Computer Networks

July – November 2022

Prof. Sukumar Nandi

CS 341- Computer Networks

Welcome to CS 341 (Theory)

- Course components:
  - Lectures
  - Class Notes
  - Assignments
  - Quizes
  - End sem

Quiz : 2 in total (2 x 10% weightage)

Assignments: minimum 2 (2 x 5% Weightage)

Mid-sem: 20% Weightage

End sem: 40% Weightage

Textx/References:

- **Computer Networking- A Top-Down Approach, Jim Kurose and Keith W. Ross**

- Computer Networks, by Andrew S Tanenbaum and David J Wetherall, 5th Ed and above, Prentice Hall.

Syllabus:

- Classification of Communication Networks. Standard models of communication: OSI and TCP/IP. Importance of layering and service models.
- Application layer services and protocols.
- Transport layer services, principles and protocols: study of TCP and UDP. Principles of reliability: sliding window protocols, selective repeat and go-back-N. Principles of congestion control: TCP case study. Details of TCP working.
- Network layer services, algorithms and protocols. Study of routing algorithms. Study of Internet router architecture. IP addressing principles: assignment and aggregation. Study of DHCP, ICMP, NAT , QoS, ARP, NDP
- Link Layer protocols: CSMA, DLL, HDLC, MPLS, Token Ring etc.

The Internet: a "nuts and bolts" view

Billions of connected computing *devices*:

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

Packet switches: forward packets (chunks of data)

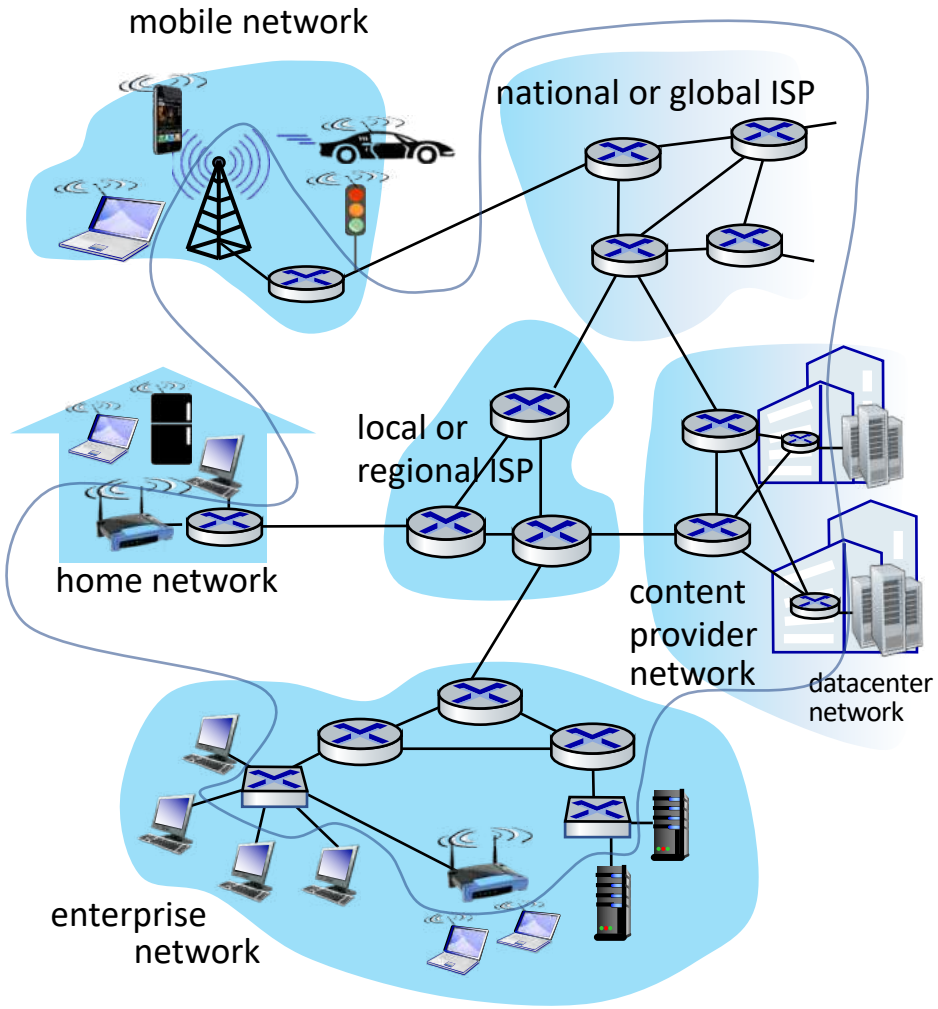
- *routers, switches*

Communication links

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

Networks

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



“Fun” Internet-connected devices

Amazon Echo

Internet refrigerator

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

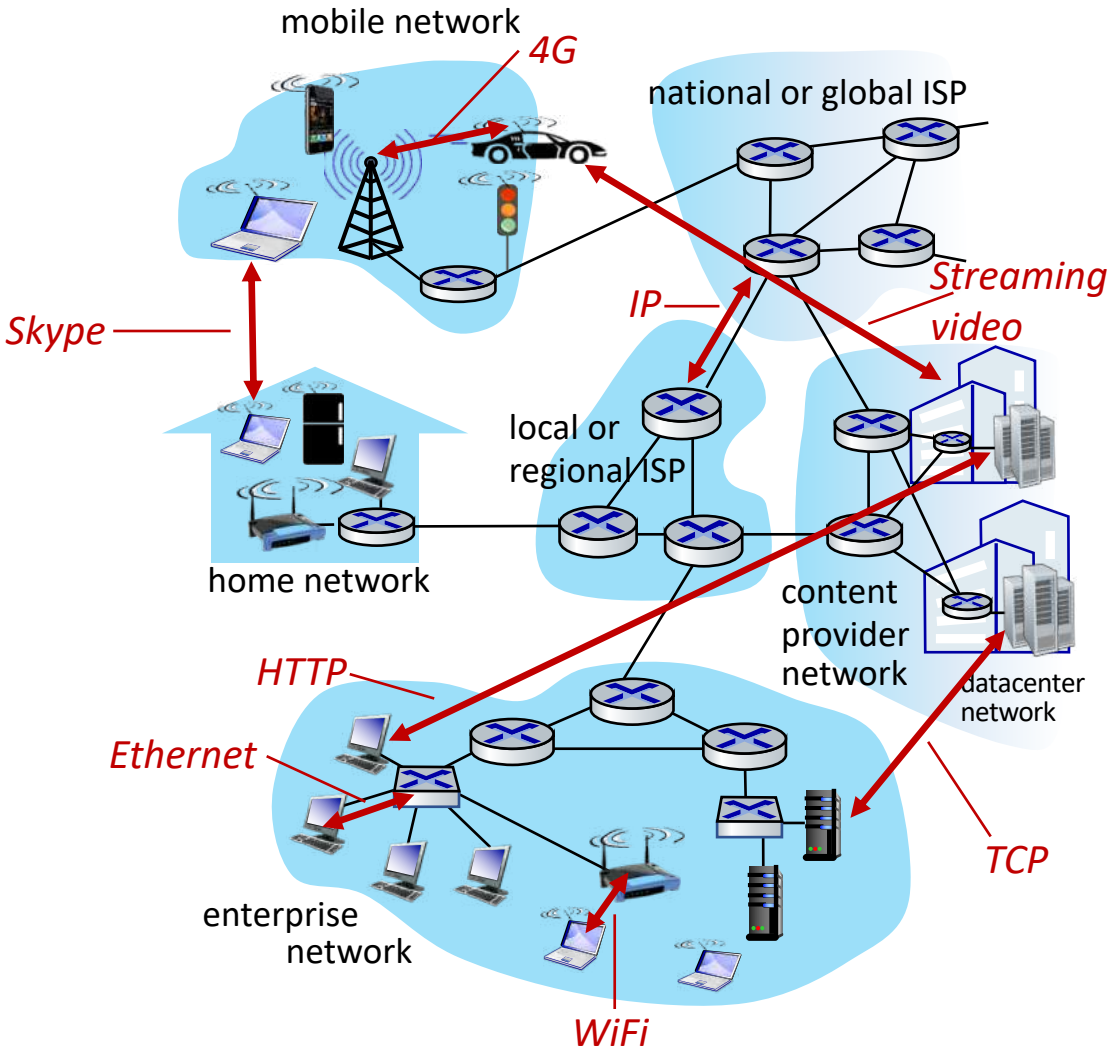
sensorized, bed mattress

Fitbit

Others?

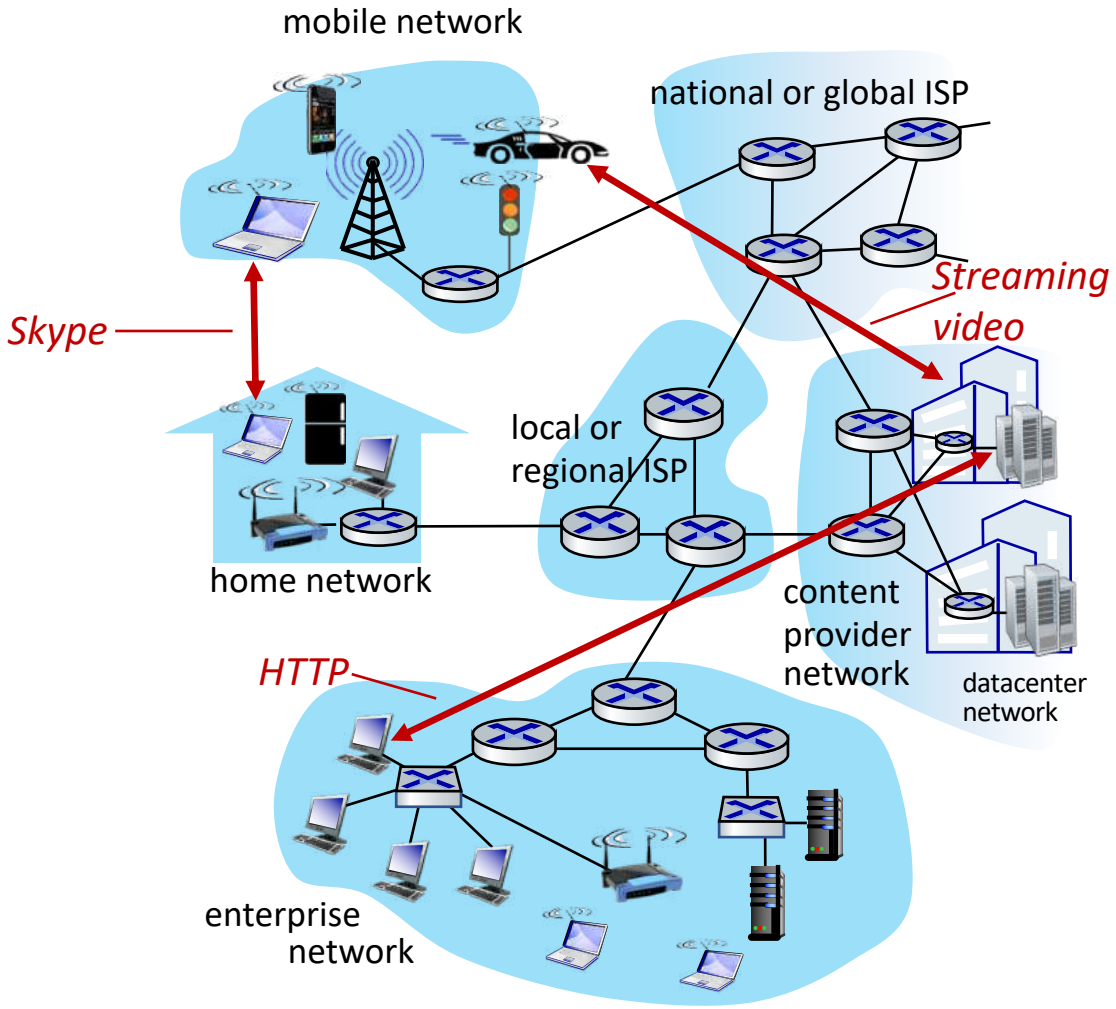
The Internet: a "nuts and bolts" view

- *Internet:* "network of networks"
  - Interconnected ISPs
- *protocols* are *everywhere*
  - control sending, receiving of messages
  - e.g., HTTP (Web), streaming video, Skype, TCP, IP, WiFi, 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, inter-connected appliances, …
- provides *programming interface* to distributed applications:
  - "hooks" allowing sending/receiving apps to "connect" to, use Internet transport service
  - provides service options, analogous to postal service



What's a protocol?

Human protocols:

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

Rules for:

… specific messages sent

… specific actions taken when message received, or other events

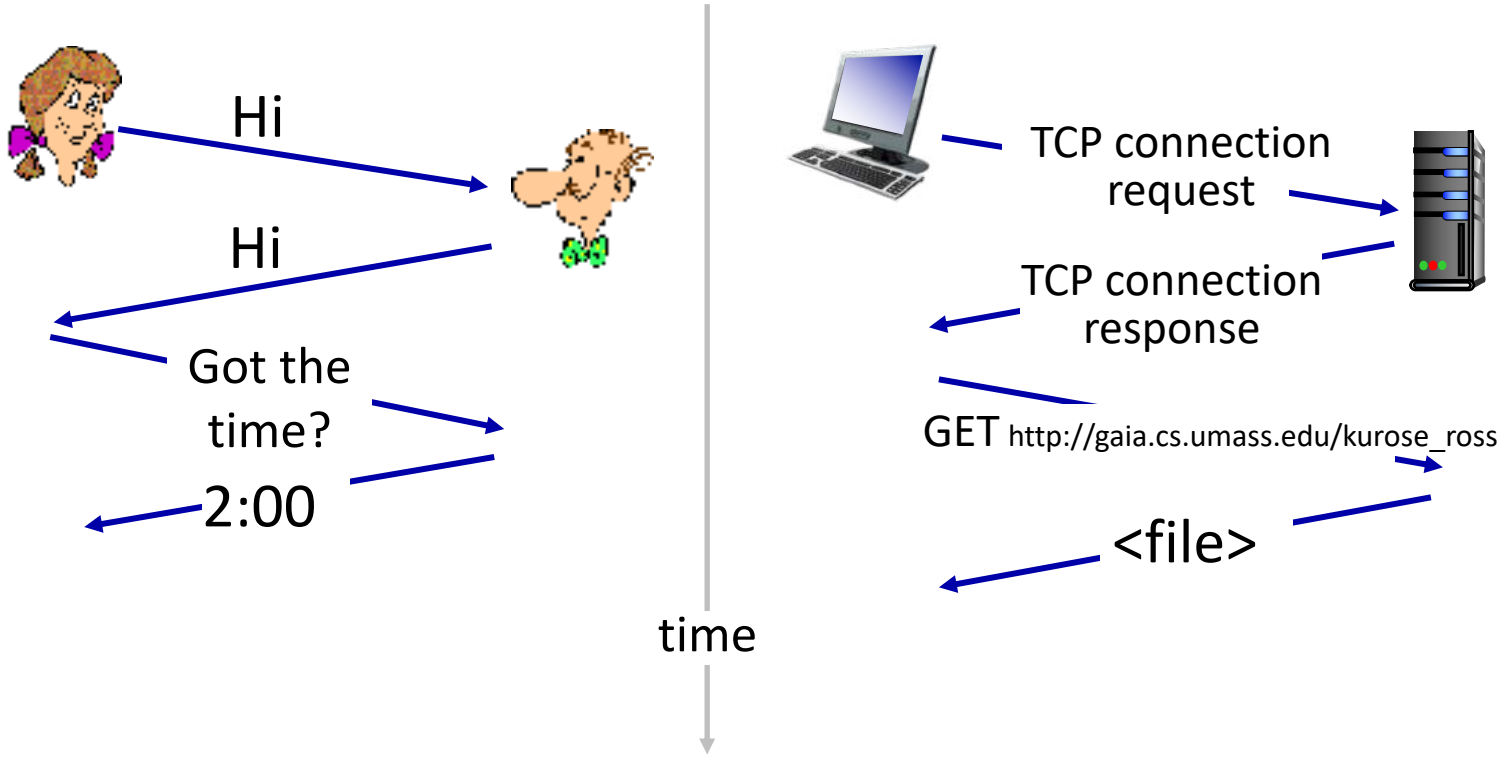
Network protocols:

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

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

What's a protocol?

A human protocol and a computer network protocol:



Q: other human protocols?

Protocols and interfaces

- A **Protocol** is a set of rules required for two or more ***similar*** processes to communicate with each other.

- An **Interface** is a set of rules required for two or more ***dissimilar*** processes to communicate with each other.

- A protocol is a ***logical*** concept, while an interface is a ***physical*** one.

Functions of Protocol

1. Help in establishing necessary CONVENTIONS.

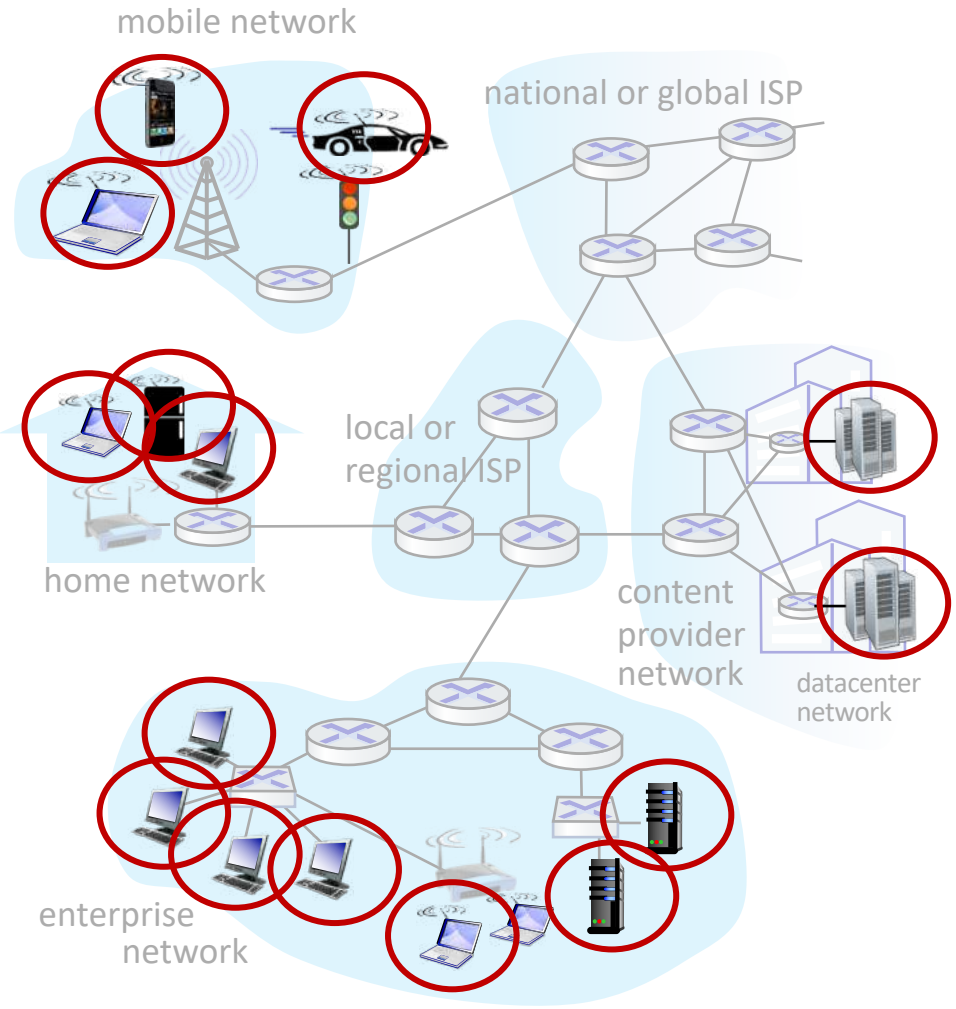
2. Help in establishing the STANDARDS.

3. Help in establishment of STANDARD DATA ELEMENTS

A closer look at Internet structure

Network edge:

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



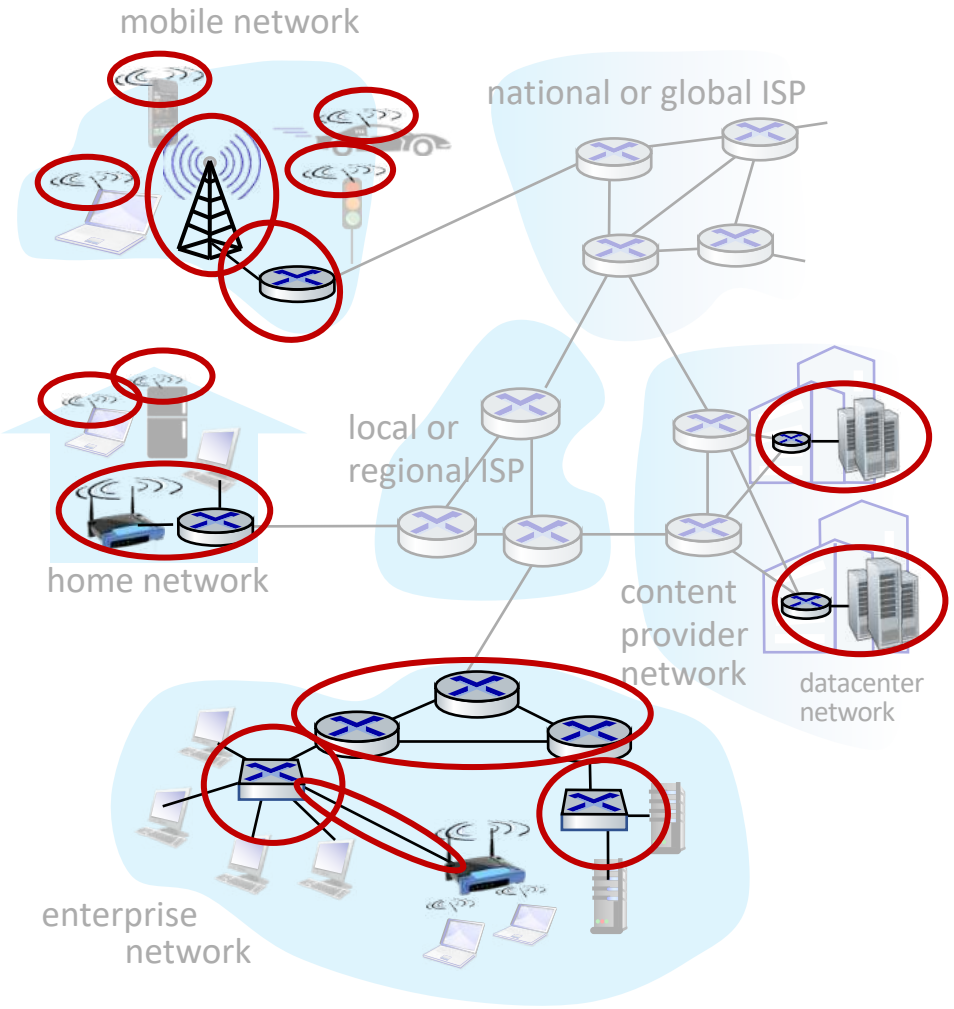
A closer look at Internet structure

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:

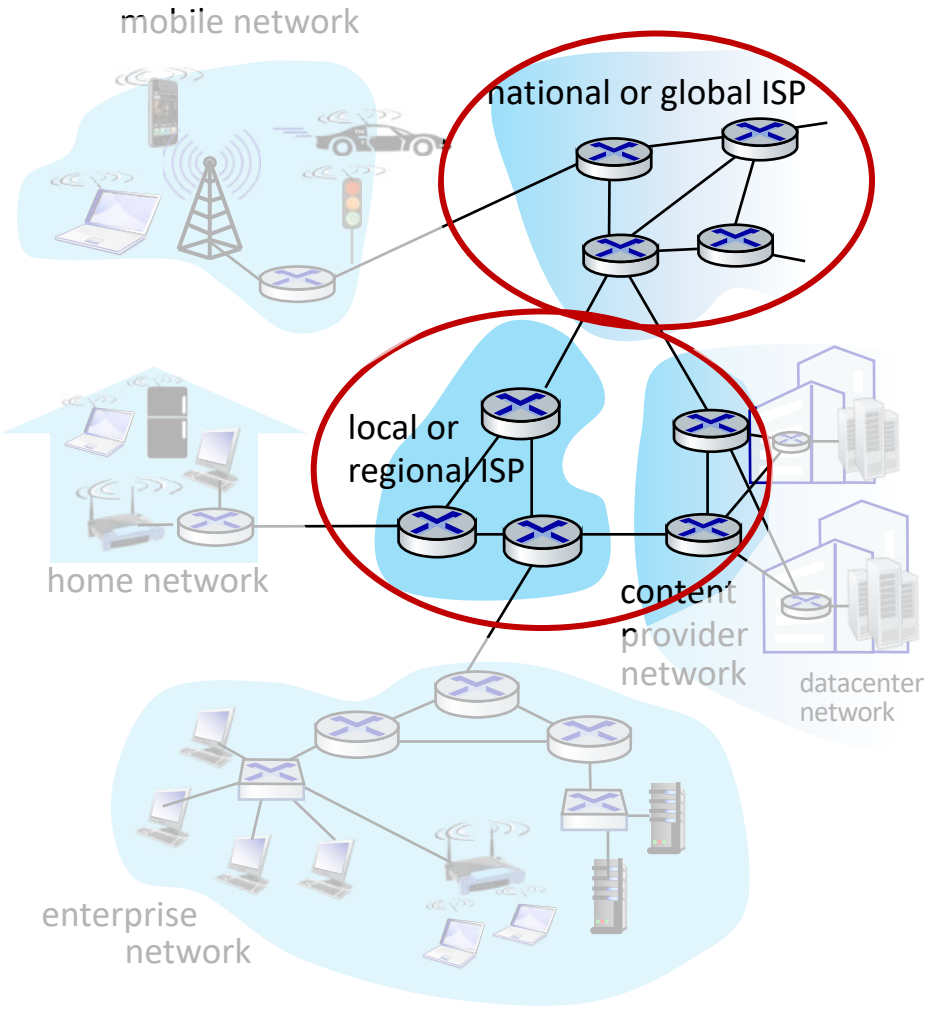
- hosts: clients and servers
- servers often in data centers

Access networks, physical media:

- wired, wireless communication links

Network core:

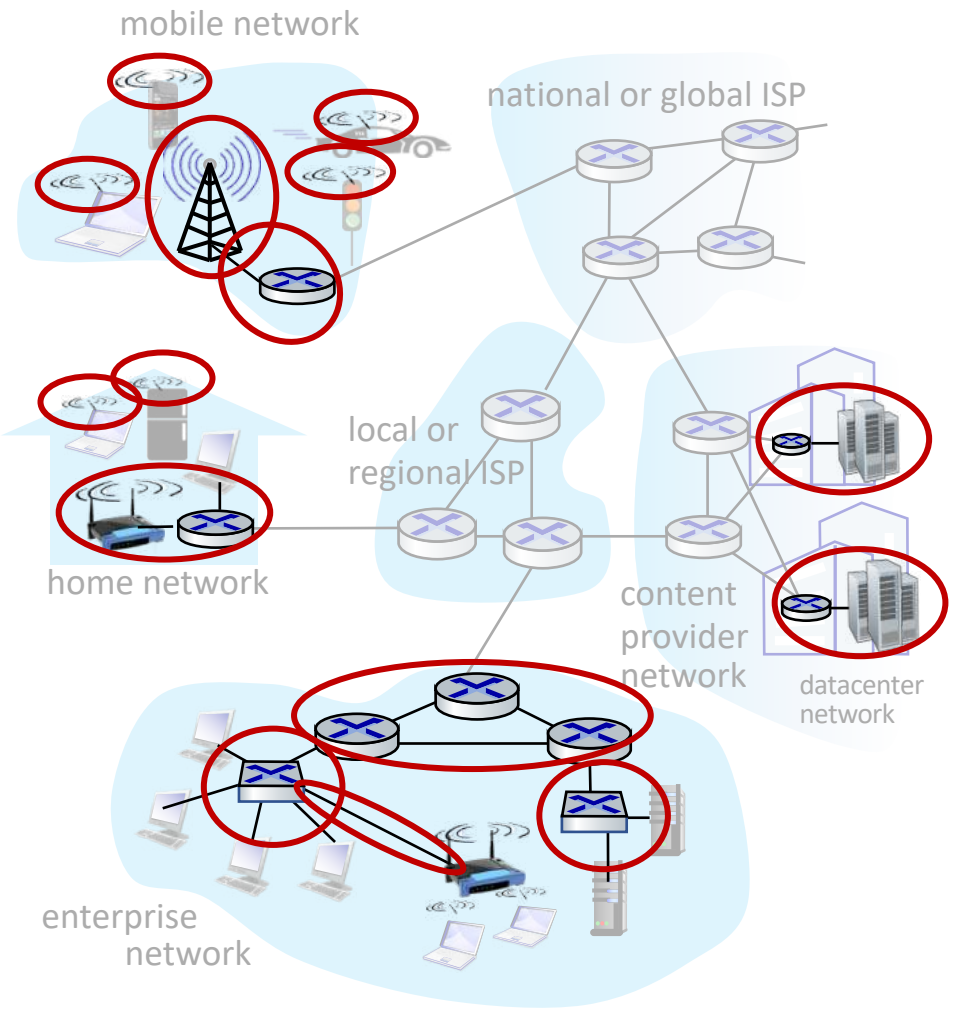
- interconnected routers
- network of networks



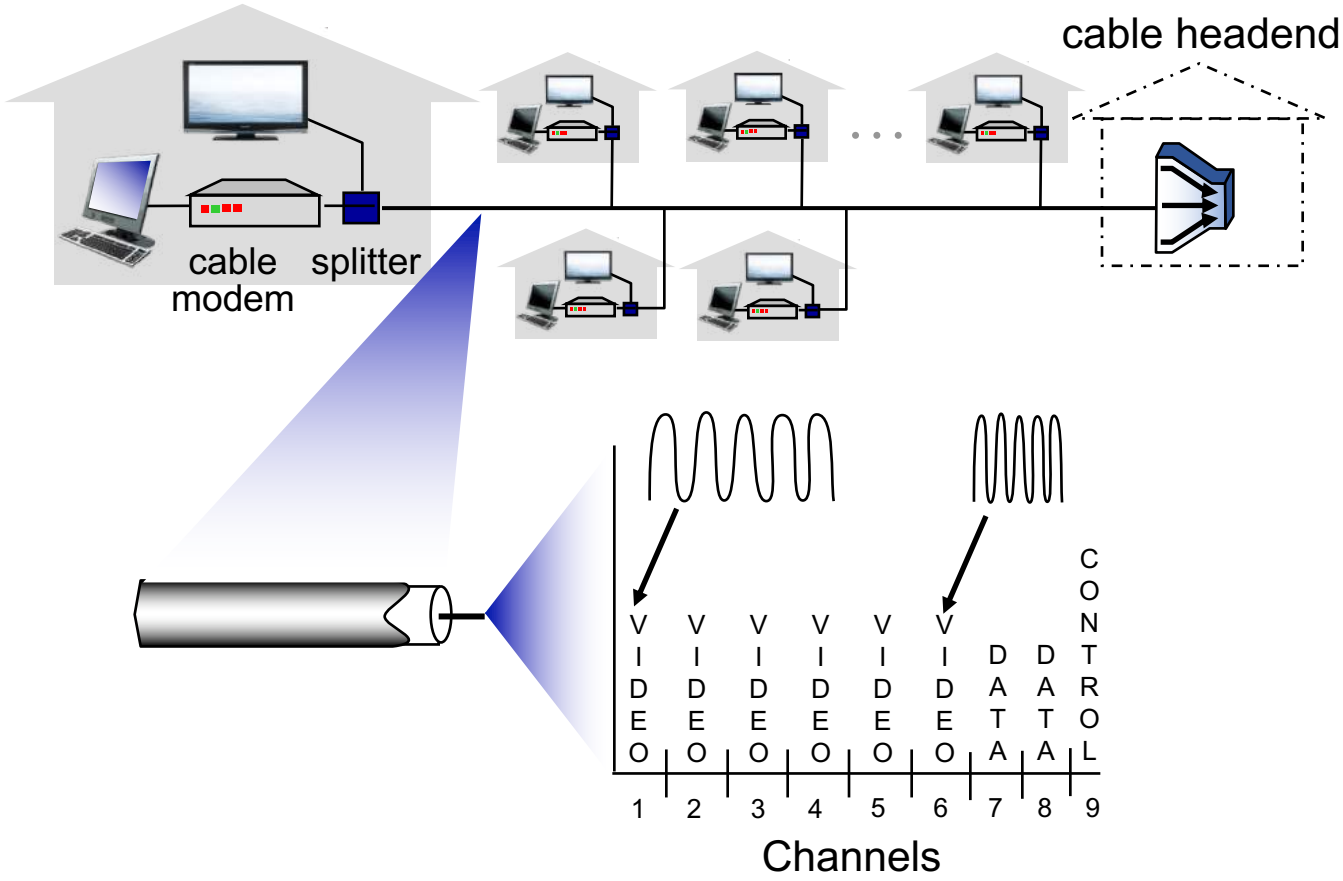
Access networks and physical media

Q: How to connect end systems to edge router?

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

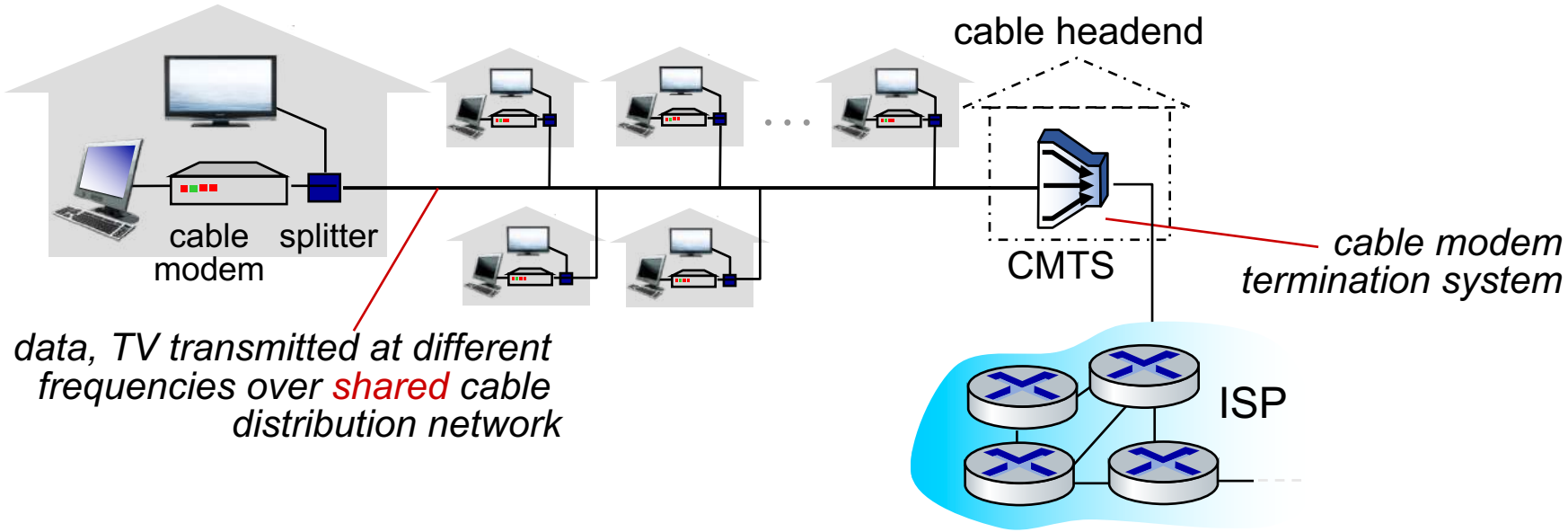


Access networks: cable-based access



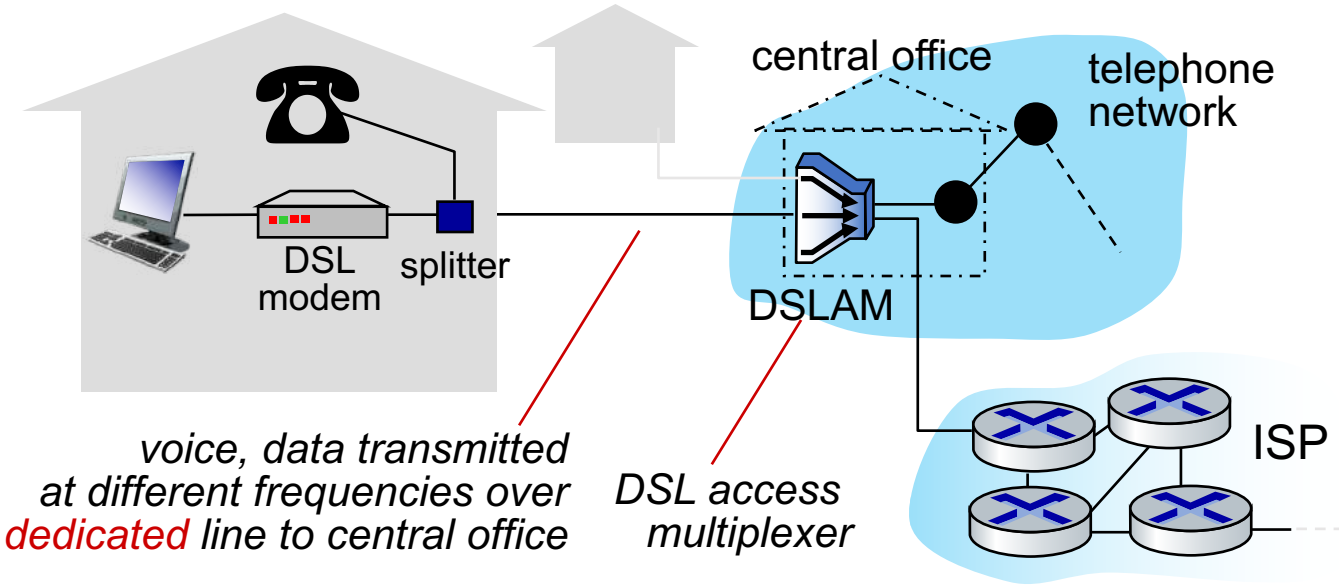
frequency division multiplexing (FDM): different channels transmitted in different frequency bands

Access networks: cable-based access



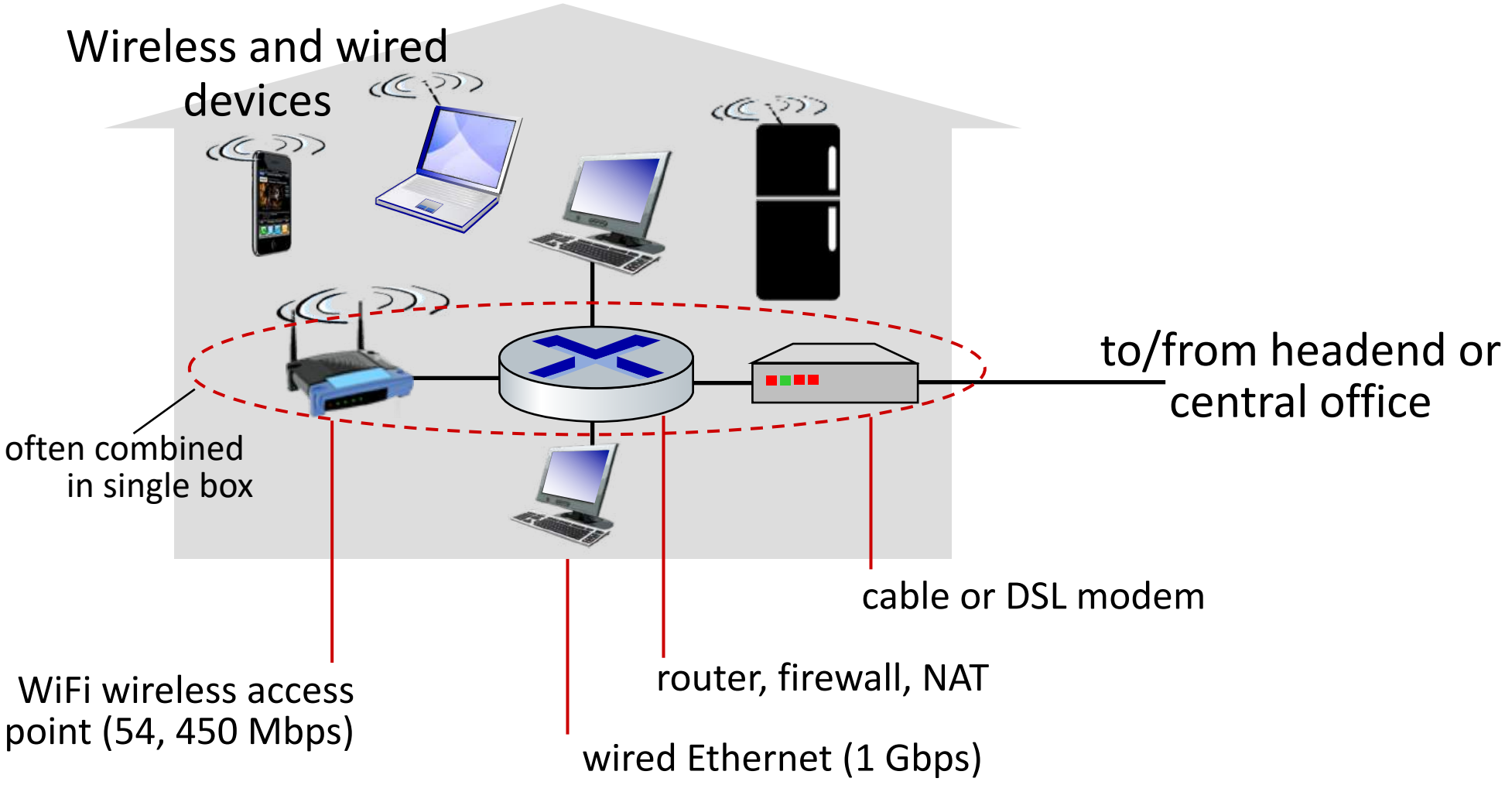
- HFC: hybrid fiber coax
  - asymmetric: up to 40 Mbps – 1.2 Gbps downstream transmission rate, 30-100 Mbps upstream transmission rate
- network of cable, fiber attaches homes to ISP router
  - homes *share access network* to cable headend

Access networks: digital subscriber line (DSL)



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

Access networks: home networks



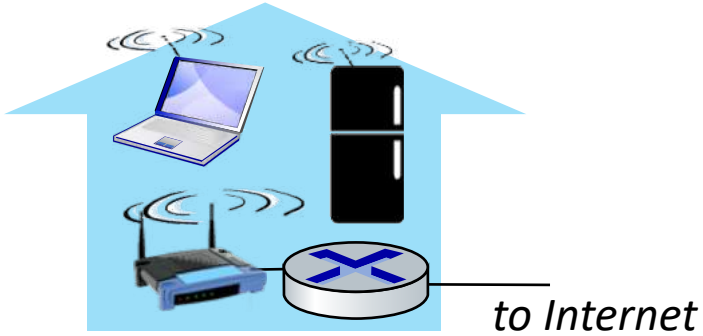
Wireless access networks

Shared *wireless* access network connects end system to router

- via base station aka “access point”

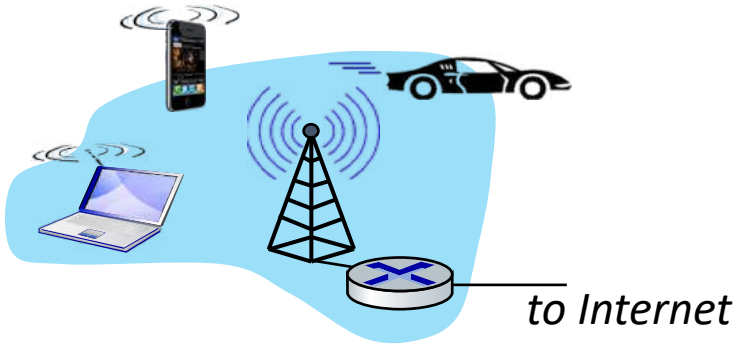
Wireless local area networks (WLANs)

- typically within or around building (~100 ft)
- 802.11b/g/n/ac (WiFi): 11, 54, 450, 3000 Mbps transmission rate

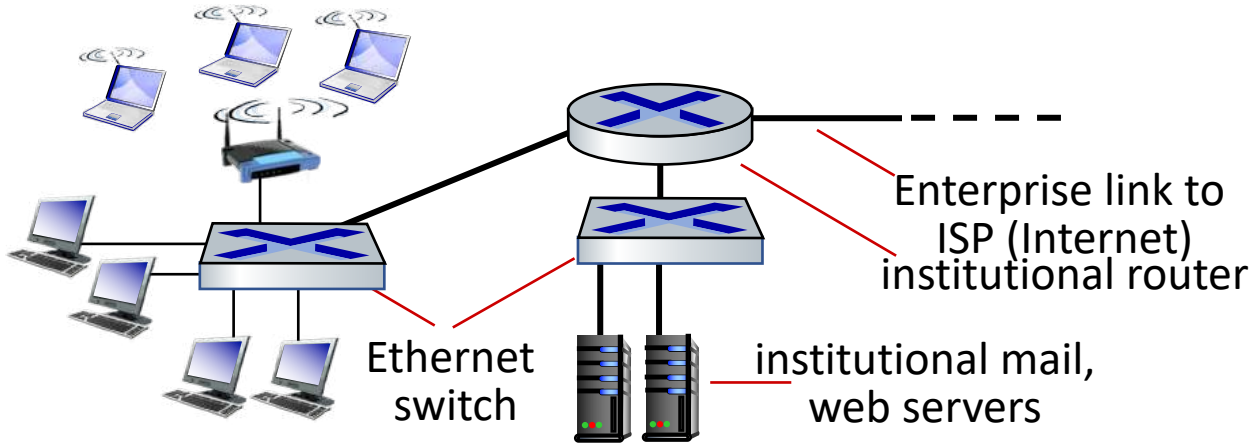


Wide-area cellular access networks

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



Access networks: enterprise networks

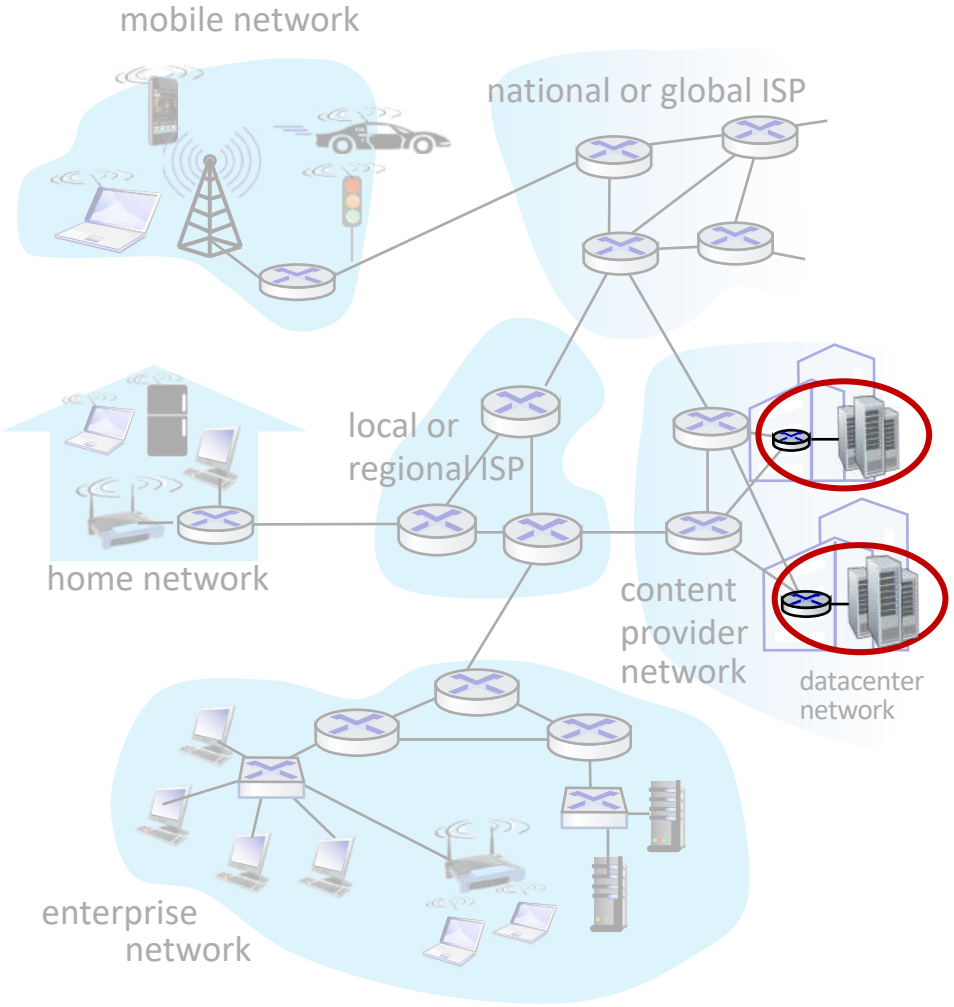


- companies, universities, etc.
- mix of wired, wireless link technologies, connecting a mix of switches and routers (we'll cover differences shortly)
  - Ethernet: wired access at 100Mbps, 1Gbps, 10Gbps, 25Gbps, 40Gbps
  - WiFi: wireless access points at 11, 54, 450, 3000 Mbps

Access networks: data center networks

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

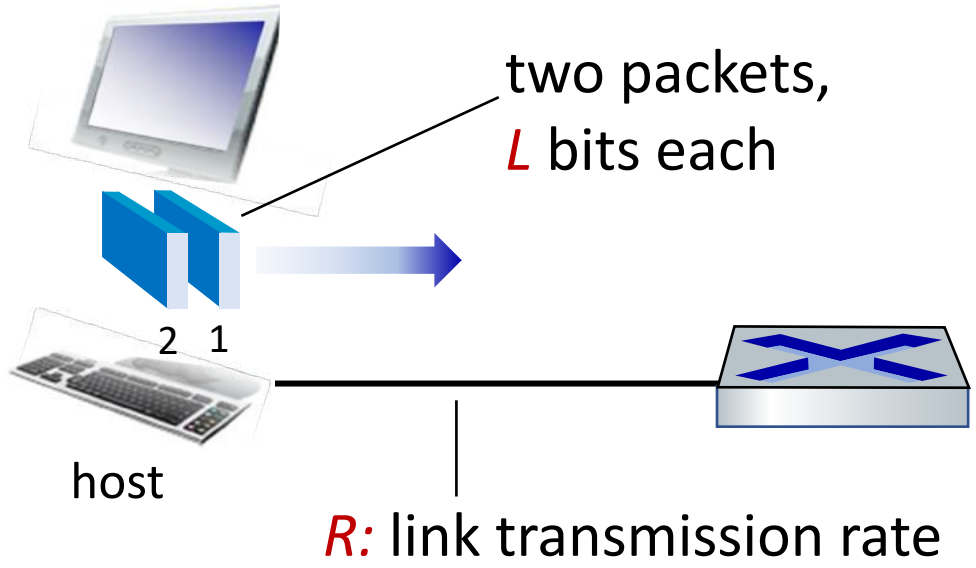
Courtesy: Massachusetts Green High Performance Computing Center (mghpcc.org)



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*



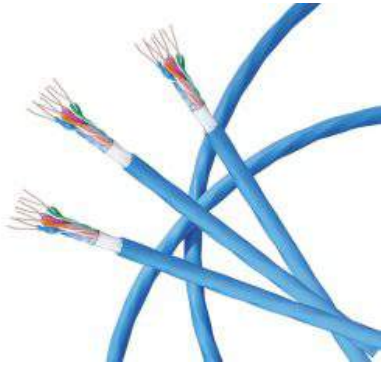
$$\text{packet transmission delay} = \text{time needed to transmit } L\text{-bit packet into link} = \frac{L \text{ (bits)}}{R \text{ (bits/sec)}}$$

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: 10Gbps 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

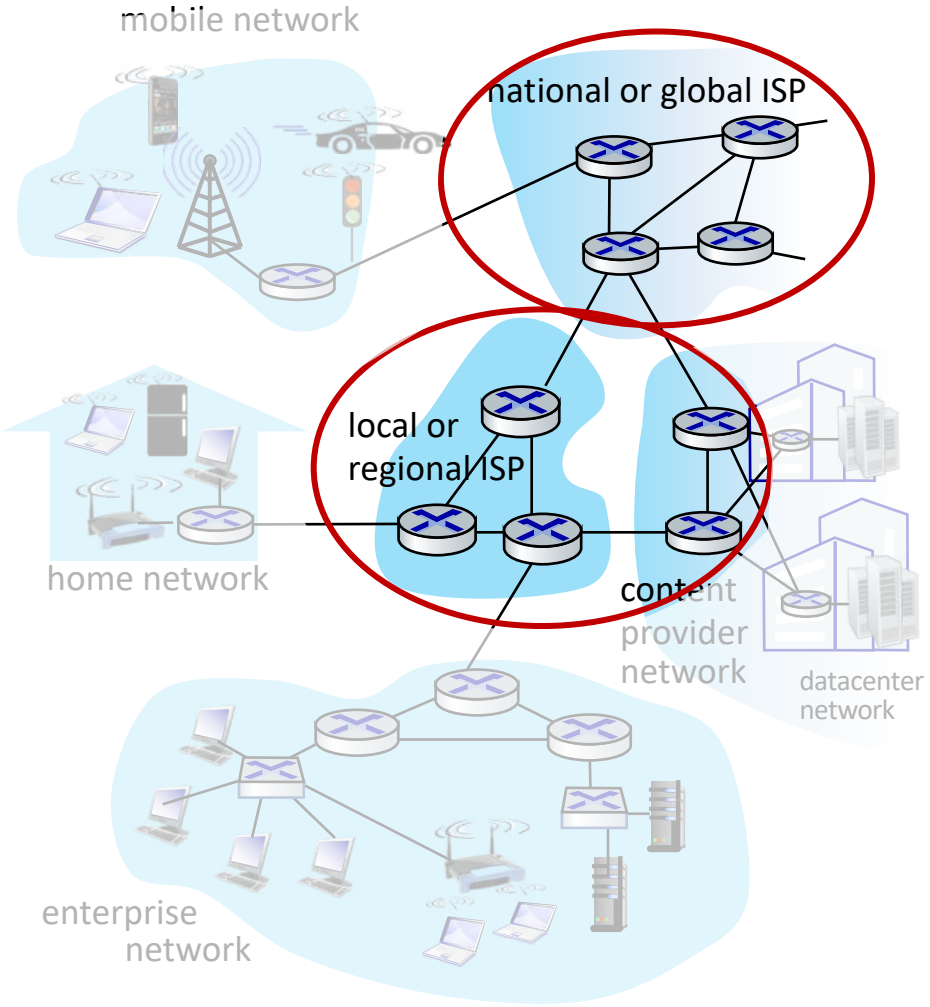
- signal carried in various "bands" in electromagnetic spectrum
- no physical "wire"
- broadcast, "half-duplex" (sender to receiver)
- propagation environment effects:
  - reflection
  - obstruction by objects
  - Interference/noise

Radio link types:

- Wireless LAN (WiFi)
  - 10-100's Mbps; 100's of meters
- wide-area (e.g., 5G cellular)
  - 100'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

The network core

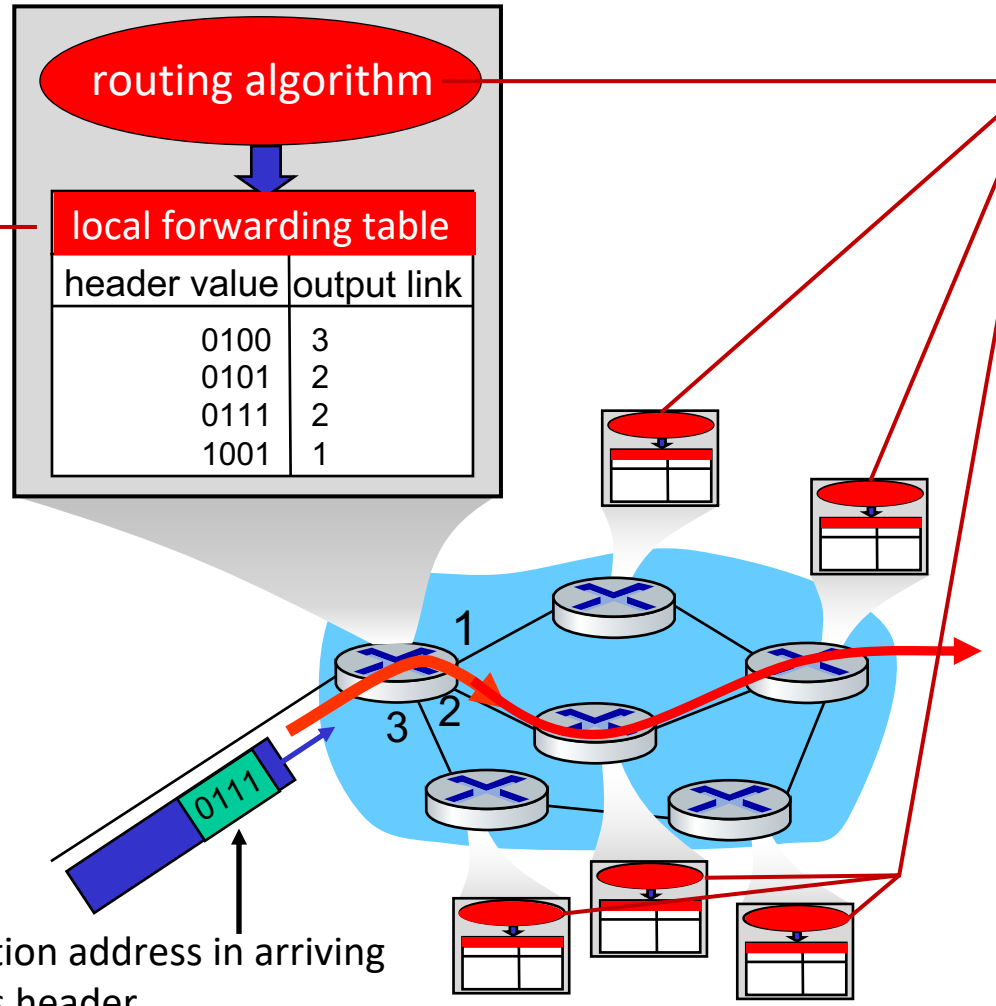
- mesh of interconnected routers
- packet-switching: hosts break application-layer messages into *packets*
  - network forwards packets from one router to the next, across links on path from source to destination



Two key network-core functions

Forwarding:

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



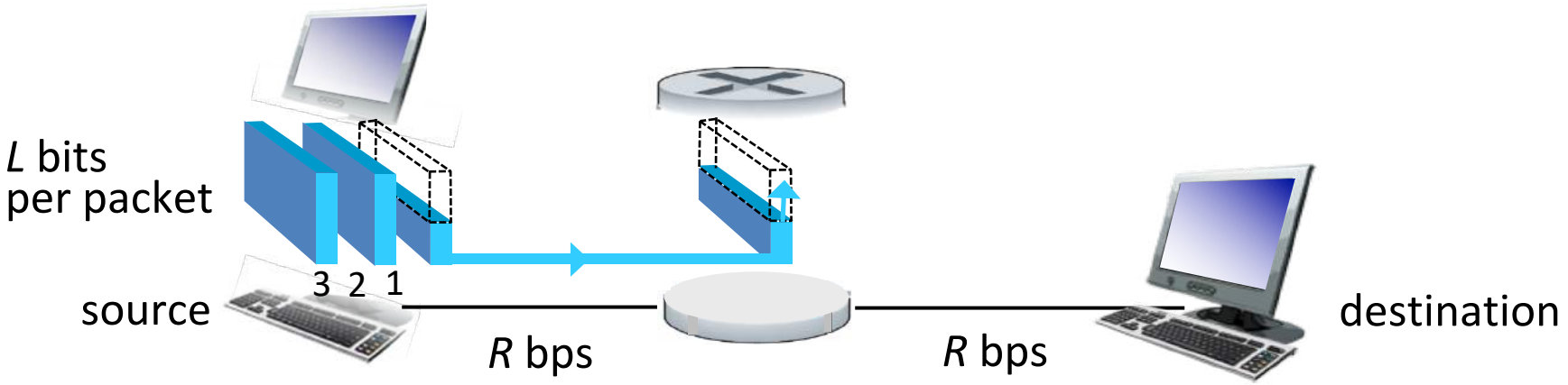
Routing:

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


routing
45 h
3,053 miles
48 h
3,309 miles
United States
Victoria
Seattle
WASHINGTON
Portland
OREGON
NORTH DAKOTA
MONTANA
MINNESOTA
SOUTH DAKOTA
IDAHO
WYOMING
Minneapolis
WISCONSIN
MICHIGAN
Toronto
Quebec City
Montreal
Ottawa
MAINE
VERMONT
NEW HAMPSHIRE
Detroit
Chicago
IOWA
Northampton
CT
RI
PENNSYLVANIA
New York
Philadelphia
NJ
MARYLAND
DE
ILLINOIS
INDIANA
OHIO
Indianapolis
Kansas City
NEVADA
UTAH
COLORADO
MISSOURI
WEST VIRGINIA
KENTUCKY
VIRGINIA
San Francisco
San Jose
CALIFORNIA
Las Vegas
Nashville
TENNESSEE
OKLAHOMA
NORTH CAROLINA
Charlotte
ARKANSAS
Los Angeles
ARIZONA
NEW MEXICO
Atlanta
SOUTH CAROLINA
MISSISSIPPI
ALABAMA
GEORGIA
San Diego
Phoenix
Dallas
Tucson
El Paso
TEXAS
BAJA CALIFORNIA
SONORA
Austin
Houston
LOUISIANA
New Orleans
Jacksonville
CHIHUAHUA
San Antonio
Orlando
Tampa
FLORIDA
COAHUILA


forwarding
forwarding
45 h
3,053 miles
48 h
3,309 miles
United States
Victoria
Seattle
WASHINGTON
Portland
NORTH DAKOTA
MINNESOTA
Minneapolis
SOUTH DAKOTA
WISCONSIN
MICHIGAN
Toronto
Quebec City
Montreal
Ottawa
MAINE
VERMONT
NEW HAMPSHIRE
Detroit
Chicago
Northampton
WYOMING
IOWA
ILLINOIS
INDIANA
OHIO
Kansas City
Indianapolis
COLORADO
KANSAS
MISSOURI
KENTUCKY
Nashville
TENNESSEE
San Francisco
San Jose
CALIFORNIA
Las Vegas
OKLAHOMA
ARKANSAS
Atlanta
Los Angeles
ARIZONA
NEW MEXICO
Dallas
MISSISSIPPI
ALABAMA
San Diego
Phoenix
Tucson
El Paso
TEXAS
BAJA CALIFORNIA
SONORA
Austin
Houston
LOUISIANA
New Orleans
CHIHUAHUA
San Antonio
COAHUILA
Tampa
FLORIDA
Cleveland
DOWNTOWN
CENTRAL
OHIO CITY

Packet-switching: store-and-forward

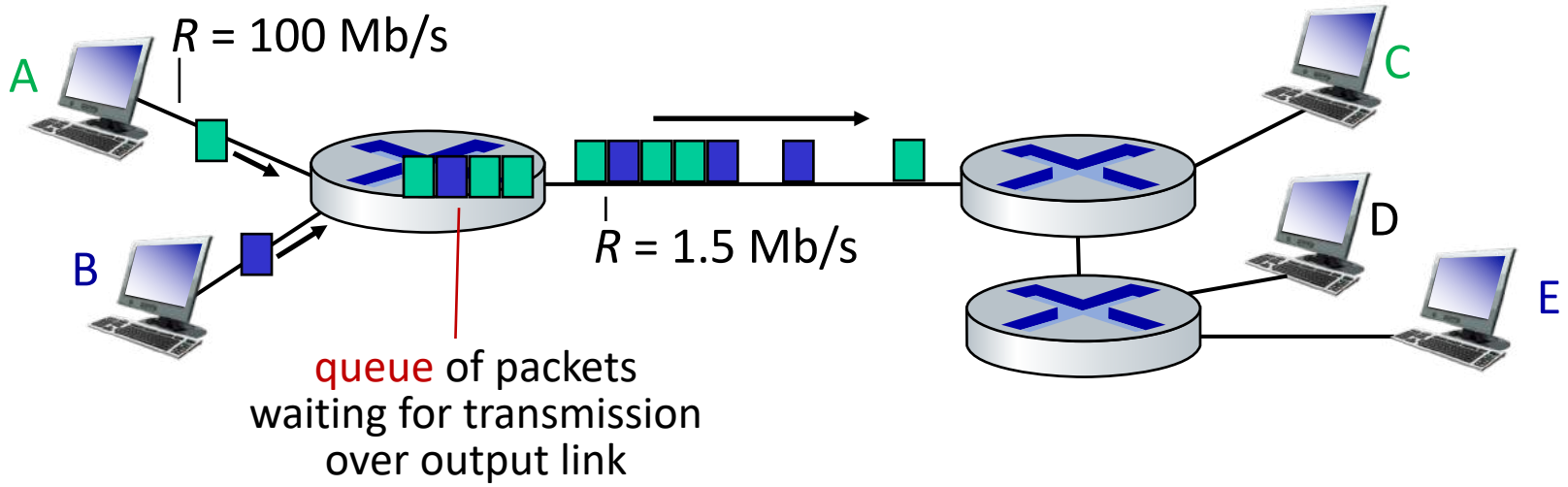


- packet transmission delay: takes L/R seconds to transmit (push out) L-bit packet into link at R bps
- *store and forward: entire* packet must arrive at router before it can be transmitted on next link

One-hop numerical example:

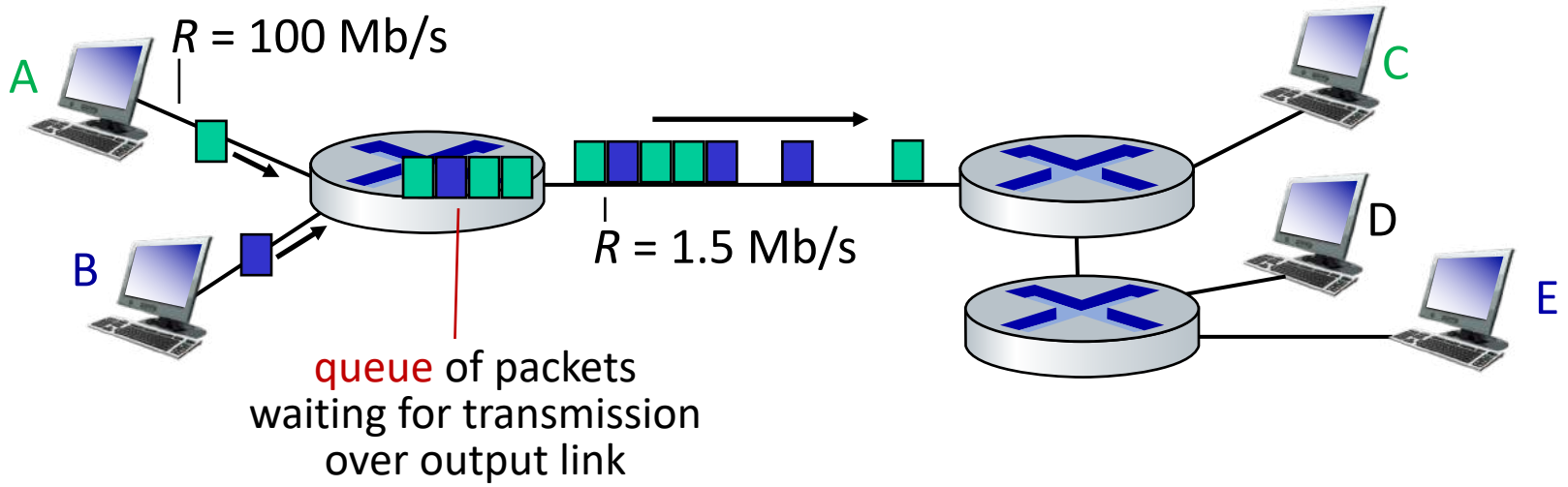
- L = 10 Kbits
- R = 100 Mbps
- one-hop transmission delay = 0.1 msec

Packet-switching: queueing



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

Packet-switching: queueing



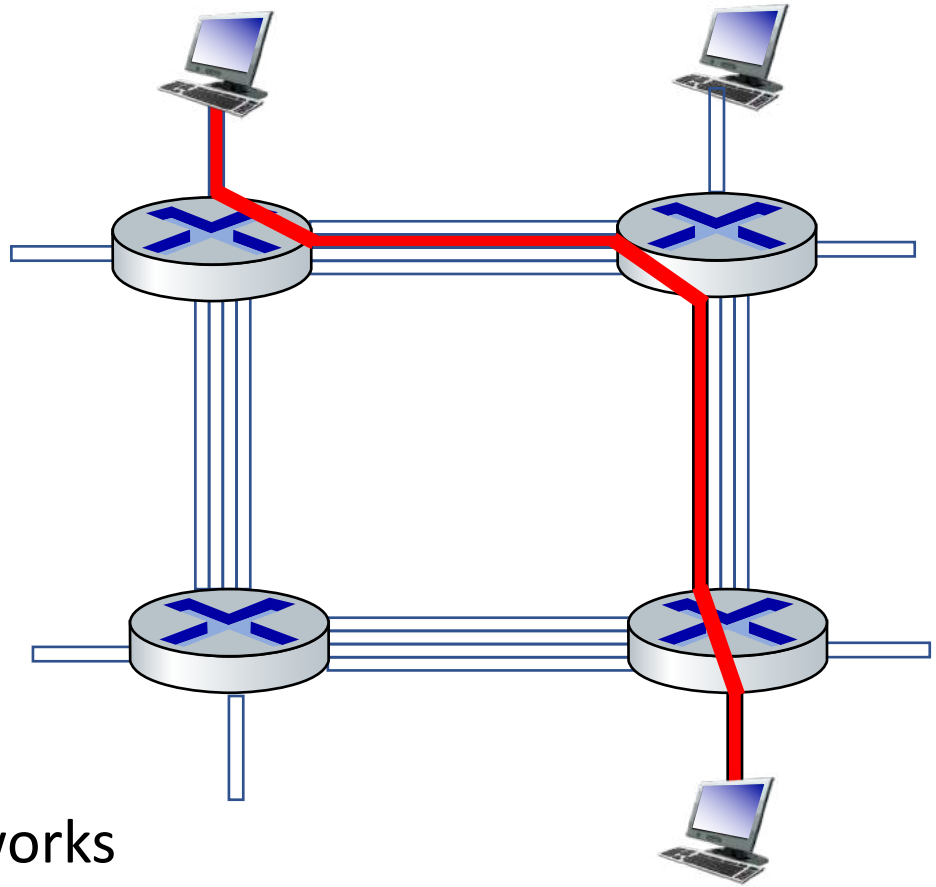
Packet queuing and loss: if arrival rate (in bps) to link exceeds transmission rate (bps) of link for some period of time:

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

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

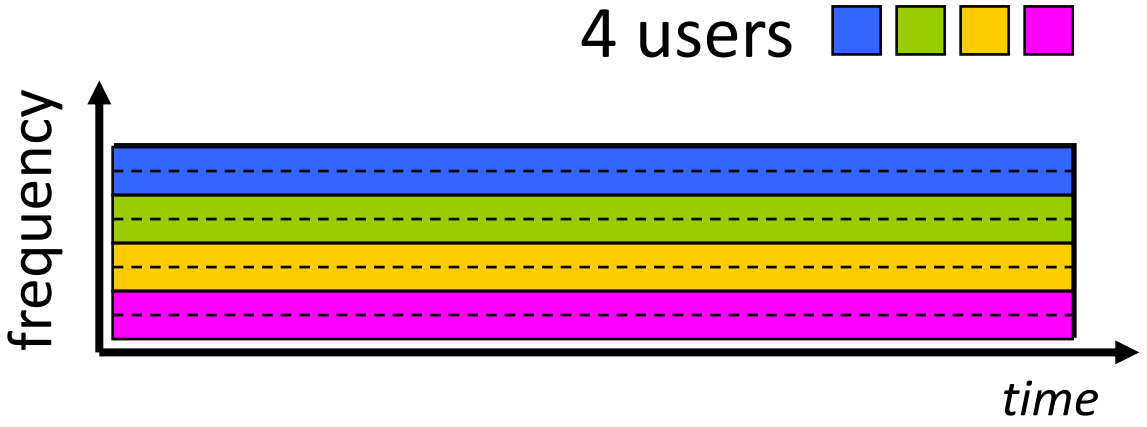


* Check out the online interactive exercises for more examples: http://gaia.cs.umass.edu/kurose_ross/interactive

Circuit switching: FDM and TDM

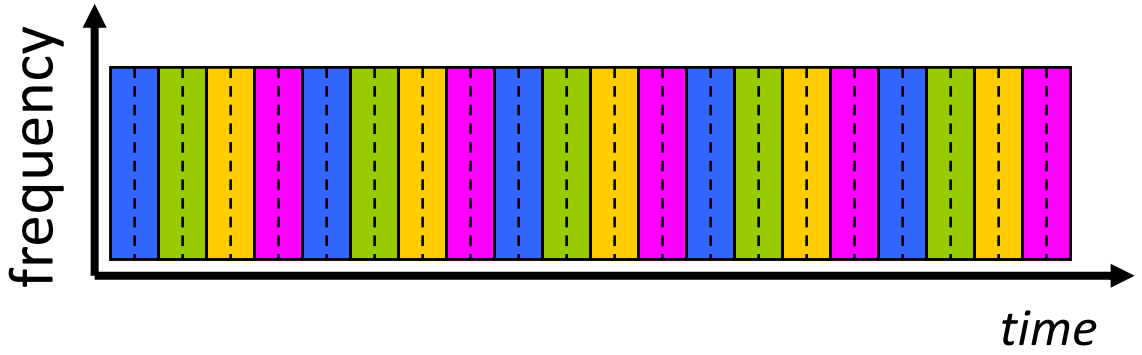
Frequency Division Multiplexing (FDM)

- 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)

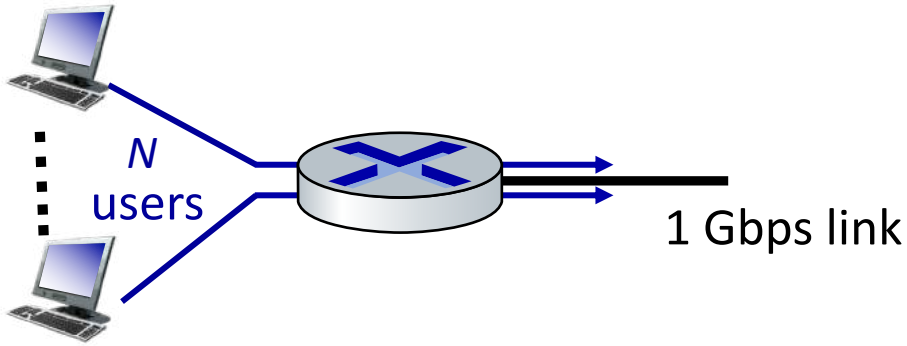
- time divided into slots
- each call allocated periodic slot(s), can transmit at maximum rate of (wider) frequency band (only) during its time slot(s)



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)

* Check out the online interactive exercises for more examples: http://gaia.cs.umass.edu/kurose_ross/interactive

Packet switching versus circuit switching

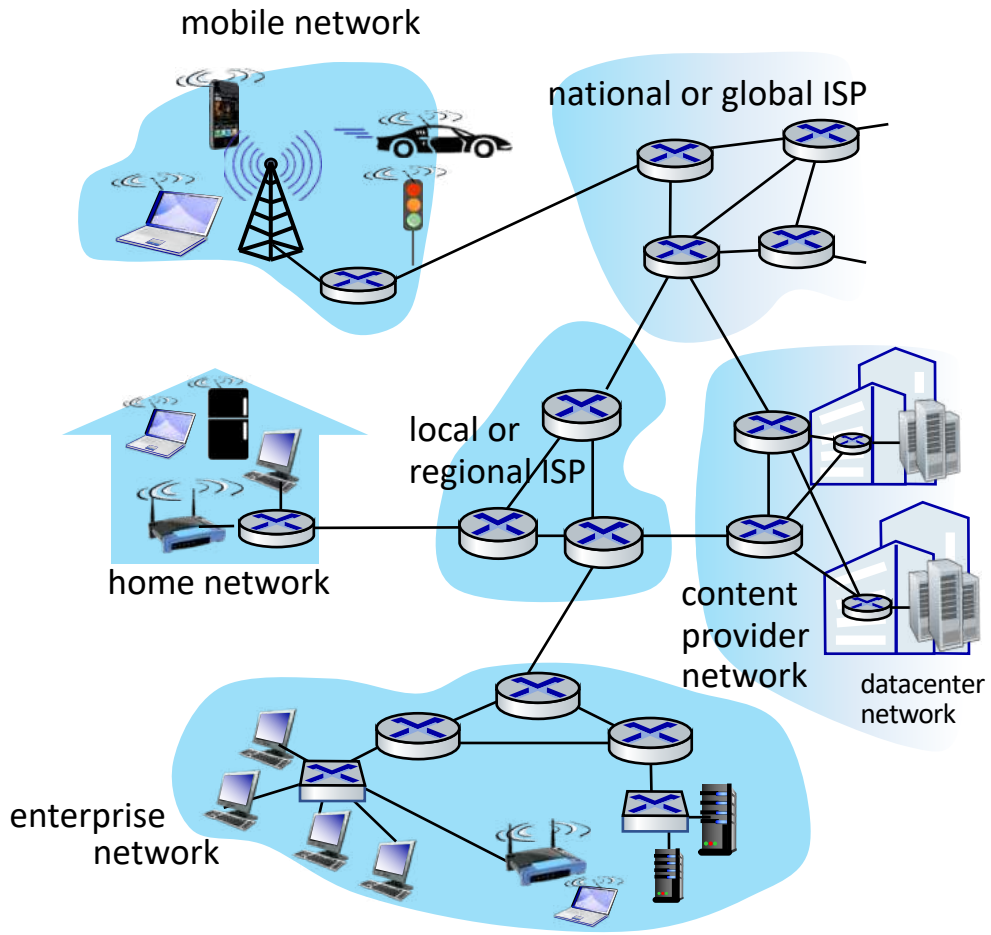
Is packet switching a “slam dunk winner”?

- 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
- *Q:* How to provide circuit-like behavior with packet-switching?
  - “It’s complicated.” We’ll study various techniques that try to make packet switching as “circuit-like” as possible.

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

Internet structure: a "network of networks"

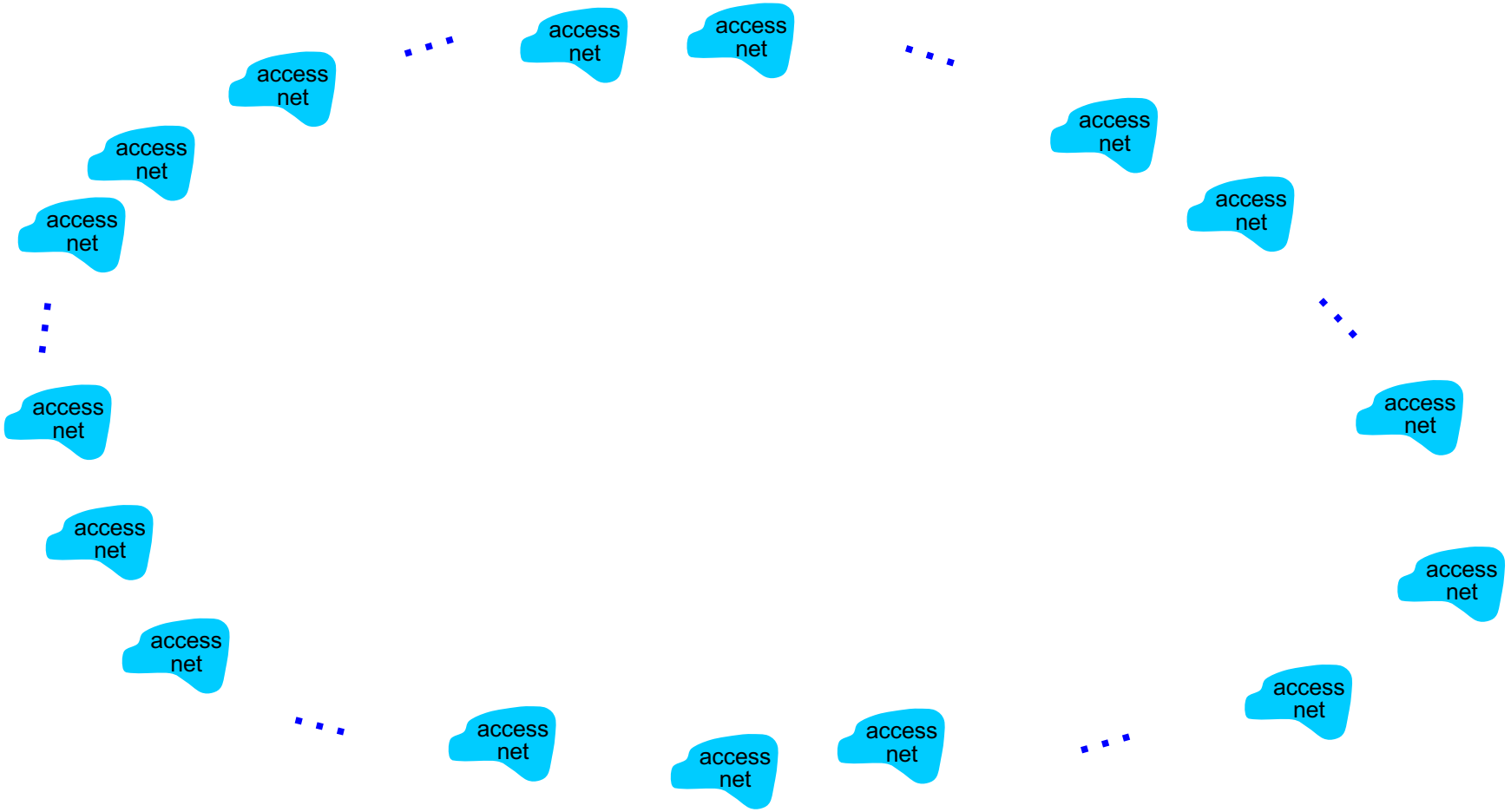
- hosts connect to Internet via access Internet Service Providers (ISPs)
- access ISPs in turn must be interconnected
  - so that *any* two hosts *(anywhere!)* can send packets to each other
- resulting network of networks is very complex
  - evolution driven by economics, national policies



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

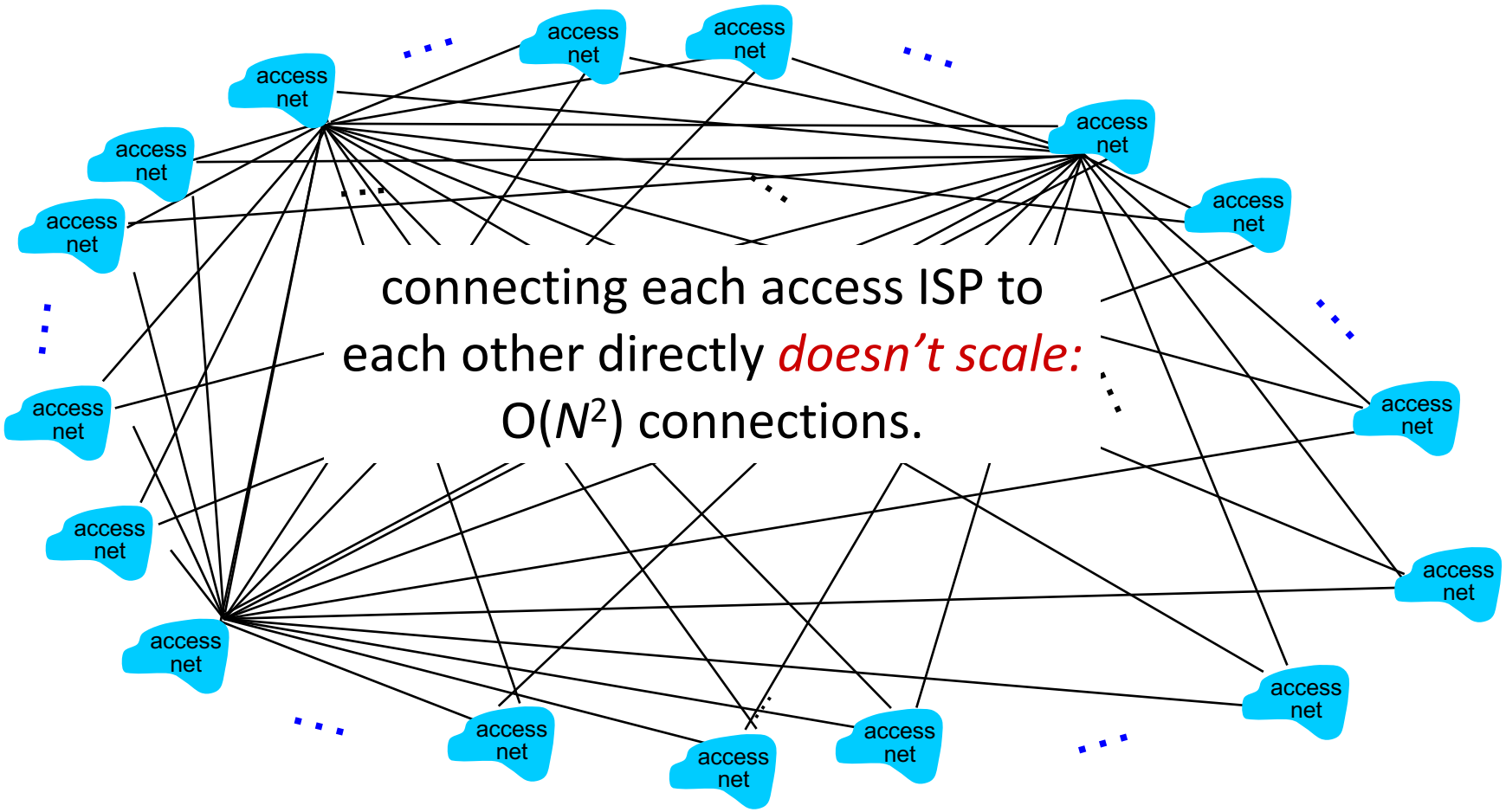
Internet structure: a "network of networks"

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



Internet structure: a "network of networks"

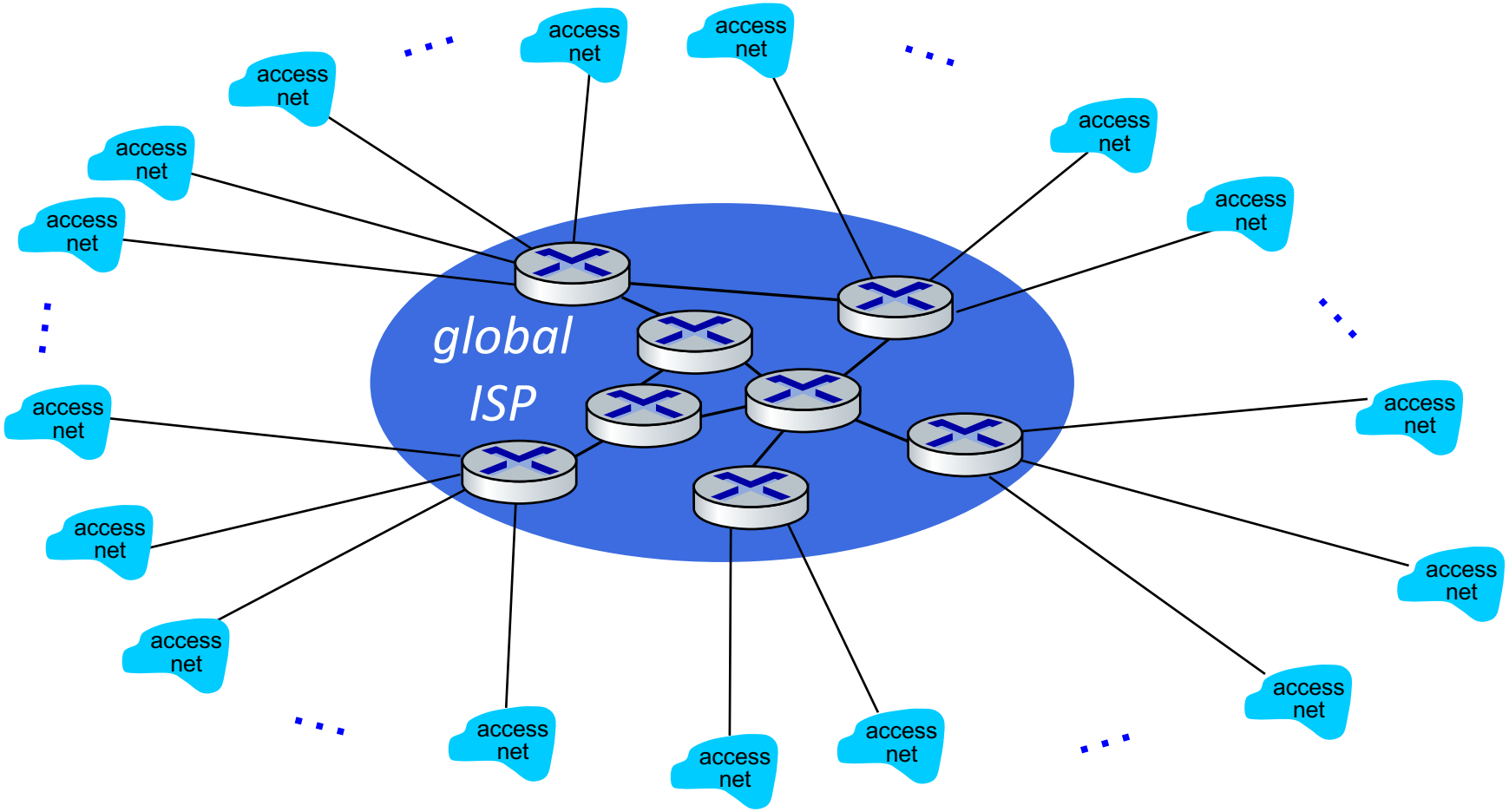
Question: given *millions* of access ISPs, how to connect them together?



Internet structure: a "network of networks"

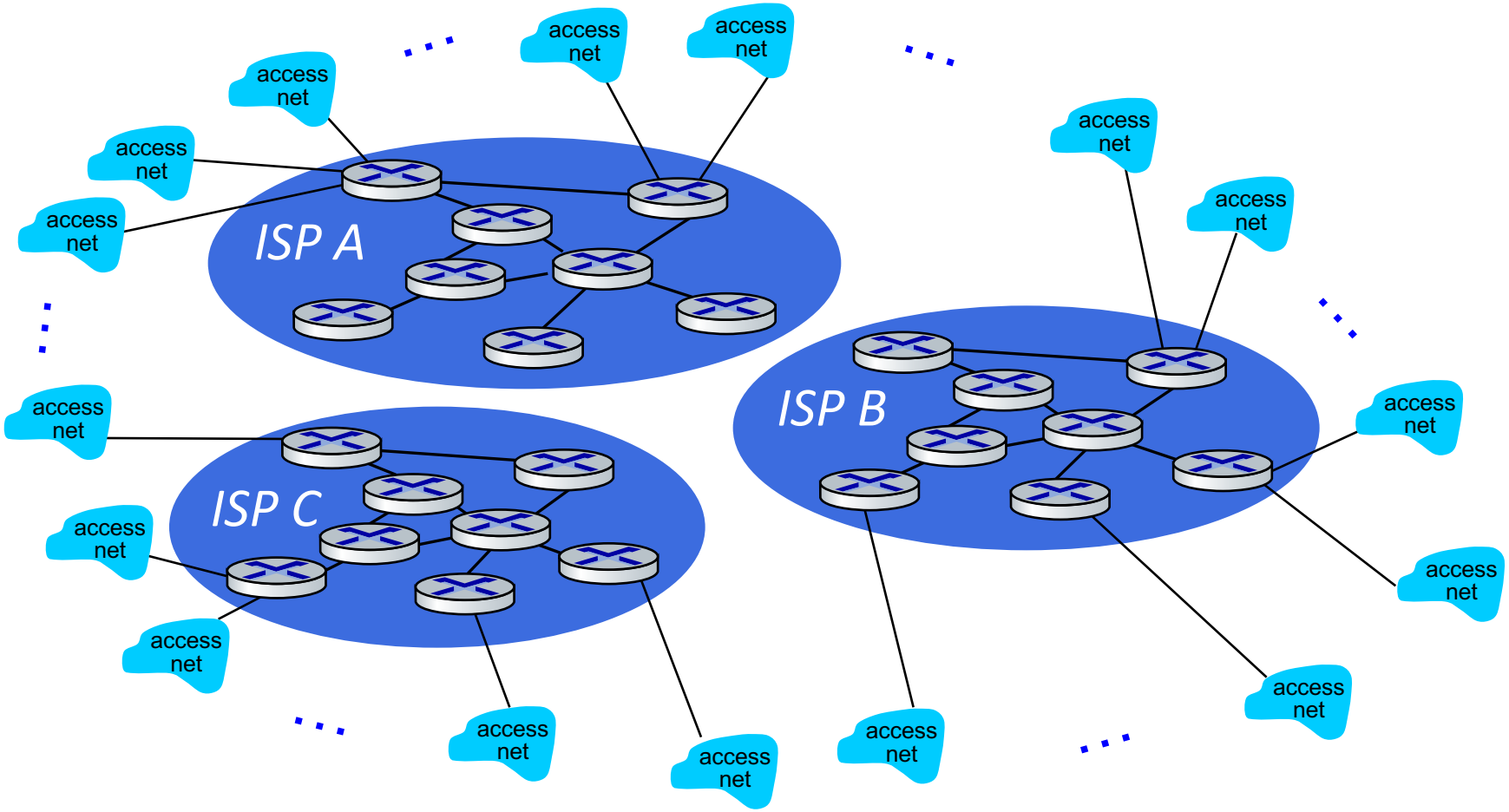
Option: connect each access ISP to one global transit ISP?

Customer and provider ISPs have economic agreement.



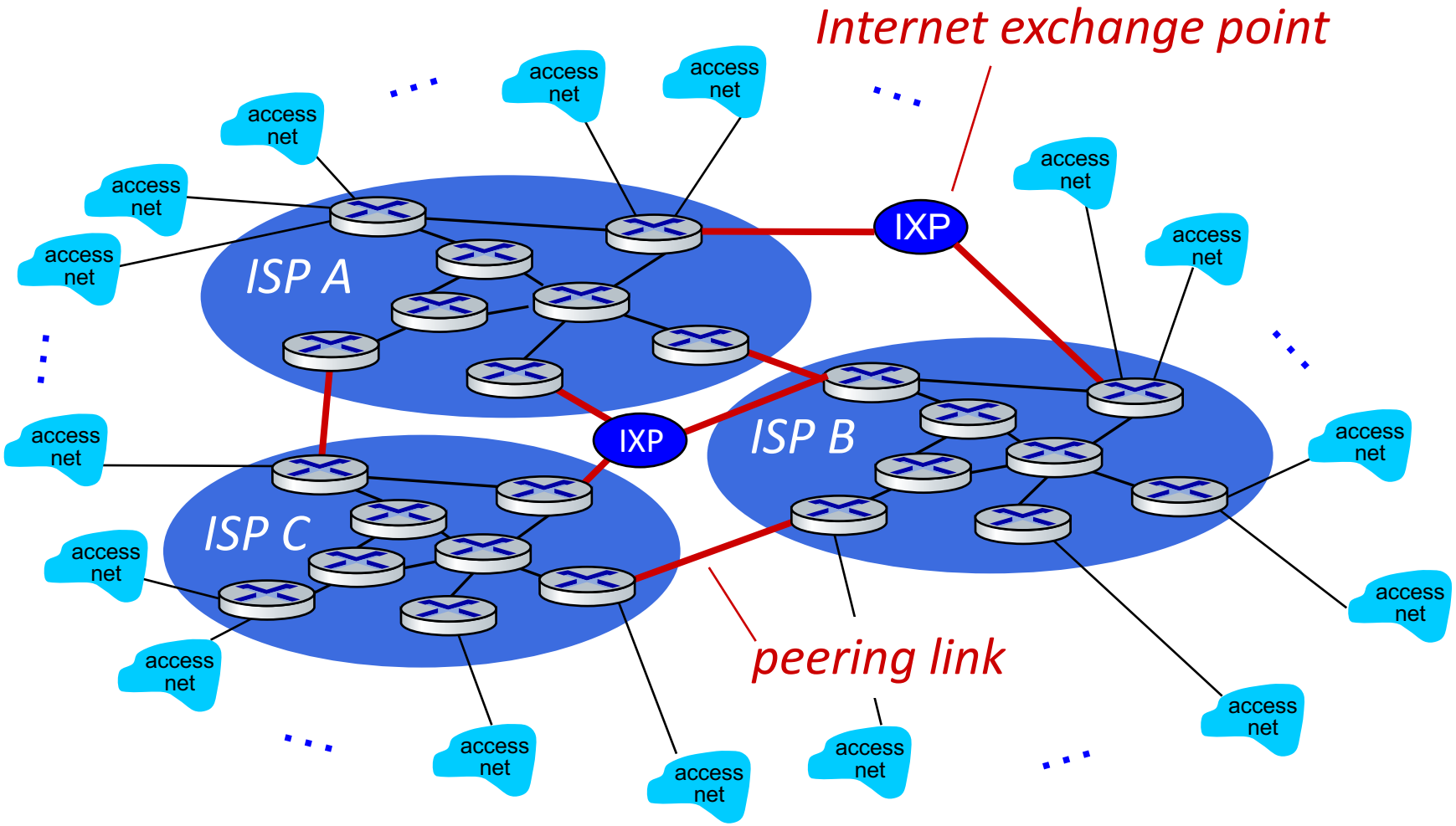
Internet structure: a "network of networks"

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



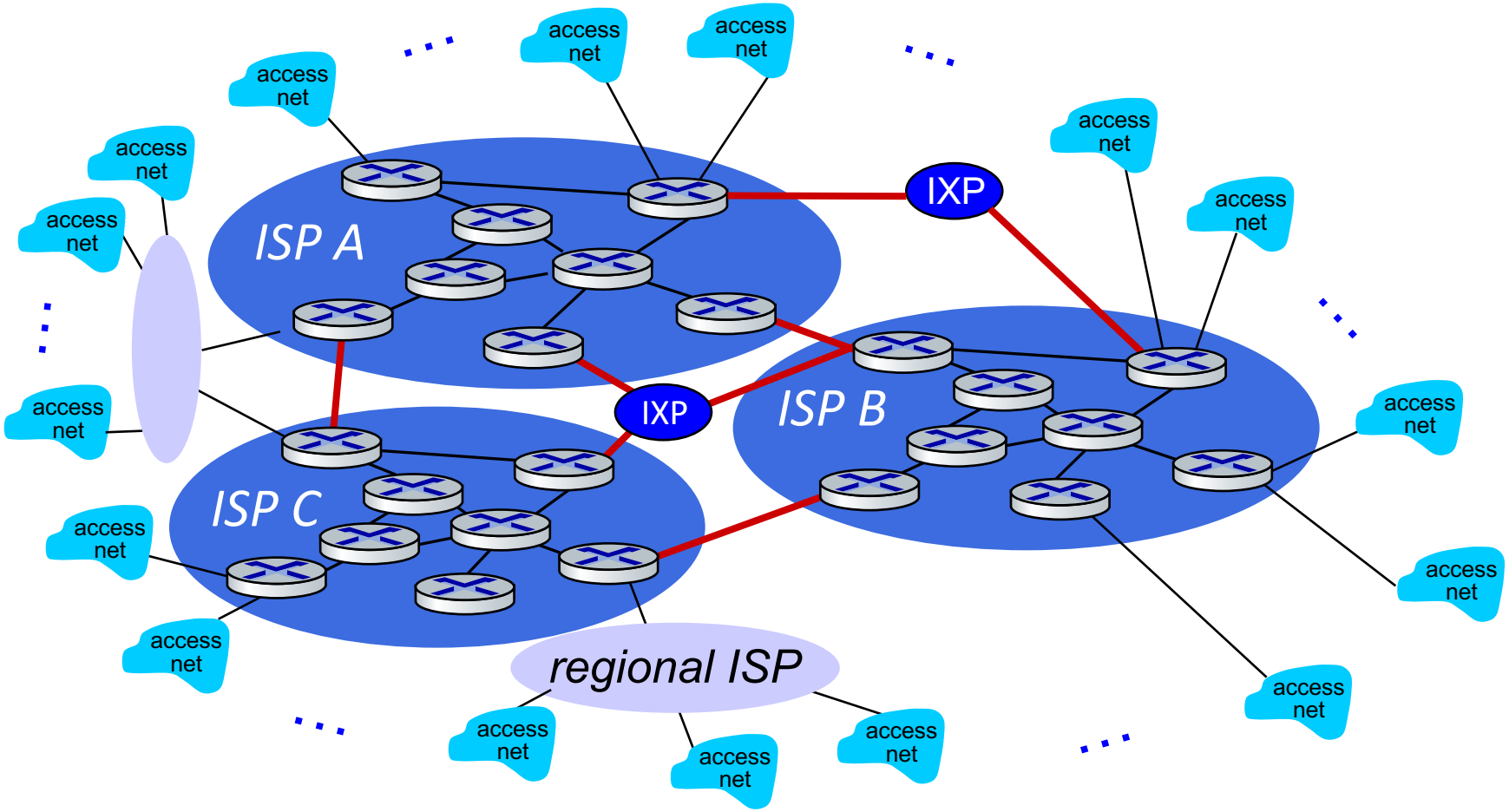
Internet structure: a "network of networks"

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



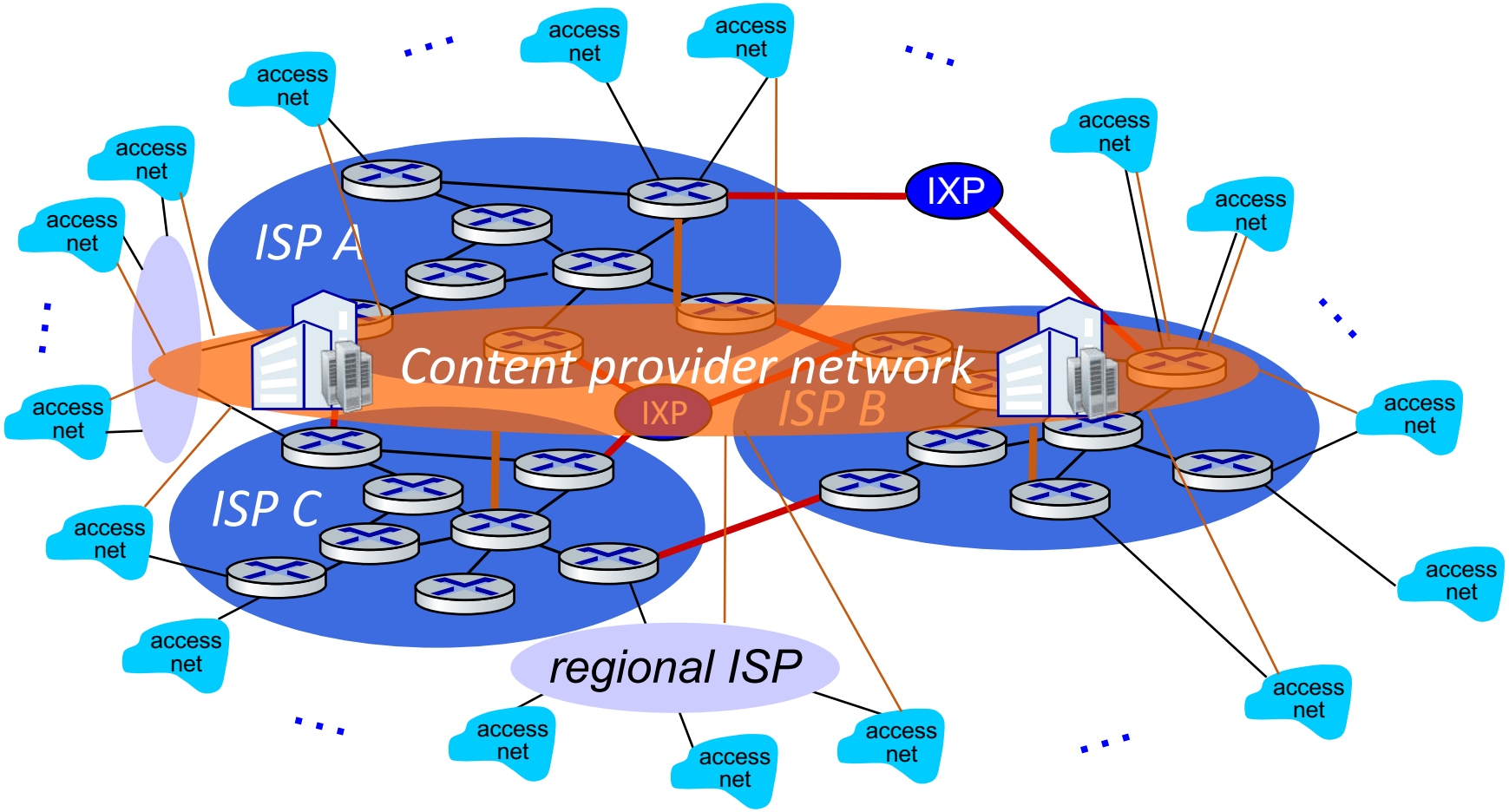
Internet structure: a "network of networks"

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

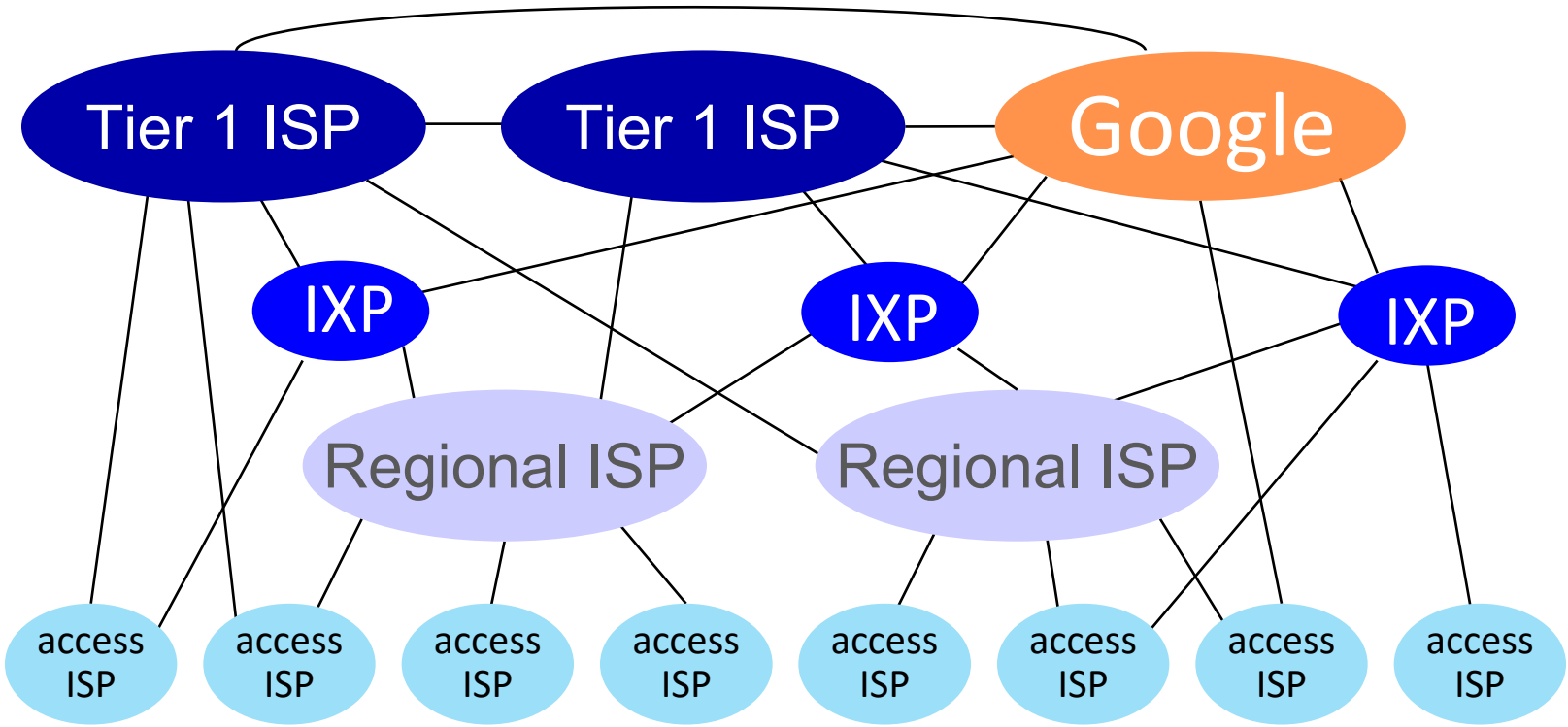


Internet structure: a "network of networks"

… 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"

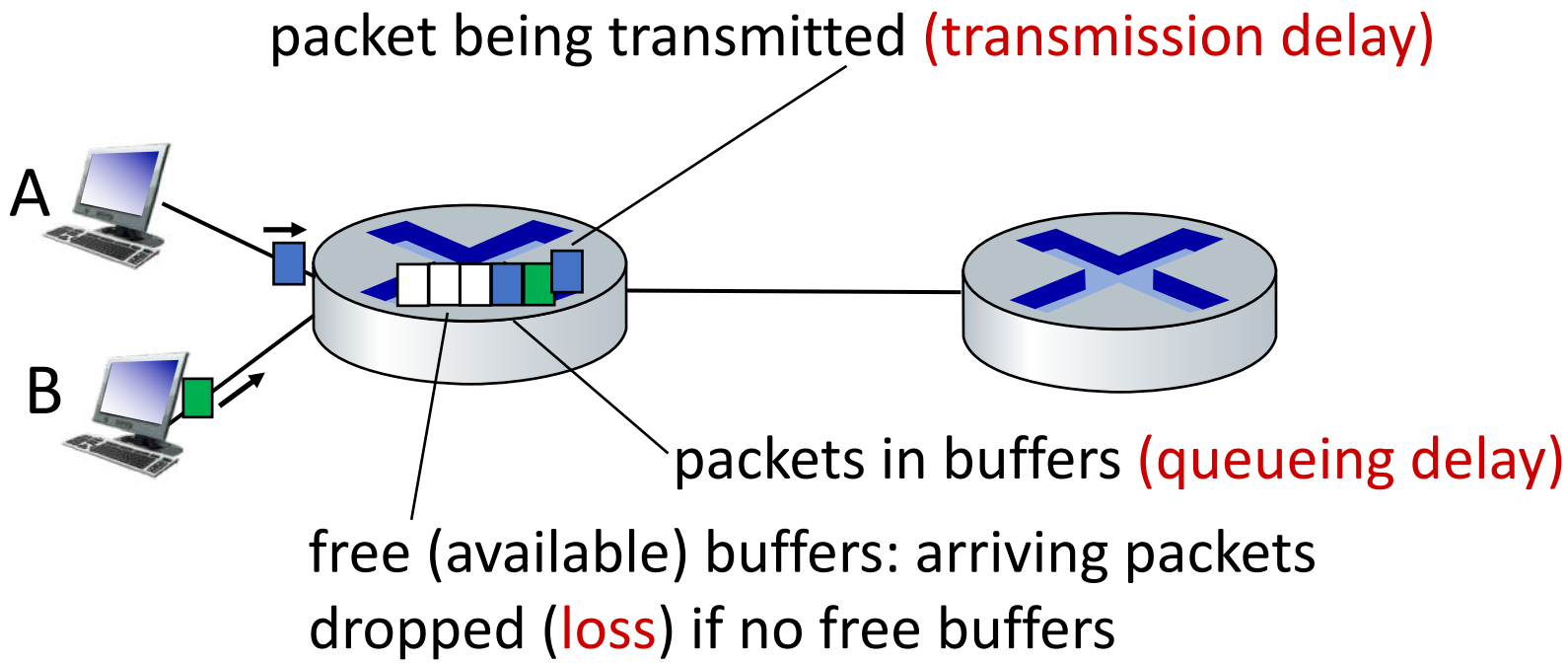


At "center": small # of well-connected large networks

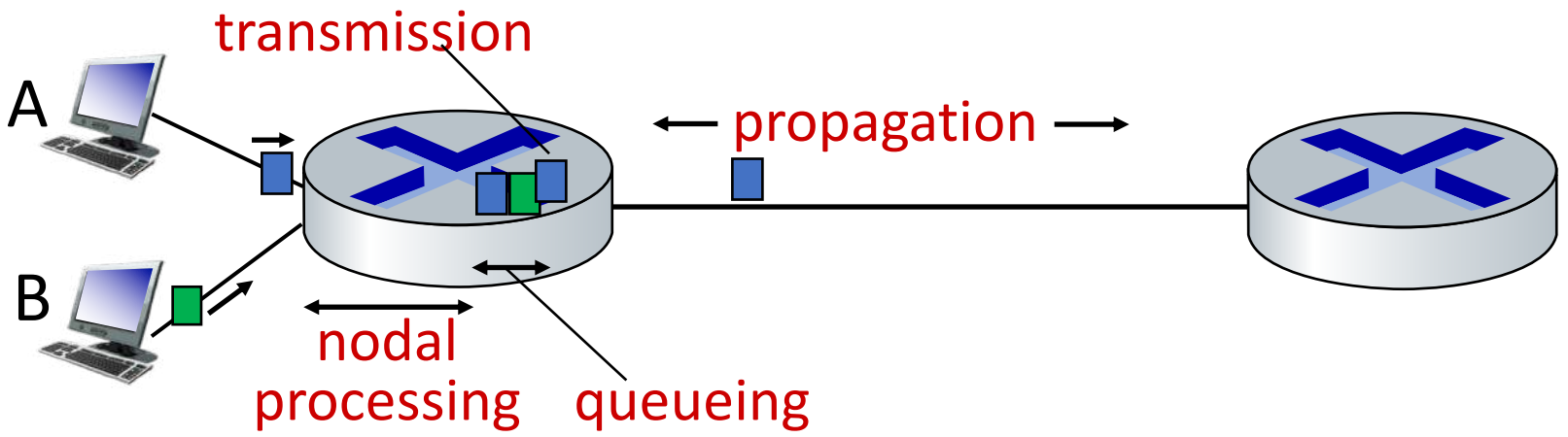
- "tier-1" commercial ISPs (e.g., Level 3, Sprint, AT&T, NTT), national & international coverage
- content provider networks (e.g., Google, Facebook): private network that connects its data centers to Internet, often bypassing tier-1, regional ISPs

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



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

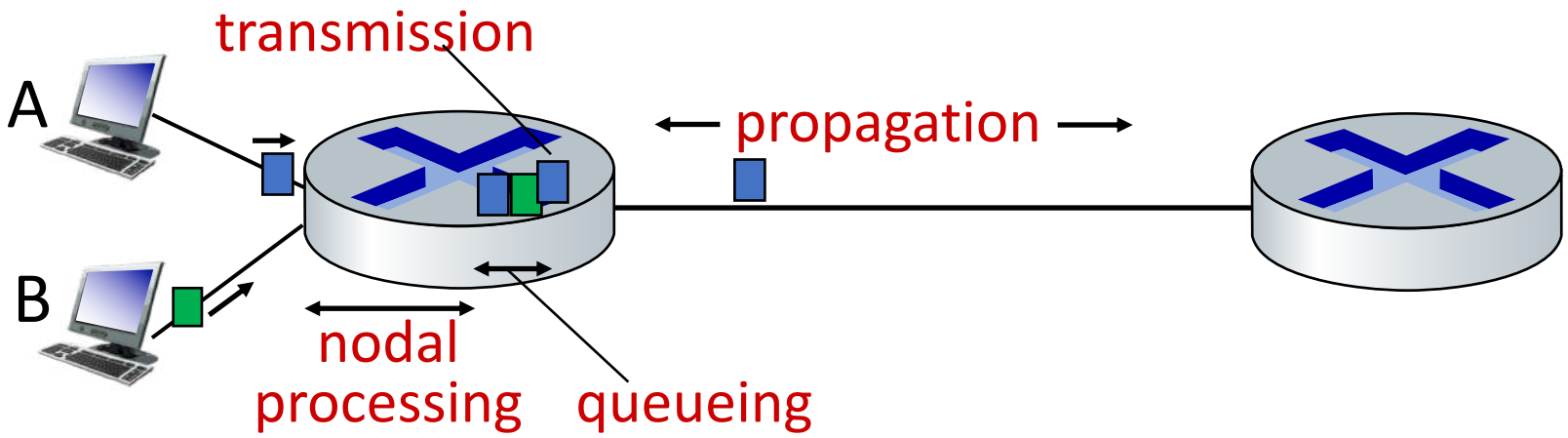
d_{proc}: nodal processing

- 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

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 (~2×10^8 m/sec)
- $d_{prop} = d/s$

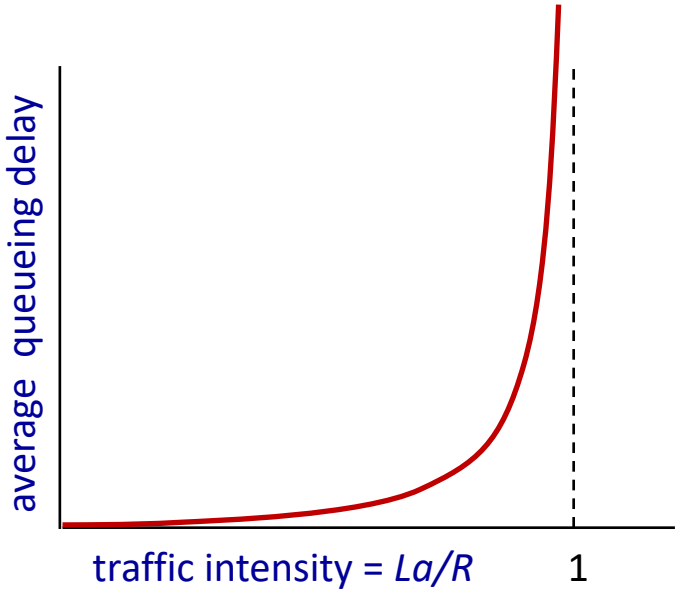
d_{trans} and d_{prop} *very* different

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"



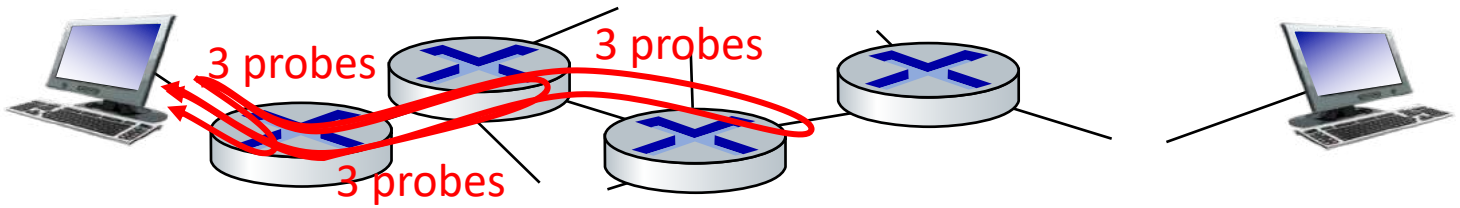
- *La/R* ~ 0: avg. queueing delay small
- *La/R* -> 1: avg. queueing delay large
- *La/R* > 1: more "work" arriving is more than can be serviced - average delay infinite!

La/R ~ 0

La/R -> 1

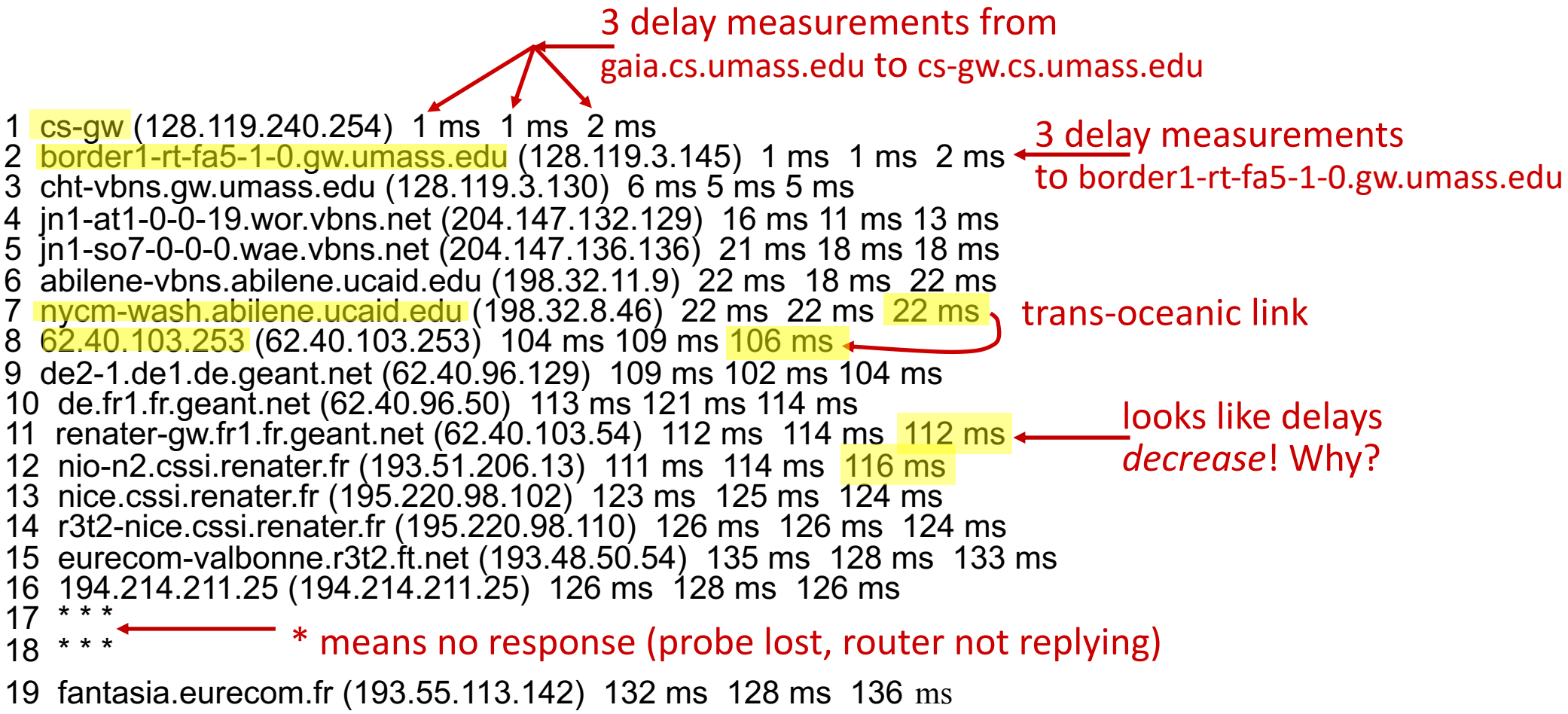
“Real” Internet delays and routes

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



Real Internet delays and routes

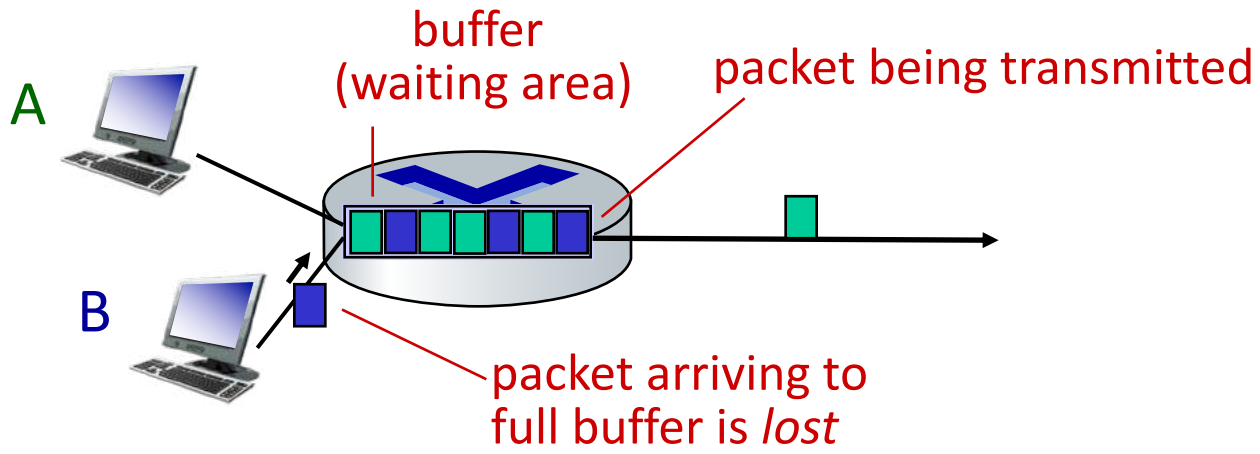
traceroute: gaia.cs.umass.edu to www.eurecom.fr



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

Packet loss

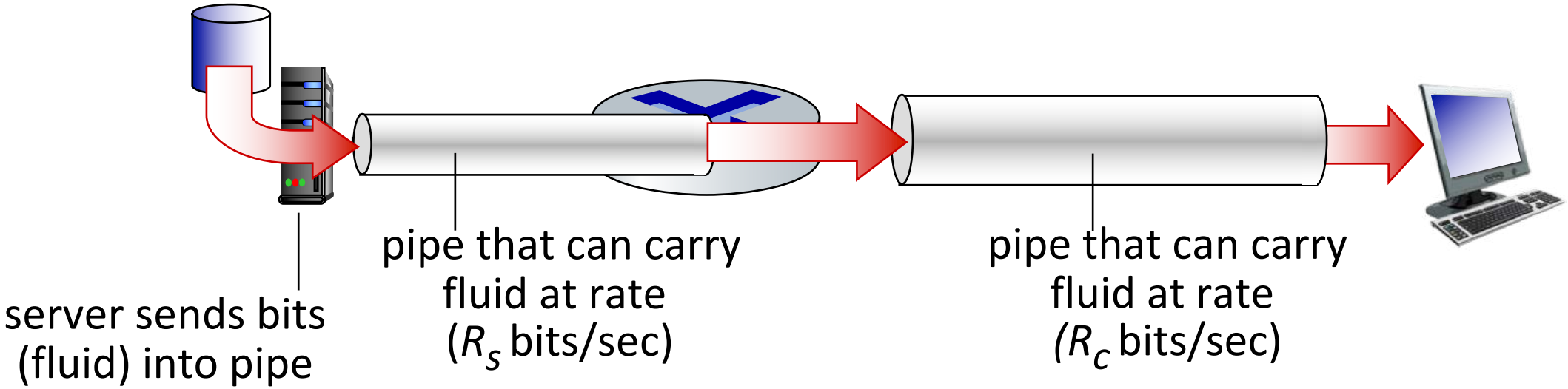
- queue (aka buffer) preceding link in buffer has finite capacity
- packet arriving to full queue dropped (aka lost)
- lost packet may be retransmitted by previous node, by source end system, or not at all



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

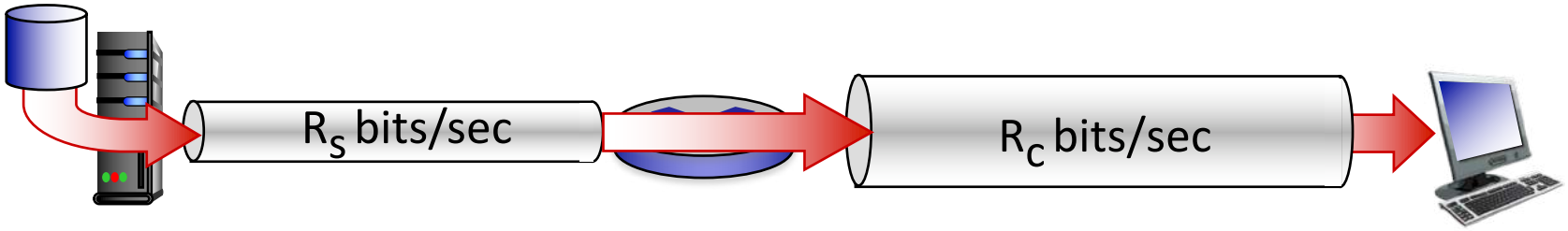
Throughput

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

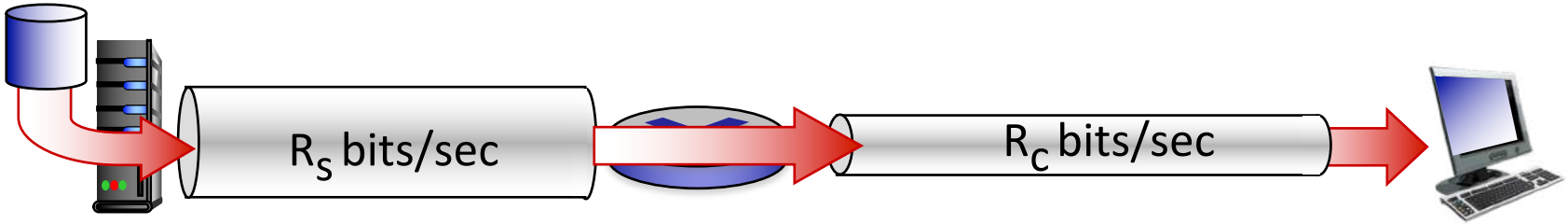


Throughput

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



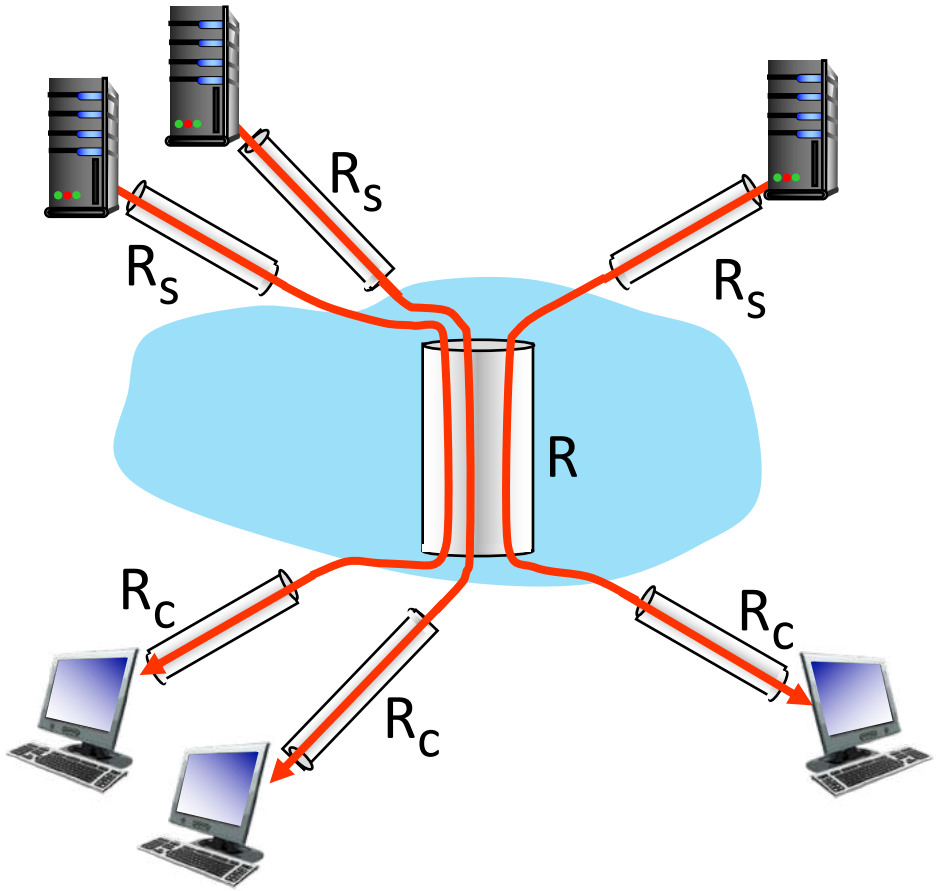
$R_s > R_c$ What is average end-end throughput?



bottleneck link

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

Throughput: network scenario



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

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

* Check out the online interactive exercises for more examples: http://gaia.cs.umass.edu/kurose_ross/

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

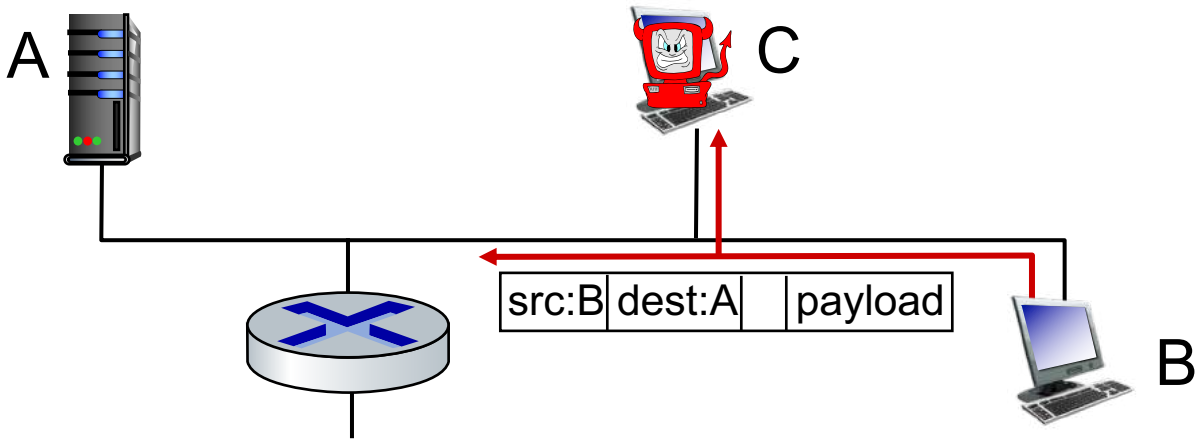
Network security

- 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: packet interception

packet "sniffing":

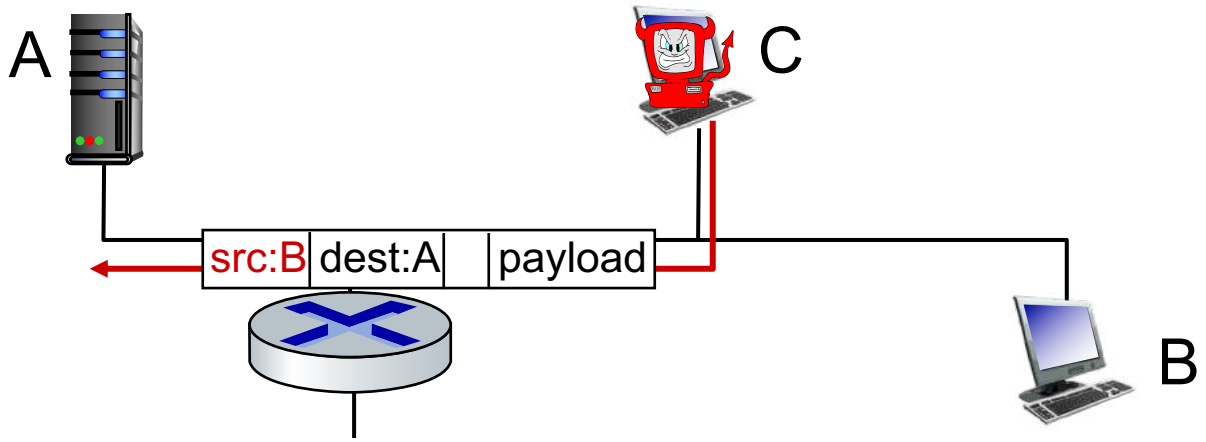
- broadcast media (shared Ethernet, wireless)
- promiscuous network interface reads/records all packets (e.g., including passwords!) passing by



Wireshark software used for our end-of-chapter labs is a (free) packet-sniffer

Bad guys: fake identity

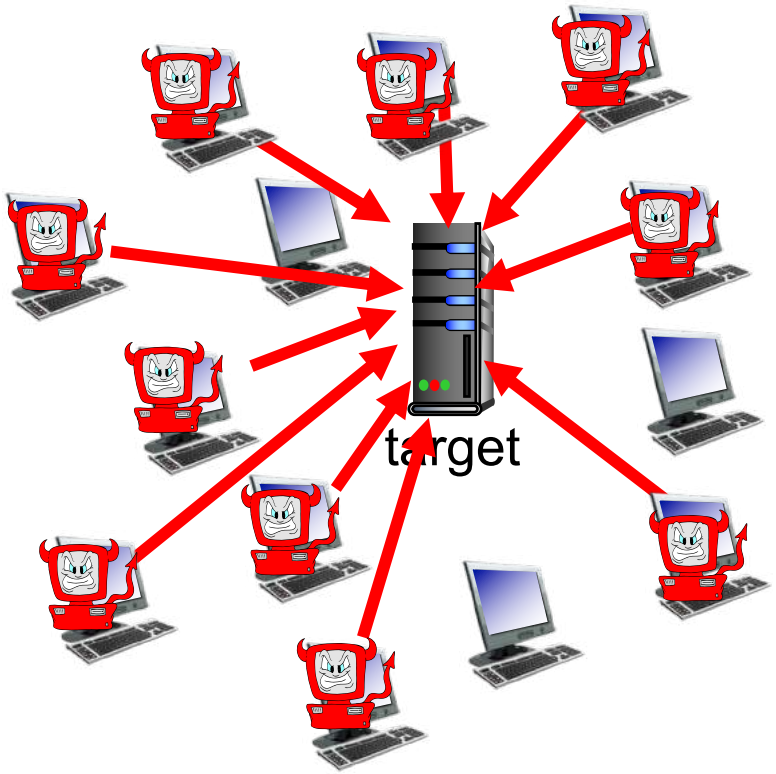
IP spoofing: injection of packet with false source address



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 (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)

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?

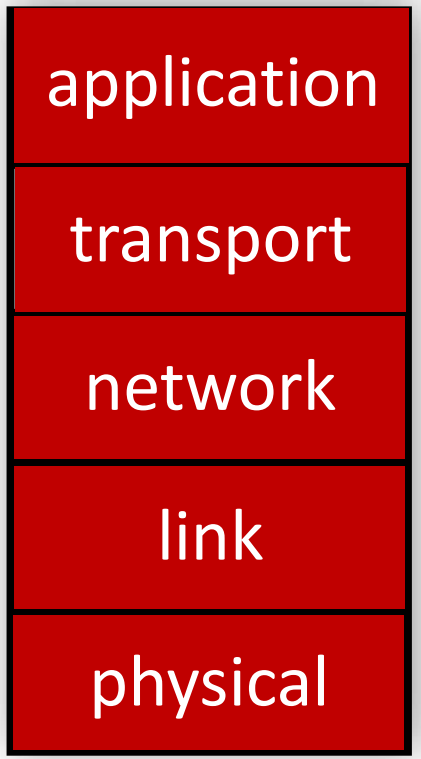
Why layering?

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”



Layers and Their Functionalities

❑ The Physical Layer (Following parameters are specified)

- Voltage and current levels
- Timings of voltage changes – how many microsec a bit occupies
- Physical data rates
- Maximum transmission distances
- Physical Connectivity – RJ45, SFP etc.
- Connection types - (i) Point-to-point (ii) Multipoint
- Physical topology – (i) Bus, (ii) Ring, (iii) Star (iv) Mesh etc. **(Note: Physical topology indicates actual physical connectivity)**
- Digital and Analog signalling
- Bit synchronization – (i) Synchronous (ii) Asynchronous
- Bandwidth usage – (i) Broadband (ii) Baseband
- Multiplexing on – (i) Frequency (ii) Time (iii) Statistical time division

❑ Network components

- Connectors and cables
- Electrical and data communication interfaces – NICs
- Concentrators, hubs and repeaters
- Modems, Transmission media converters etc.

Layers and Their Functionalities – Contd.

- The Link Layer
  - Media Access Control (MAC) Sub Layer – controlling the transmission
    - Logical topology – Bus, Ring
    - Media Access (i.e. Contention, Token passing, polling)
    - Addressing methodology – w.r.t. the actual physical device
  - Logical Link Control (LLC) sub layer – establishes and maintains the link for transmitting data frames from one device to another.
    - Transmission synchronization – Synchronous, Asynchronous, Isochronous
    - Connection services – Flow control and error handling
  - Responsibilities
    - Organizing the 1's and 0's supplied by physical layer into groups of logical information called 'frames'
    - Utilization of line links
    - Error notification (and correction)
    - Ordered delivery of frames
  - Network components

- Bridges
- Network interface boards
- Switches

Layers and Their Functionalities – Contd.

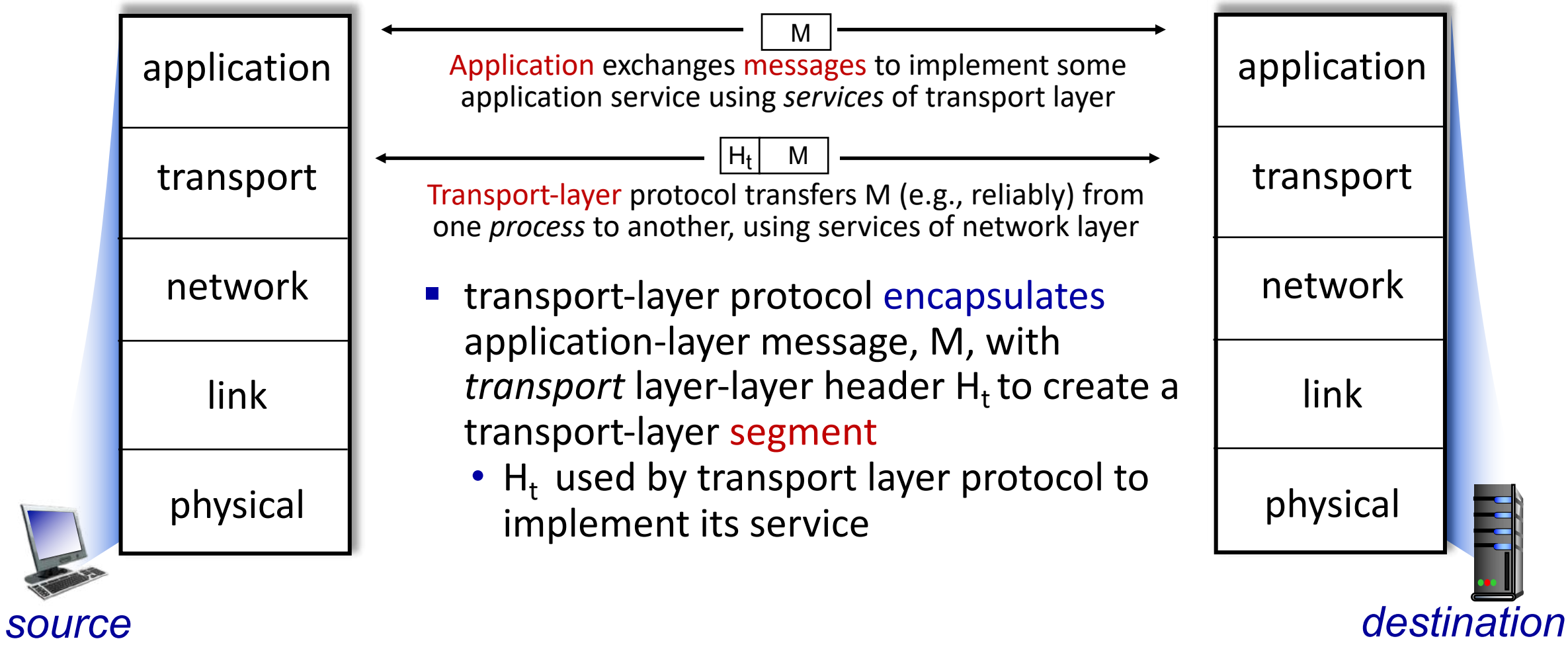
- The Network Layer
  - Providing connectivity and path selection between two end systems that may be located on geographically diverse ‘subnetworks’
  - ROUTING
  - Responsibilities
    - Addressing – Logical Network address and service address
    - Switching
      - Circuit Switching
      - Message switching
      - Packet Switching
    - Route analysis
    - Route selection
    - Connection services – Network layer flow, error handling and packet sequence control
    - Network services – Network layer translation.
  - Network components

- Router
- Layer-3 switches

Layers and Their Functionalities – Contd.

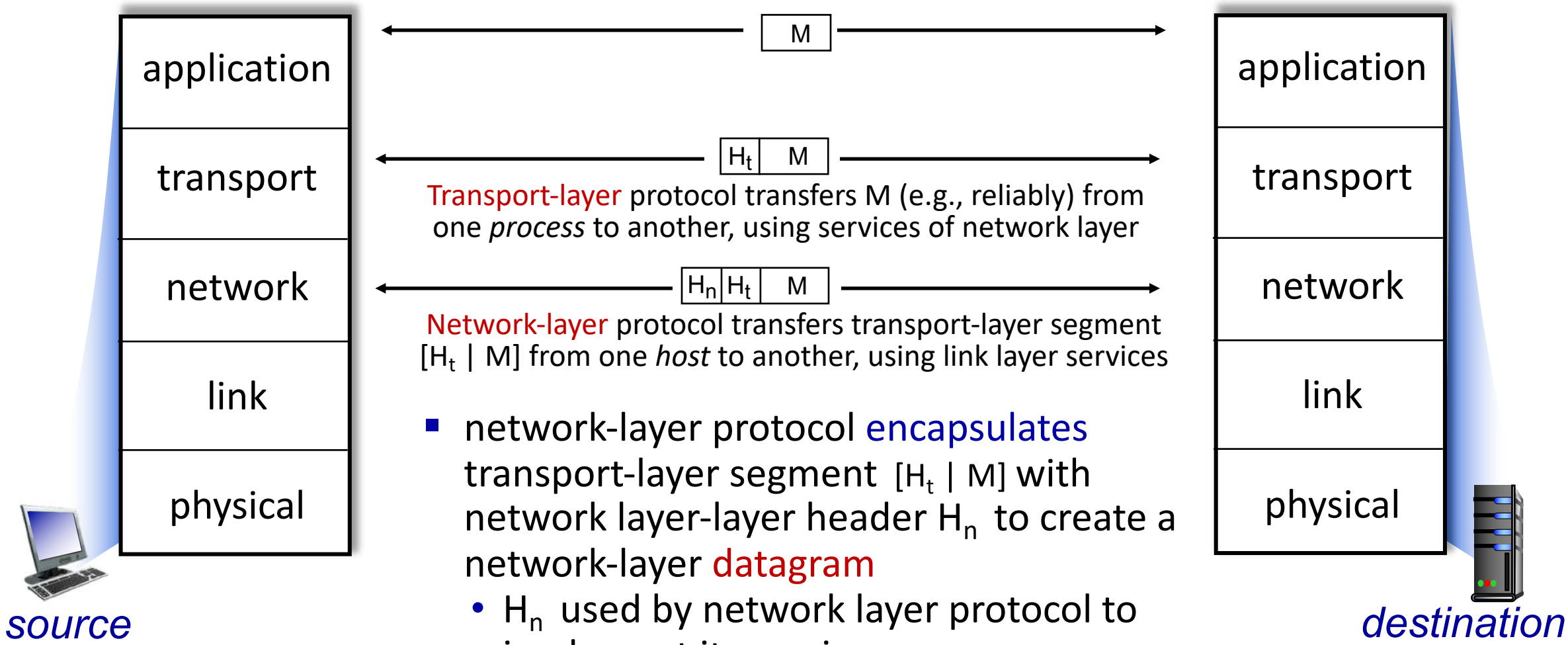
- The Transport Layer
  - Address resolution
  - Establishment, maintenance and termination of Virtual Circuits (VCs)
  - Fault detection during transportation and recovery procedures i.e. preservation of data integrity
  - Information flow control.
  - Prevention of message duplication and losses
  - Proper identification and addressing of user equipment.

Services, Layering and Encapsulation



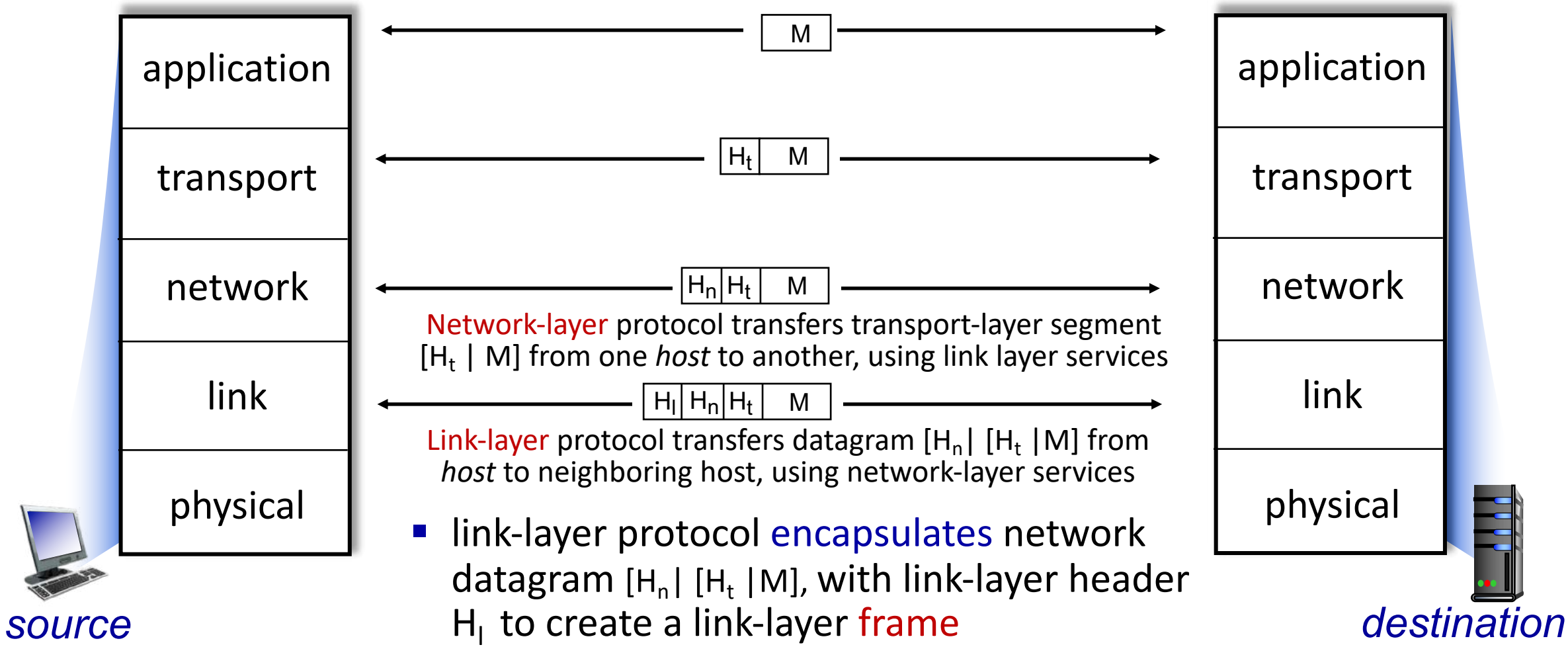
- transport-layer protocol encapsulates application-layer message, M, with *transport* layer-layer header H_t to create a transport-layer segment
  - H_t used by transport layer protocol to implement its service

Services, Layering and Encapsulation



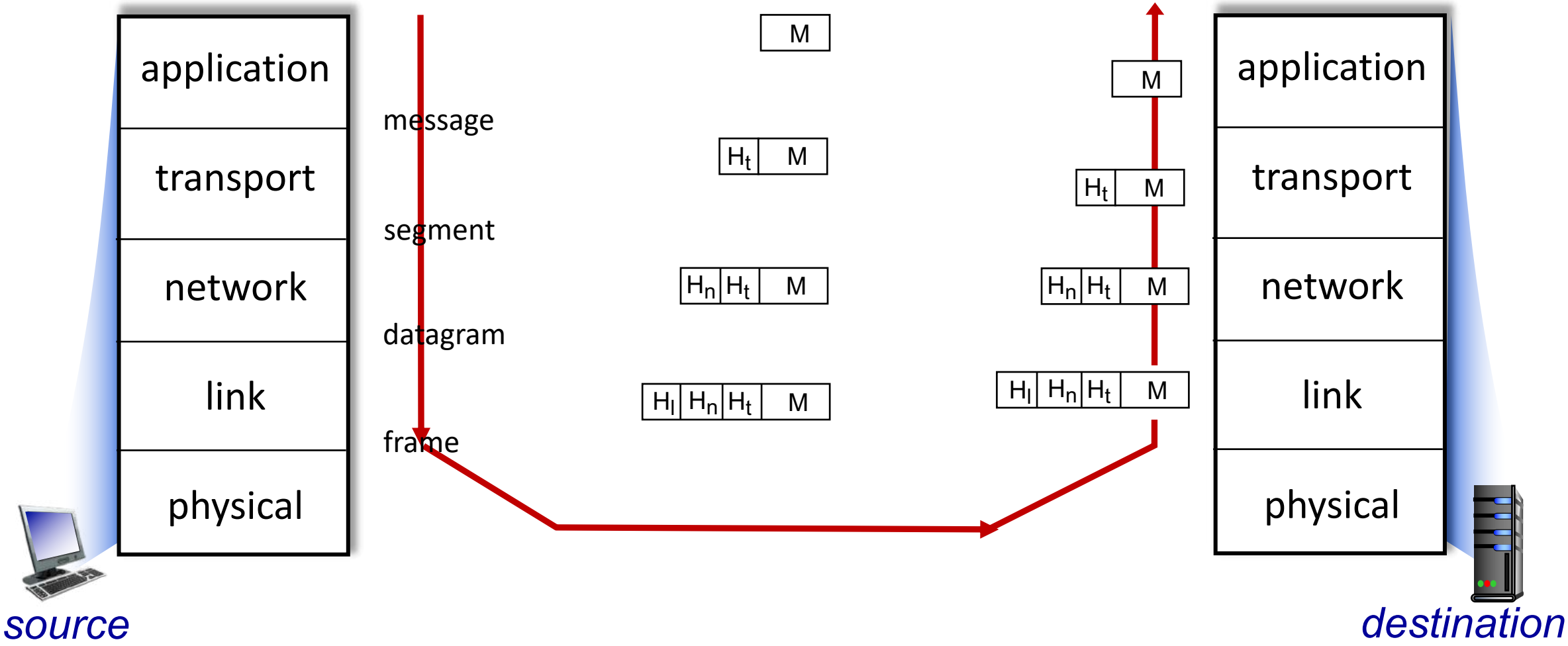
- network-layer protocol encapsulates transport-layer segment [H_t | M] with network layer-layer header H_n to create a network-layer datagram
  - H_n used by network layer protocol to implement its service

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