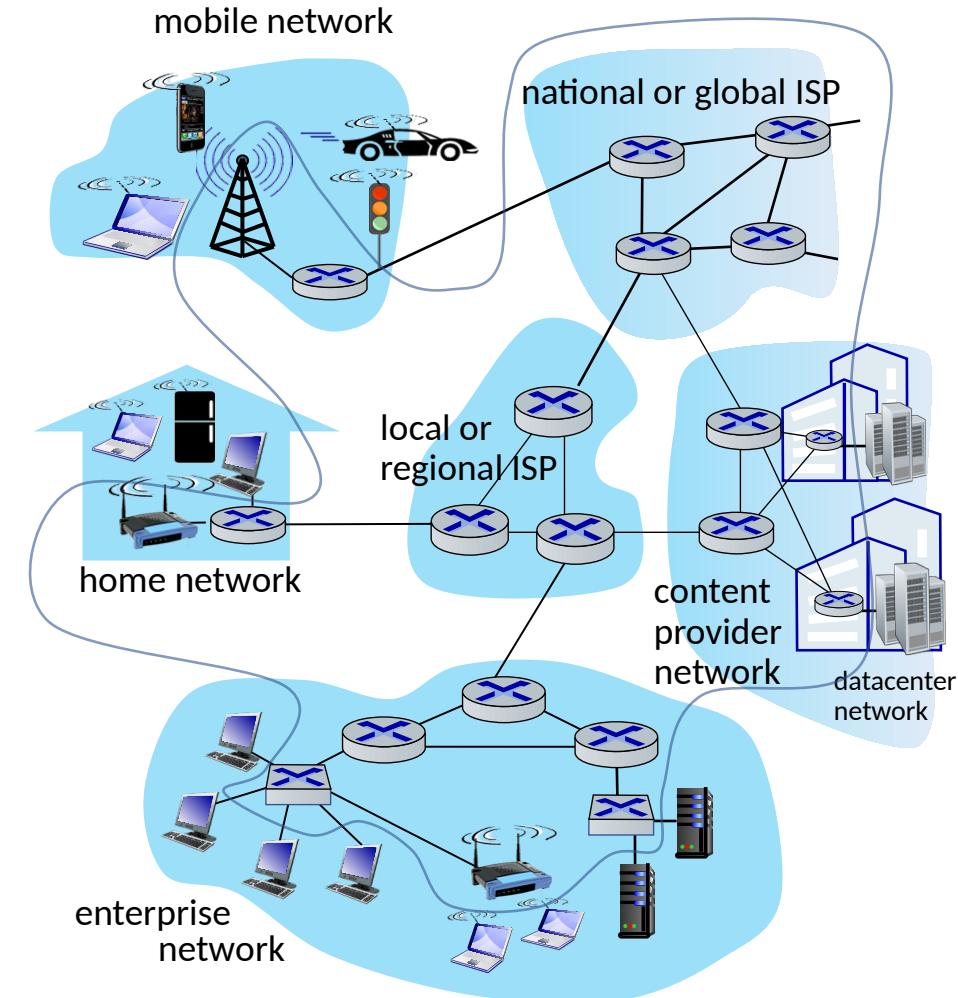


Communication links

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

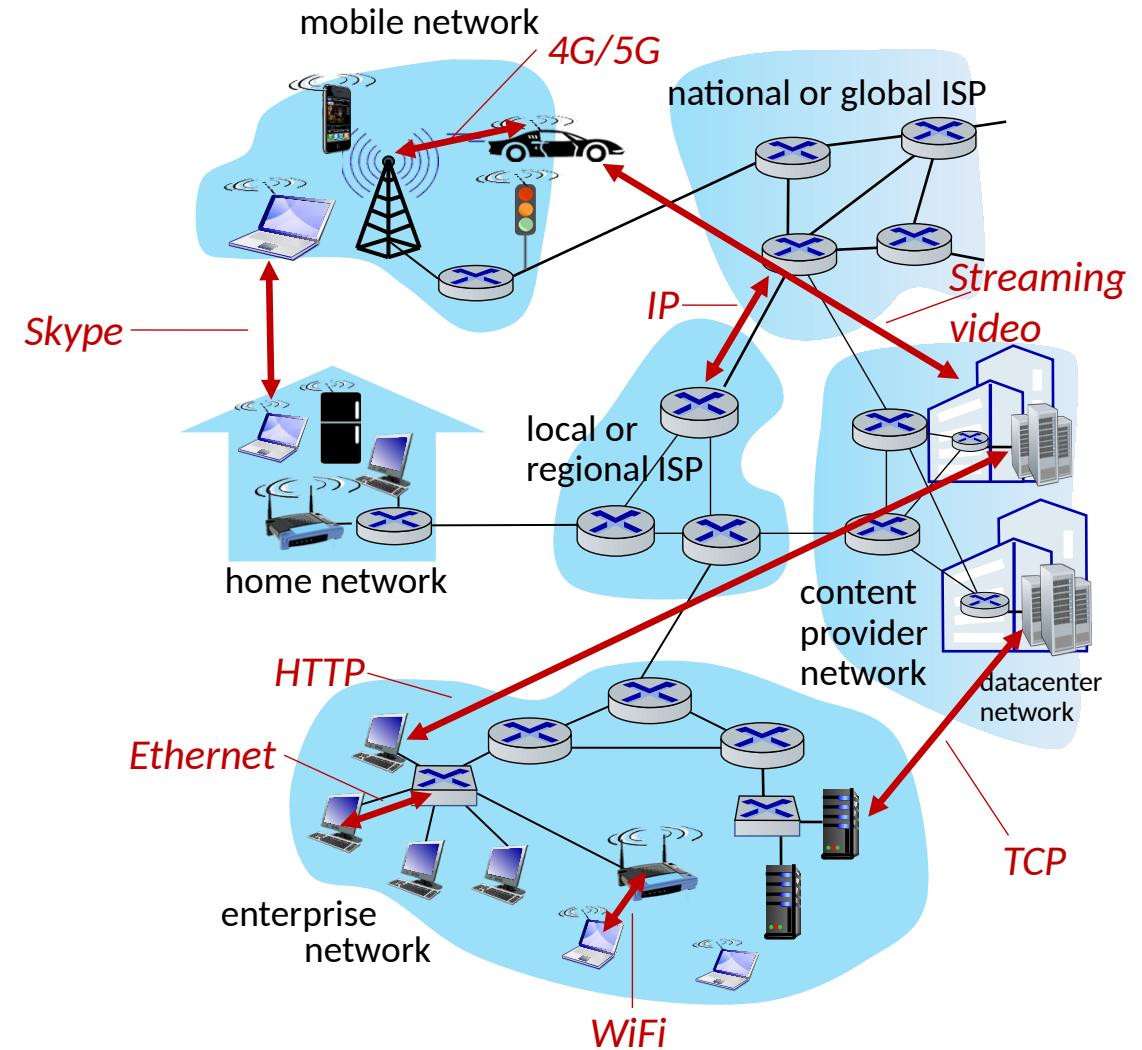
Networks

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



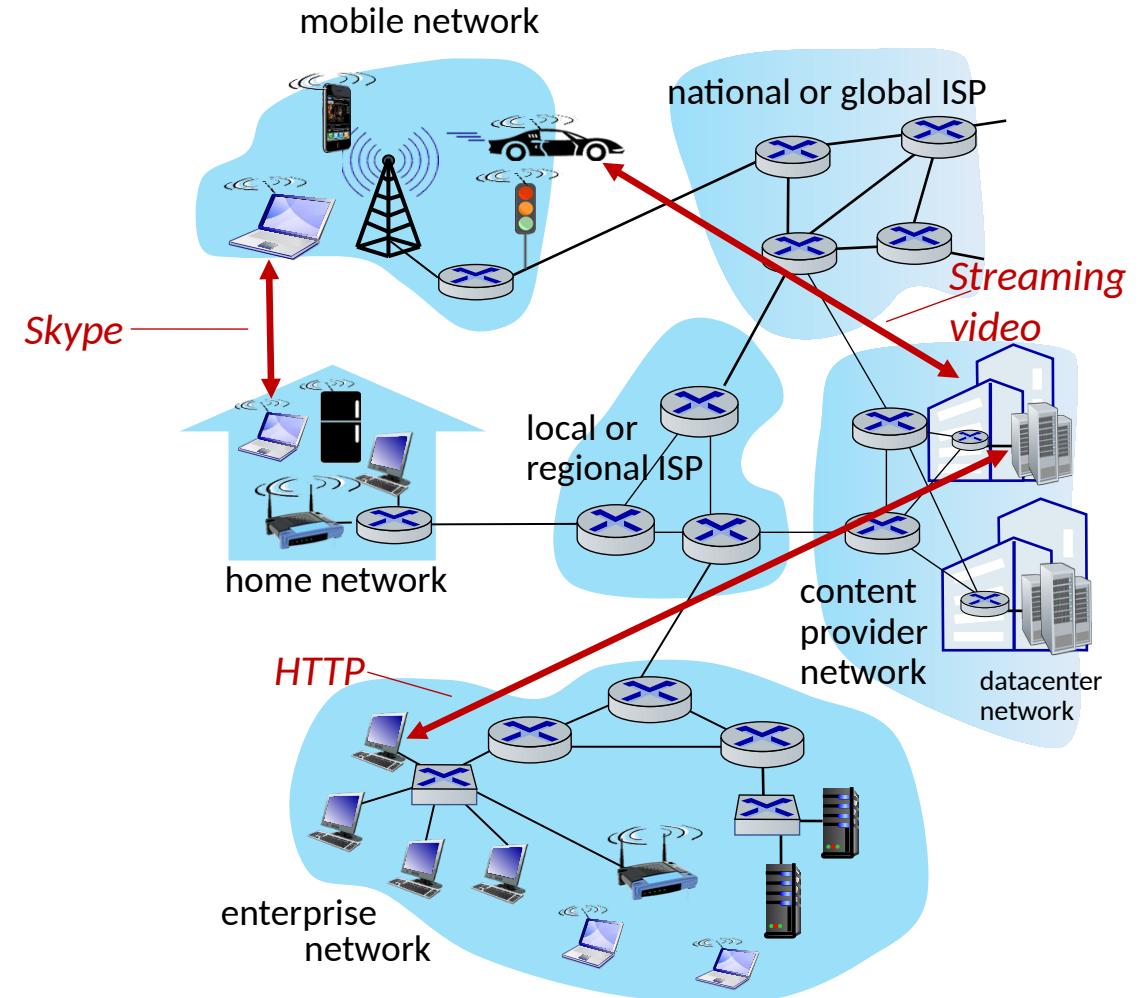
The Internet: a “nuts and bolts” view

- *Internet: “network of networks”*
 - Interconnected ISPs
- *protocols are everywhere*
 - control sending, receiving of messages
 - e.g., HTTP (Web), streaming video, Skype, TCP, IP, WiFi, 4G, 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, interconnected 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

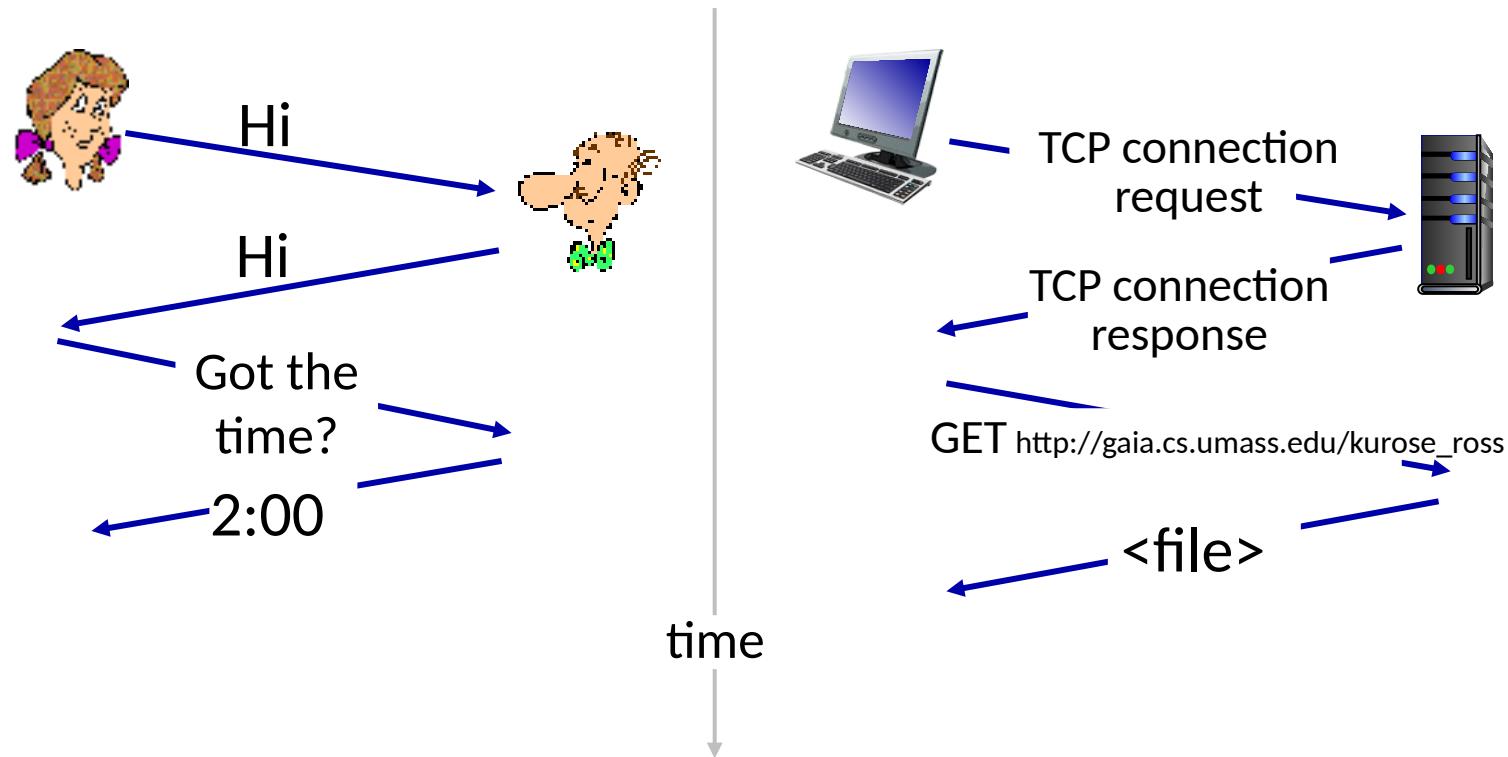
Network protocols:

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

*Protocols define the **format** and the **order of messages exchanged** between two or more communicating entities, as well as the **actions taken** on the transmission and/or receipt of a message or other event*

What's a protocol?

A human protocol and a computer network protocol:



Q: other human protocols?

Chapter 1: roadmap

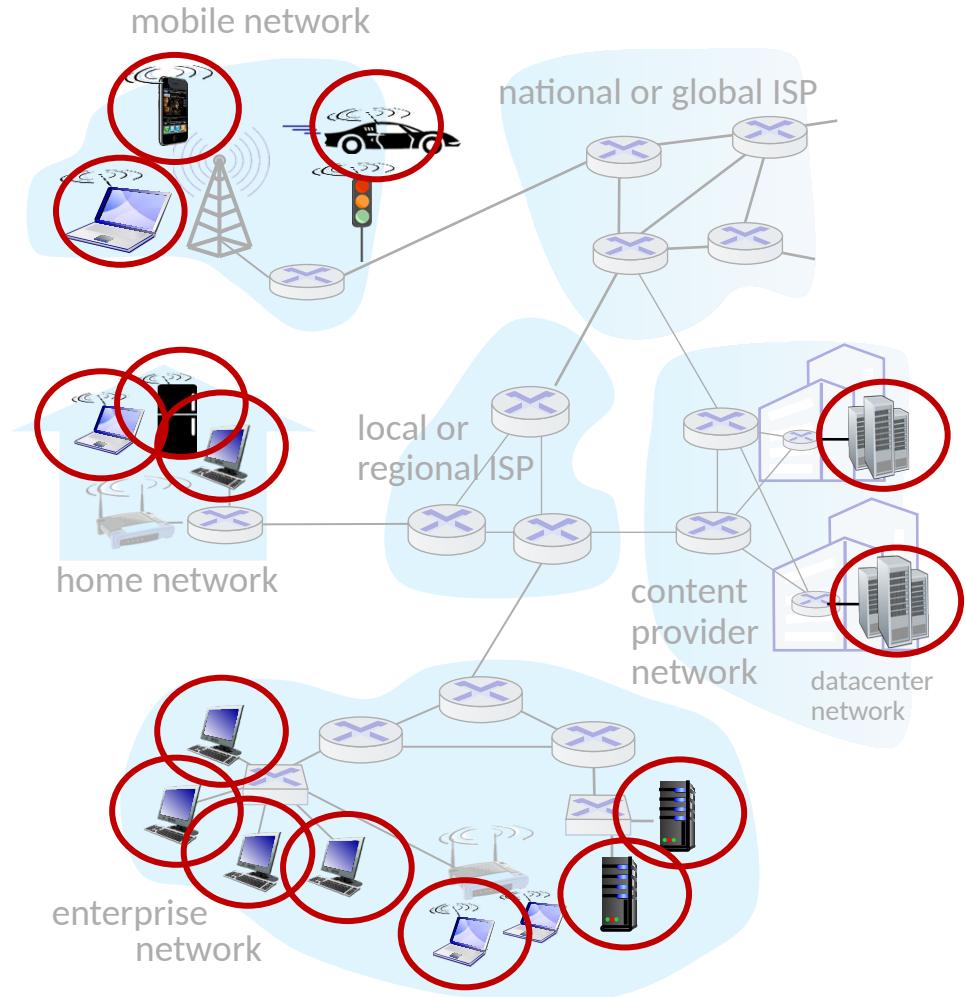
- What *is* the Internet?
- What *is* a protocol?
- **Network edge:** hosts, access network, physical media
- Network core: packet/circuit switching, internet structure
- Performance: loss, delay, throughput
- Security
- Protocol layers, service models
- History



A closer look at Internet structure

Network edge:

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



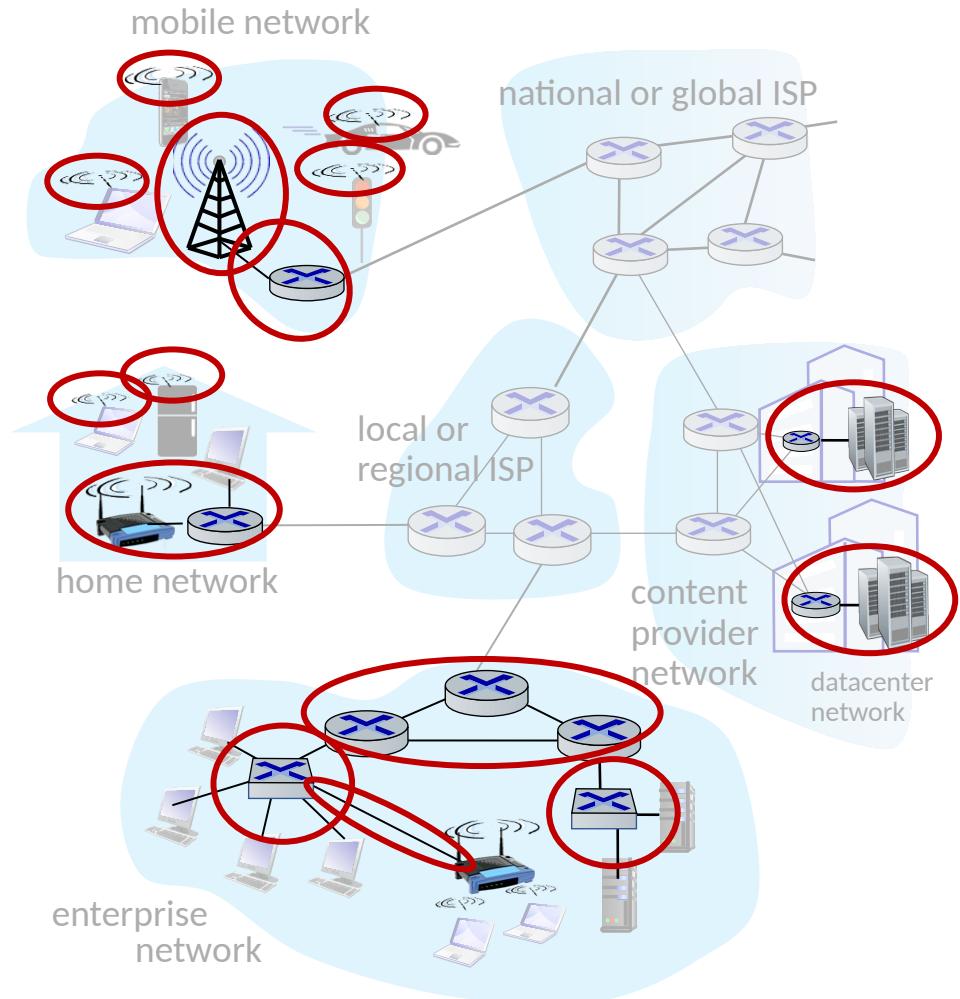
A closer look at Internet structure

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:

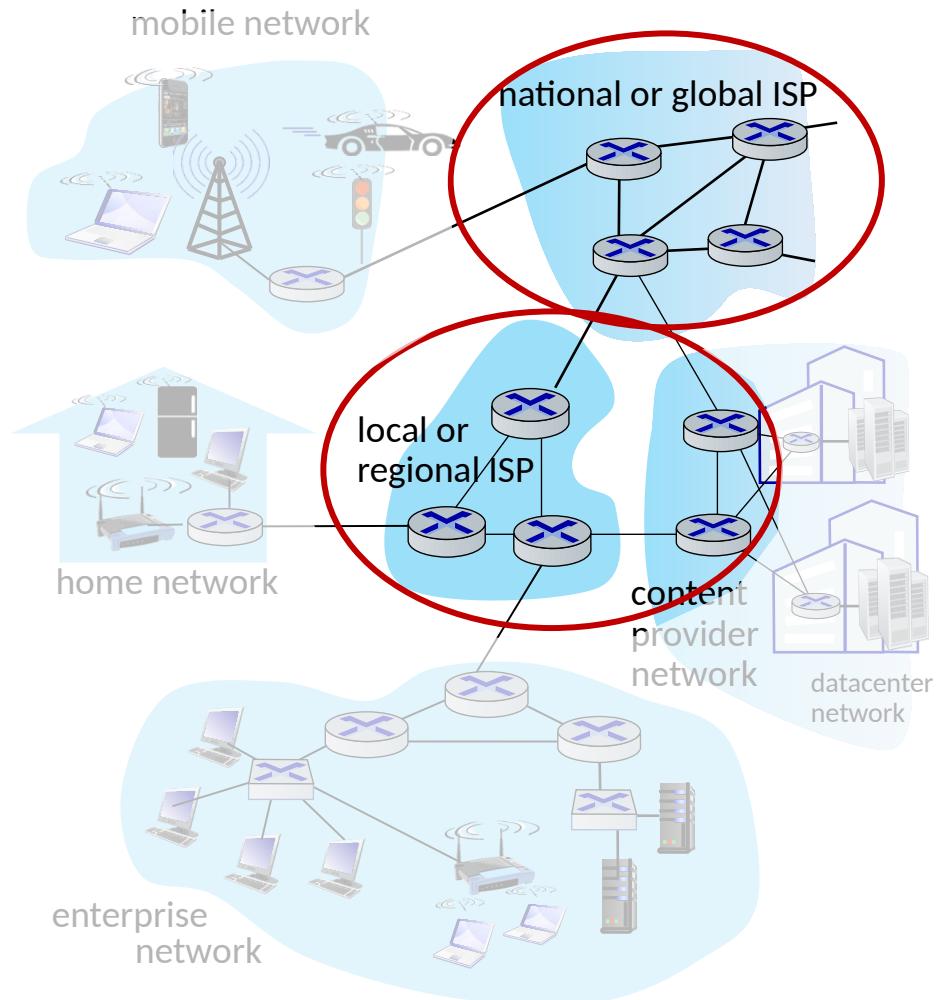
- hosts: clients and servers
- servers often in data centers

Access networks, physical media:

- wired, wireless communication links

Network core:

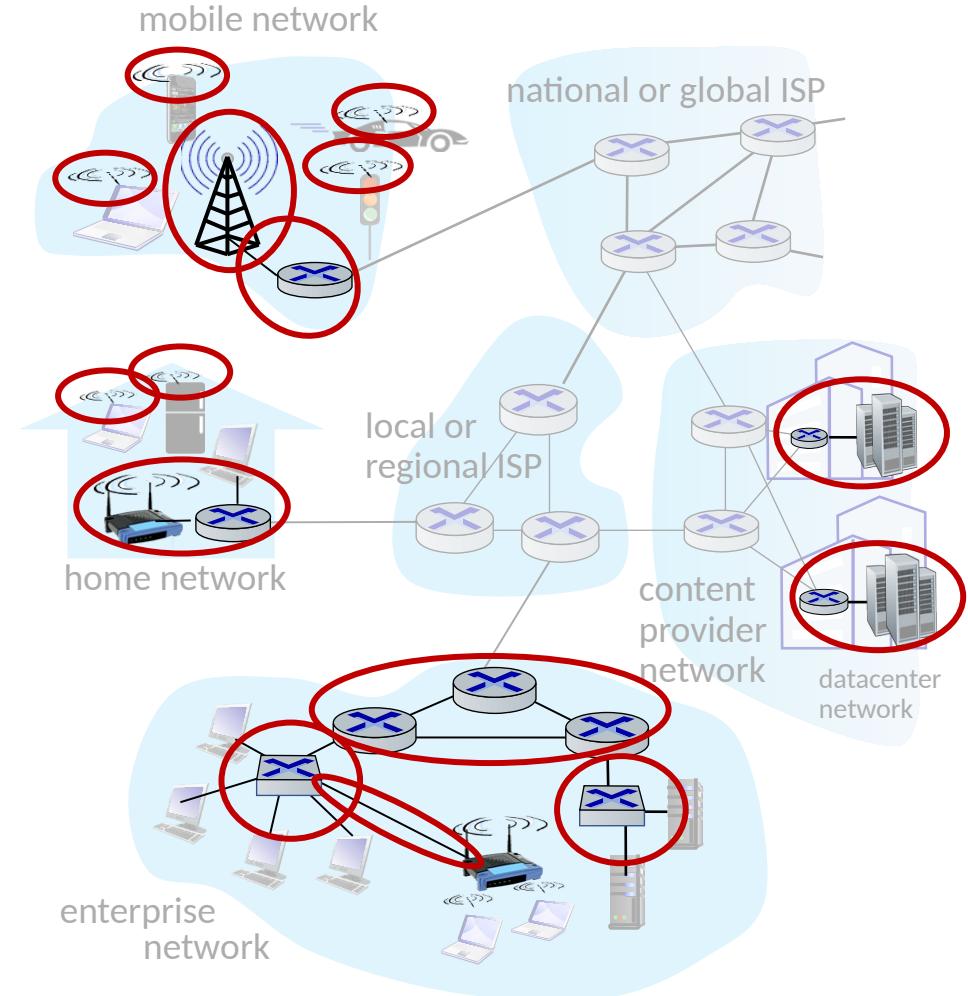
- interconnected routers
- network of networks



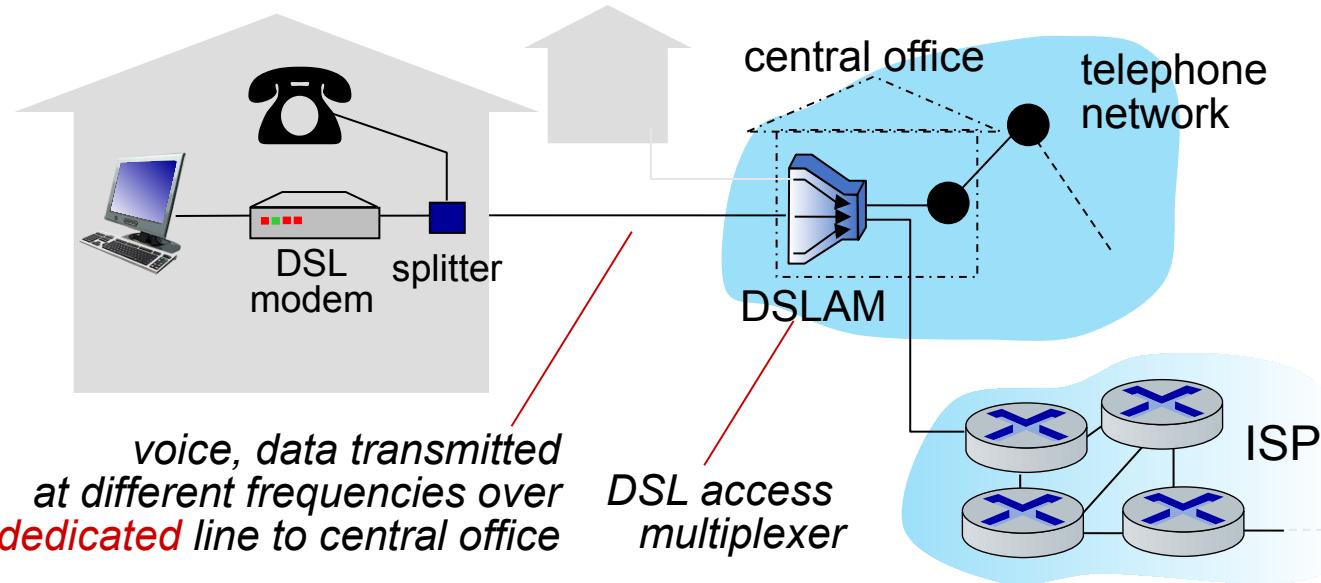
Access networks and physical media

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

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

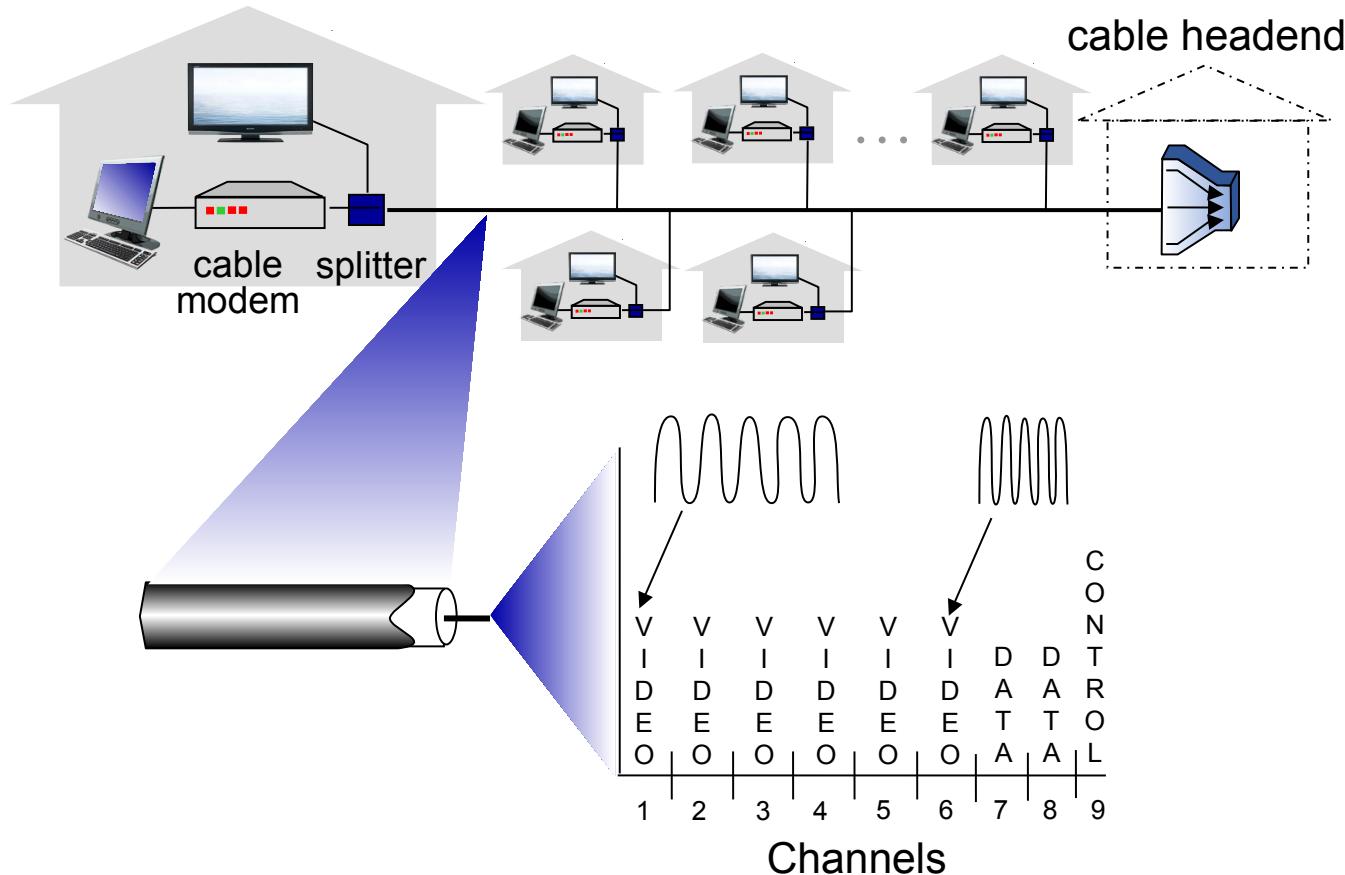


Access networks: digital subscriber line (DSL)



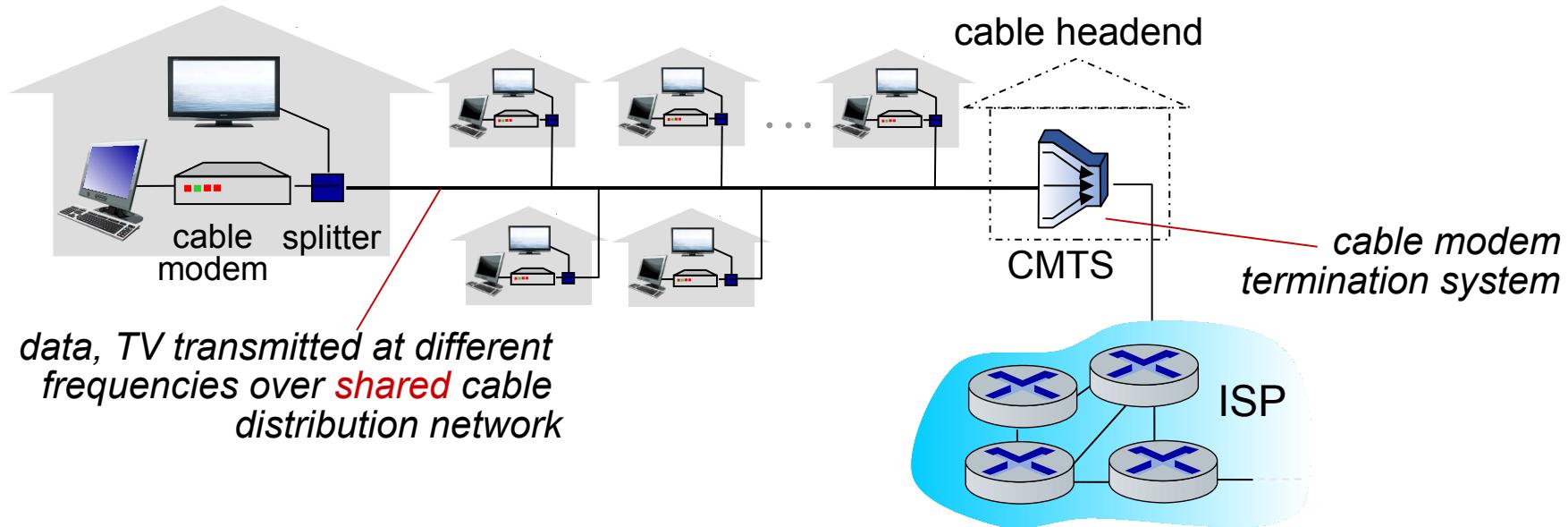
- use **existing** telephone line to central office DSLAM
 - data over DSL phone line goes to Internet
 - voice over DSL phone line goes to telephone net
- 24-52 Mbps dedicated downstream transmission rate
- 3.5-16 Mbps dedicated upstream transmission rate

Access networks: cable-based access



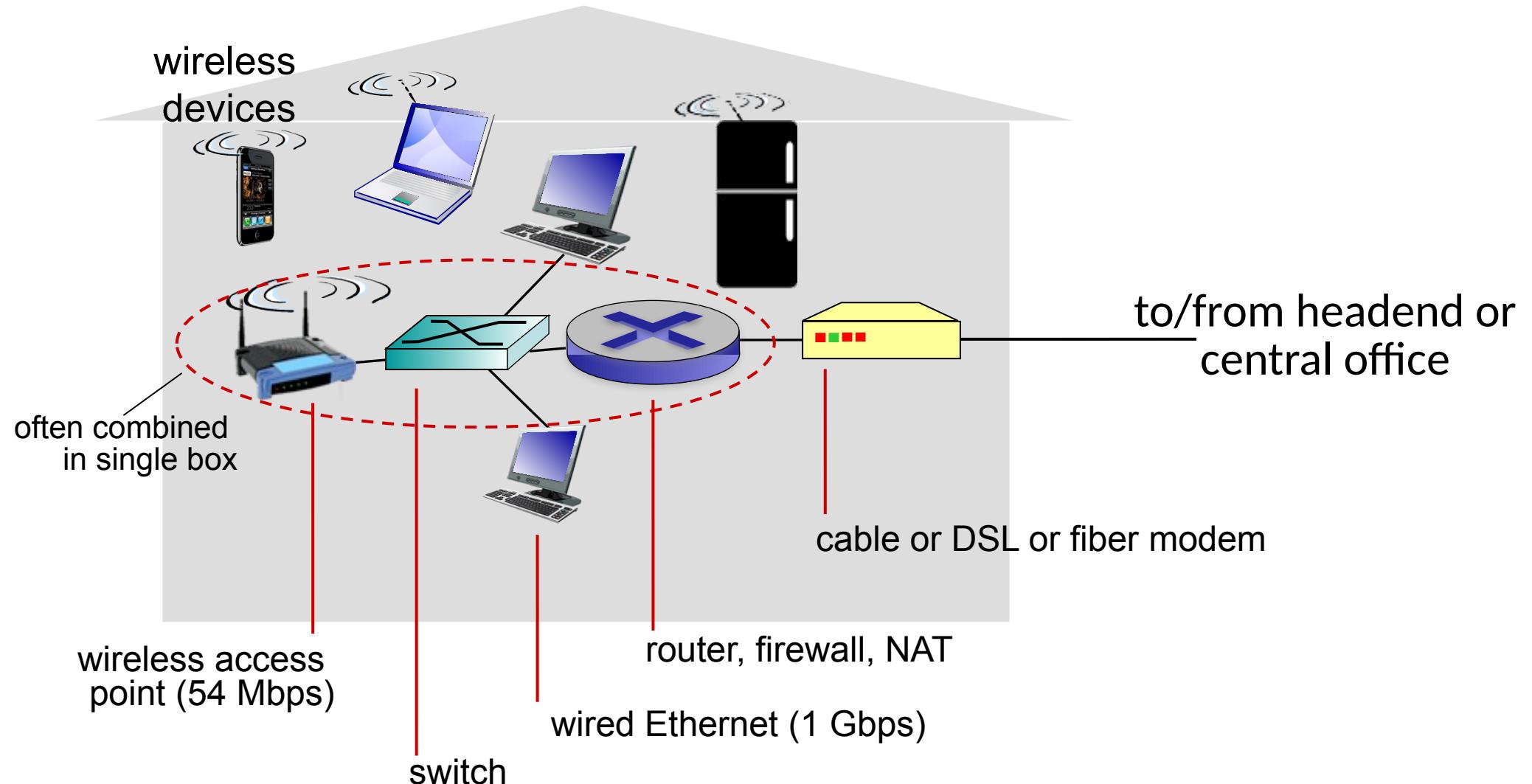
frequency division multiplexing (FDM): different channels transmitted in different frequency bands

Access networks: cable-based access



- HFC: hybrid fiber coax
 - asymmetric: up to 40 Mbps - 1.6 Gbps downstream transmission rate, 30 Mbps - 1 Gbps upstream transmission rate, per channel
- network of cable, fiber attaches homes to ISP router
 - homes **share access network** to cable headend

Access networks: home networks



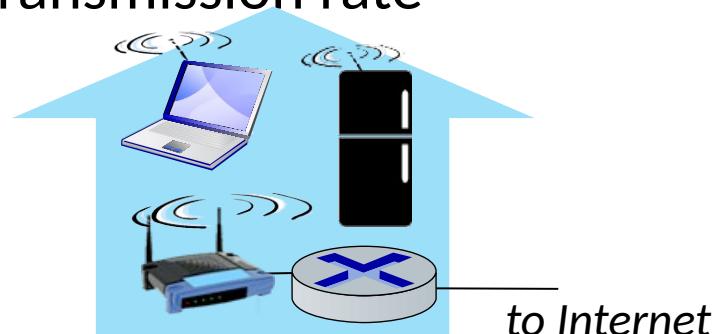
Wireless access networks

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 (~30 m)
- IEEE 802.11b/g/n/ac/ax (Wi-Fi): 11, 54, 450, 1730, 2400 Mbps transmission rate

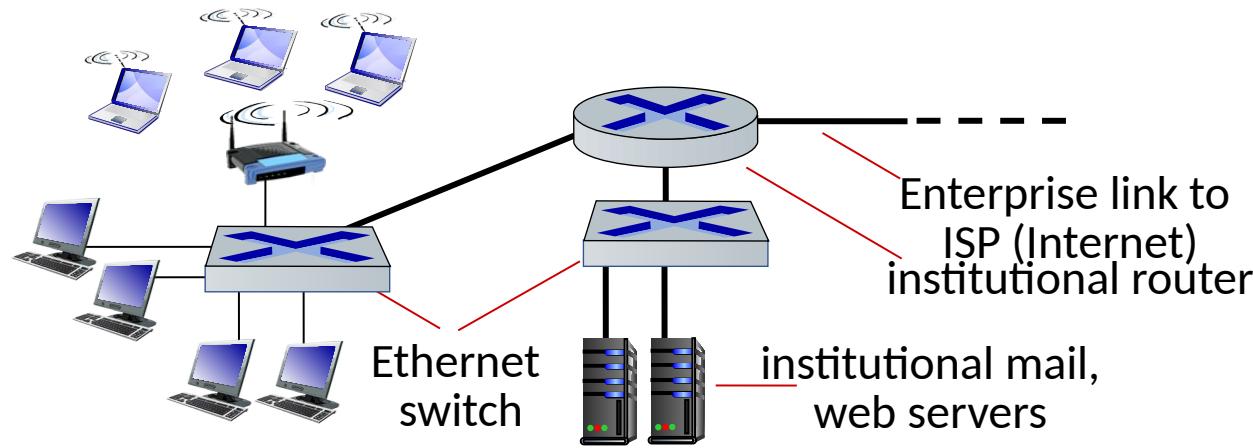


Wide-area cellular access networks

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



Access networks: enterprise networks



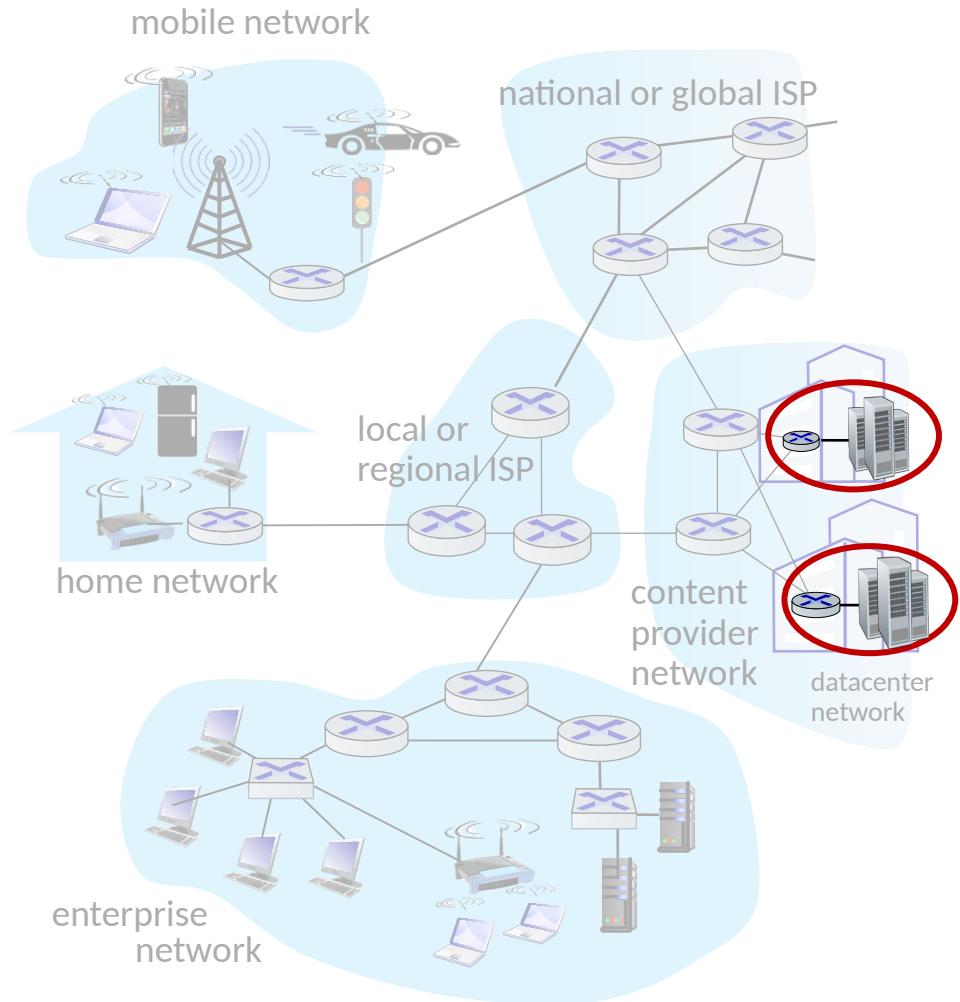
- companies, universities, etc.
- mix of wired, wireless link technologies, connecting a mix of switches and routers (we'll cover differences shortly)
 - Ethernet: wired access at 100Mbps, 1Gbps, 10Gbps
 - WiFi: wireless access points at 11, 54, 450 Mbps

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)



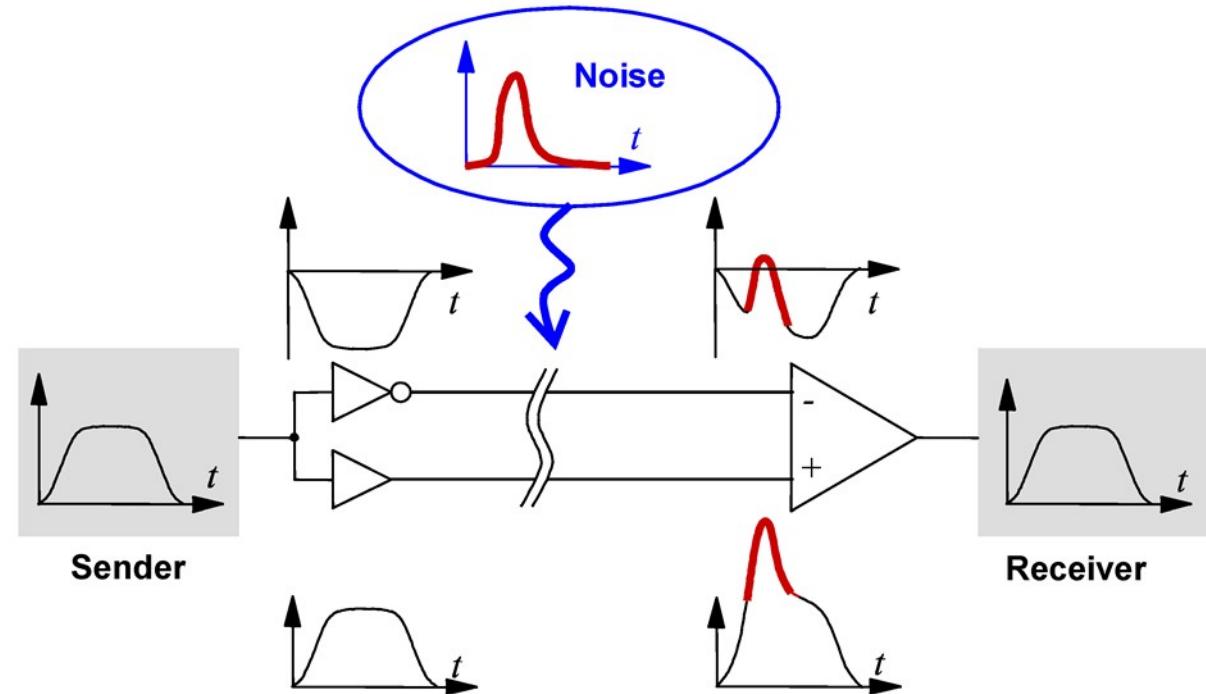
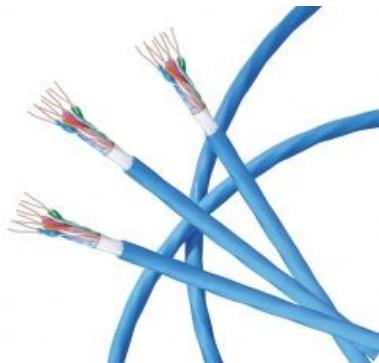
Links: physical media

- **bit:** propagates between transmitter/receiver pairs
- **physical link:** what lies between transmitter & receiver
- **guided media:**
 - signals propagate in solid media: copper, fiber, coax
- **unguided media:**
 - signals propagate freely, e.g., radio

Links: physical media (twisted pair)

Twisted pair (TP)

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



<https://resourcespcb.cadence.com/blog/differential-signaling-demystified>

Links: physical media (coax. fiber)

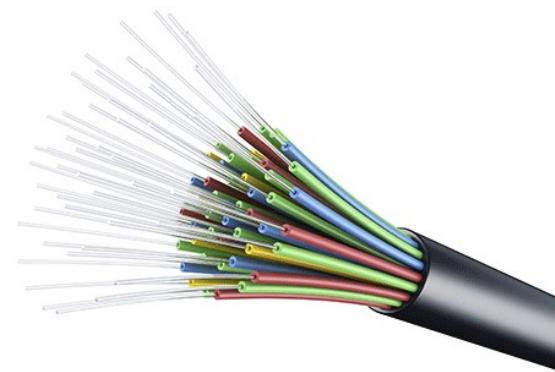
Coaxial cable:

- two concentric copper conductors
- bidirectional
- 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)
- low error rate:
 - repeaters spaced far apart
 - immune to electromagnetic noise



Links: physical media (radio)

Wireless radio

- signal carried in various “bands” in electromagnetic spectrum
- no physical “wire”
- broadcast, “half-duplex” (sender to receiver)
- propagation environment effects:
 - reflection
 - obstruction by objects
 - Interference/noise

Radio link types:

- **Wireless LAN (Wi-Fi)**
 - 10-100's Mbps; 10's of meters
- **Wide-area** (e.g., 4G cellular)
 - 10's Mbps over ~10 Km
- **Bluetooth:** cable replacement
 - short distances, limited rates
- **Terrestrial microwave**
 - point-to-point; 45 Mbps channels
- **Satellite**
 - up to 45 Mbps per channel
 - 270 msec end-end delay

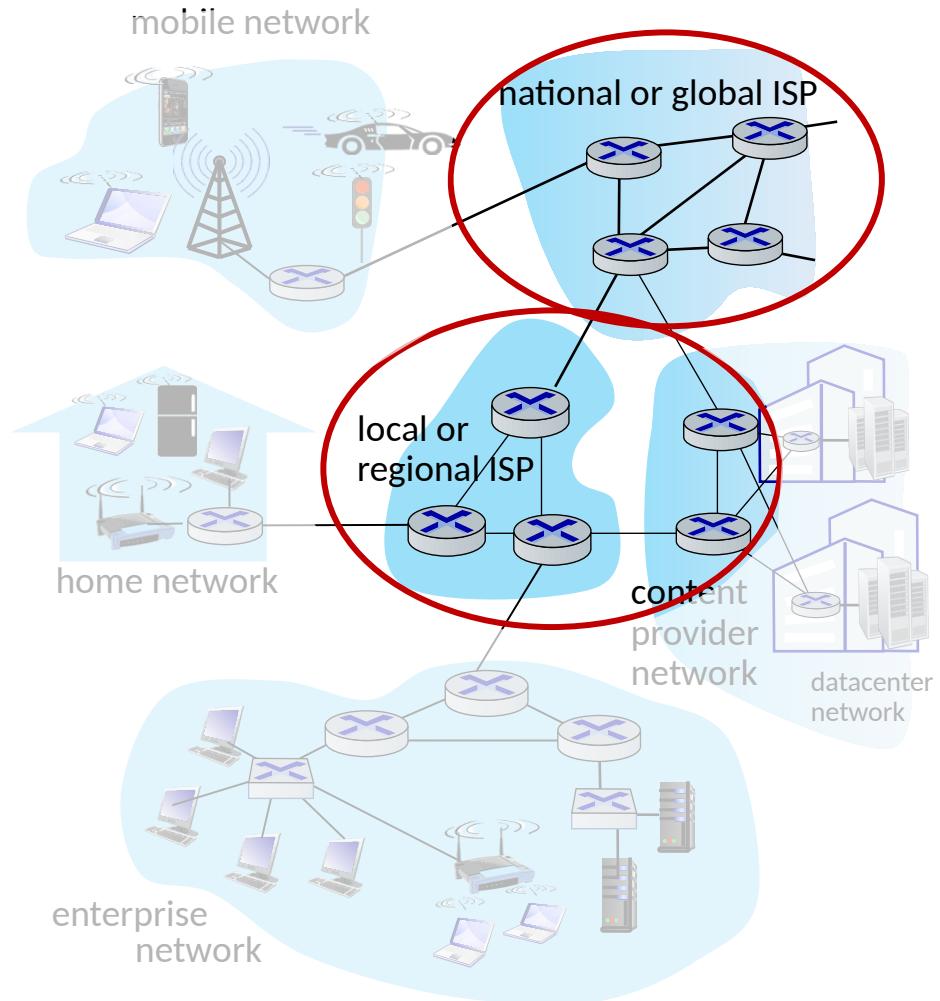
Chapter 1: roadmap

- 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



The network core

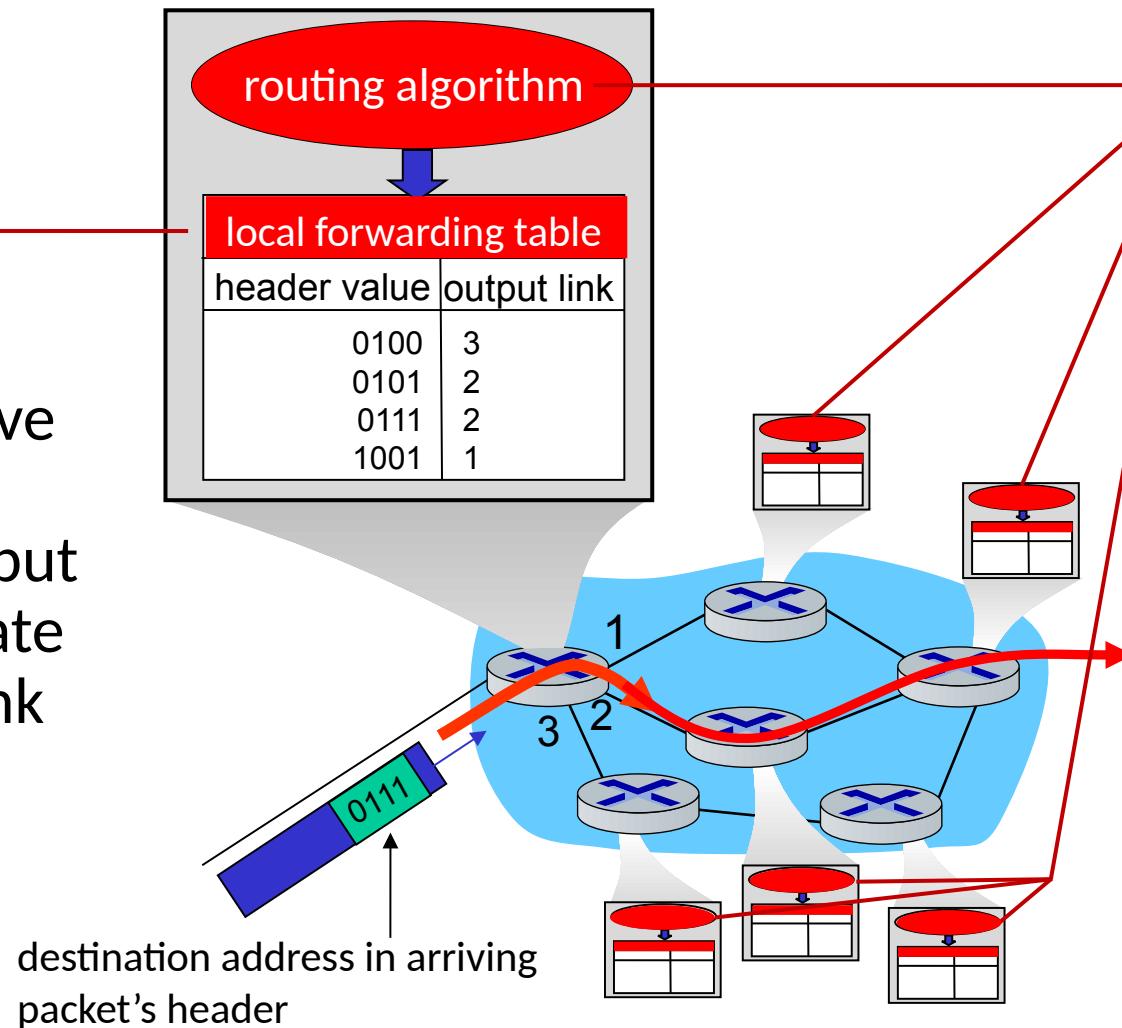
- mesh of interconnected routers
- **packet-switching**: hosts break application-layer messages into *packets*
 - network **forwards** packets from one router to the next, across links on path from **source to destination**



Two key network-core functions

Forwarding:

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



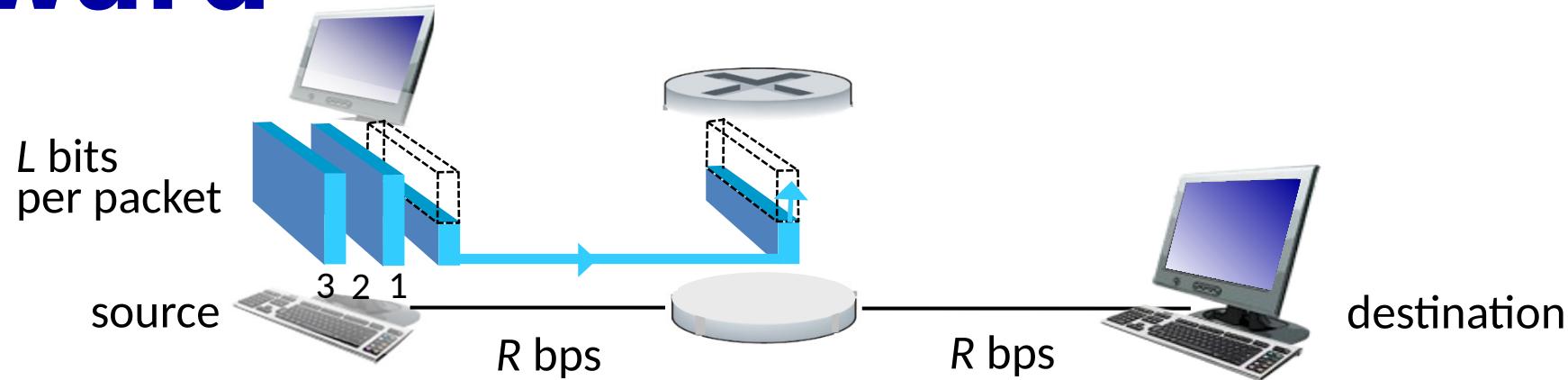
Routing:

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





Packet-switching: store-and-forward

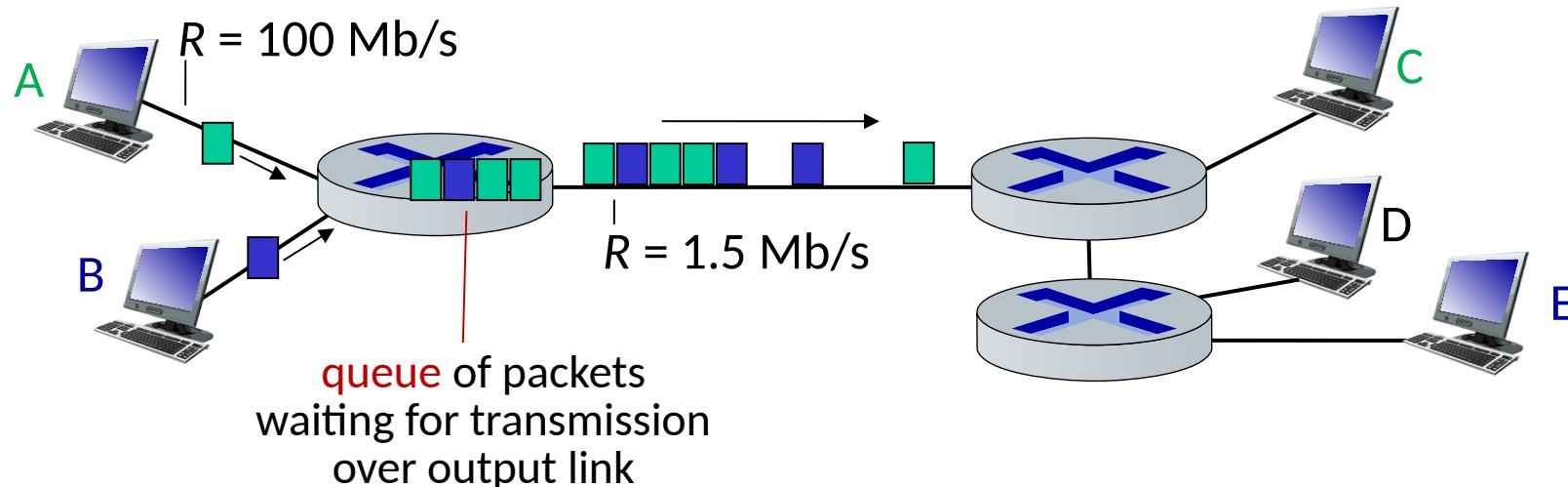


- **packet transmission delay:** takes L/R seconds to transmit (push out) L -bit packet into link at R bps
- **store and forward:** entire packet must arrive at router before it can be transmitted on next link

One-hop numerical example:

- $L = 10 \text{ kbits}$
- $R = 100 \text{ Mbps}$
- one-hop transmission delay = 0.1 msec

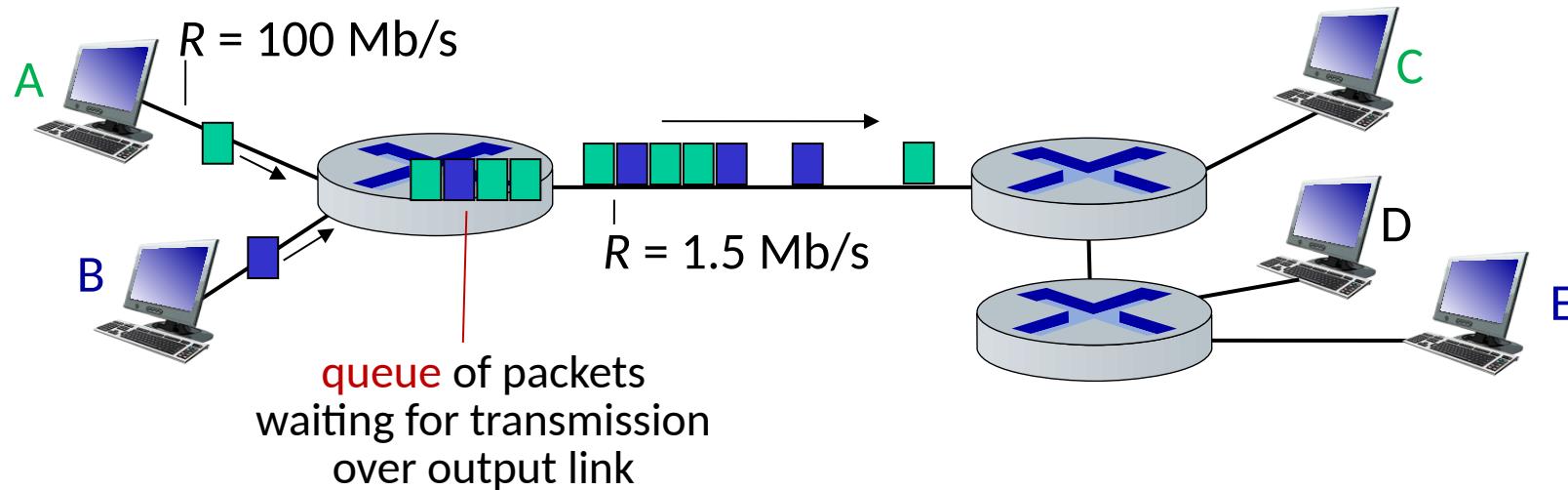
Packet-switching: queueing



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



Packet-switching: queueing



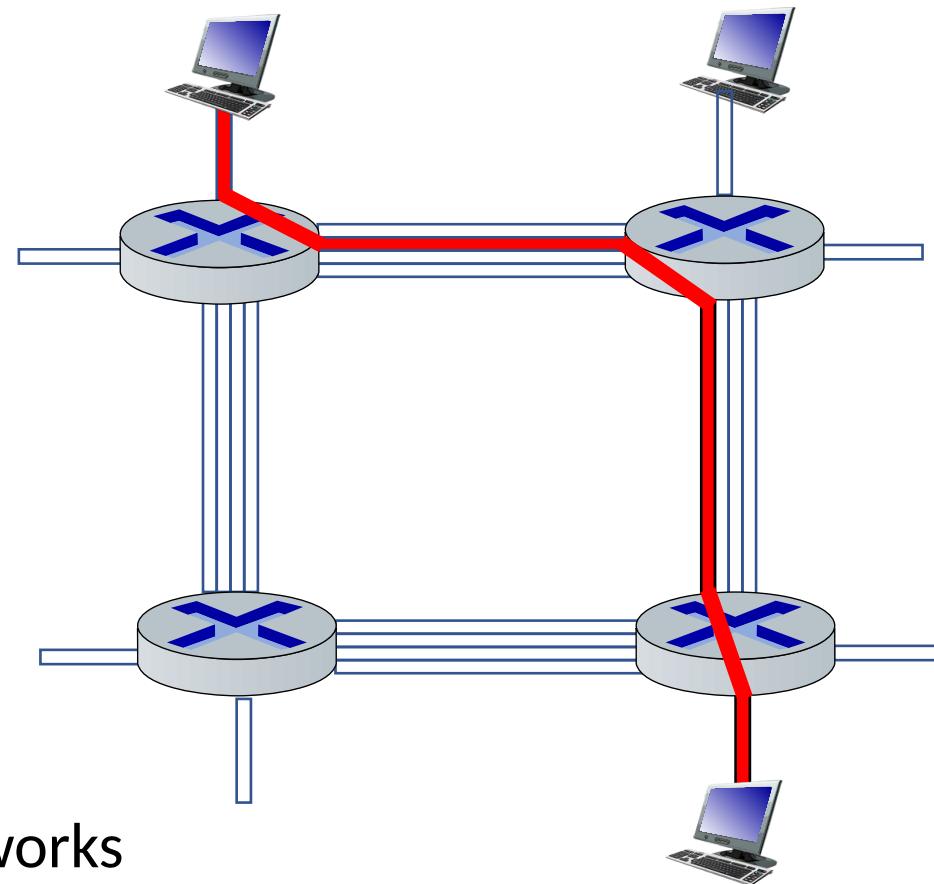
Packet queuing and loss: if arrival rate (in bps) to link exceeds transmission rate (bps) of link for some period of time:

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

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



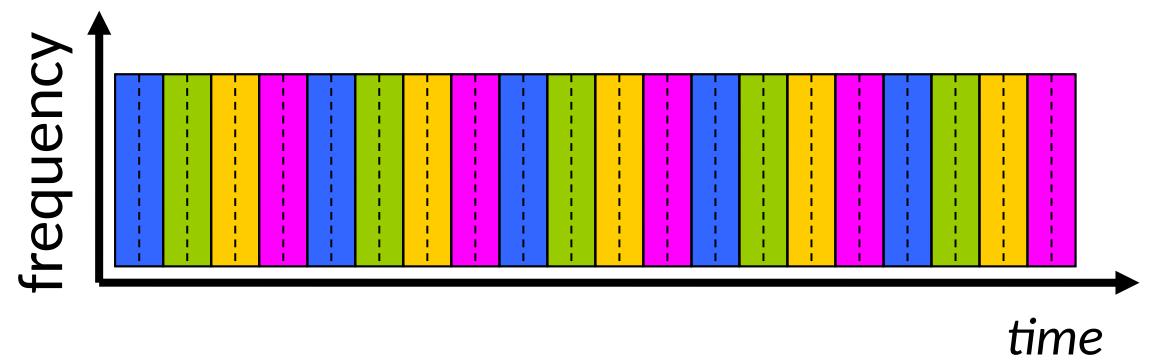
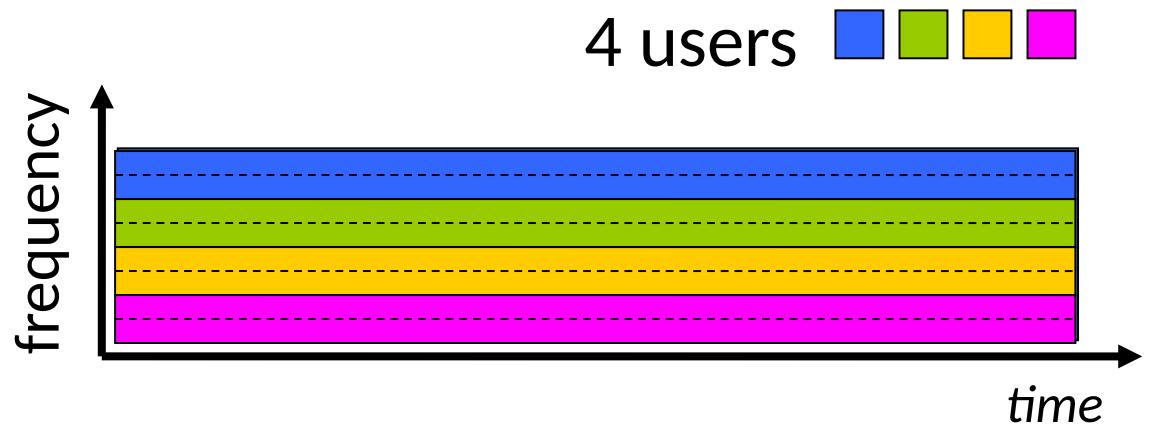
Circuit switching: FDM and TDM

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)

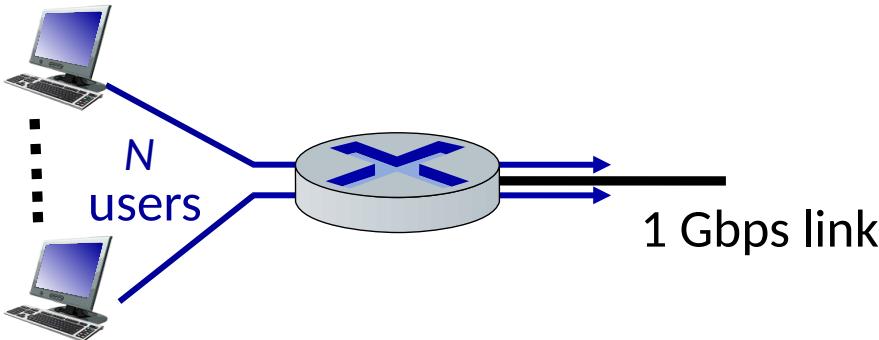
- time divided into slots
- each call allocated periodic slot(s), can transmit at maximum rate of (wider) frequency band (only) during its time slot(s)



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)

Packet switching versus circuit switching

En binomisk fordeling eller binomialfordeling er en diskret fordeling (et begrep innen sannsynlighetsteori og matematisk statistikk) som håndterer hyppige (diskrete) forsøk med fast sannsynlighet.

Dersom en stokastisk variabel X er binomisk fordelt, med n =totale antall forsøk, k =antall lykkede forsøk og p =sannsynligheten for å lykkes i hvert forsøk, skriver man :

$$X \in Bin(n, p)$$

X har sannsynlighetsfunksjonen

$$p_X(k) = \binom{n}{k} p^k (1-p)^{n-k}.$$

der p er sannsynligheten for at hendelsen skal inntreffe og $1 - p = q$ således sannsynligheten for at hendelsen ikke skal inntreffe. Slik dukker binomialkoeffisientene opp i fordelingen.

p	0,1	Hver bruker aktiv 10 prosent av tiden
n	35	Antall brukere totalt
k		Antall aktive brukere

k	P(k)	P(1 <= X <= k)	P(X >= k)
0	0,02503	0,02503	1,00000
1	0,09734	0,12238	0,97497
2	0,18387	0,30625	0,87762
3	0,22473	0,53098	0,69375
4	0,19976	0,73075	0,46902
5	0,13762	0,86836	0,26925
6	0,07645	0,94482	0,13164
7	0,03519	0,98001	0,05518
8	0,01369	0,99370	0,01999
9	0,00456	0,99826	0,00630
10	0,00132	0,99958	0,00174
11	0,00033	0,99991	0,00042 0,042 % sannsynlighet for 11 eller flere aktive bruker
12	0,00007	0,99998	0,00009
13	0,00001	1,00000	0,00002

Packet switching versus circuit switching

En binomisk fordeling eller binomialfordeling er en **diskret fordeling** (et begrep innen sannsynlighetsteori og matematisk statistikk) som håndterer hyppige (diskrete) forsøk med fast **sannsynlighet**.

Dersom en stokastisk variabel X er binomisk fordelt, med n =totale antall forsøk, k =antall lykkede forsøk og p =sannsynligheten for å lykkes i hvert forsøk, skriver man :

$$X \in Bin(n, p)$$

X har **sannsynlighetsfunksjonen**

$$p_X(k) = \binom{n}{k} p^k (1-p)^{n-k}.$$

der p er sannsynligheten for at **hendelsen** skal inntreffe og $1 - p = q$ således sannsynligheten for at hendelsen ikke skal inntreffe. Slik dukker **binomialkoeffisientene** opp i fordelingen.

p	0,1	Hver bruker aktiv 10 prosent av tiden
n	70	Antall brukere totalt
k		Antall aktive brukere

k	$P(k)$	$P(1 \leq X \leq k)$	$P(X \geq k)$
0	0,00063	0,00063	1,00000
1	0,00487	0,00550	0,99937
2	0,01868	0,02418	0,99450
3	0,04705	0,07123	0,97582
4	0,08756	0,15879	0,92877
5	0,12843	0,28722	0,84121
6	0,15459	0,44181	0,71278
7	0,15704	0,59885	0,55819
8	0,13741	0,73626	0,40115
9	0,10518	0,84144	0,26374
10	0,07129	0,91273	0,15856
11	0,04320	0,95594	0,08727 8,727 % sannsynlighet for 11 eller flere aktive bruker
12	0,02360	0,97954	0,04406
13	0,01170	0,99124	0,02046

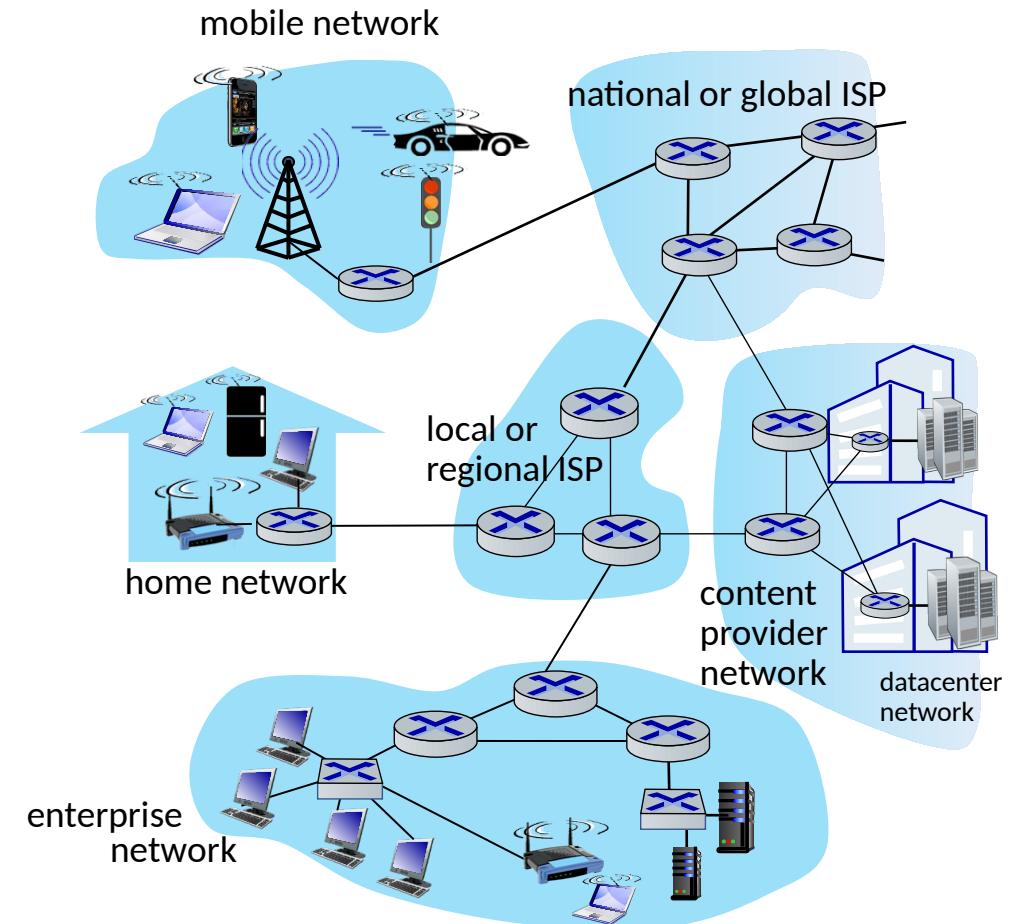
Packet switching versus circuit switching

Is packet switching a “slam dunk winner”?

- great for “bursty” data - sometimes has data to send, but at other times not
 - resource sharing
 - simpler, no call setup
- **excessive congestion possible:** packet delay and loss due to buffer overflow
 - protocols needed for reliable data transfer, congestion control

Internet structure: a “network of networks”

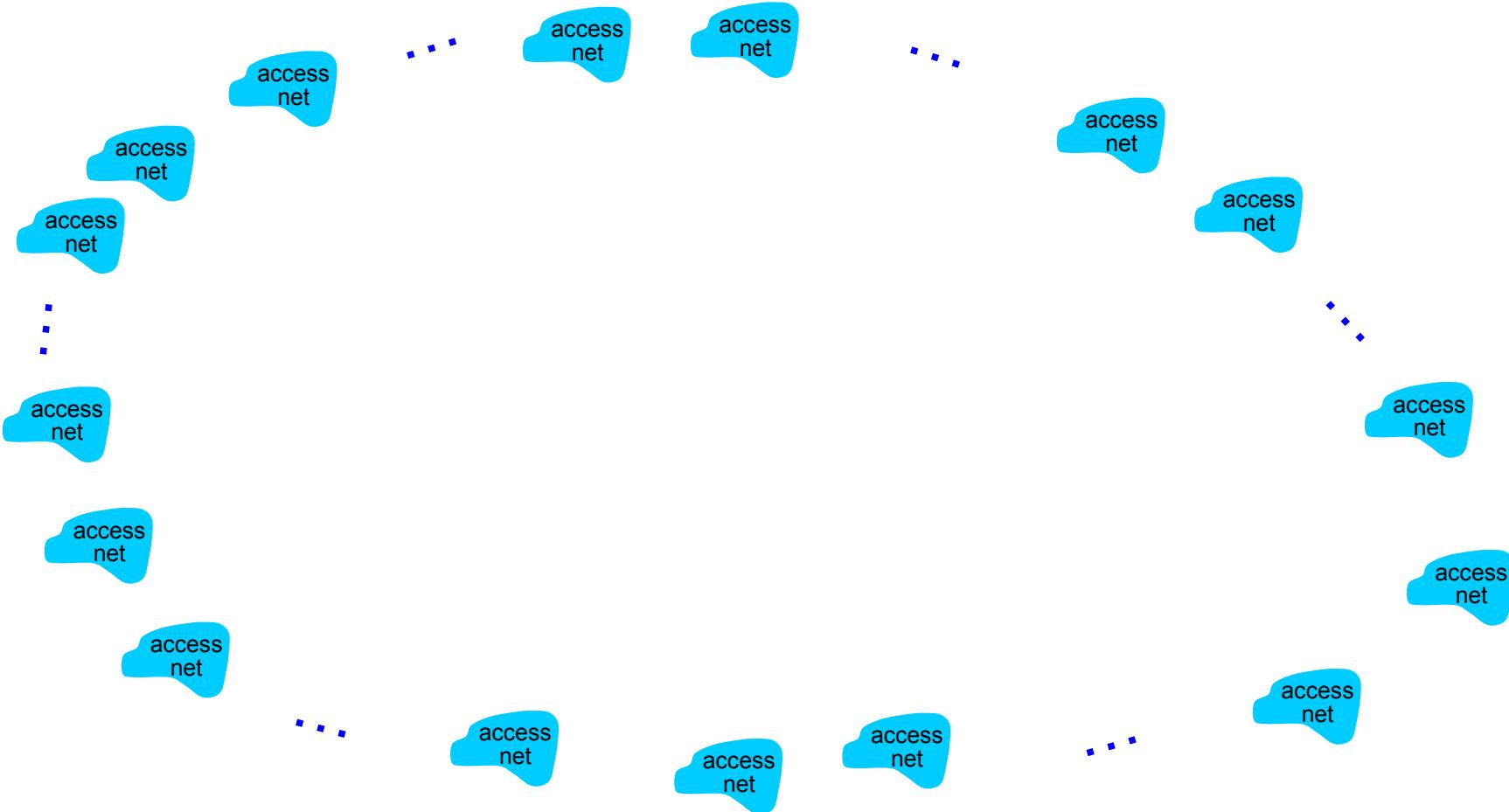
- hosts connect to Internet via **access** Internet Service Providers (ISPs)
- access ISPs in turn must be interconnected
 - so that *any two hosts (anywhere!)* can send packets to each other
- resulting network of networks is very complex
 - evolution driven by **economics, national policies**



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

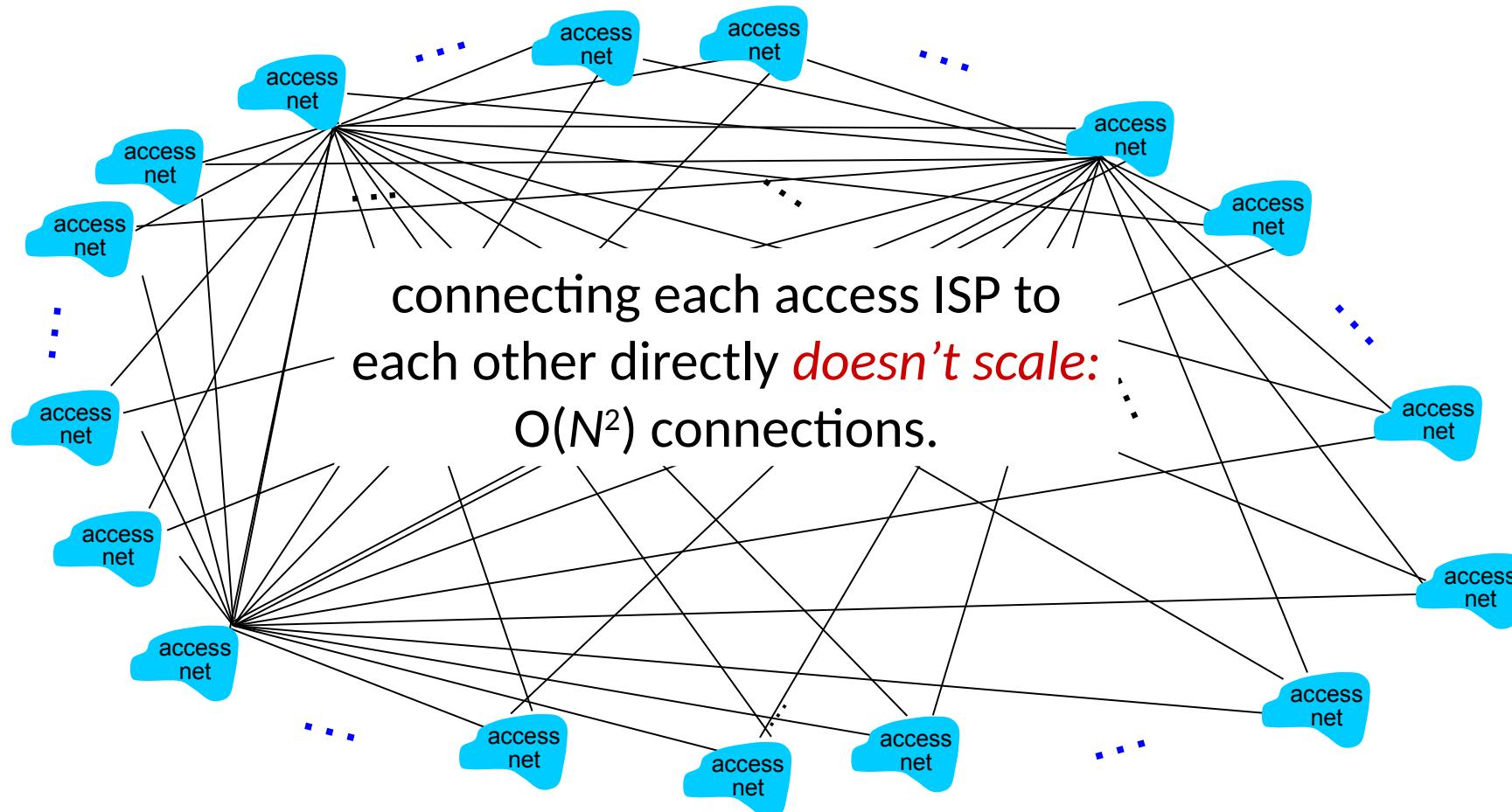
Internet structure: a “network of networks”

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



Internet structure: a “network of networks”

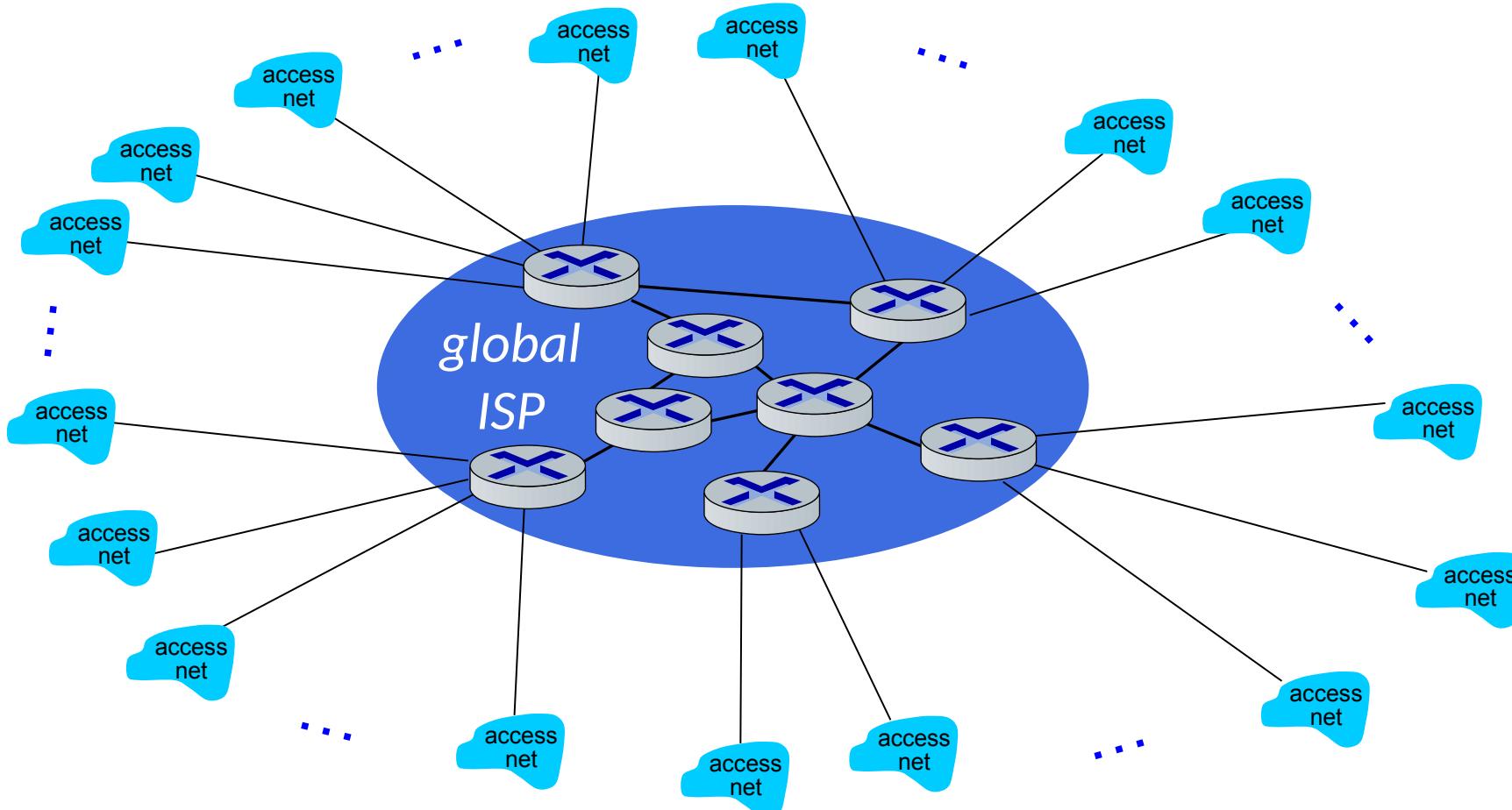
Question: given millions of access ISPs, how to connect them together?



Internet structure: a “network of networks”

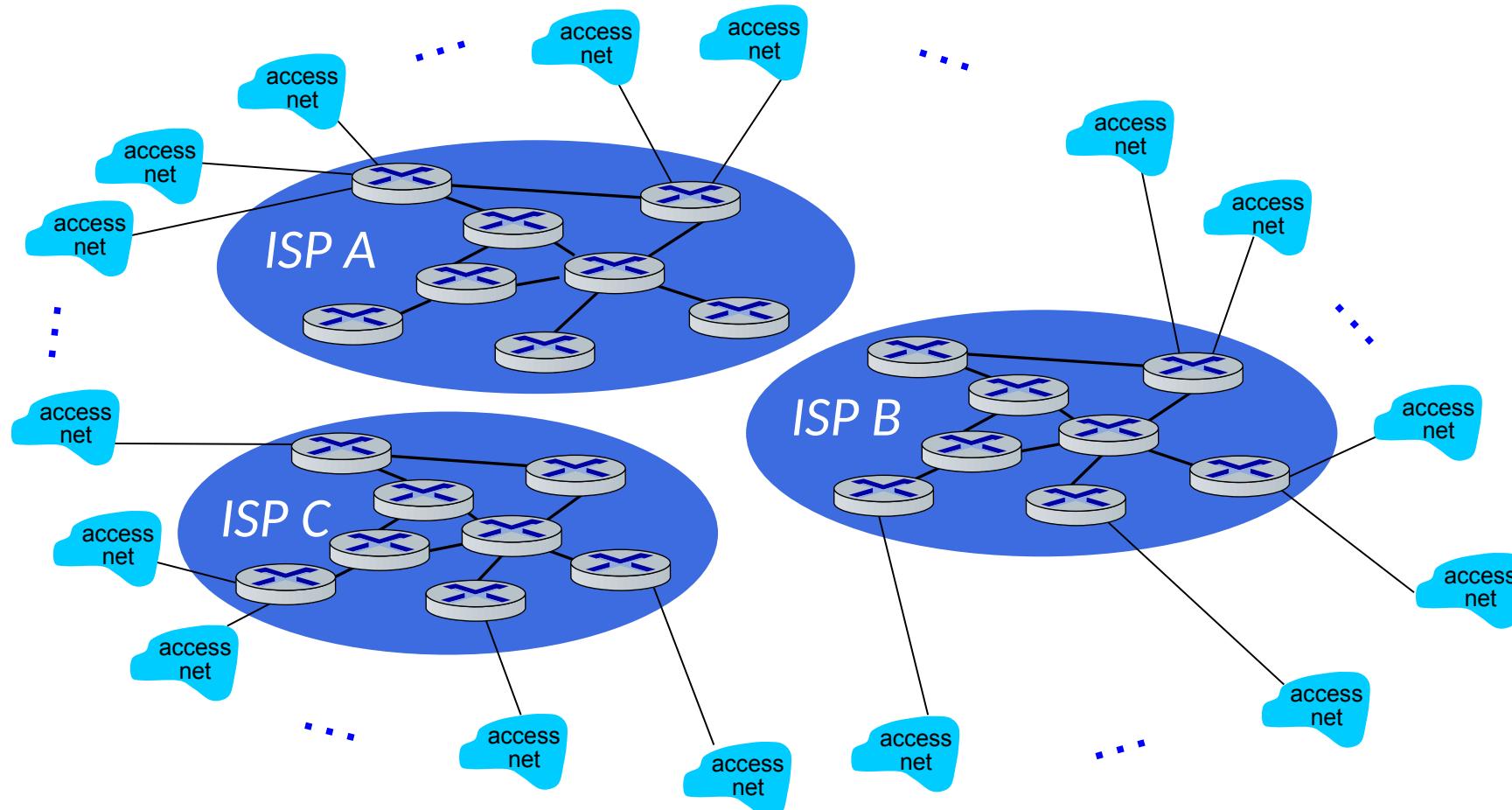
Option: connect each access ISP to one global transit ISP?

Customer and provider ISPs have economic agreement.



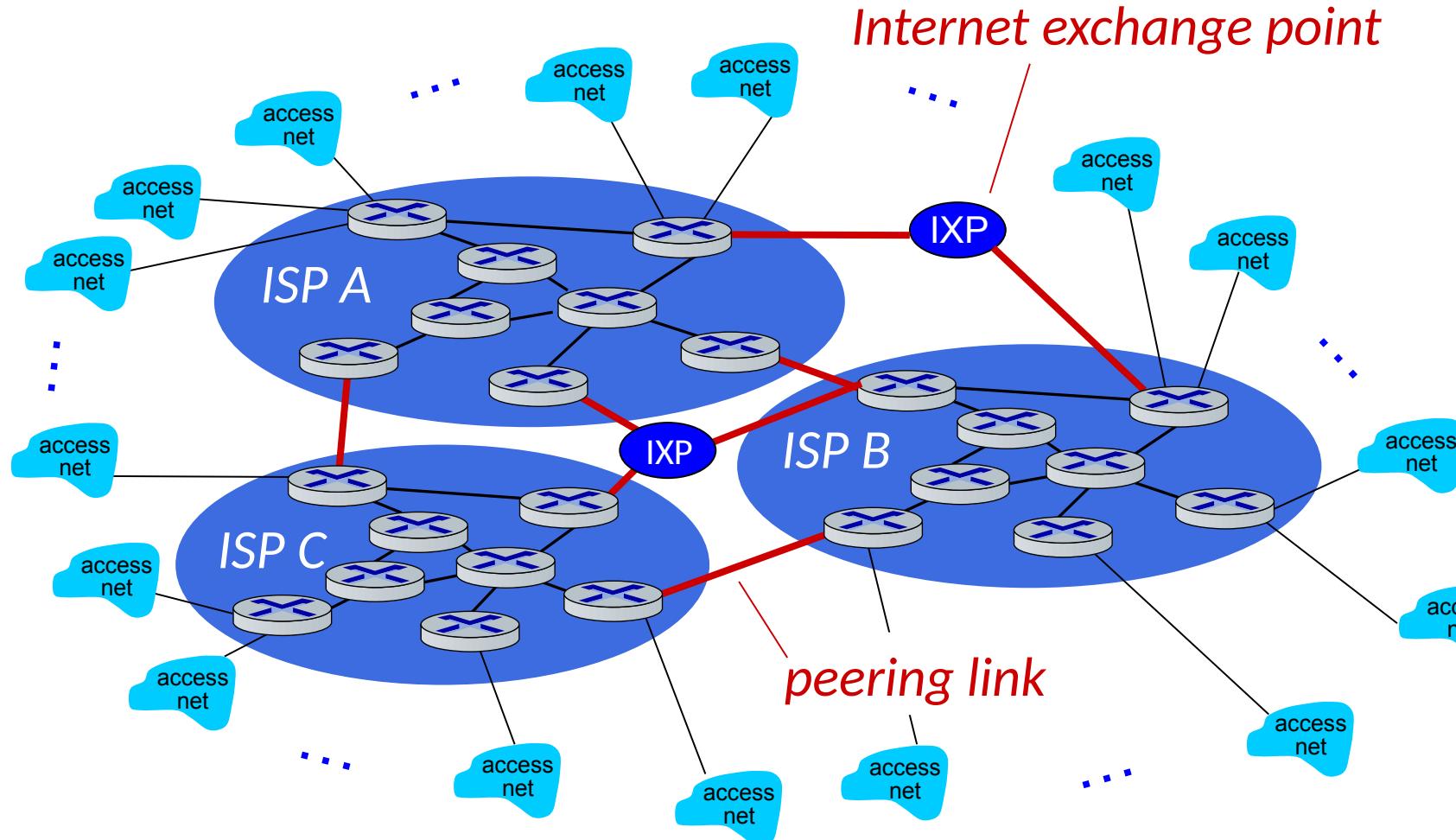
Internet structure: a “network of networks”

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



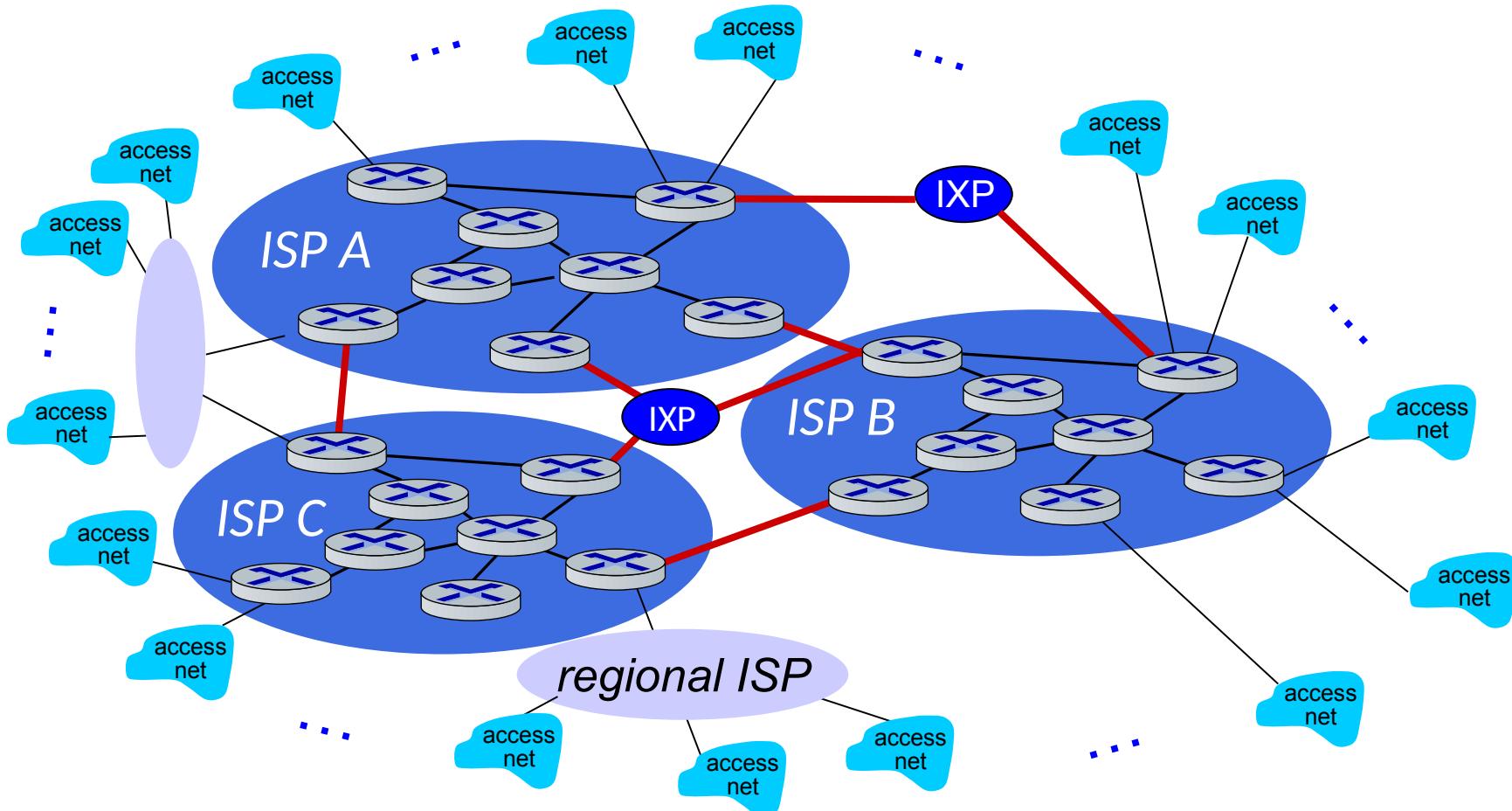
Internet structure: a “network of networks”

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



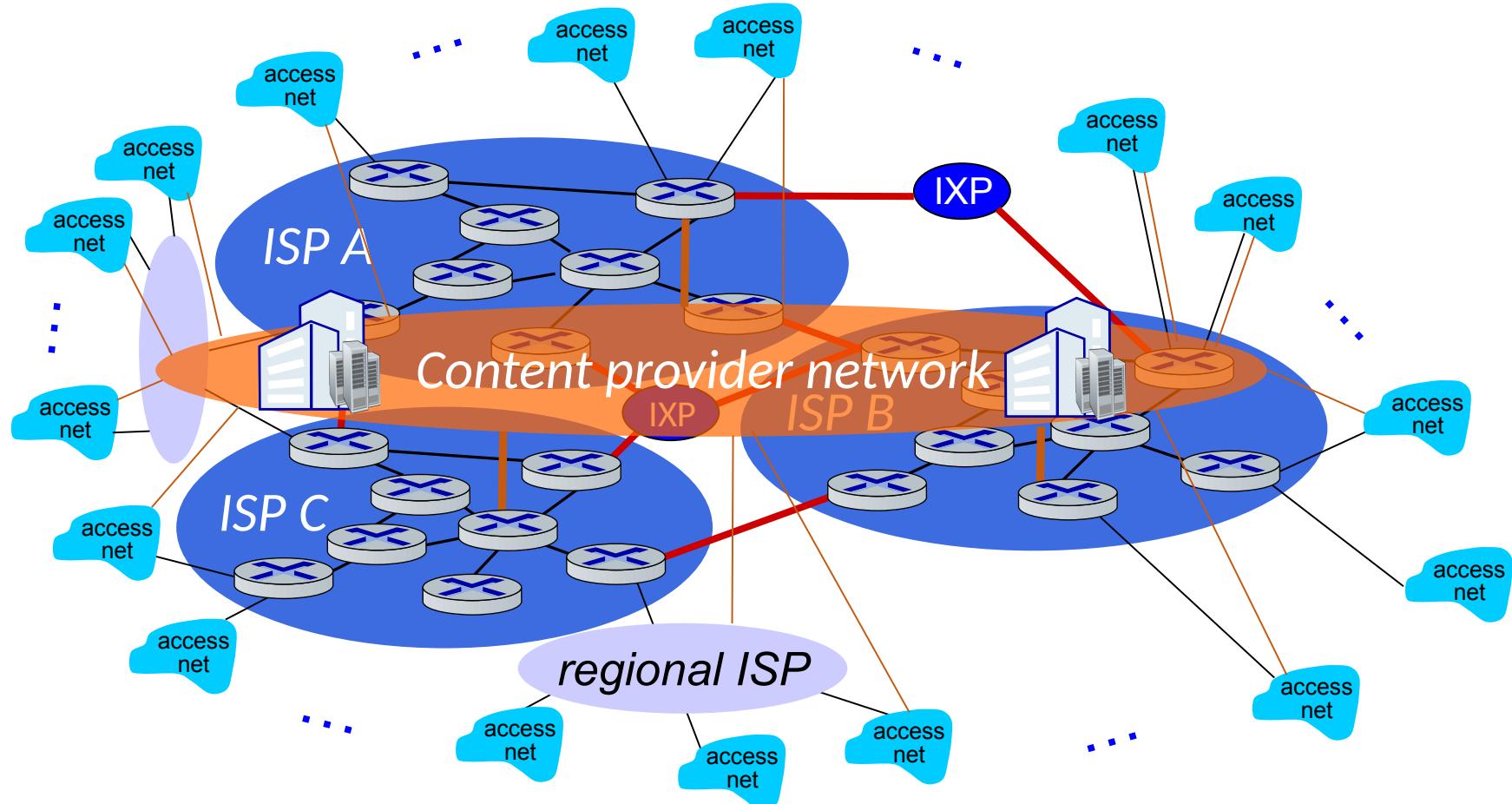
Internet structure: a “network of networks”

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

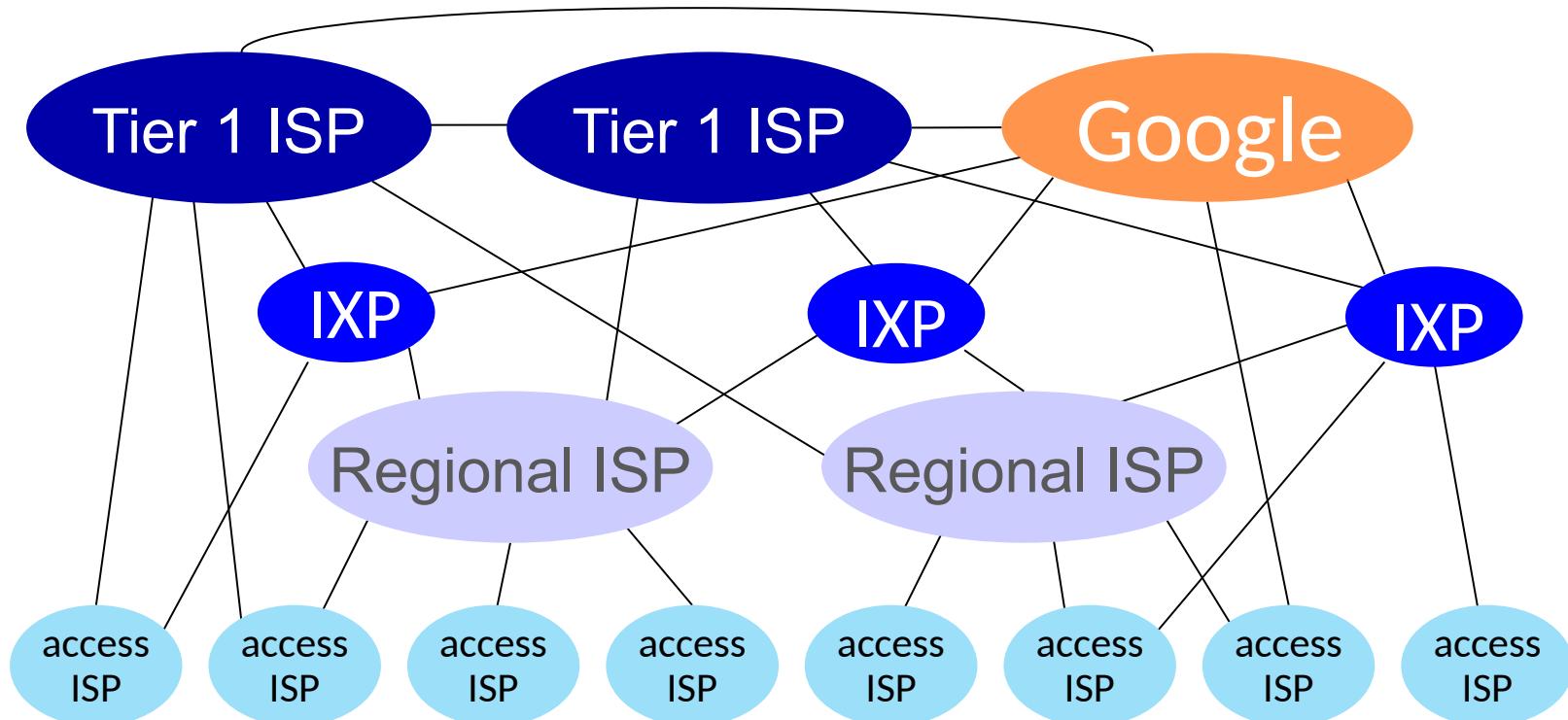


Internet structure: a “network of networks”

... 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

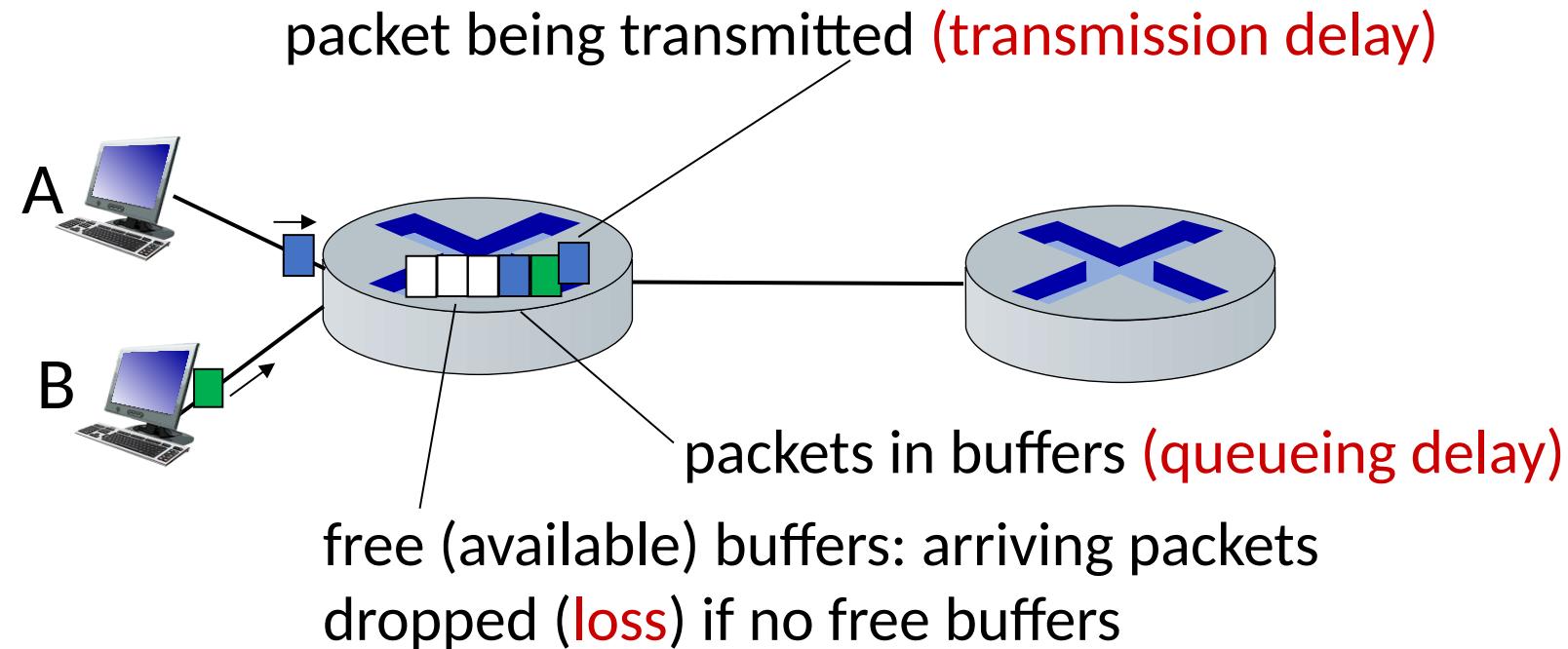
Chapter 1: roadmap

- 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

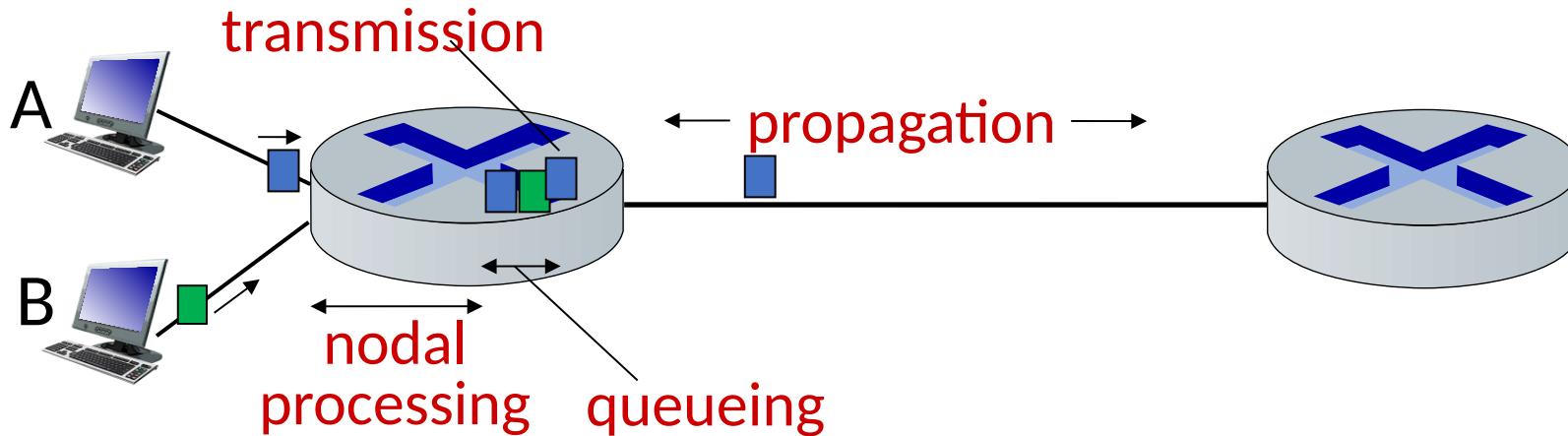


How do packet delay and loss occur?

- 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



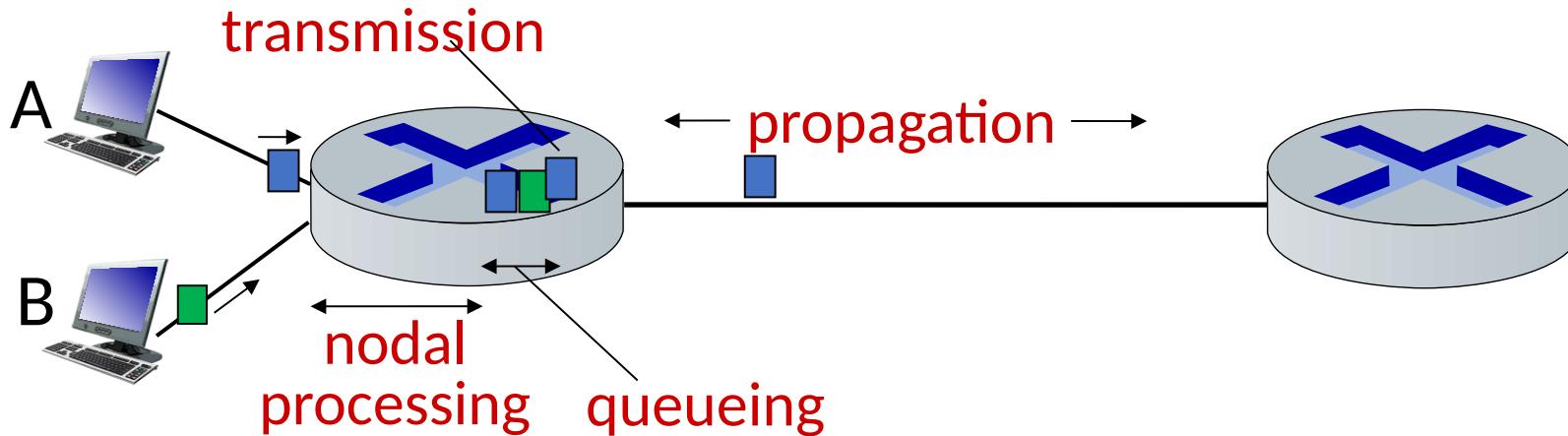
d_{proc} : nodal processing

- 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



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

d_{trans} : transmission delay:

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

d_{prop} : propagation delay:

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

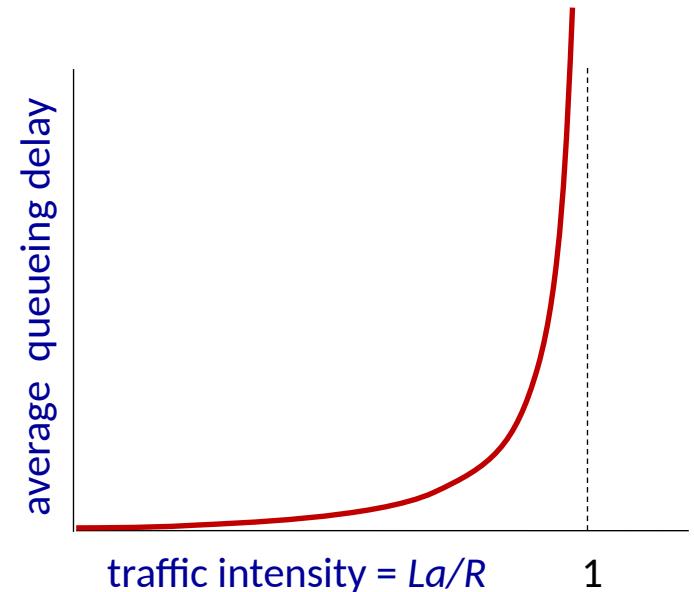
d_{trans} and d_{prop}
very different

Packet queueing delay (revisited)

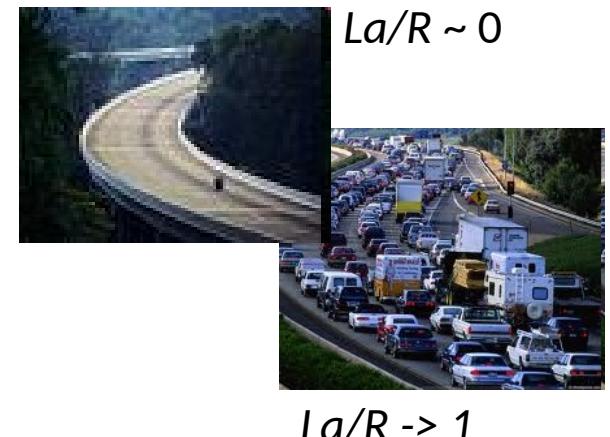
- a : average packet arrival rate
- L : packet length (bits)
- R : link bandwidth (bit transmission rate)

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

“traffic
intensity”

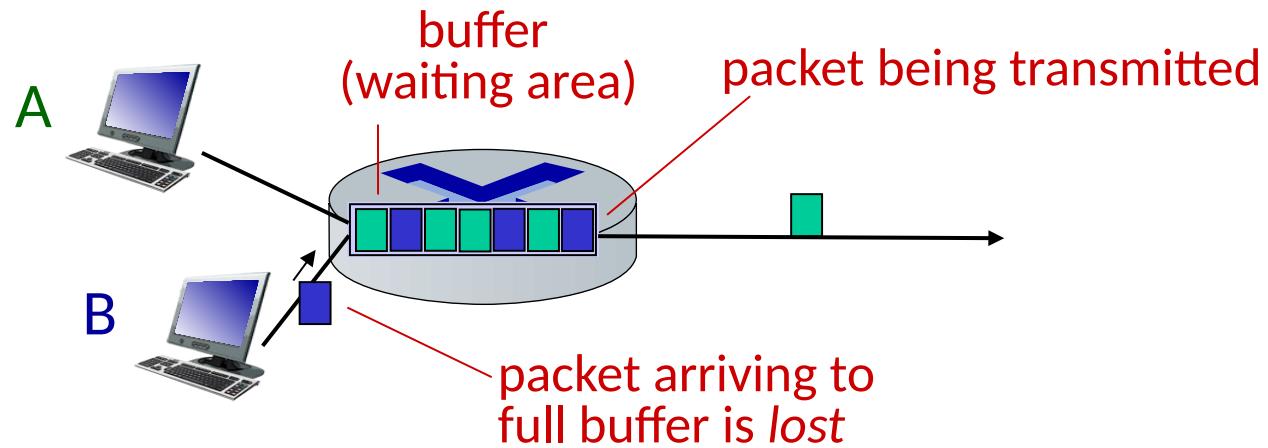


- $La/R \sim 0$: avg. queueing delay small
- $La/R \rightarrow 1$: avg. queueing delay large
- $La/R > 1$: more “work” arriving is more than can be serviced - average delay infinite!



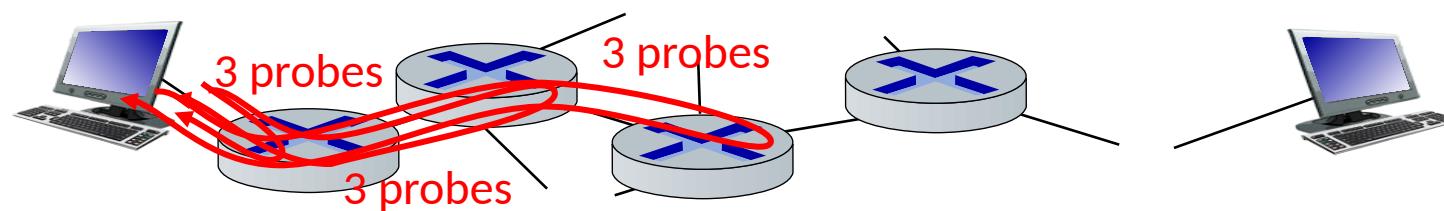
Packet loss

- 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



“Real” Internet delays and routes

- what do “real” Internet delay & loss look like?
- **traceroute** (**tracert** in Windows) 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 (TTL) field value of i)
 - router i will return packets to sender
 - sender measures time interval between transmission and reply



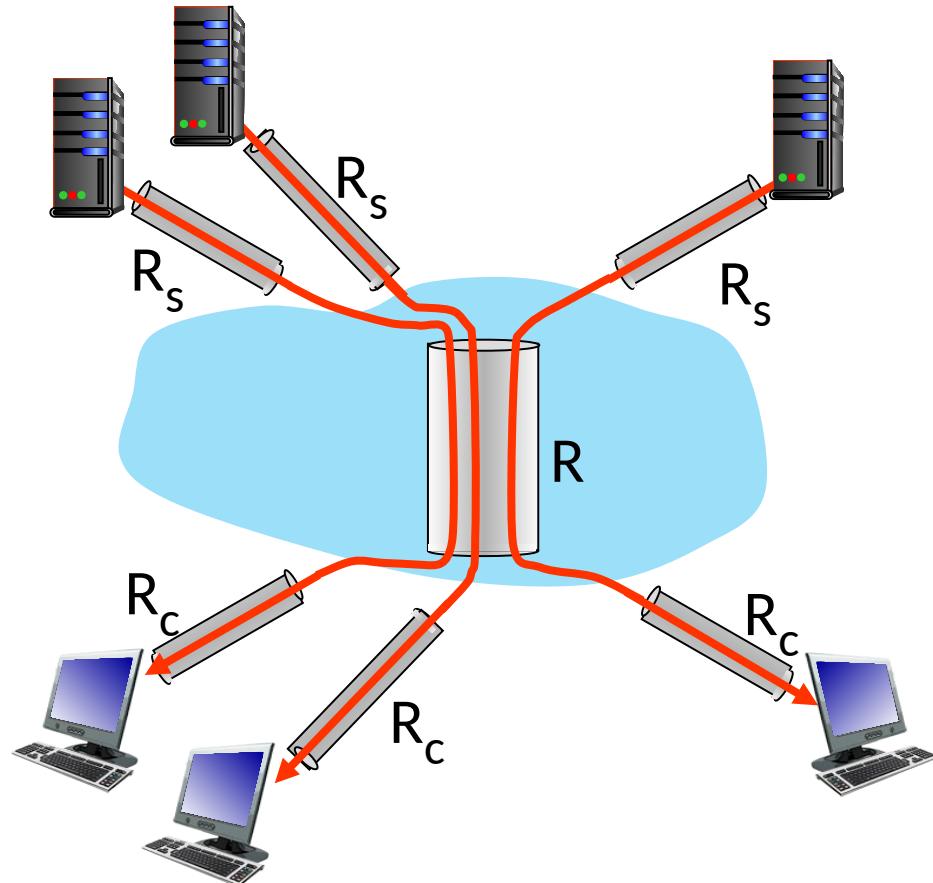
Real Internet delays and routes

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

		3 delay measurements from gaia.cs.umass.edu to cs-gw.cs.umass.edu			
1	cs-gw (128.119.240.254)	1 ms	1 ms	2 ms	
2	border1-rt-fa5-1-0.gw.umass.edu (128.119.3.145)	1 ms	1 ms	2 ms	3 delay measurements to border1-rt-fa5-1-0.gw.umass.edu
3	cht-vbns.gw.umass.edu (128.119.3.130)	6 ms	5 ms	5 ms	
4	jn1-at1-0-0-19.wor.vbns.net (204.147.132.129)	16 ms	11 ms	13 ms	
5	jn1-so7-0-0-0.wae.vbns.net (204.147.136.136)	21 ms	18 ms	18 ms	
6	abilene-vbns.abilene.ucaid.edu (198.32.11.9)	22 ms	18 ms	22 ms	
7	nycm-wash.abilene.ucaid.edu (198.32.8.46)	22 ms	22 ms	22 ms	
8	62.40.103.253 (62.40.103.253)	104 ms	109 ms	106 ms	trans-oceanic link
9	de2-1.de1.de.geant.net (62.40.96.129)	109 ms	102 ms	104 ms	
10	de.fr1.fr.geant.net (62.40.96.50)	113 ms	121 ms	114 ms	
11	renater-gw.fr1.fr.geant.net (62.40.103.54)	112 ms	114 ms	112 ms	looks like delays decrease! Why?
12	nio-n2.cssi.renater.fr (193.51.206.13)	111 ms	114 ms	116 ms	
13	nice.cssi.renater.fr (195.220.98.102)	123 ms	125 ms	124 ms	
14	r3t2-nice.cssi.renater.fr (195.220.98.110)	126 ms	126 ms	124 ms	
15	eurecom-valbonne.r3t2.ft.net (193.48.50.54)	135 ms	128 ms	133 ms	
16	194.214.211.25 (194.214.211.25)	126 ms	128 ms	126 ms	
17	***				
18	***	* means no response (probe lost, router not replying)			
19	fantasia.eurecom.fr (193.55.113.142)	132 ms	128 ms	136 ms	

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

Throughput: network scenario



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

- per-connection end-end throughput:
 $\min(R_c, R_s, R/3)$
- 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/

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



Network security

- Internet not originally designed with (much) security in mind
 - *original vision*: “a group of mutually trusting users attached to a transparent network” 
 - Internet protocol designers playing “catch-up”
 - security considerations in all layers!
- We now need to think about:
 - how bad guys can attack computer networks
 - how we can defend networks against attacks
 - how to design architectures that are immune to attacks

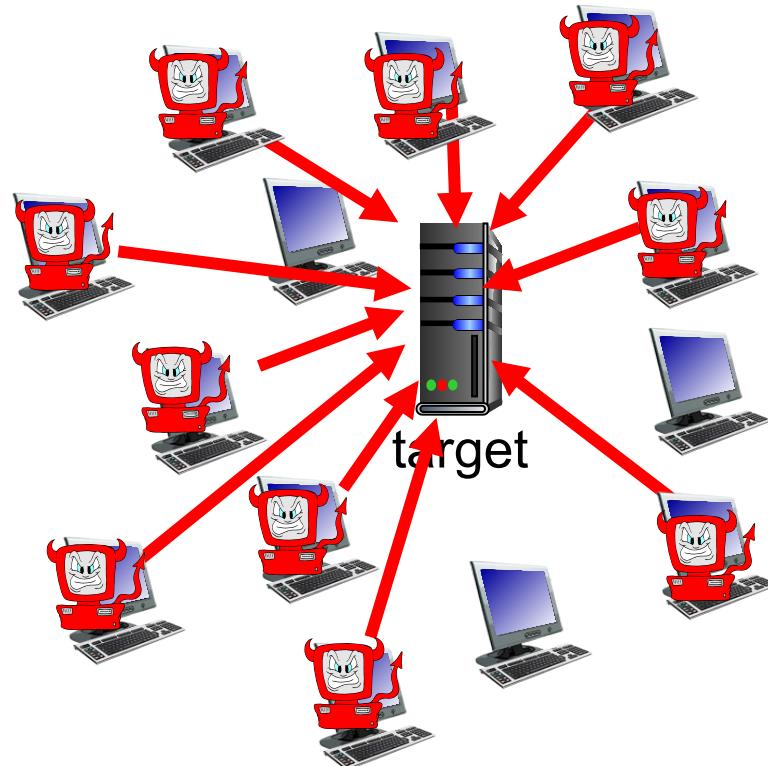
Bad guys: put malware into hosts via Internet

- **malware** can get in host from:
 - *virus*: self-replicating infection targeting a software or system vulnerability
 - *worm*: self-replicating infection by passively receiving object that gets itself executed
 - *trojan*: tricks a user into running malicious software by receiving/executing object (e.g., e-mail attachment)
- **keyloggers** can record keystrokes, **spyware** logs web sites visited, upload info to collection site
- infected host can be enrolled in **botnet**, used for spam, DDoS attacks and others...

Bad guys: denial of service

Denial of Service (DoS): attackers make resources (server, bandwidth) unavailable to legitimate traffic by overwhelming resource with bogus traffic

1. select target
2. break into hosts
around the network
(see botnet)
3. send packets to target
from compromised
hosts



Lines of defense:

- **authentication**: proving you are who you say you are
 - cellular networks provides hardware identity via SIM card; no such hardware assist in traditional Internet
- **confidentiality**: via encryption
- **integrity checks**: digital signatures prevent/detect tampering
- **access restrictions**: password-protected VPNs
- **firewalls**: specialized “middleboxes” in access and core networks:
 - off-by-default: filter incoming packets to restrict senders, receivers, applications
 - detecting/reacting to DOS attacks

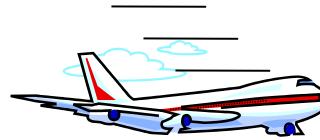
... lots more on security (throughout, Chapter 8)

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



Example: organization of air travel



end-to-end transfer of person plus baggage

ticket (purchase)

baggage (check)

gates (load)

runway takeoff

airplane routing

ticket (complain)

baggage (claim)

gates (unload)

runway landing

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

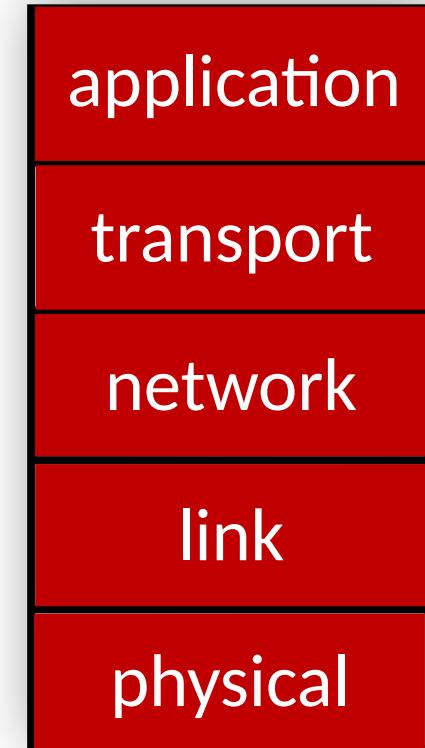
Why layering?

Approach to designing/discussing complex systems:

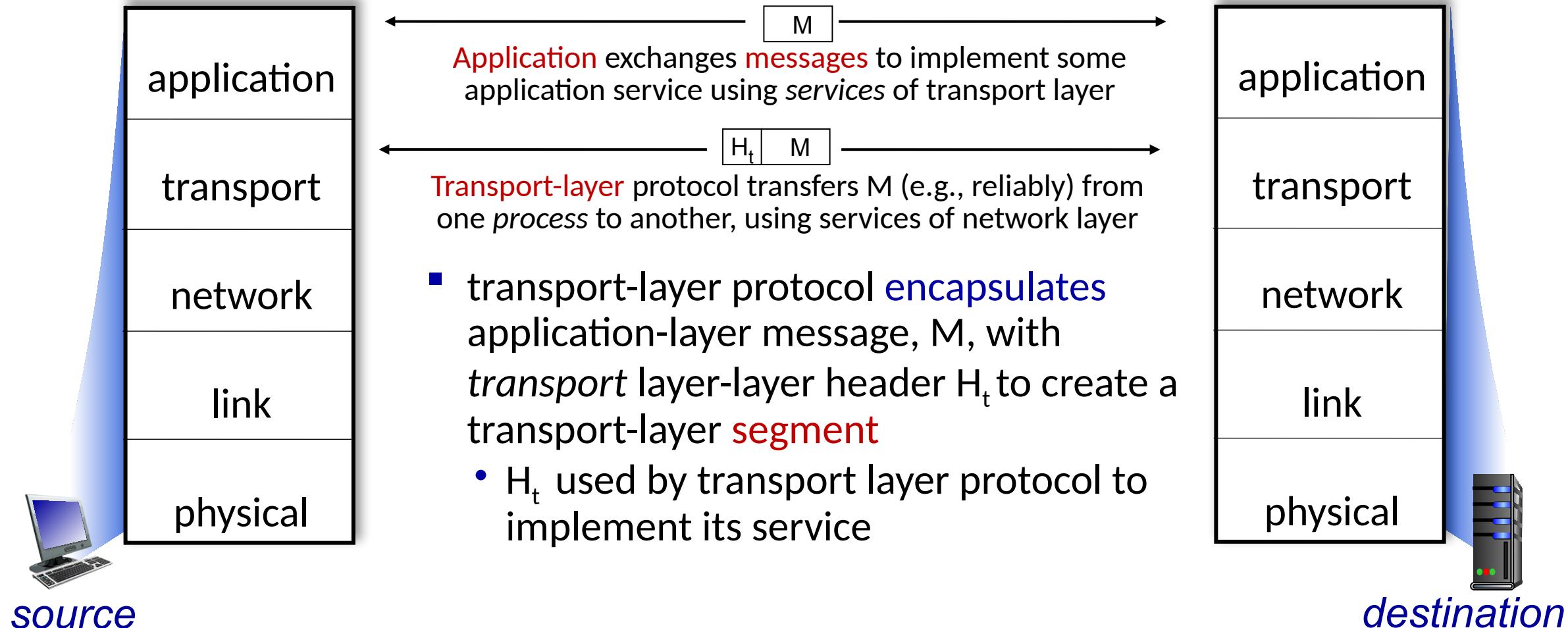
- explicit structure allows identification, relationship of system's pieces
 - layered *reference model* for discussion
- modularization eases maintenance, updating of system
 - change in layer's service *implementation*: transparent to rest of system
 - e.g., change in gate procedure doesn't affect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

Layered Internet protocol stack

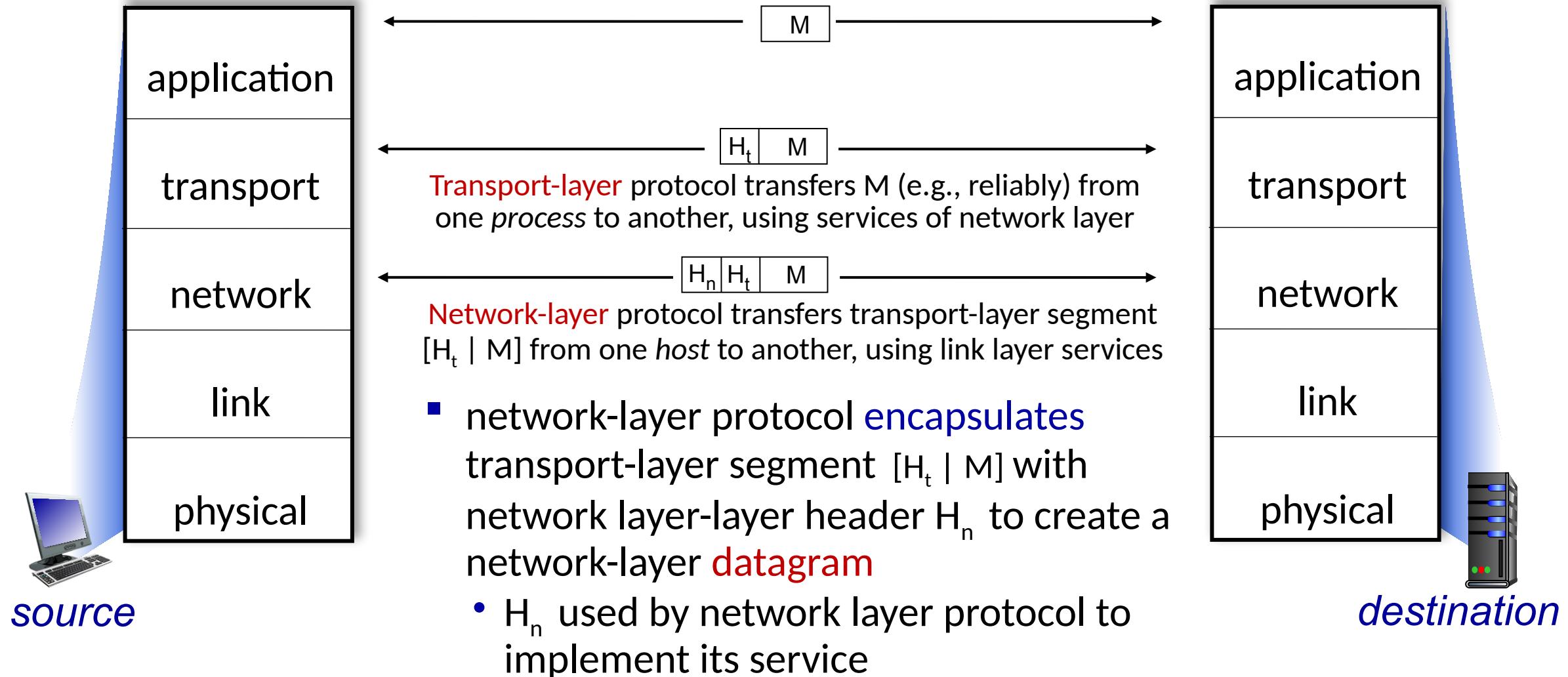
- *application*: supporting network applications
 - HTTP, IMAP, SMTP, DNS
- *transport*: process-process data transfer
 - TCP, UDP
- *network*: routing of datagrams from source to destination
 - IP, routing protocols
- *link*: data transfer between neighboring network elements
 - Ethernet, 802.11 (WiFi), PPP
- *physical*: bits “on the wire”



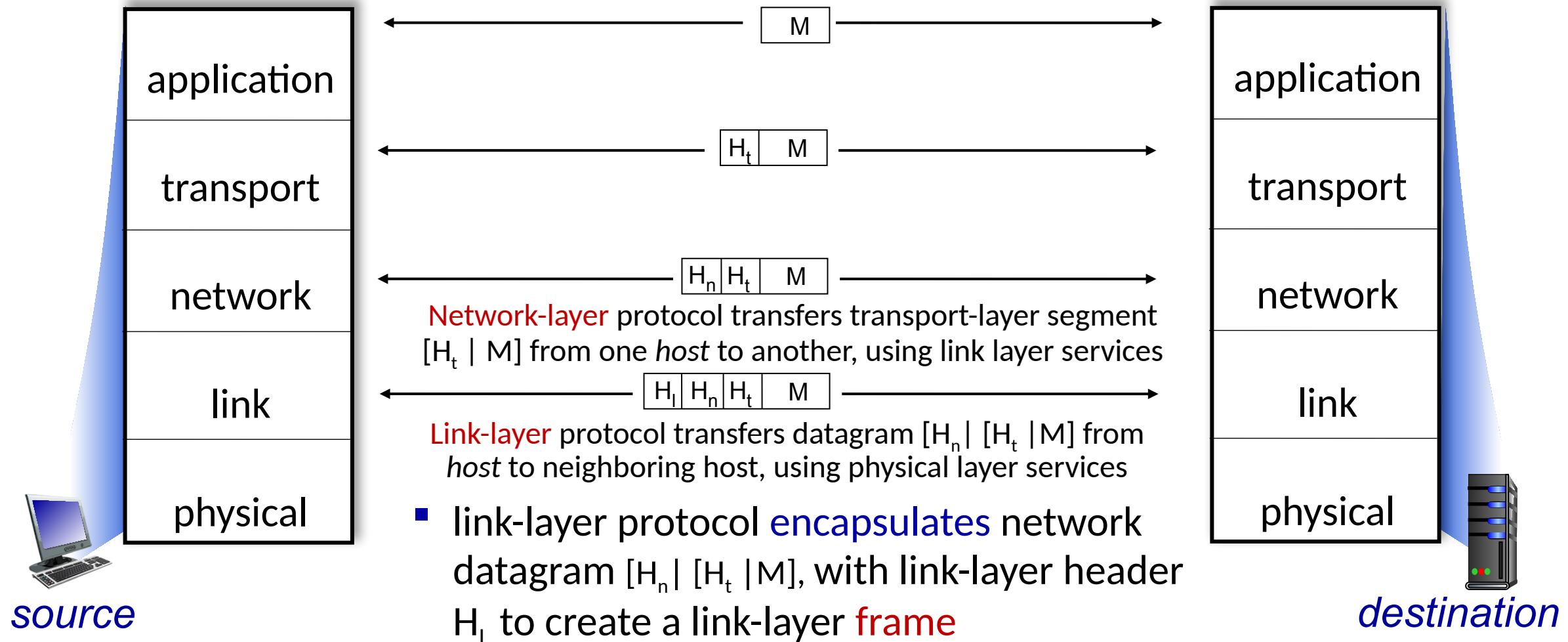
Services, Layering and Encapsulation



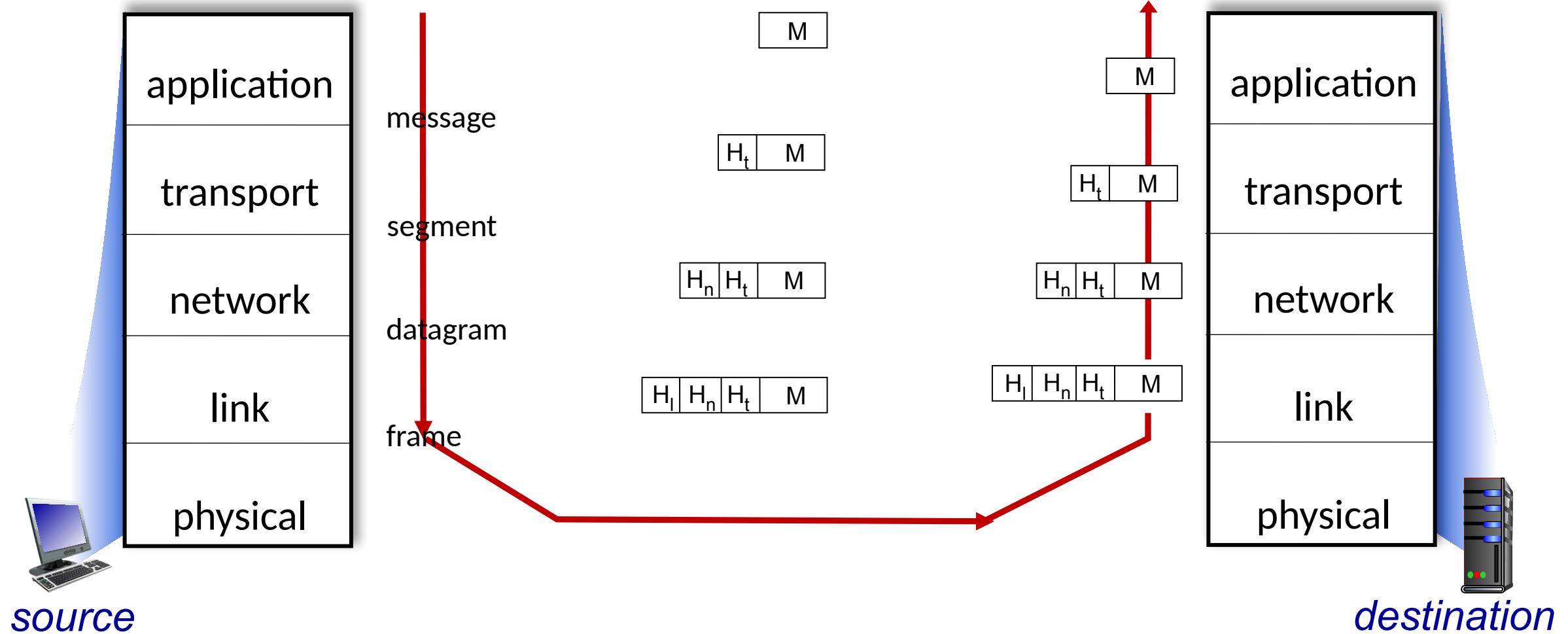
Services, Layering and Encapsulation



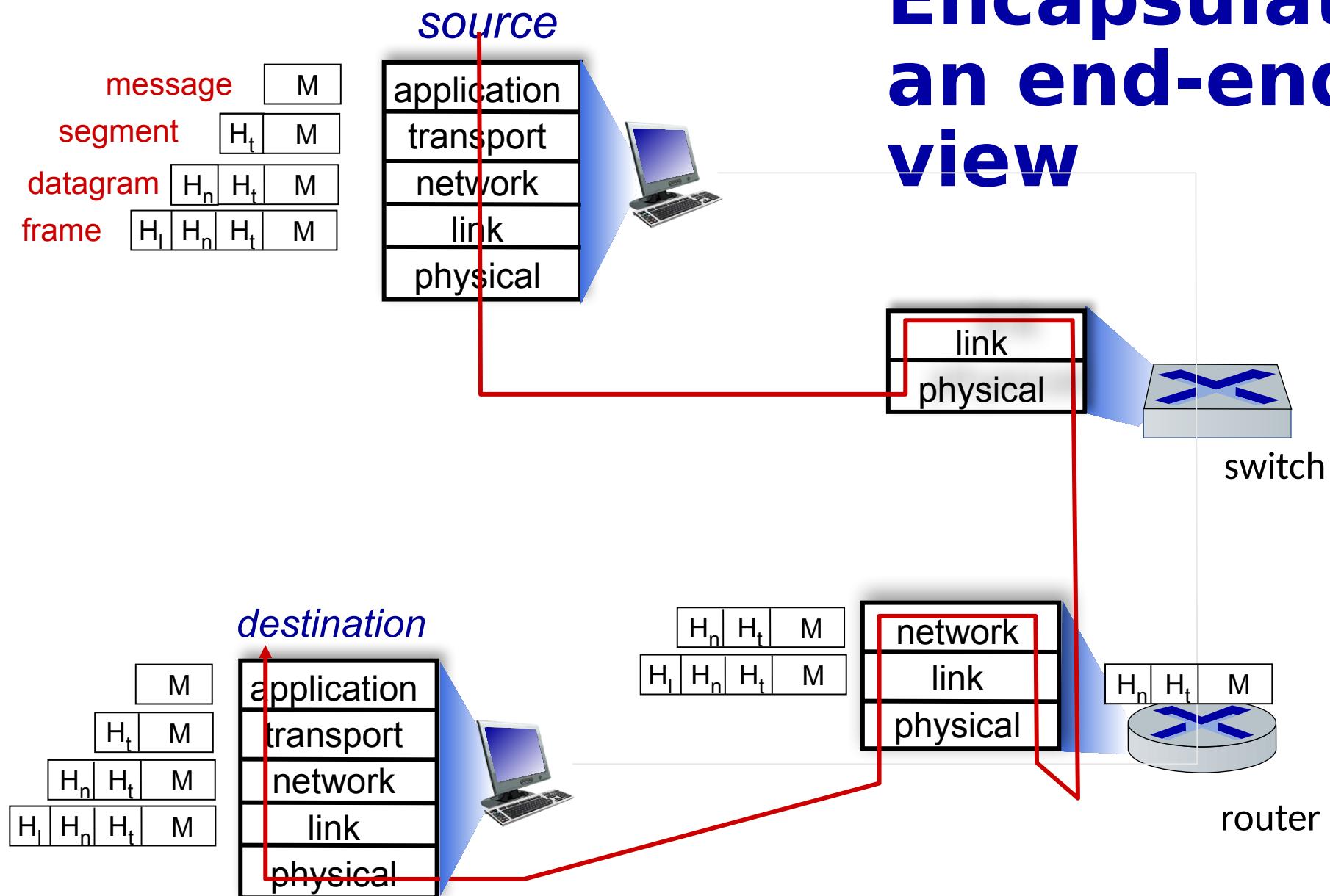
Services, Layering and Encapsulation



Services, Layering and Encapsulation



Encapsulation: an end-end view



Chapter 1: roadmap

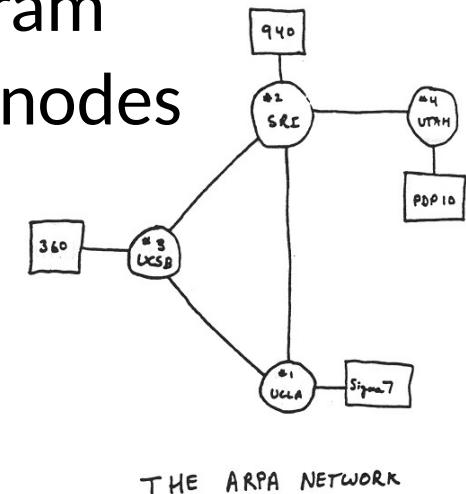
- What *is* the Internet?
- What *is* a protocol?
- Network edge: hosts, access network, physical media
- Network core: packet/circuit switching, internet structure
- Performance: loss, delay, throughput
- Security
- Protocol layers, service models
- History



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

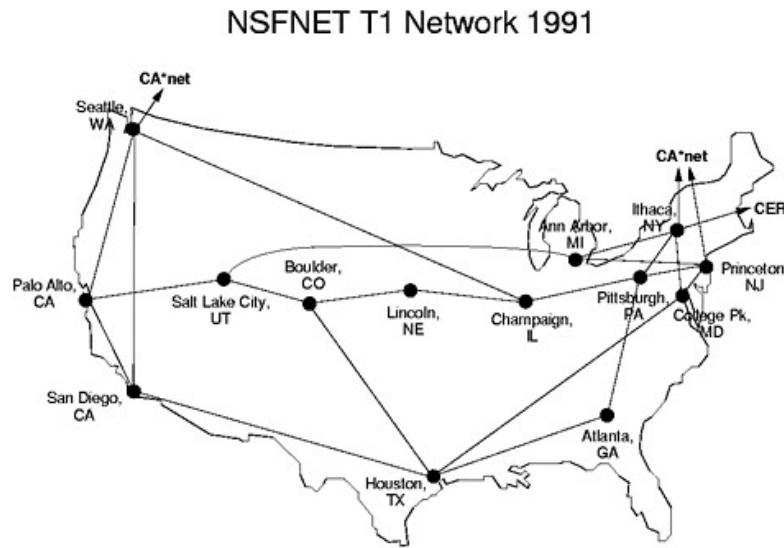
Cerf and Kahn's internetworking principles:

- 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

- Jan 1, 1983: birth of the Internet as ARPANET switched to TCP/IP protocols
- 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, 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: Timothy 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!

Additional Chapter 1 slides

Wireshark

