Chapter 1: Introduction - CNs and the Internet

Chapter goal:

- presenting a broad overview of computer networking and the Internet

- getting “feel”, “big picture”, and introduction to terminology

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:

- public Internet will be our principal vehicle along the course for discussing computer networks and their protocols

- 2 ways to explain

* a “nuts and bolts” view (hardware and software components)

* a services view: networking infrastucture, provides services to distributed apps

The Internet:

- a computer network that interconnects billions of computing devices throughout the world

- computers, TVs, gaming consoles, thermostats, home security systems, home appliances, watches, eye glasses, cars, traffic control systems, and more *(hosts or end systems)* are being connected to the Internet

- end systems are connected together by a network of communication links and packet switches

The Internet: a "nuts and bolts" view

(hardware and software components)

Billions of connected computing *devices*:

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

Packet switches: forward packets (chunks of data)

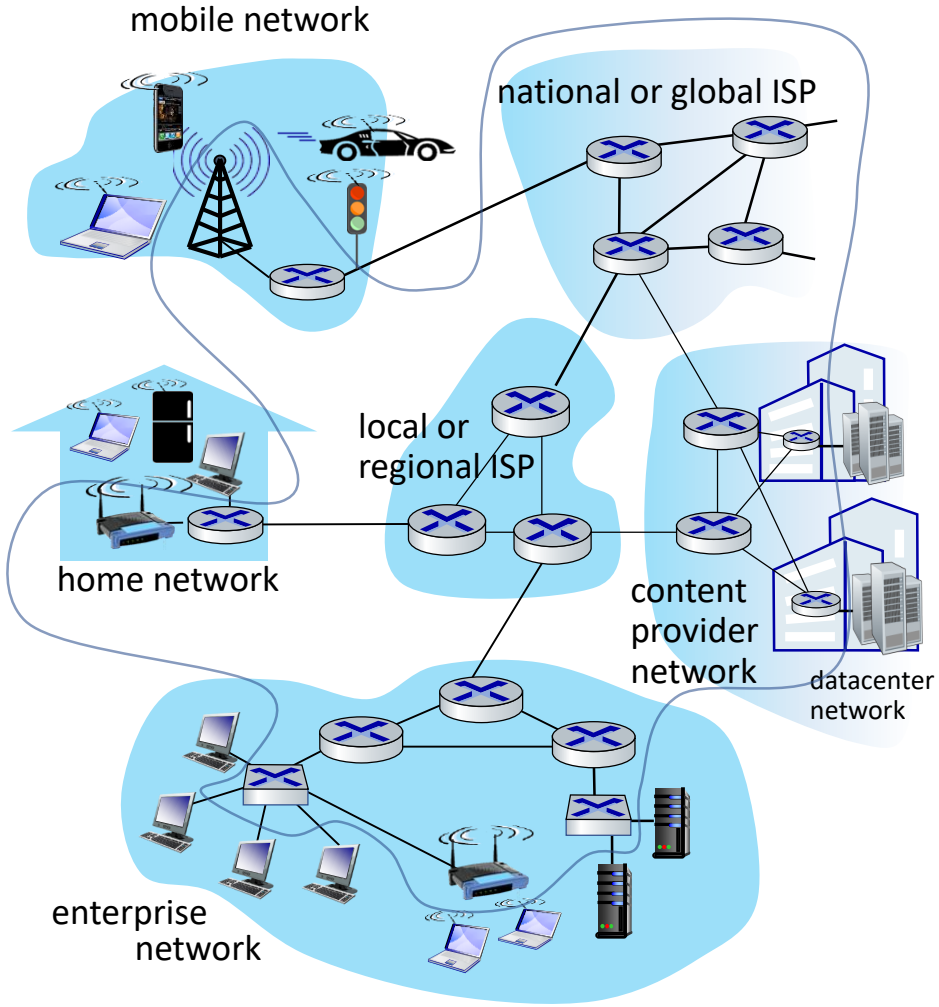
- *routers, switches*

Communication links

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

Networks

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



Internet-connected devices

Amazon Echo

Internet refrigerator

IP picture frame

Pacemaker & Monitor

Tweet-a-watt: monitor energy use

bikes

cars

Security Camera

Slingbox: remote control cable TV

Web-enabled toaster + weather forecaster

AR devices

scooters

Internet phones

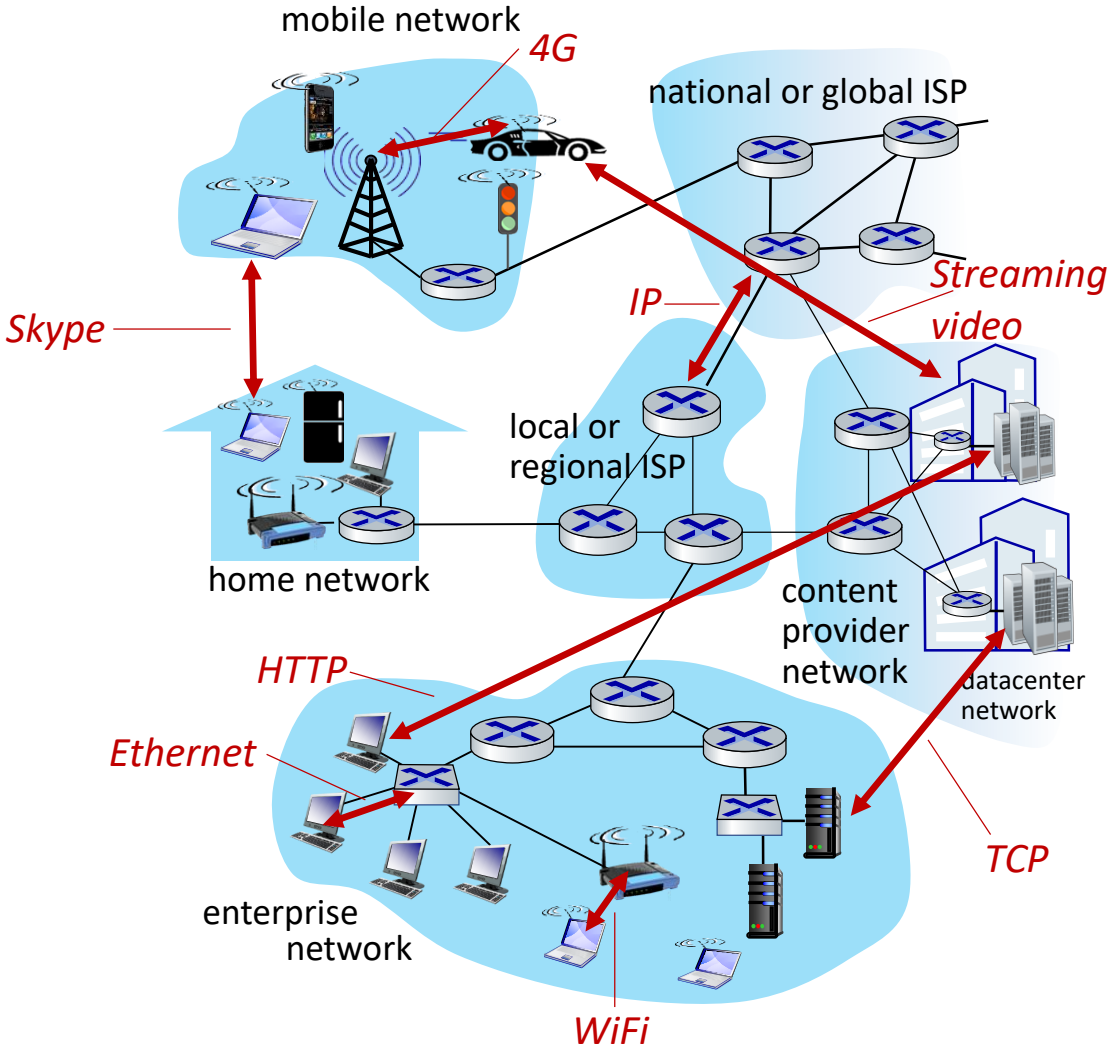
Gaming devices

sensorized, bed mattress

Fitbit

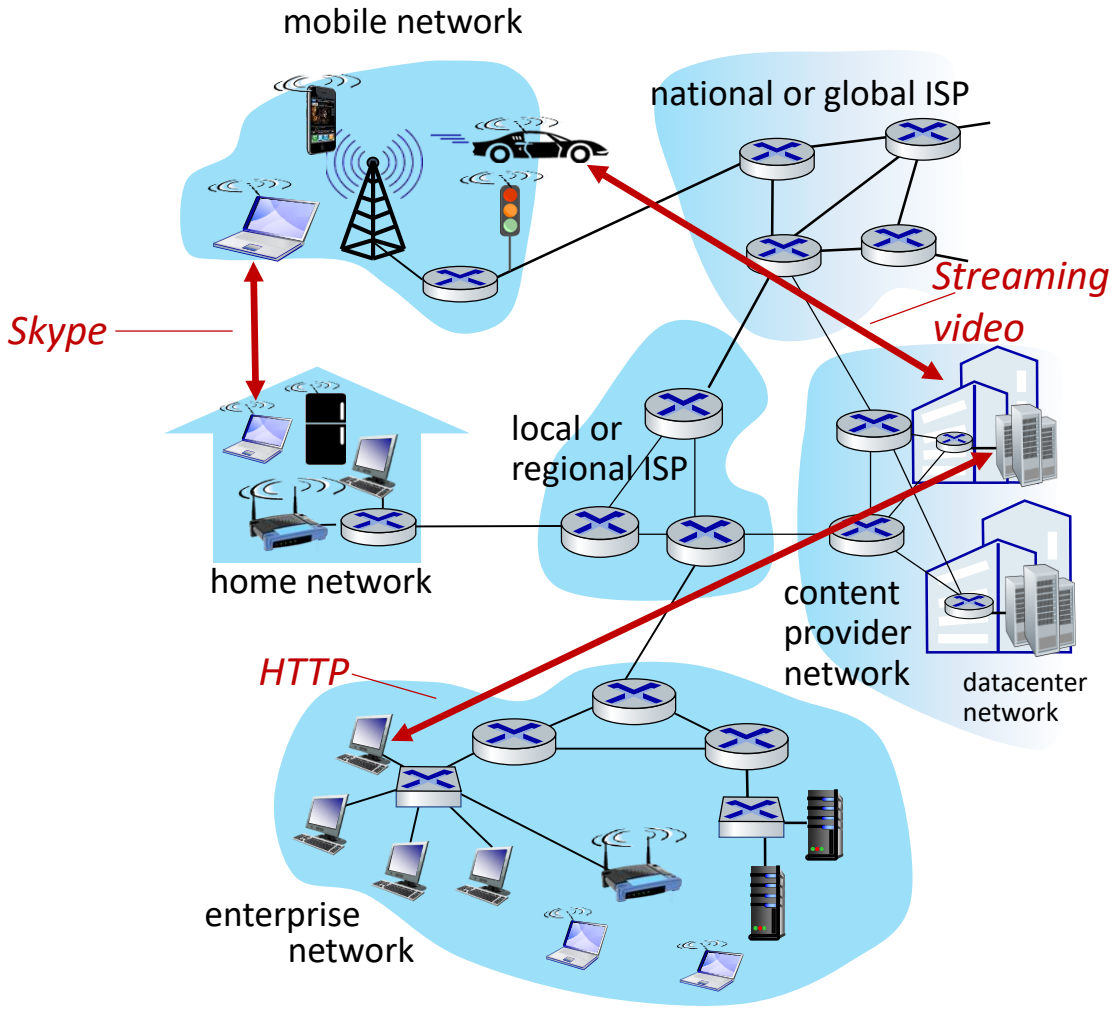
The Internet: a "nuts and bolts" view

- *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 standarts documents, 9000 docs)*
  - IETF: *Internet Engineering Task Force*



The Internet: a "services" view (2nd way to define/explain)

- *Infrastructure* that provides services to applications:
  - Web, streaming video, multimedia teleconferencing, email, games, e-commerce, social media, mobile apps, inter-connected appliances, …
- provides *programming interface (socket 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:

- initiate communications
- ability or inability to communicate

Rules for:

… specific messages sent

… specific actions taken when message received, or other events

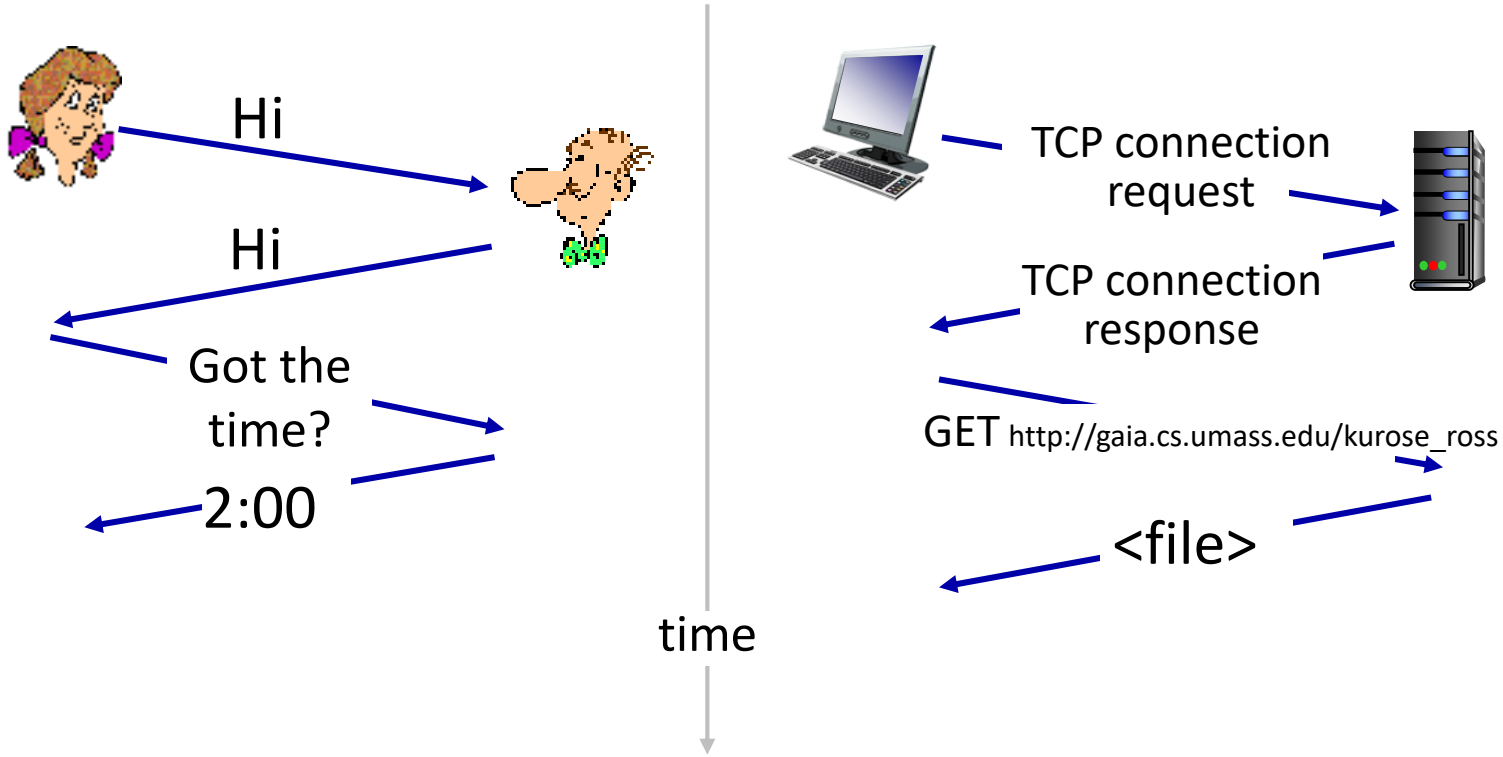
Network protocols (are like human protocols):

- 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 (simple analogy):



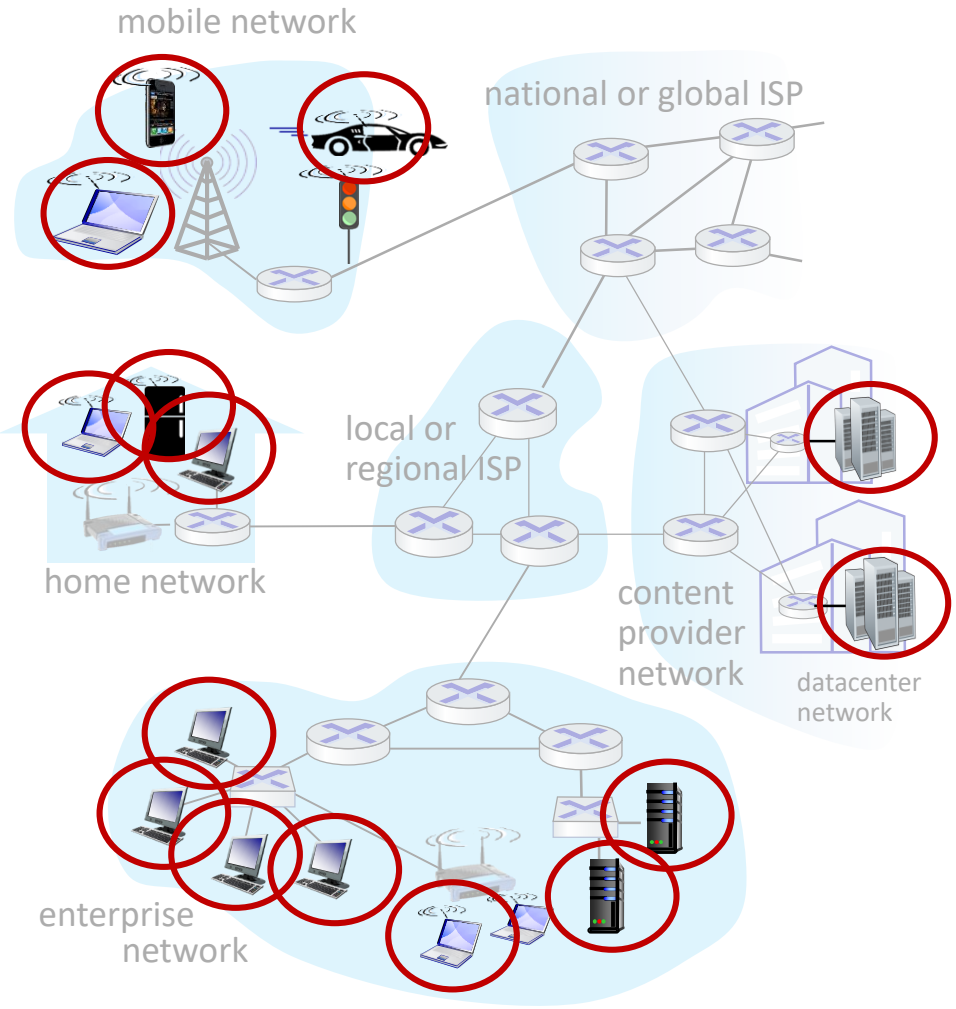
Chapter 1: roadmap

- 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 (end systems): clients and servers
- servers often in data centers



A closer look at Internet structure

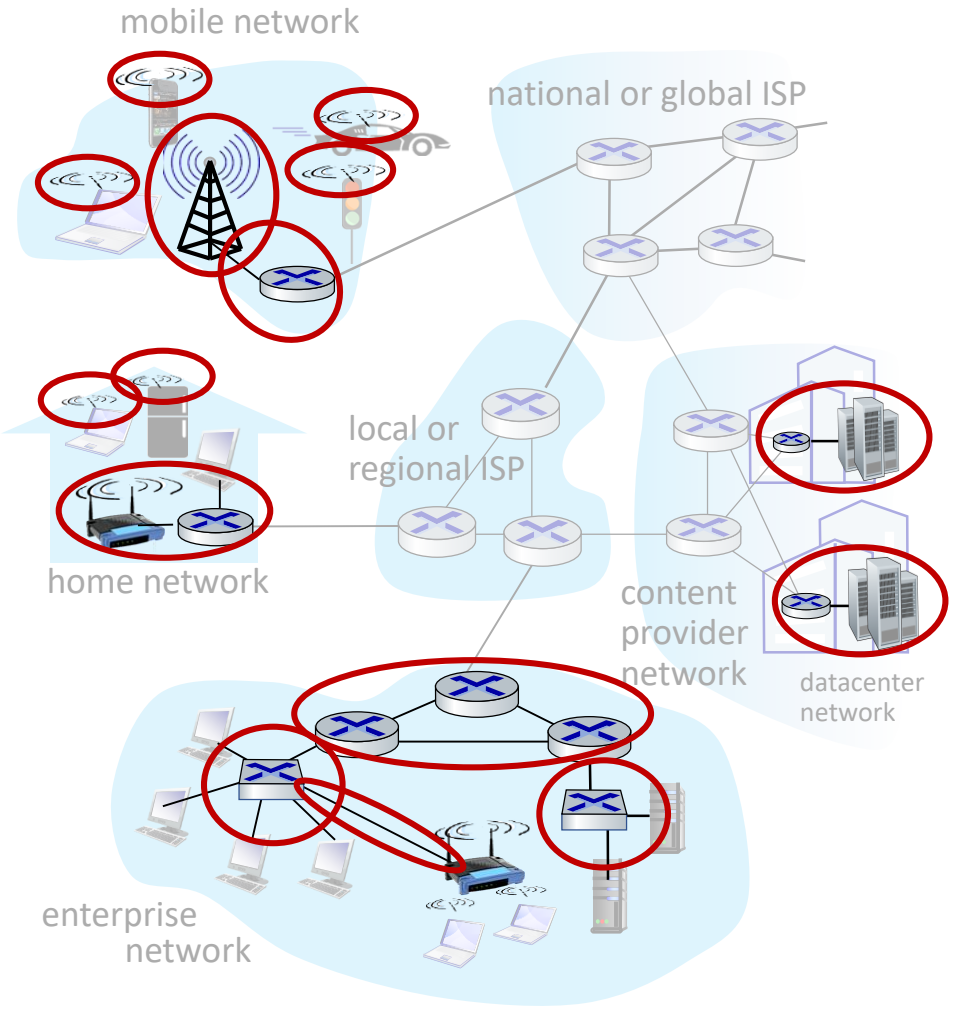
Network edge:

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

Access networks, physical media:

- wired, wireless communication links

physically connects an end system to the first router (edge router) on a path from the end system to any other distant end system.



A closer look at Internet structure

Network edge:

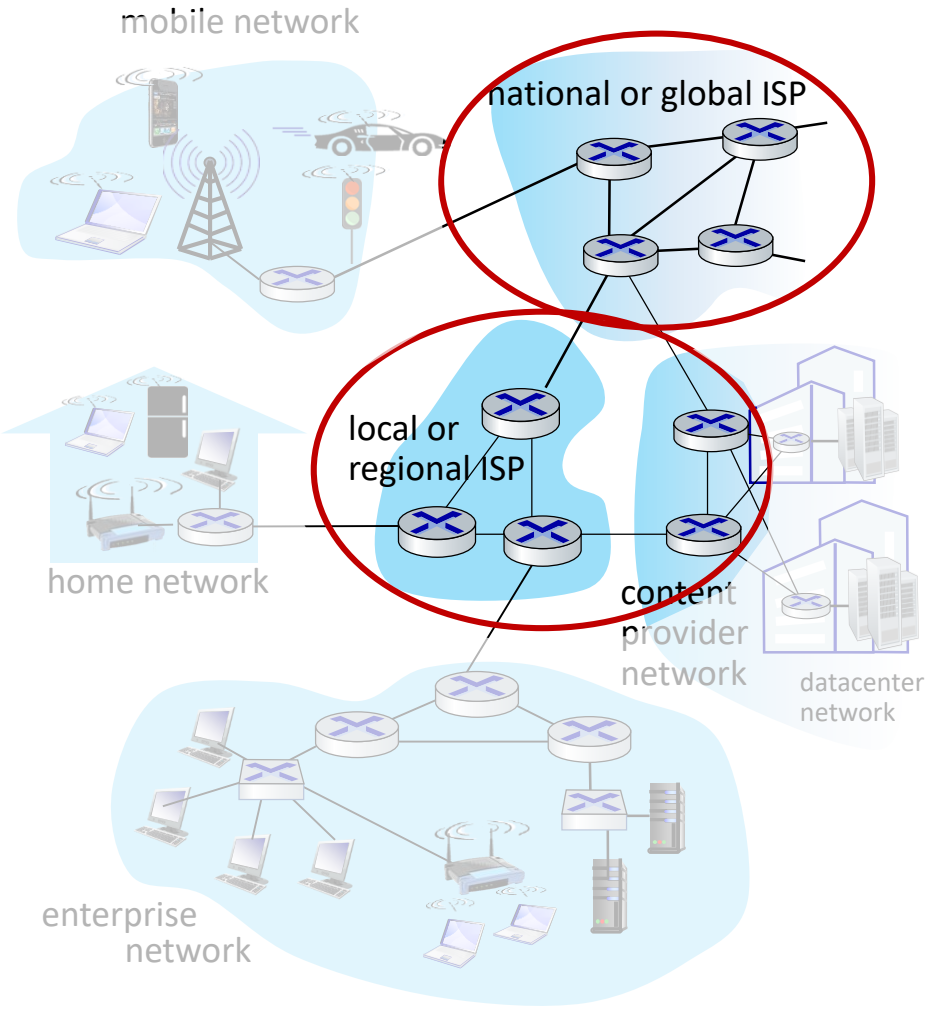
- hosts: clients and servers
- servers often in data centers

Access networks, physical media:

- wired, wireless communication links

Network core:

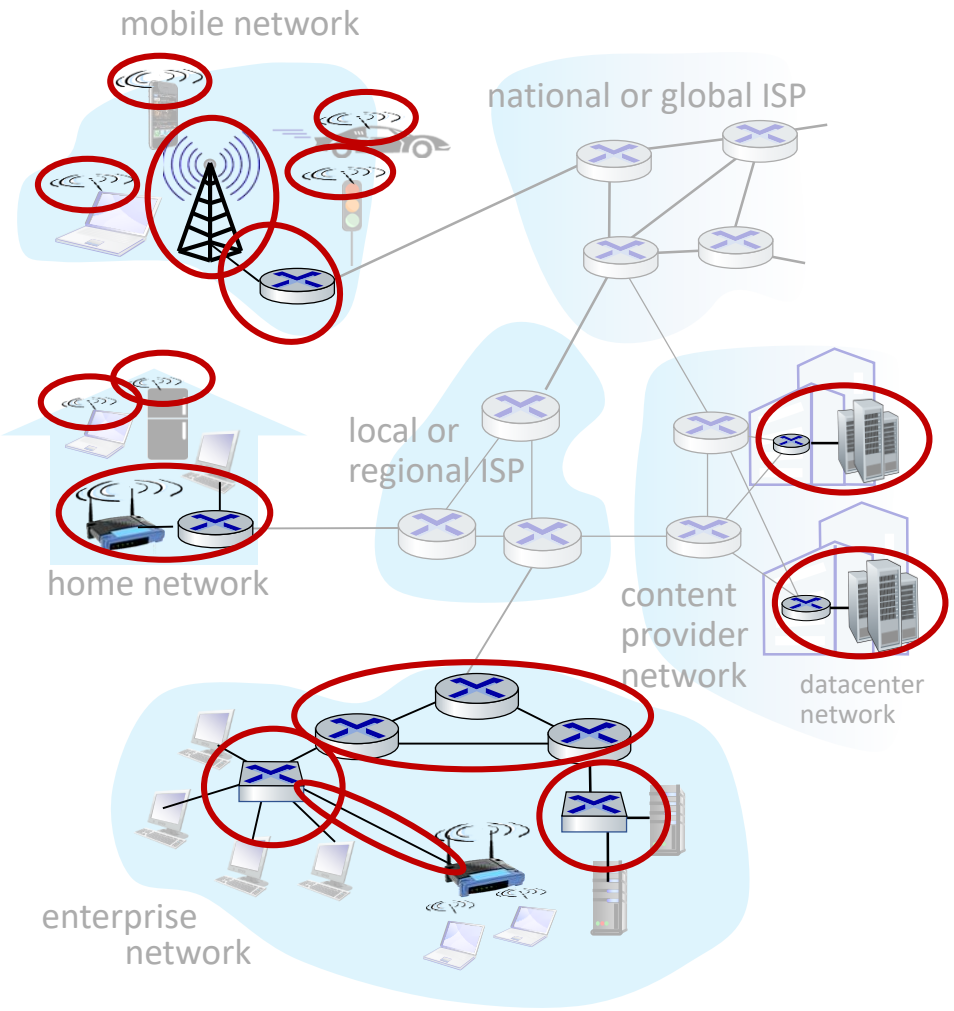
- interconnected routers
- network of networks



Access networks and physical media

Q: How to connect end systems to edge router?

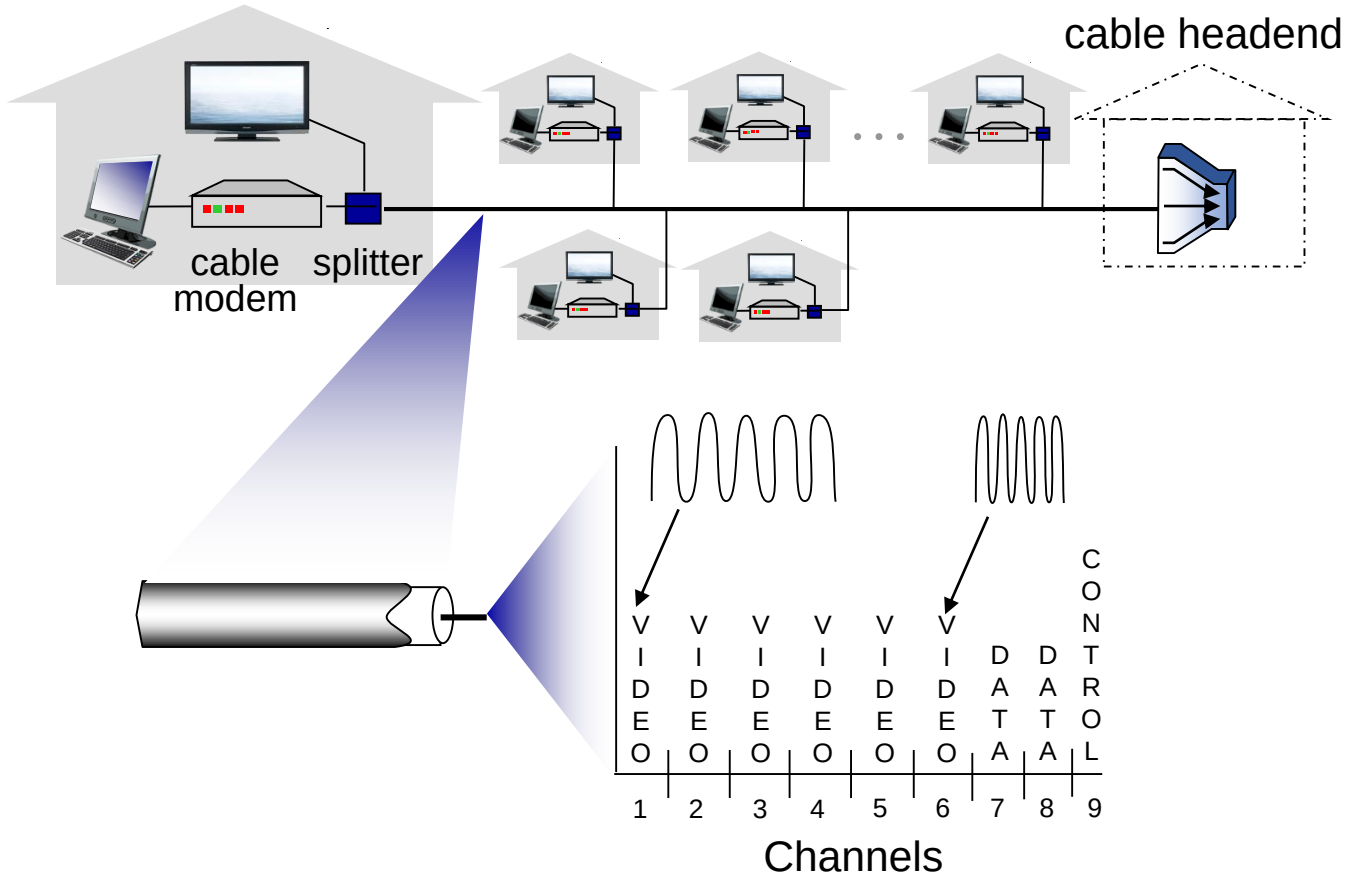
- residential access nets
- institutional access networks (school, company)
- mobile access networks (WiFi, 4G/5G)



Access networks: cable-based access

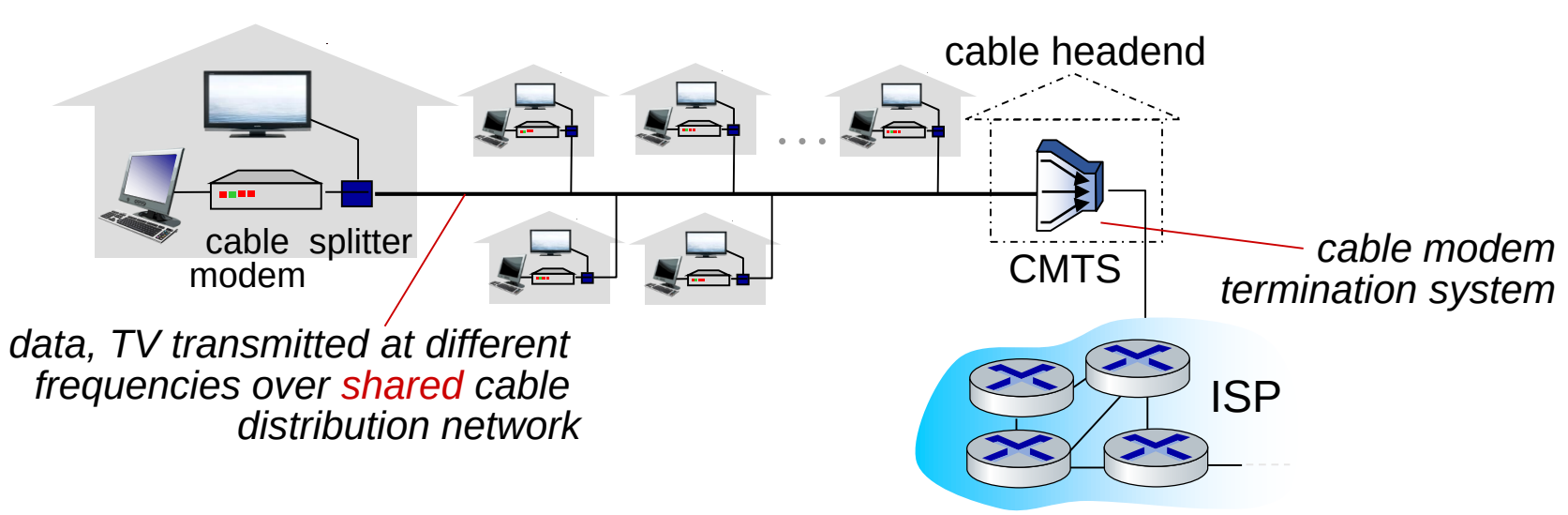
Residential or home access maybe done as cable, DSL, fiber, and wireless.

If we use the same link for different purposes as in DSL and cable connections, we need FDM.



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

Access networks: cable-based access



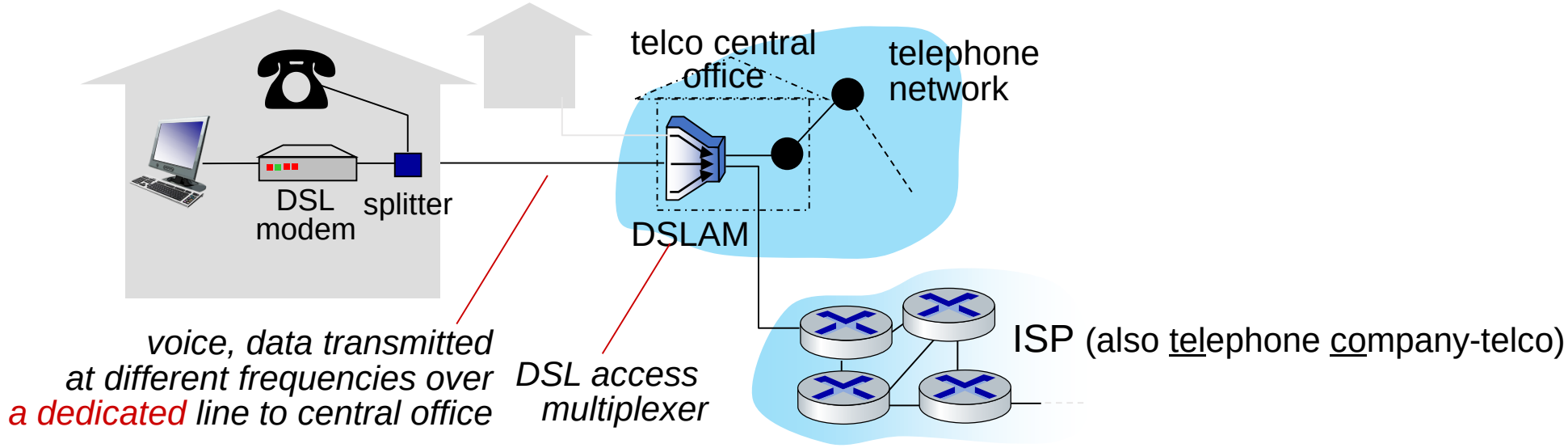
a hybrid structure, a network of cable and fiber

a shared broadcast medium Access

every packet sent by the headend travels downstream on every link to every home

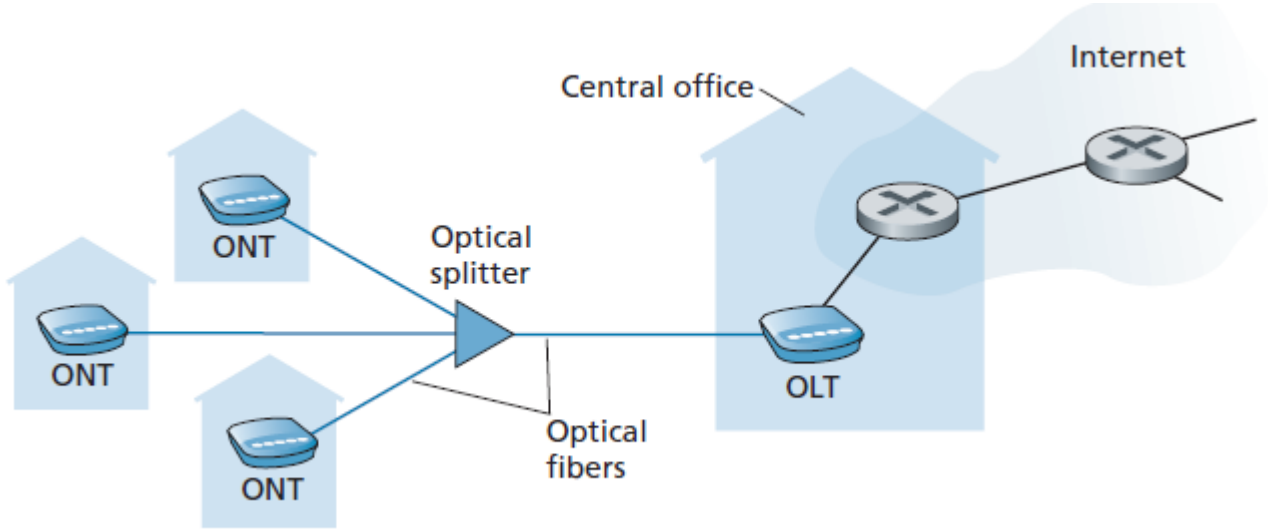
- cable Internet access makes use of the cable television company's existing cable tv infrastructure
- requires cable modems (an external device and connects to the home PC through an Ethernet port)
- HFC: hybrid fiber coax *(fiber from cable headend to junctions, coaxial from junctions to houses)*
  - asymmetric: up to 40 Mbps – 1.2 Gbps downstream transmission rate, 30-100 Mbps upstream transmission rate
- network of cable and 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 (Analog signals from houses are translated into digital format in the DSLAM)
- voice over DSL phone line goes to telephone net
- 24-52 Mbps dedicated downstream transmission rate
- 3.5-16 Mbps dedicated upstream transmission rate
- downstream and upstream rates are different, asymmetric
- on customer side there is a splitter, separates the data and telephone signals arriving to the home and forwards the data signal to the DSL modem
- DSLAM separates the data and phone signals and sends the data into the Internet, so it is like a splitter for telco

Access networks: fiber to the home (FTTH)

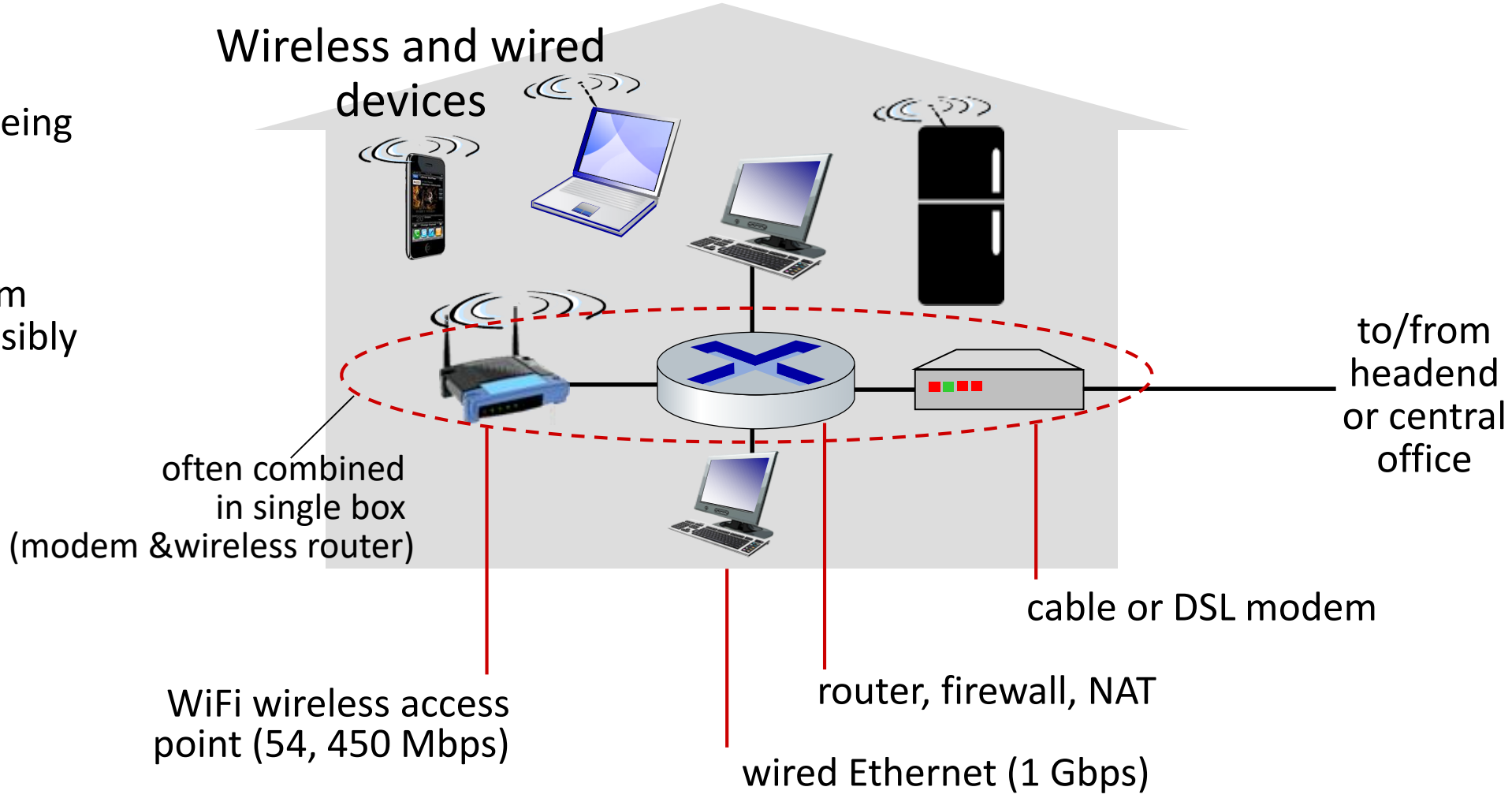


- each home has an *optical network terminator (ONT)*, which is connected by dedicated optical fiber to a neighborhood splitter
- users connect a home router to the ONT
- splitter combines a number of homes onto a single, shared optical fiber, which connects to an *optical line terminator (OLT)* in the telco's central office
- can potentially provide Internet access rates in the gigabits per second range

Access networks: home networks

wireless connection is being deployed today

a WiFi wireless router is connected to the modem (router and modem possibly bundled together)



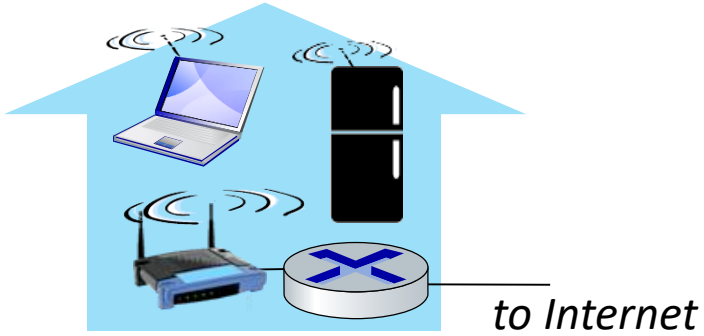
Wireless access networks

Shared *wireless* access network connects end system to router

- via base station aka “access point”

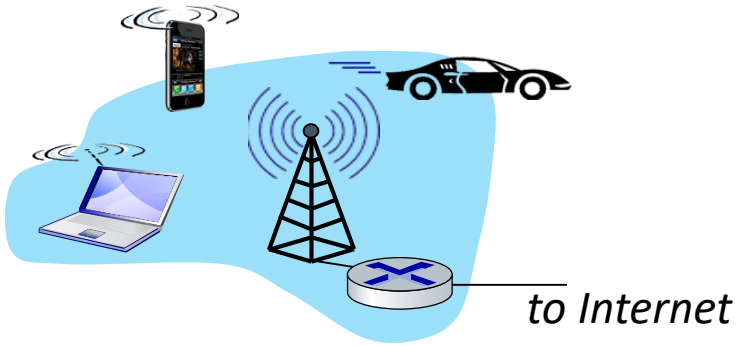
Wireless local area networks (WLANs)

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

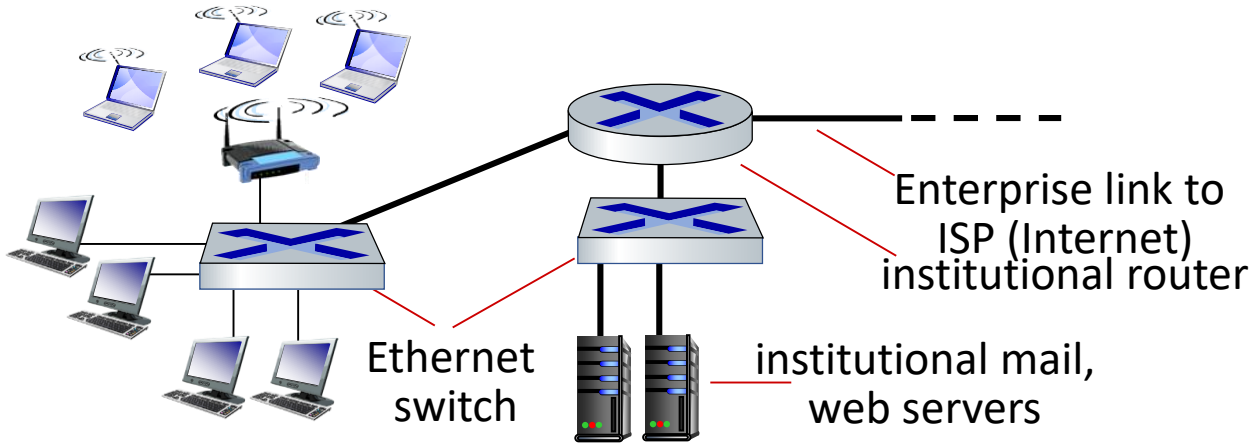


Wide-area cellular access networks

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



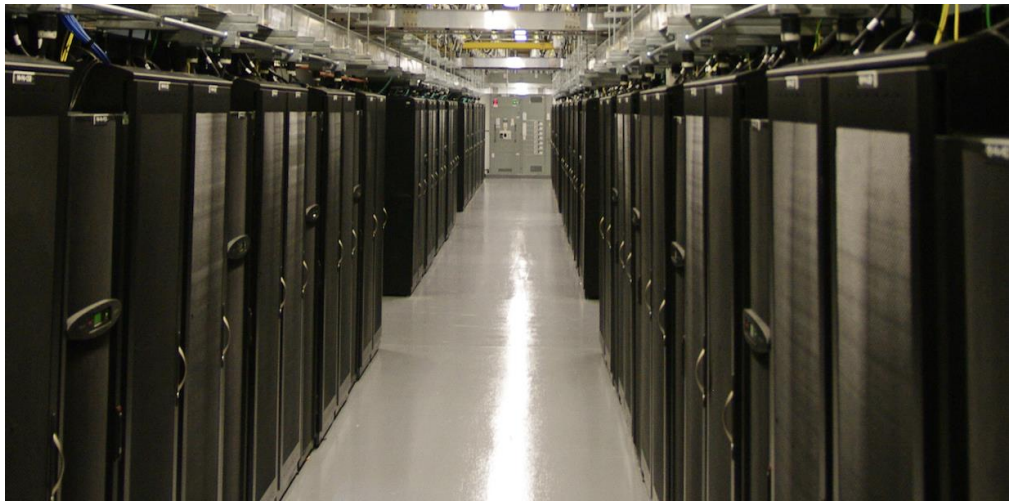
Access networks: enterprise networks



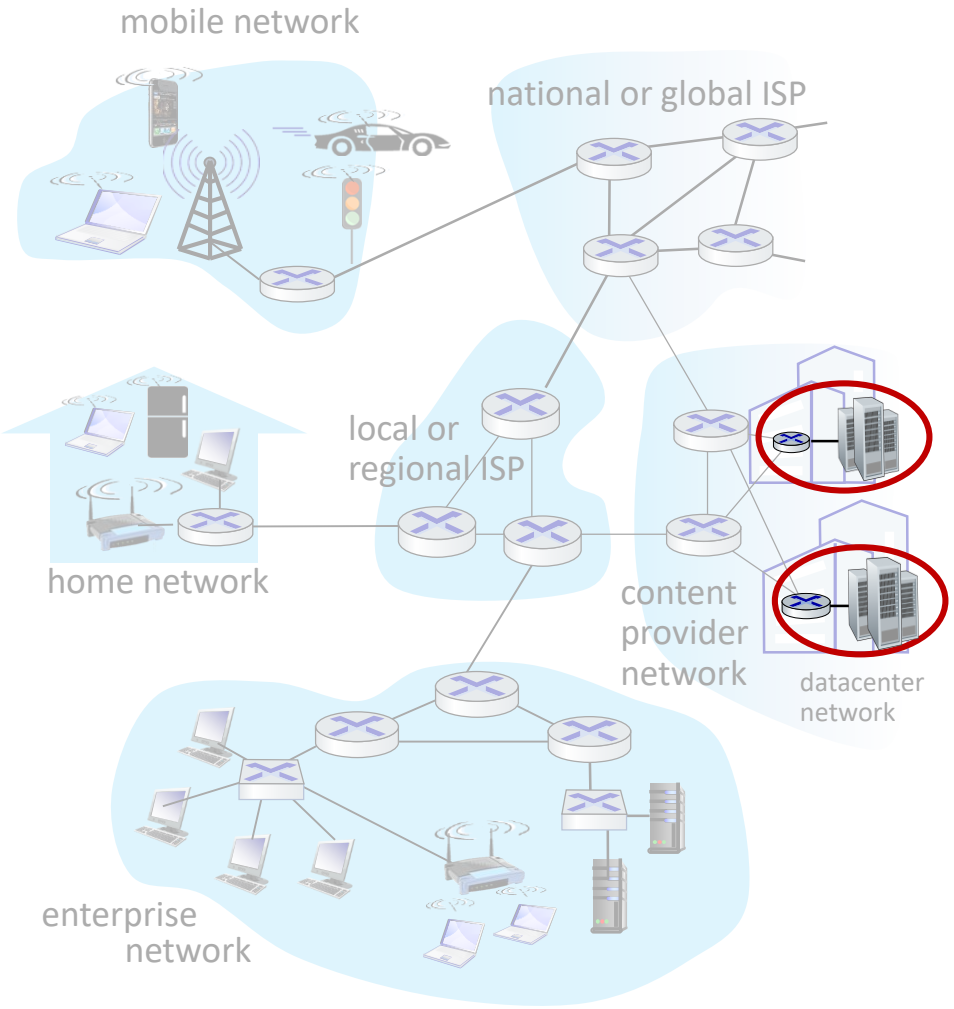
- many types of LAN technologies, Ethernet is the most prevalent access technology in corporate, university, and home networks
- mix of wired, wireless link technologies, connecting a mix of switches and routers
- wireless users transmit/receive packets to/from an access point that is connected into the enterprise's network (most likely using wired Ethernet), which in turn is connected to the wired Internet
- 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)

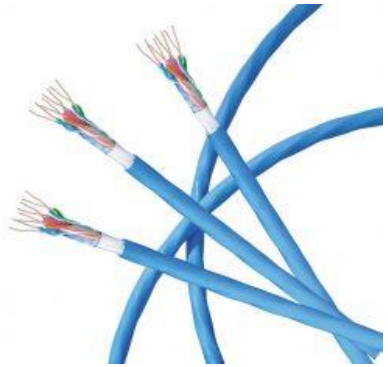


Links: physical media

- 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

Twisted pair (TP)

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



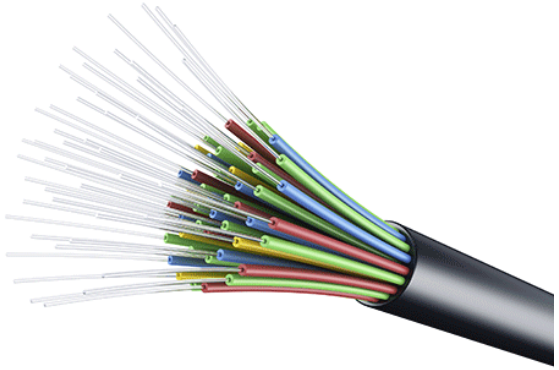
Links: physical media

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

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:

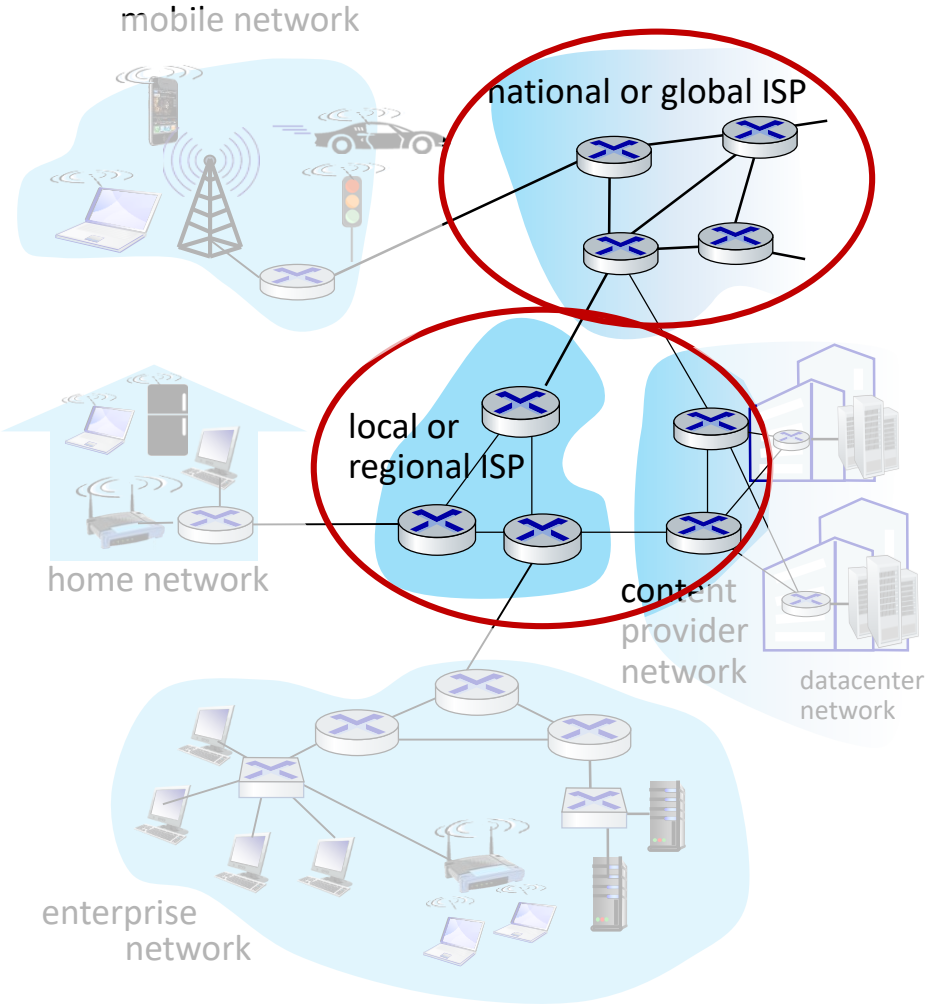
- Wireless LAN (WiFi)
  - 10-100's Mbps; 10's of meters
- Wide-area (e.g., 4G cellular)
  - 10's Mbps over ~10 Km
- Bluetooth: cable replacement
  - short distances, limited rates
- Terrestrial microwave
  - point-to-point; 45 Mbps channels
- Satellite
  - up to 45 Mbps per channel
  - 270 msec end-end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

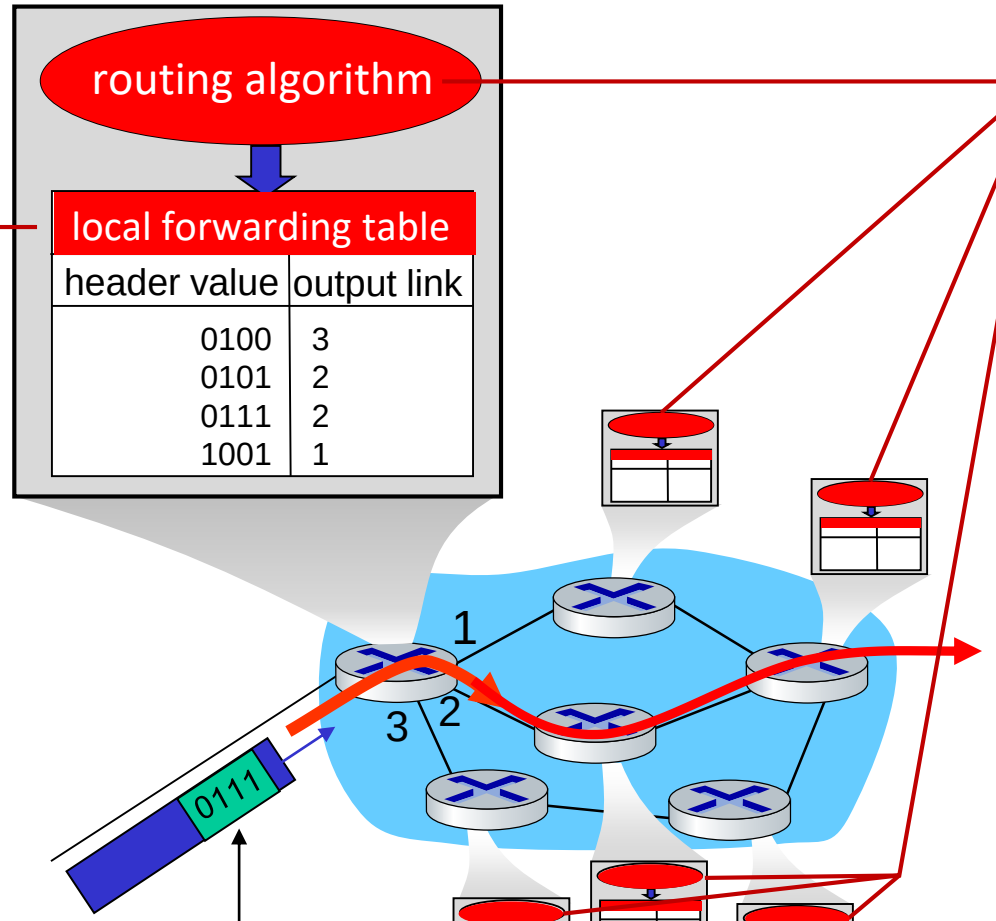
- mesh of packet switches, links, and interconnected routers
- packet-switching: hosts break application-layer messages into *packets*
- packets travel through communication links and packet switches, these are mostly routers and link-layer switches
  - network forwards packets from one router to the next, across links on path from source to destination
- a packet of L bits, a link with transmission rate R bits/sec -> time to transmit the packet is L / R seconds



Two key network-core functions

Forwarding:

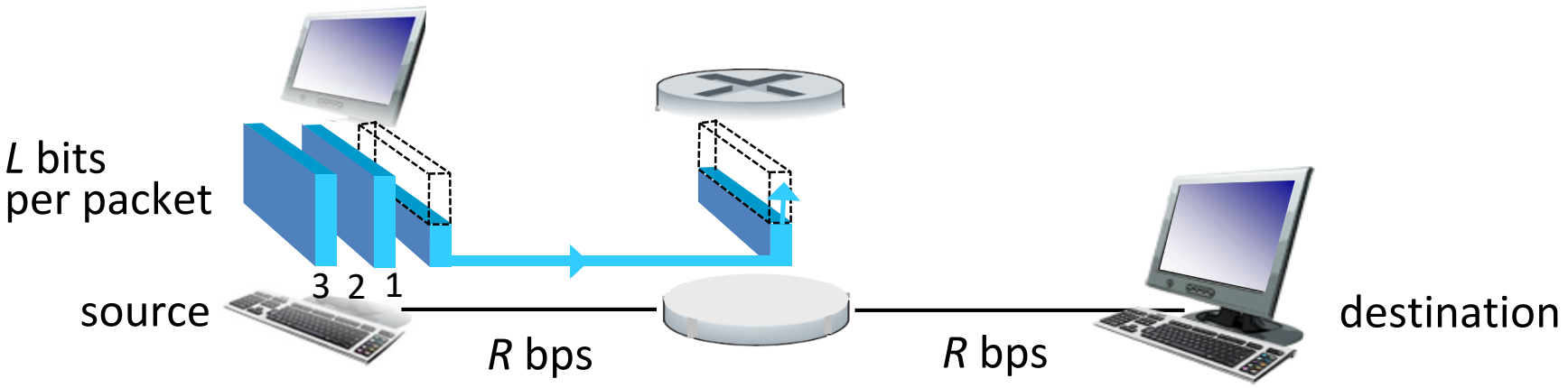
- aka "switching"
- *local* action: move arriving packets from router's input link to appropriate router output (outbound) link



Routing:

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

Packet-switching: store-and-forward



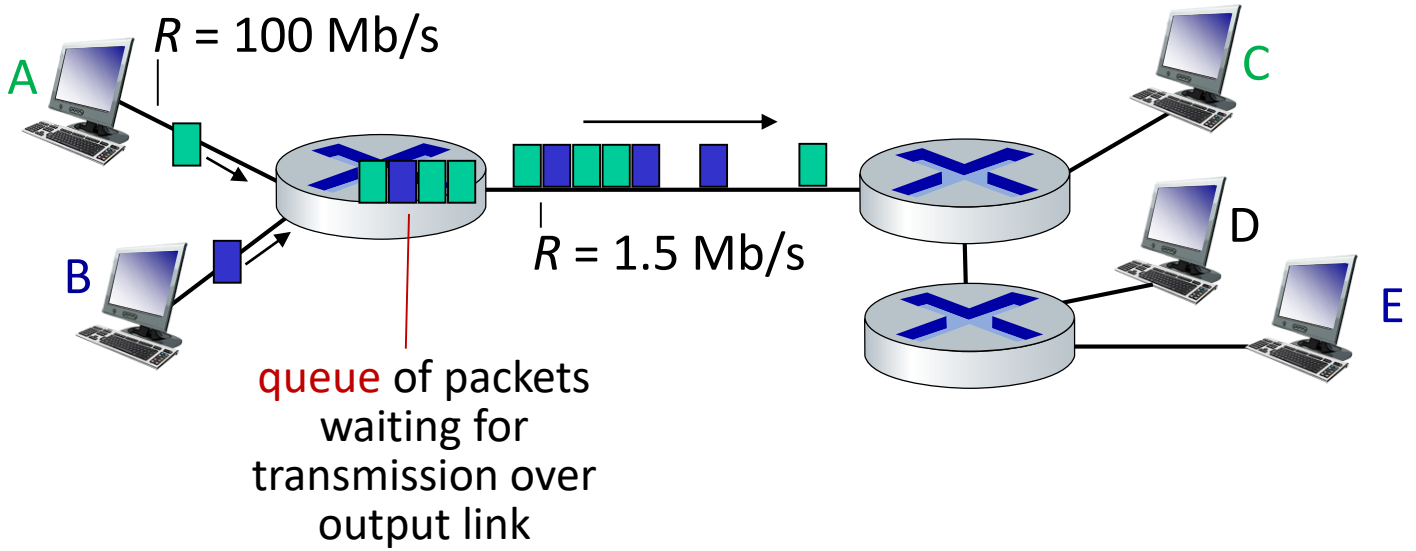
One-hop numerical example:

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

- 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

* source begins to transmit at time 0; at time L/R seconds, the source has transmitted the entire packet, and the entire packet has been received and stored at the router.
* at time L/R seconds, it can begin to transmit the packet onto the outbound link towards the destination; at time 2L/R, the router has transmitted the entire packet, and the entire packet has been received by the destination. Thus, the total delay is 2L/R. Here we assume that there is no propagation delay (we have the speed of light to travel the bits)
* we need 4L/R time for all of the three packets arrived at the destination.

Packet-switching: queueing

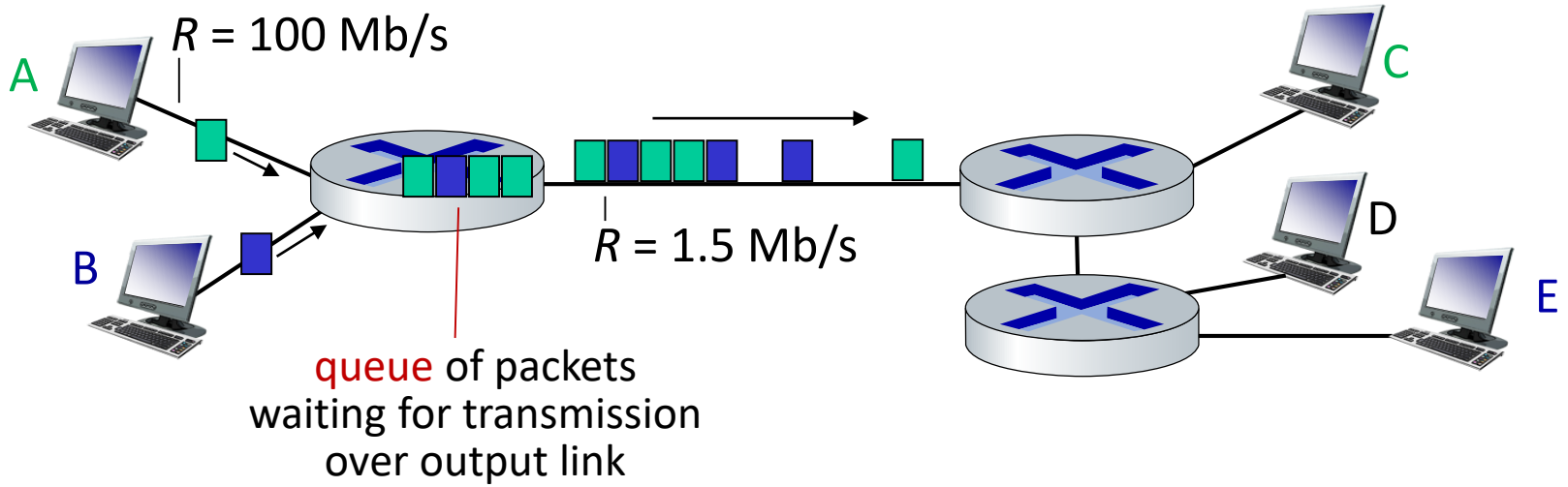


Queueing occurs when work arrives faster than it can be serviced:

* There are multiple links attached to a packet switch (A and B). For each attached link, there is an output buffer or output queue. This queue stores the packets for that link.
* If an arriving packet needs to be transmitted onto a link but if the link is busy with the transmission of another packet, the arriving packet must wait in the output buffer.. Thus, in addition to the *store-and-forward delays*, there are also *output buffer queuing delays*.

* Both A and B send packets to E, the links between the first router and A and B has capacity of 100 Mb/s. Capacity of the link between the routers is 1.5 Mb/s (Congestion will occur at this point, if the arrival rate of packets to the first router exceeds 1.5 Mb/s)

Packet-switching: queueing



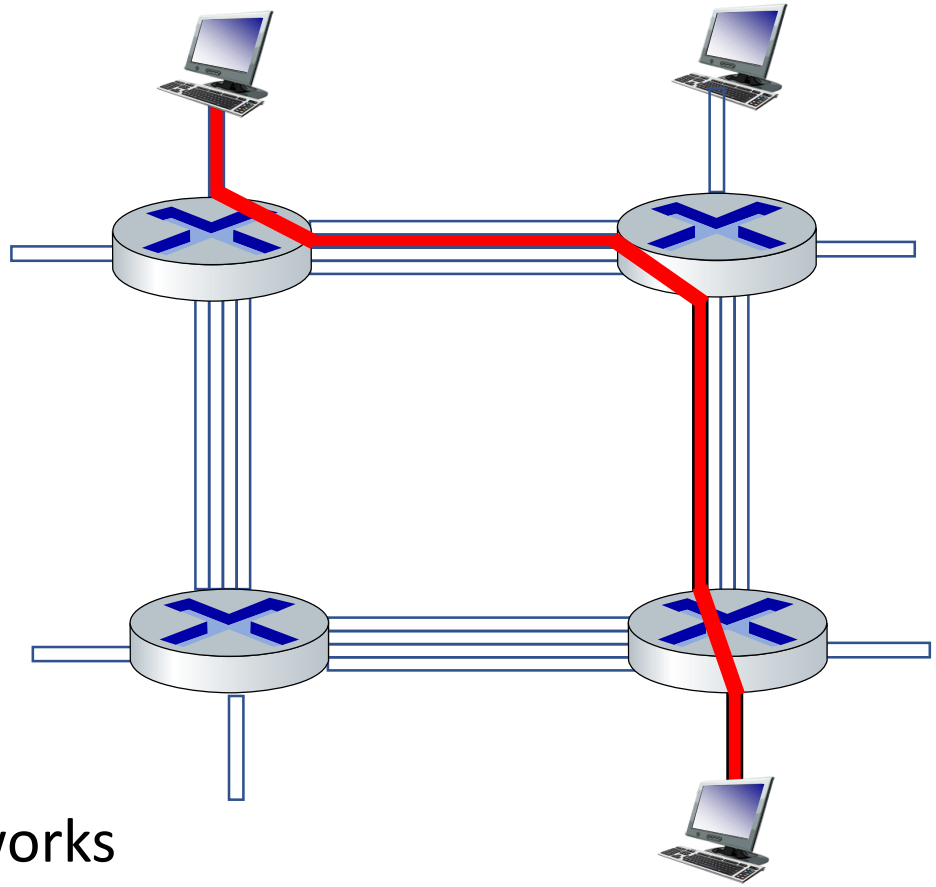
Packet queuing and loss: if arrival rate (in bps) to link exceeds transmission rate (bps) of link for some period of time:

- 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

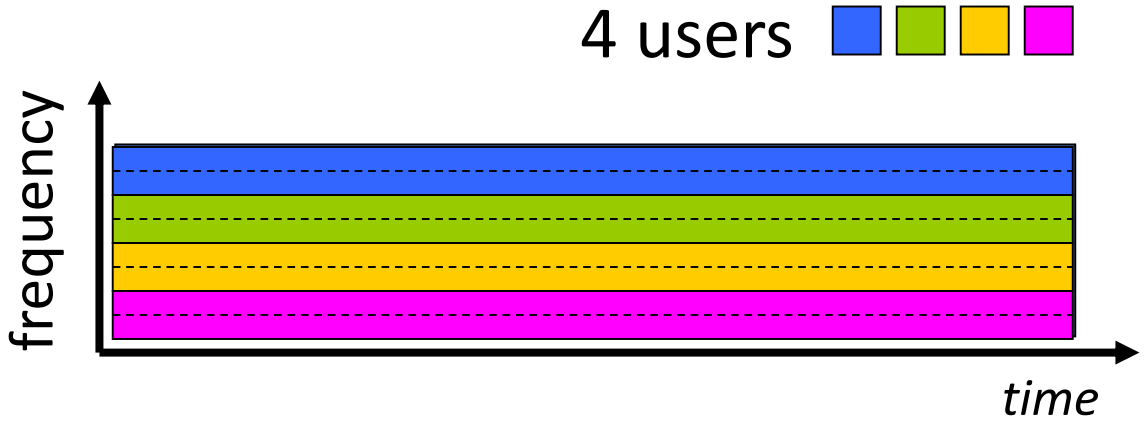


Circuit switching: FDM and TDM

A circuit in a link is implemented with either FDM or TDM:

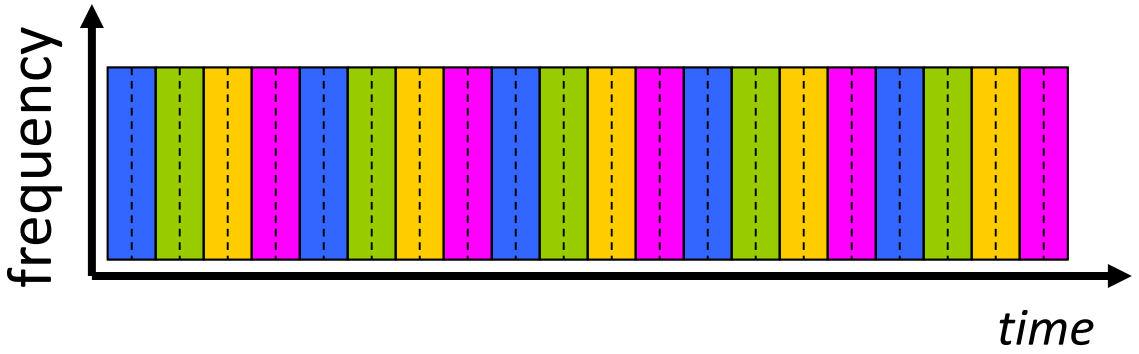
Frequency Division Multiplexing (FDM)

- 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)

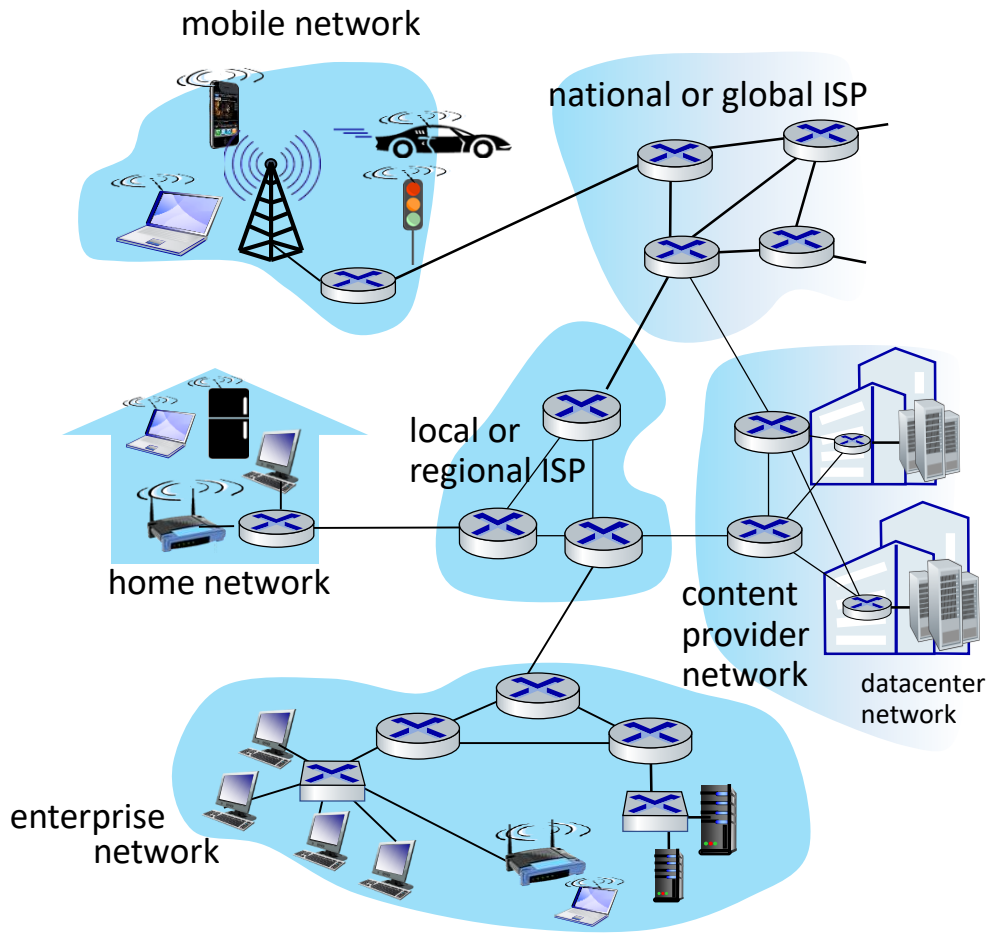


Packet switching versus circuit switching

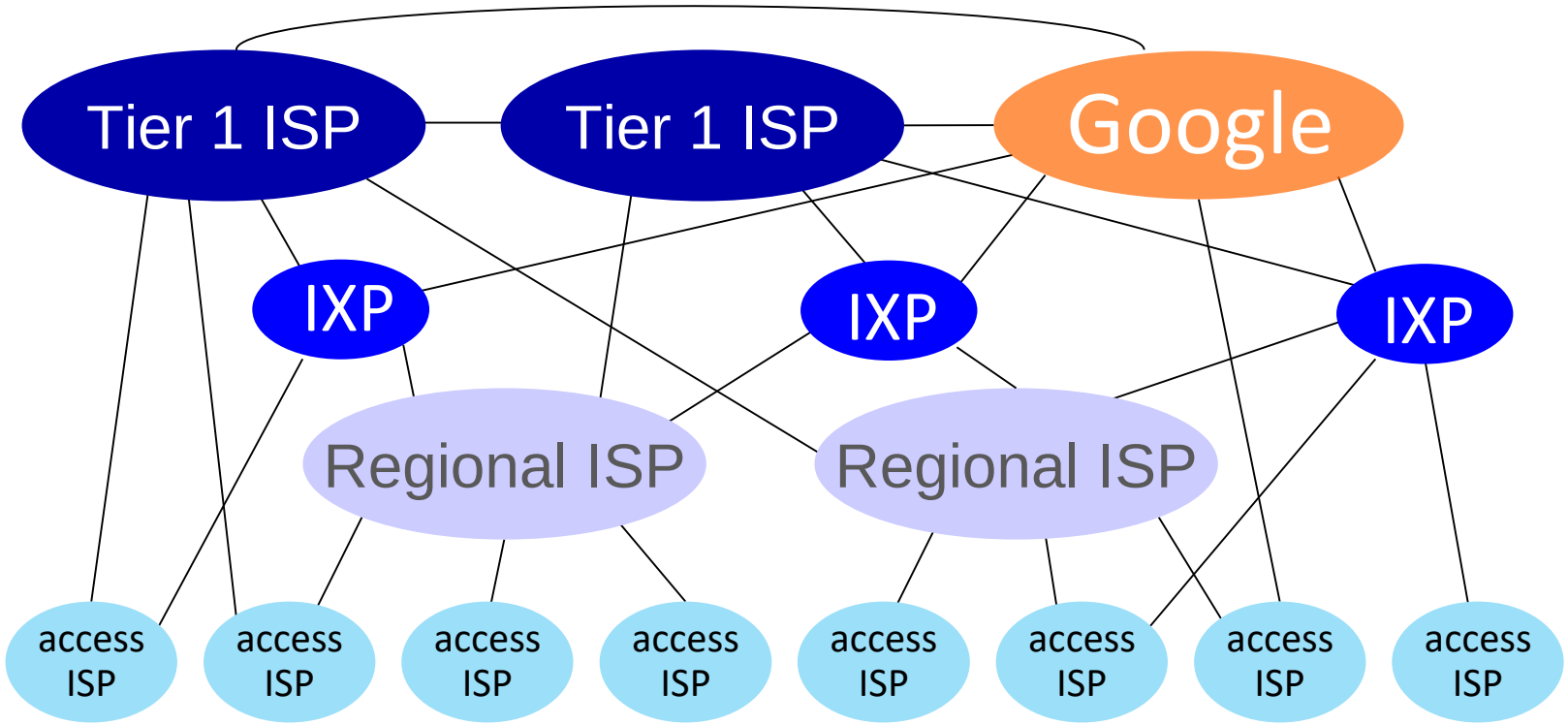
- Circuit switching pre-allocates use of the transmission link regardless of demand, with allocated but unneeded link time going unused.
- Packet switching allocates link use on demand. Link transmission capacity will be shared on a packet-by-packet basis only among those users who have packets.
- Packet switching offers better sharing of transmission capacity than circuit switching and it is simpler, more efficient, and less costly to implement than circuit switching.
- The trend has certainly been in the direction of packet switching.

Internet structure: a "network of networks"

- hosts connect to Internet via access Internet Service Providers (ISPs) or telcos or an enterprise (uni., comp.)
- 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



Internet structure: a "network of networks"



Internet exchange point (IXP):

a physical location where Internet infrastructure companies such as ISPs and CPNs connect with each other

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

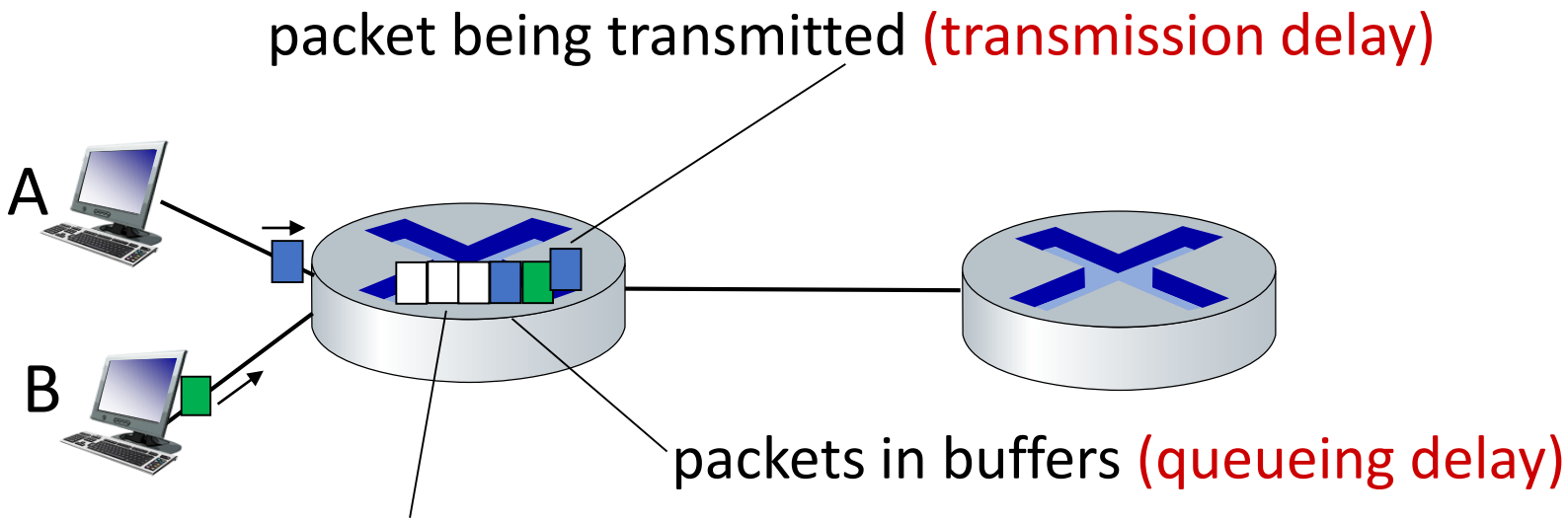
- "tier-1" commercial ISPs (e.g., Level 3, Sprint, AT&T, NTT), national & international coverage
- content provider networks (e.g., Google, Facebook): private network that connects its data centers to Internet, often bypassing tier-1 and regional ISPs

Chapter 1: roadmap

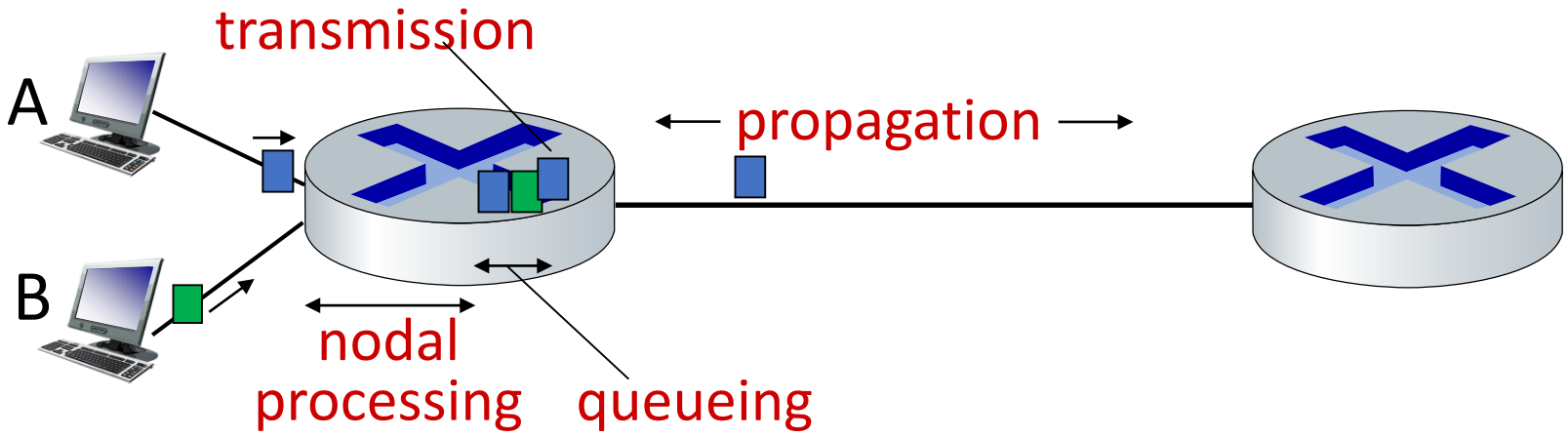
- What *is* the Internet?
- What *is* a protocol?
- Network edge: hosts, access network, physical media
- Network core: packet/circuit switching, internet structure
- Performance: loss, delay, throughput
- Security
- Protocol layers, service models
- History

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/types



(total nodal delay) $d_{nodal} = d_{proc} + d_{queue} + d_{trans} + d_{prop}$

d_{proc}: nodal processing

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

d_{queue}: queueing delay

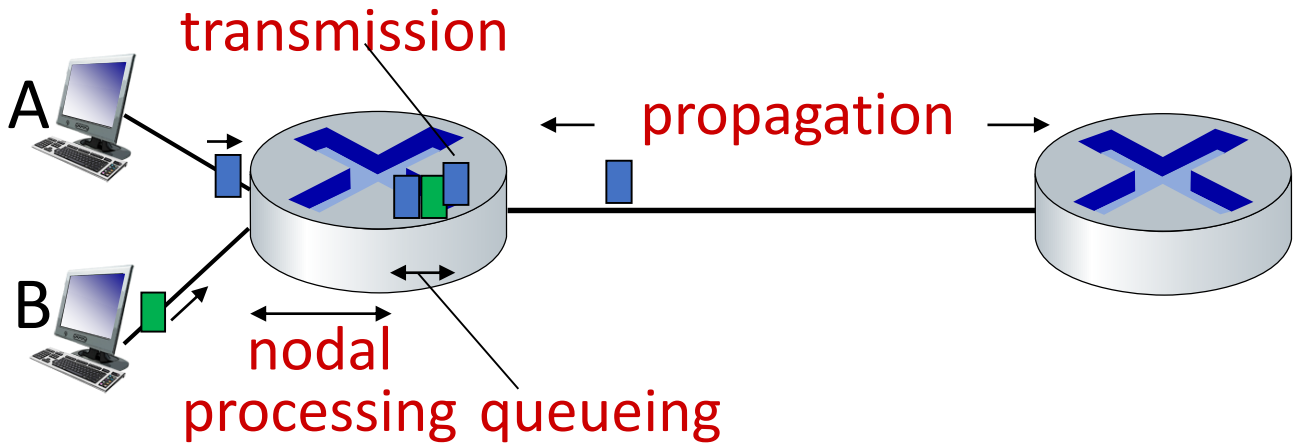
- time waiting at output link for transmission
- depends on congestion level of router
- can be on the order of microsecs to milisecs

Packet delay: four sources

- Packets are transmitted in a first-come-first-served manner, a packet can be transmitted only after all the packets that have arrived before it have been transmitted

- for a 10 Mbps Ethernet link, the rate is R = 10 Mbps; for a 100 Mbps Ethernet link, the rate is R = 100 Mbps (capacity of a link)

- transmission delay is the amount of time required to push (or transmit) all of the packet's bits into the link
- time required to propagate from the beginning of the link to destination is the propagation delay



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

d_{trans}: transmission delay:

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

d_{prop}: propagation delay:

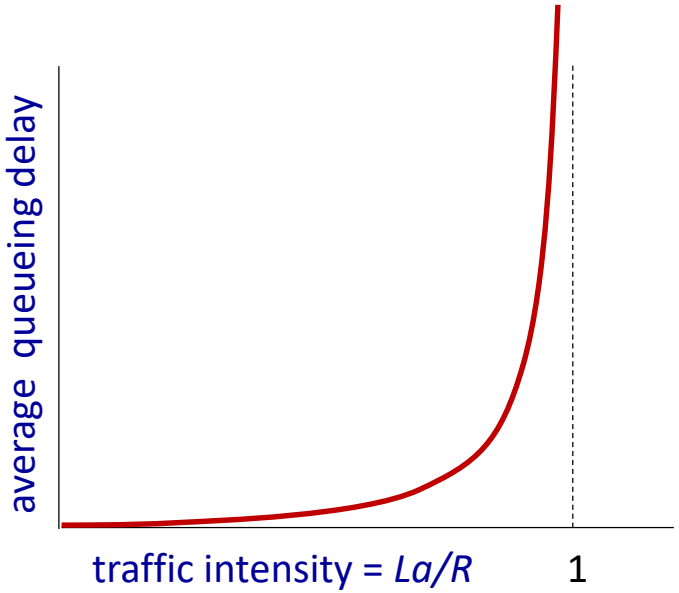
- d: length of physical link
- s: propagation speed (~2×10^8 m/sec): depends on the physical medium
- $d_{prop} = d/s$

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"*



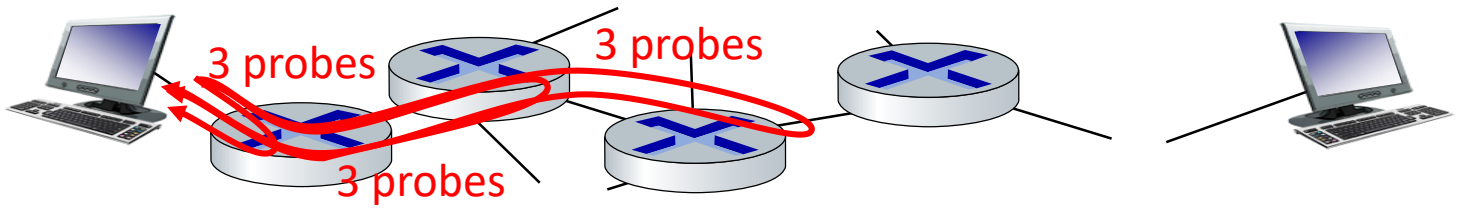
- *La/R* ~ 0: avg. queueing delay small
- *La/R* -> 1: avg. queueing delay large
- *La/R* > 1: more "work" arriving is more than can be serviced - average delay infinite! *(it is not the reality, a packet can arrive to find a full queue, with no place to store such a packet, a router will drop that packet; that is, the packet will be lost)*
- Design your system so that the traffic intensity is no greater than 1.

La/R ~ 0

La/R -> 1

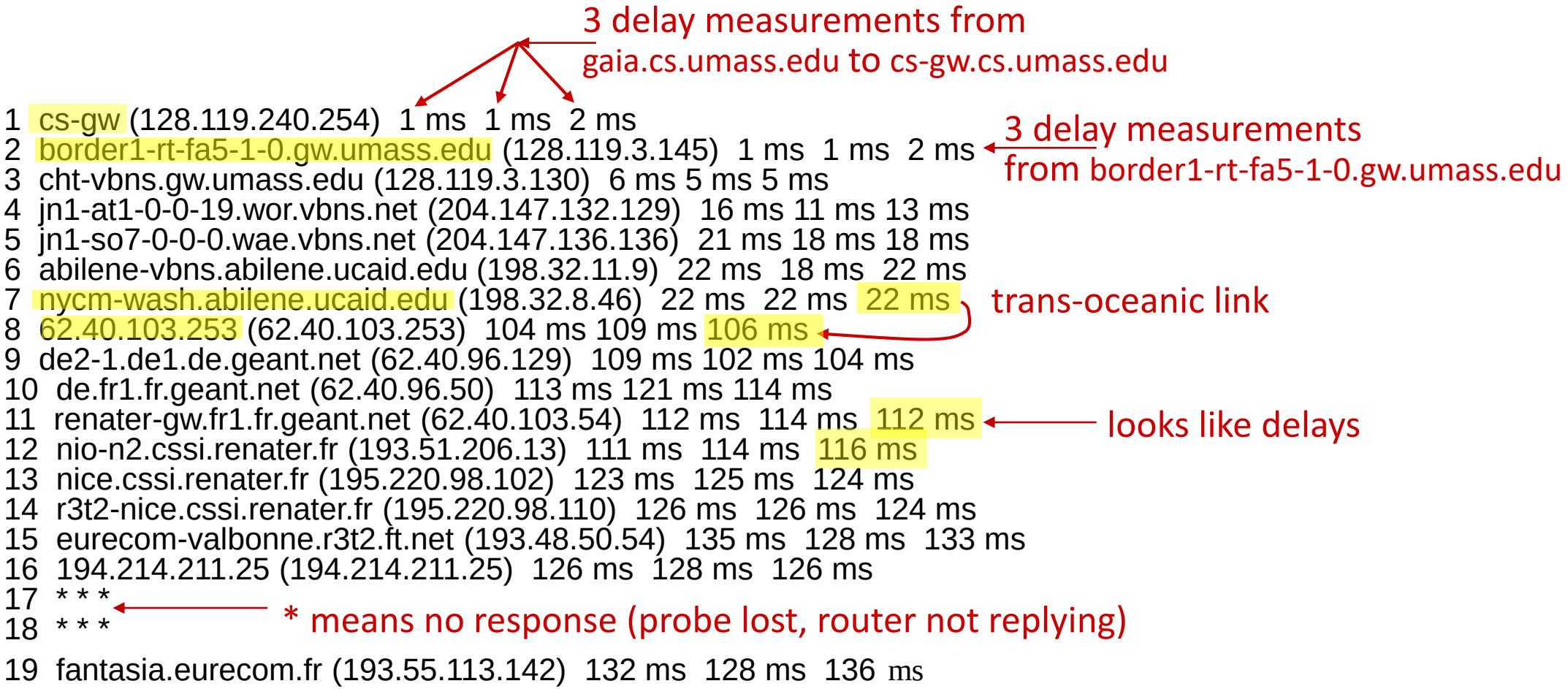
“Real” Internet delays and routes

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



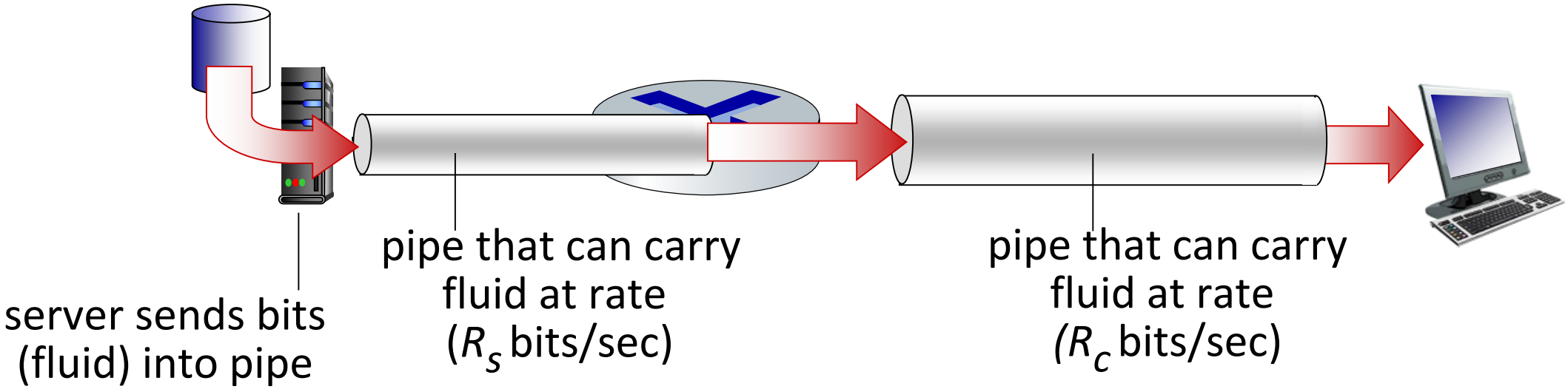
Real Internet delays and routes

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



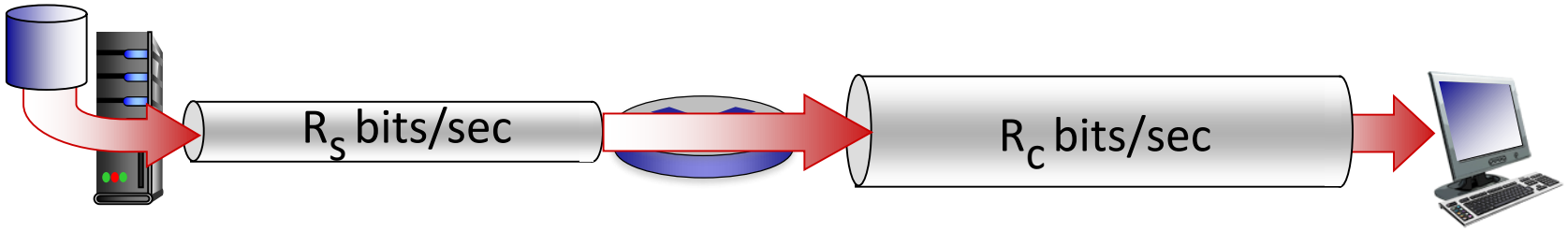
Throughput

- *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

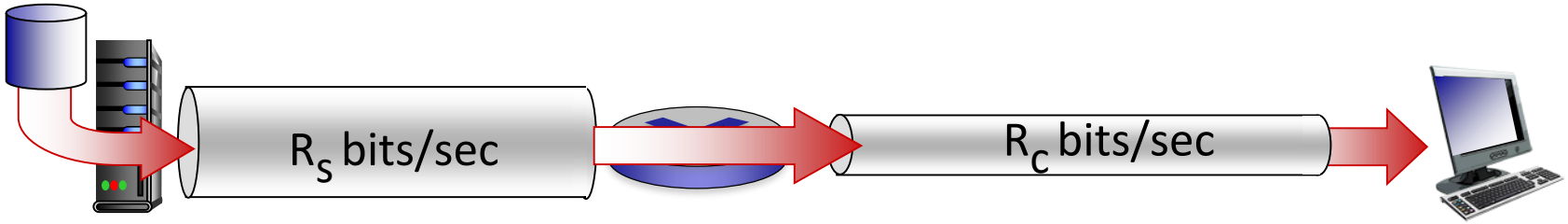


Throughput

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



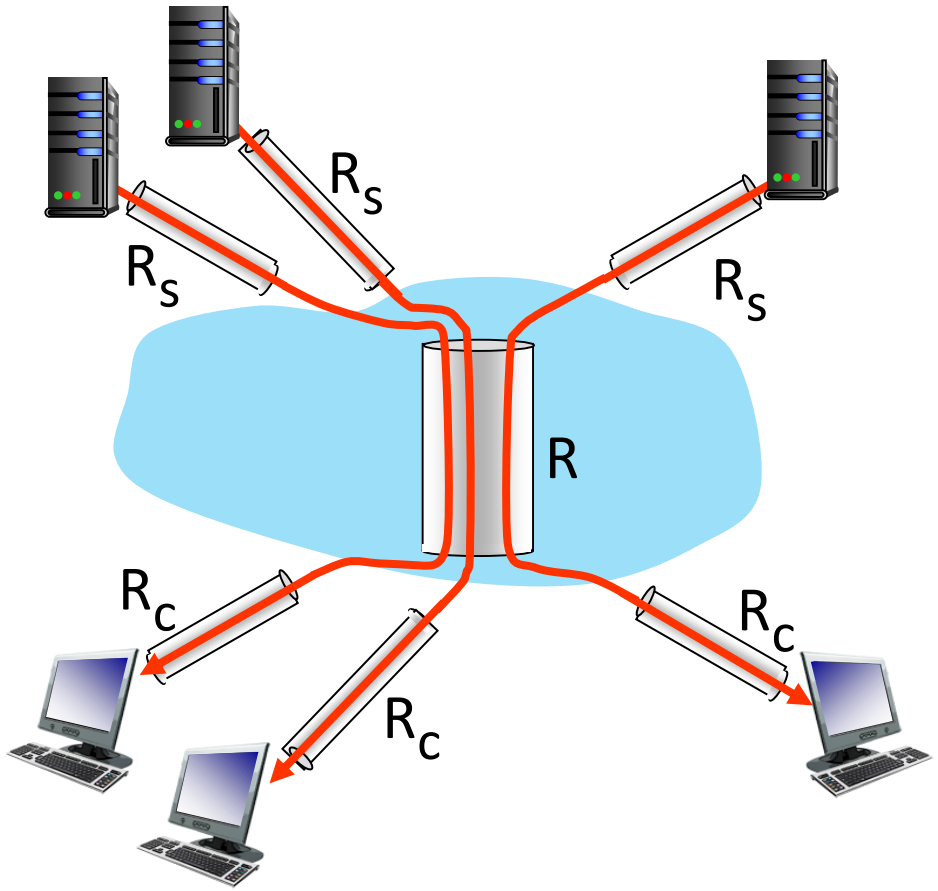
$R_s > R_c$ What is average end-end throughput?



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

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

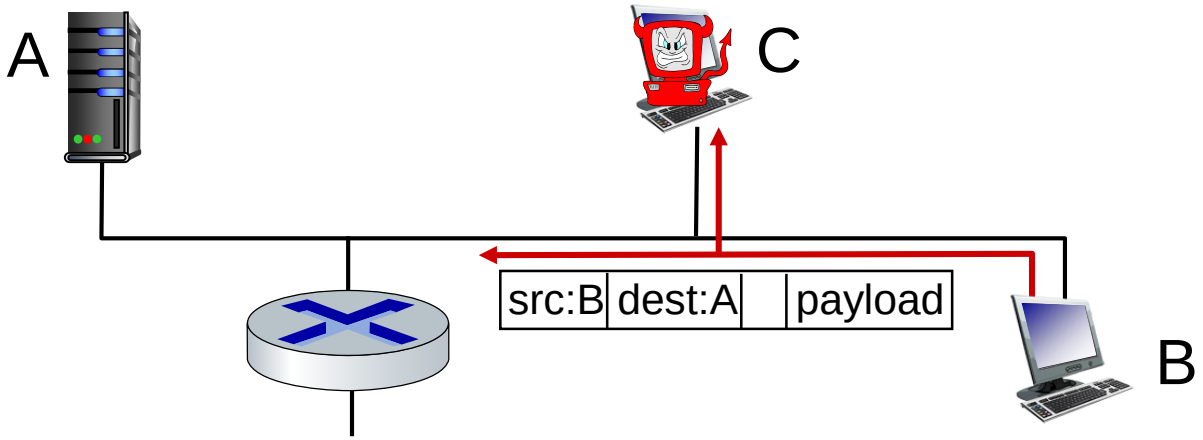
Network security

- Internet not originally designed with (much) security in mind
  - *original vision:* "a group of mutually trusting users attached to a transparent network"
- 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: packet interception

packet "sniffing": a passive receiver that records a copy of every packet that flies by (a packet sniffer)

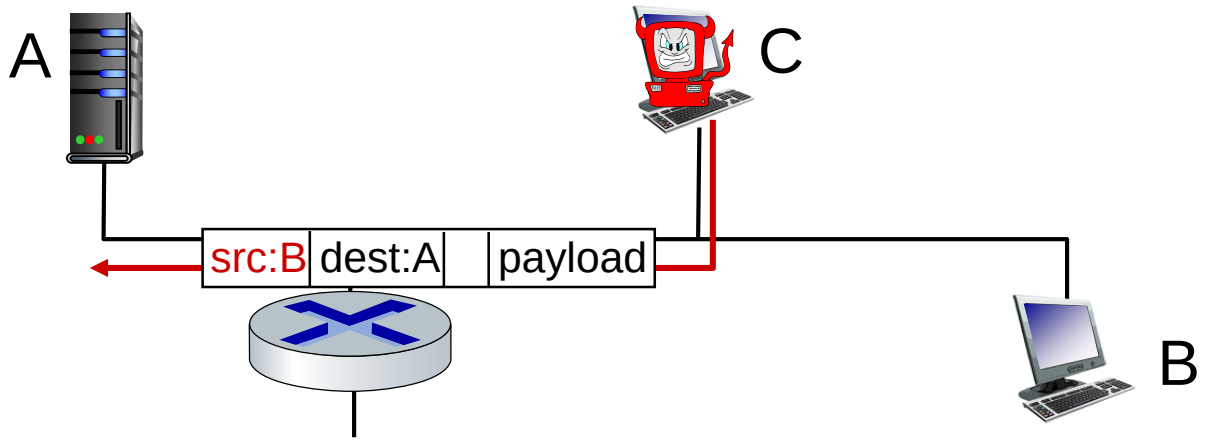
- broadcast media (shared Ethernet, wireless)
- promiscuous network interface reads/records all packets (e.g., including passwords!) passing by



Wireshark is a (free) packet-sniffer

Bad guys: fake identity

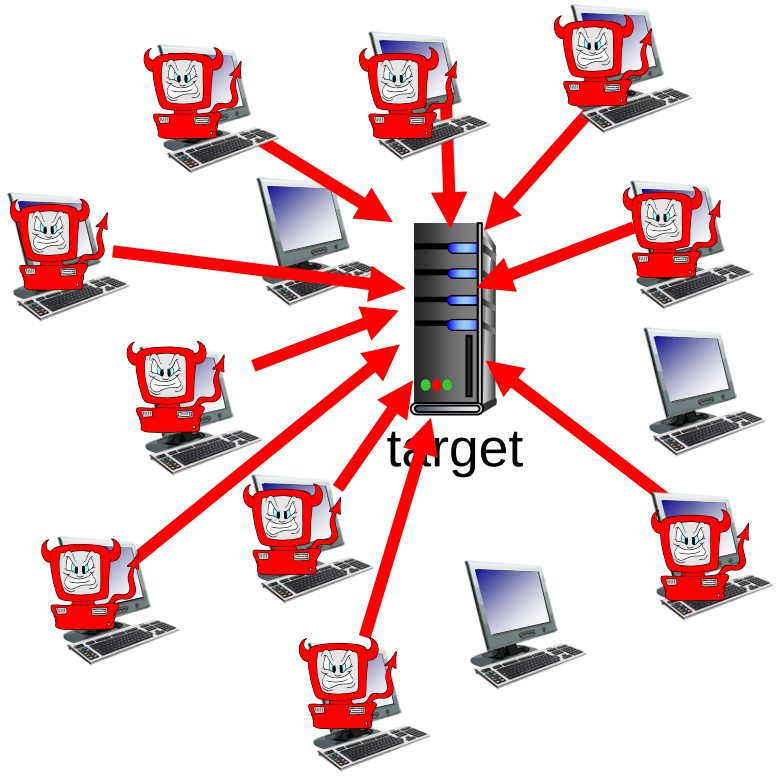
IP spoofing: injection of packet with false source address



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 (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

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

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?

(layered structures)

Why layering?

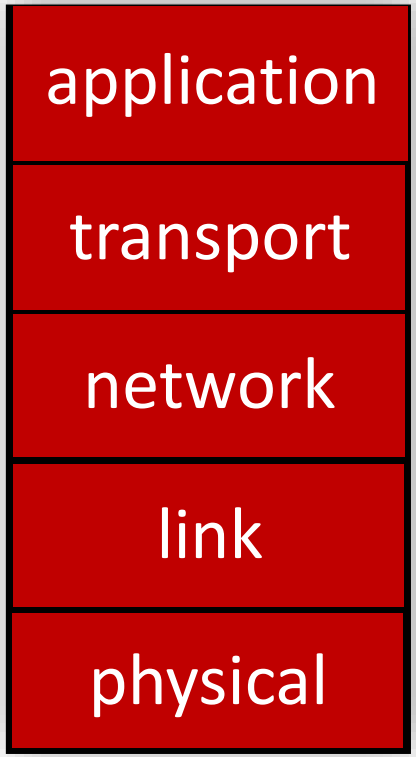
Approach to designing/discussing complex systems:

- 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

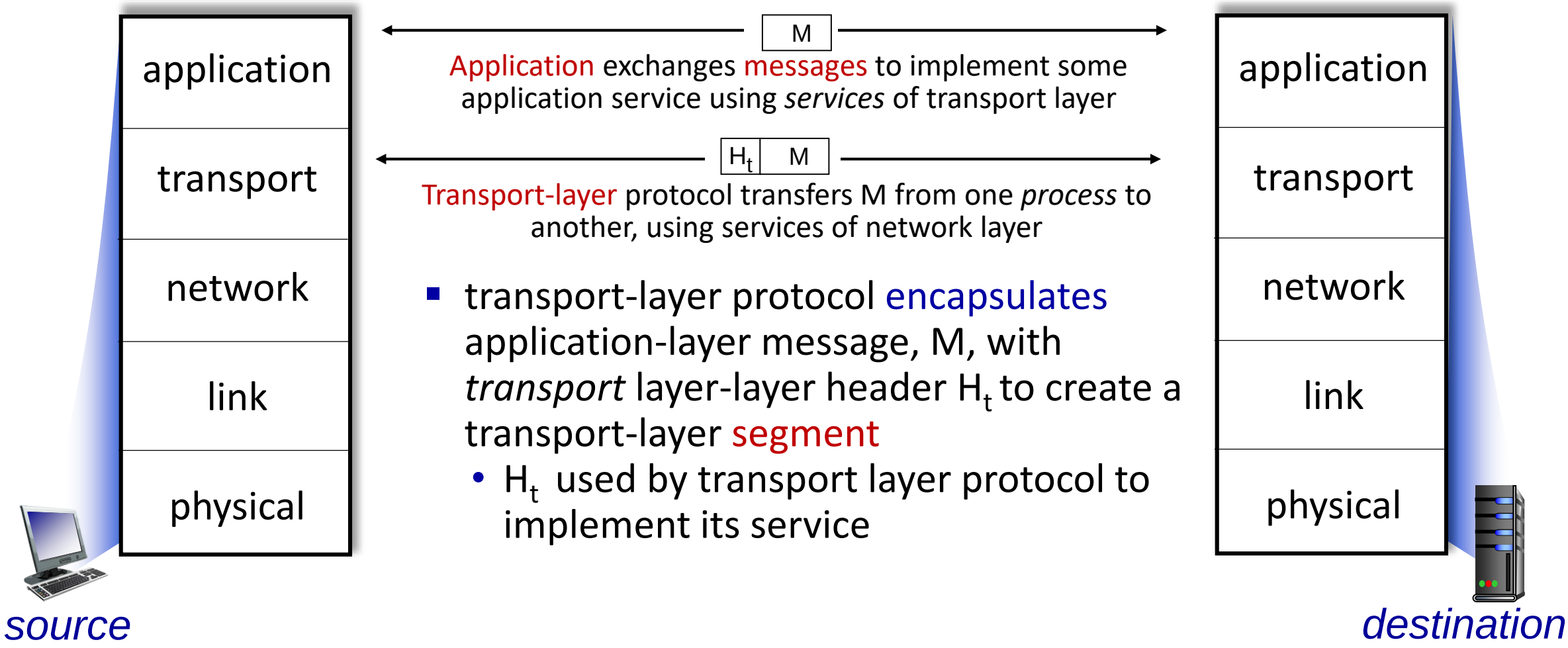
** each layer implements a service, a number of protocols in layers*

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”

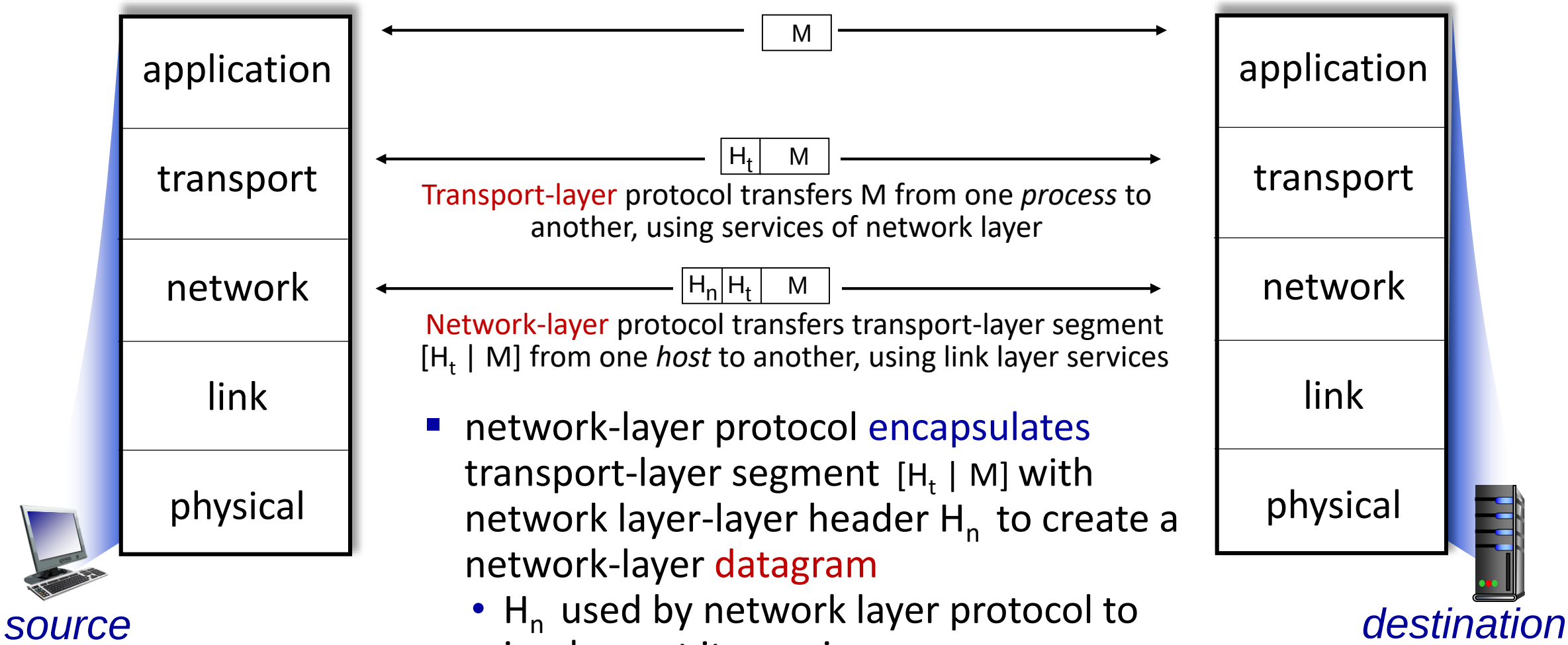


Services, Layering and Encapsulation



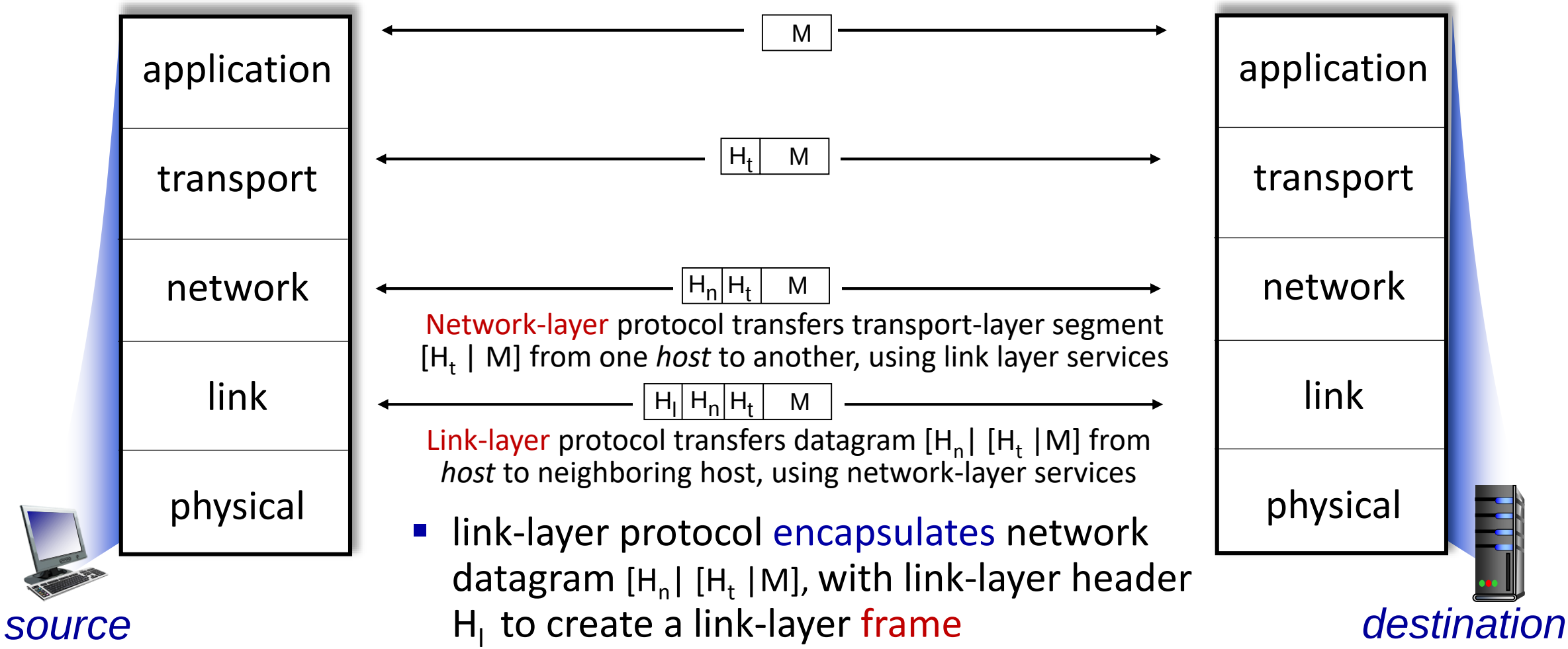
- transport-layer protocol encapsulates application-layer message, M, with *transport* layer-layer header H_t to create a transport-layer segment
  - H_t used by transport layer protocol to implement its service

Services, Layering and Encapsulation



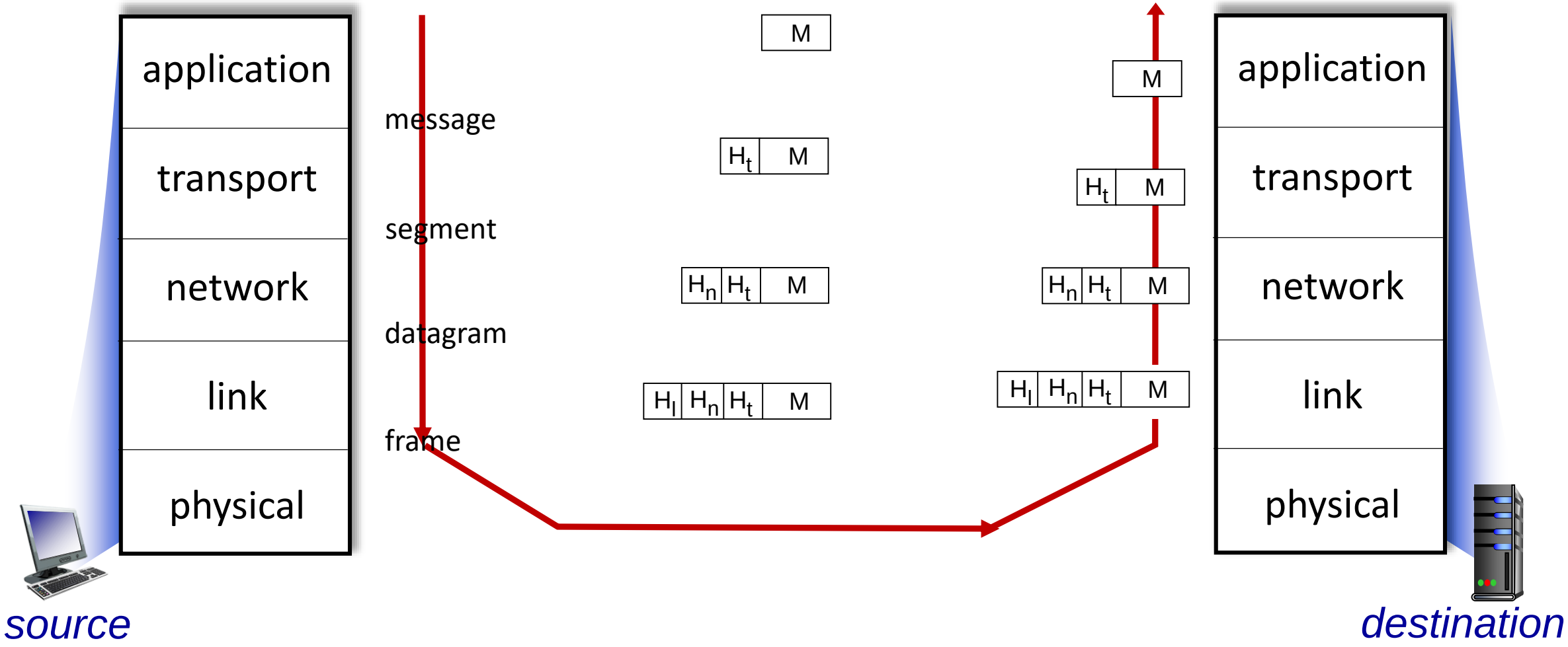
- network-layer protocol encapsulates transport-layer segment [H_t | M] with network layer-layer header H_n to create a network-layer datagram
  - H_n used by network layer protocol to implement its service

Services, Layering and Encapsulation

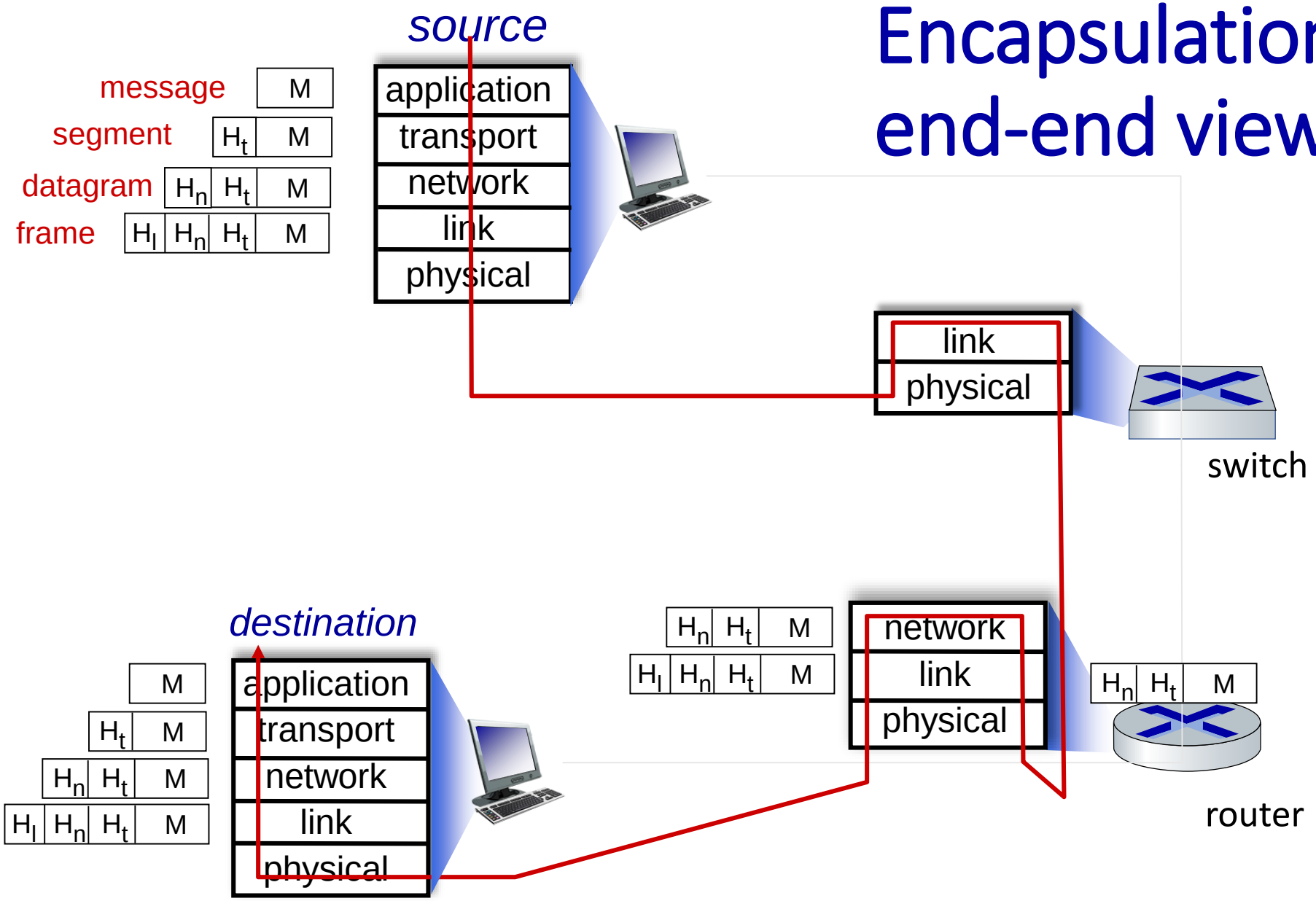


- link-layer protocol encapsulates network datagram [H_n| [H_t |M], with link-layer header H_l to create a link-layer frame

Services, Layering and Encapsulation



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

1972-1980: Internetworking, new and proprietary networks

1980-1990: new protocols, a proliferation of networks

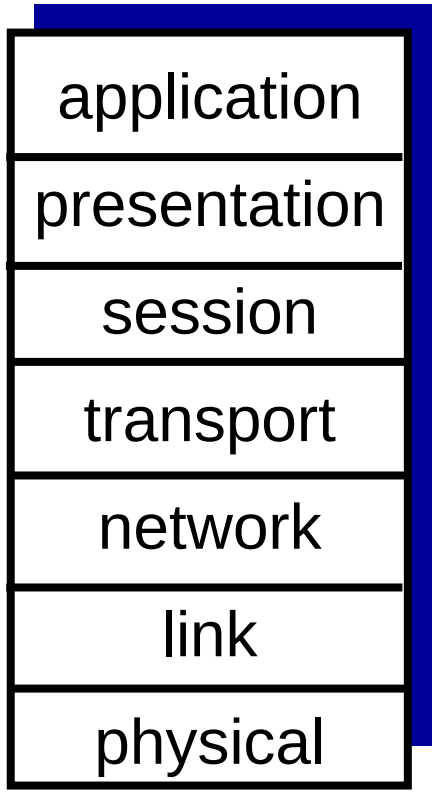
1990, 2000s: commercialization, the Web, new applications

2005-present: scale, SDN, mobility, cloud

ISO/OSI reference model

Two layers not found in Internet protocol stack!

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



The seven layer OSI/ISO reference model