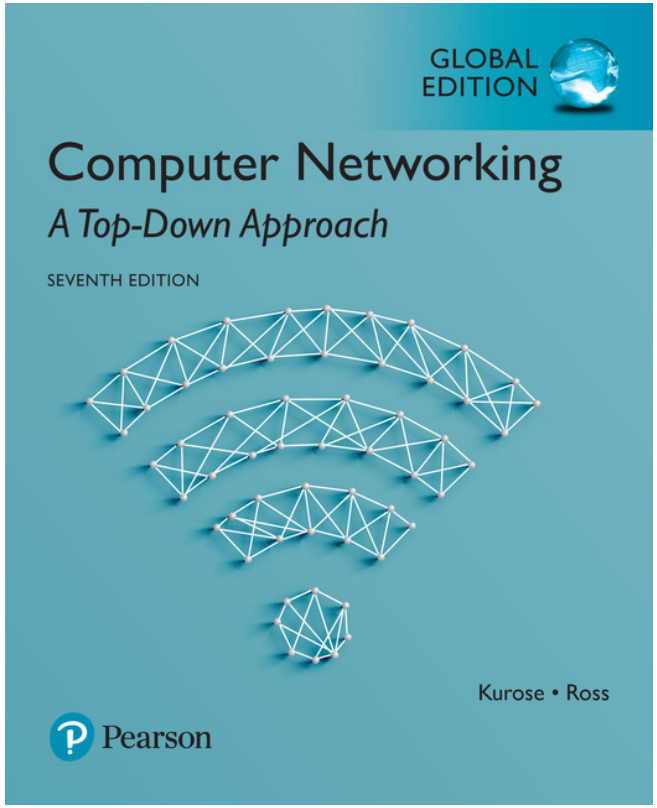
Overview



Chapter 1

Computer Networking: A Top Down Approach

Jim Kurose, Keith Ross
7th Edition, Global Edition
Pearson
April 2016

Introduction: Goal & Overview

Goal

- Provide the overview of computer networks

Overview

- What is the Internet?
- What is a protocol?
- Network performance
- Network security
- Internet history

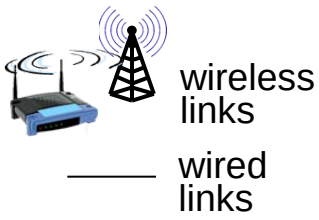
⇒ **Chapter 1 of Reference Book**

Roadmap

1. what *is* the Internet?
   - What is a protocol?
2. network edge
   - end systems, access networks, links
3. network core
   - packet switching, circuit switching, network structure
4. delay, loss, throughput in networks
5. protocol layers, service models
6. networks under attack: security
7. Internet history

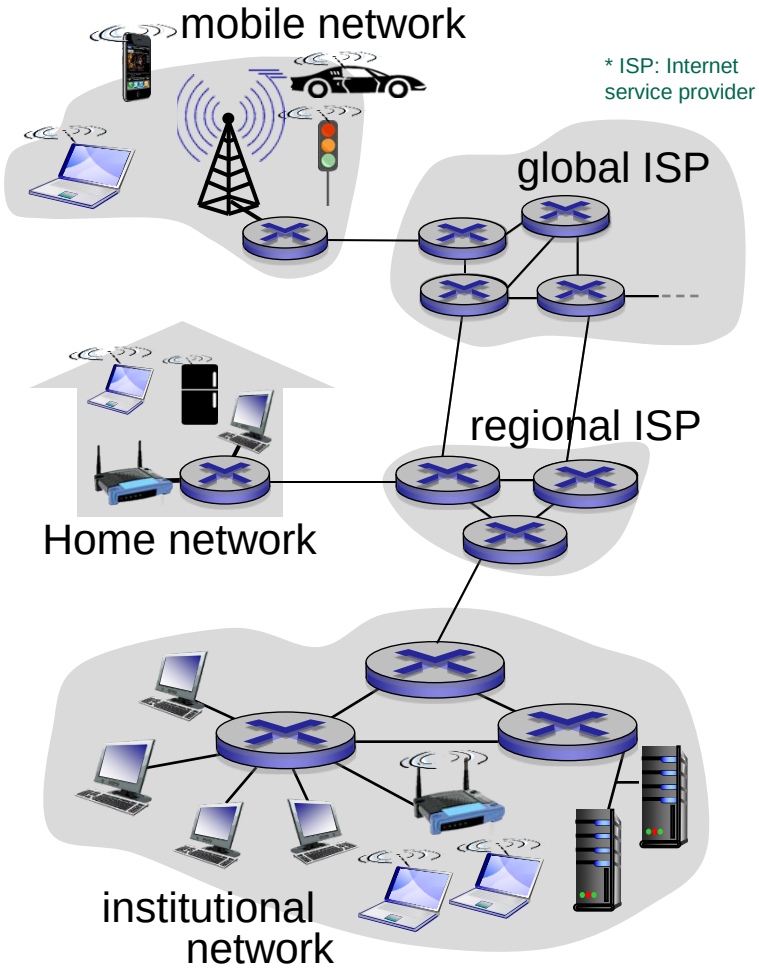
What is the Internet: "nuts and bolts" view

- billions of connected computing devices:
  - *hosts* = *end systems*
  - running *network apps*



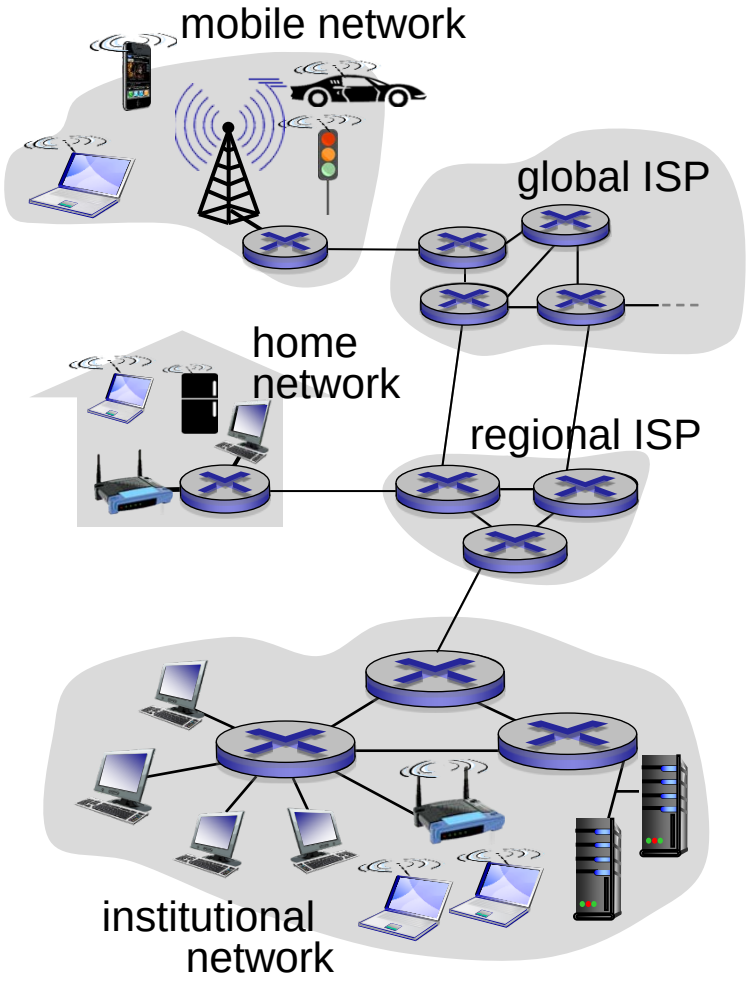
- *communication links*
  - fiber, copper, terrestrial radio, satellite radio
  - transmission rate: *bandwidth*

- *packet switches:* forward packets (chunks of data)
  - *routers* and *switches*



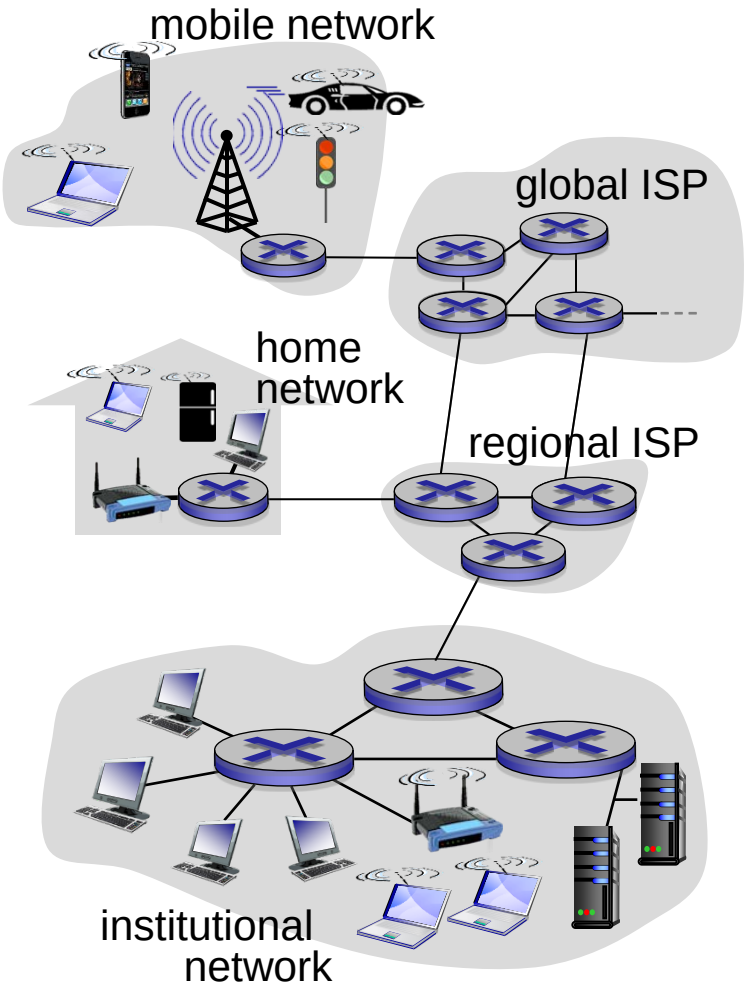
What is the Internet: "nuts and bolts" view

- *Internet:* "network of networks"
  - Interconnected ISPs
- *protocols* control sending, receiving of messages
  - e.g., TCP, IP, HTTP, 802.11, …
- *Internet standards*
  - IETF: Internet Engineering Task Force
  - RFC: Request for comments
    - IETF documents



What is the Internet: a service view

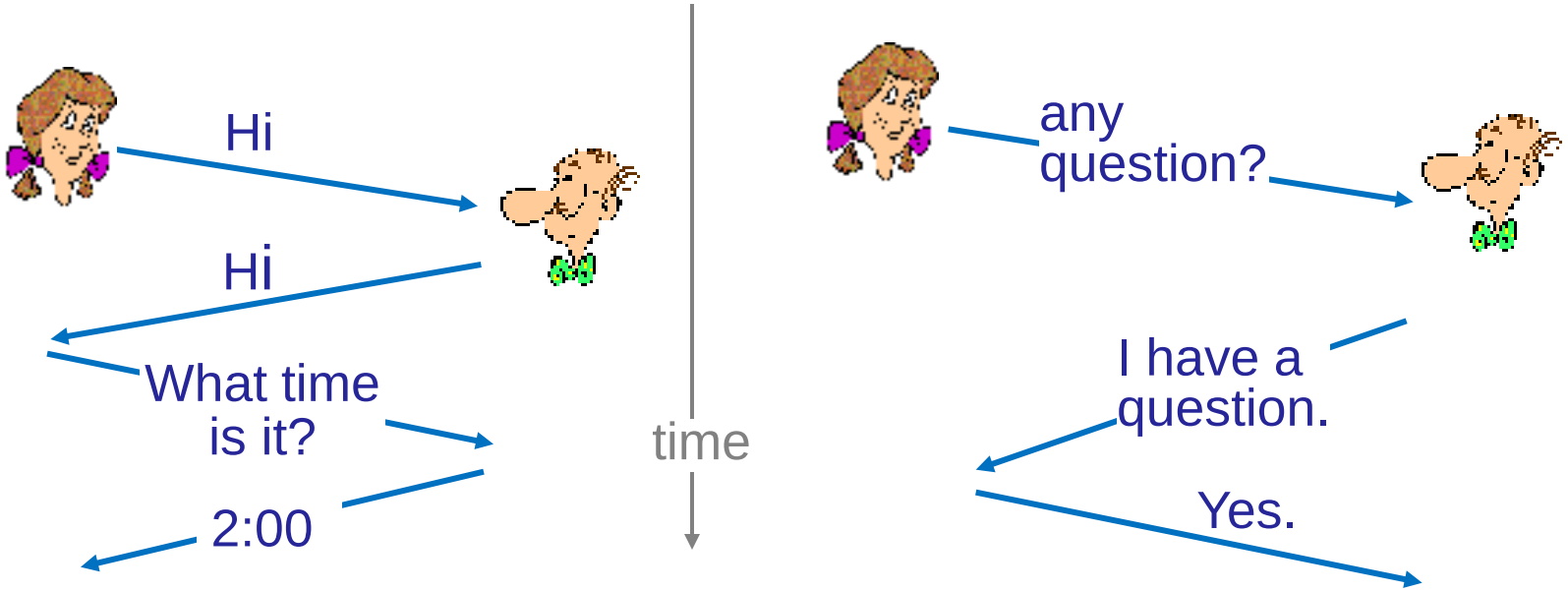
- *infrastructure that provides services to applications:*
  - Web, VoIP, email, games, e-commerce, social nets, …
- *provides programming interface to apps*
  - hooks that allow sending and receiving app programs to "connect" to Internet
  - provides service options, analogous to postal service



What is a protocol?

human protocols:

- "what time is it?"
- "I have a question"

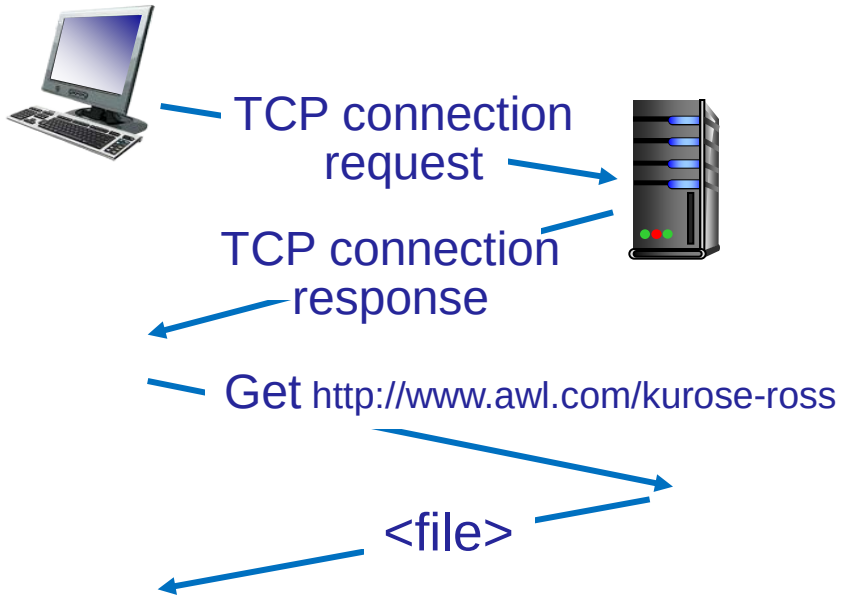


- specific messages sent
- specific actions taken when messages received, or other events

What is a protocol?

network protocols:

- machines rather than humans
- all communication activity in Internet governed by protocols



❖ *protocols define format, order of messages sent and received among network entities, and actions taken on the transmission and/or receipt of a message, or other event.*

What is a protocol?

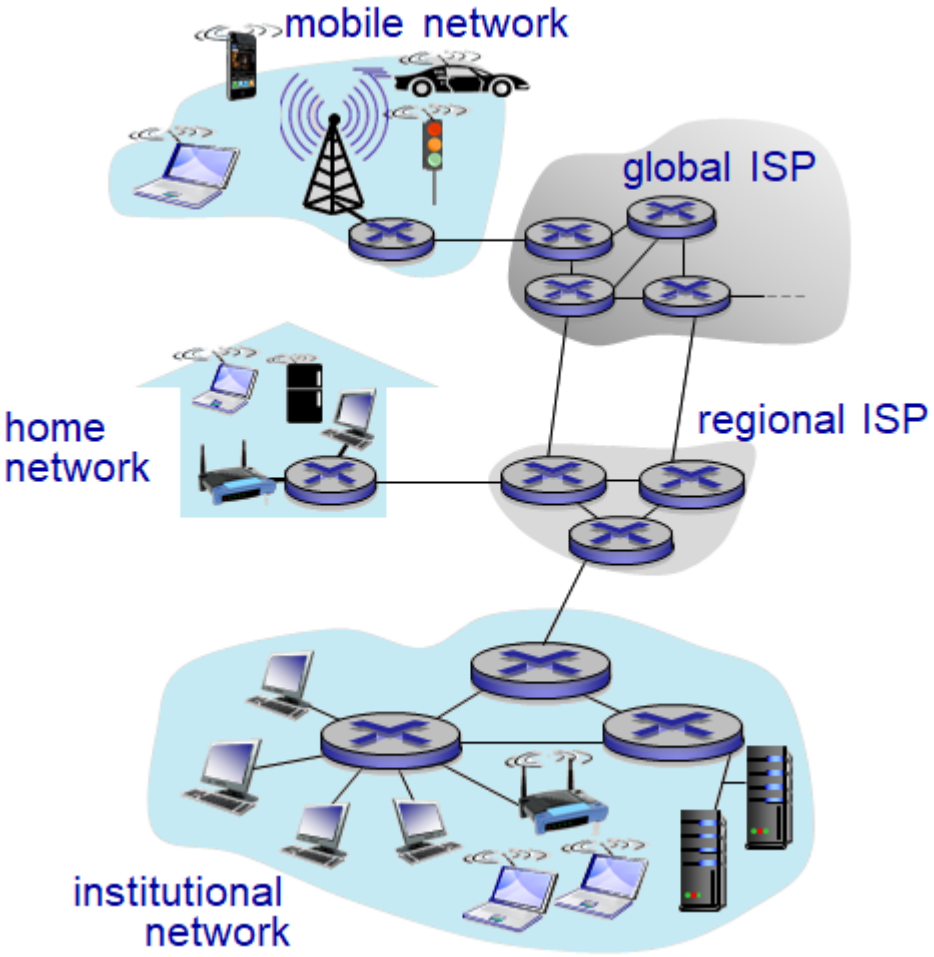
- Protocol
  - A set of rules governing the exchange of data between two or more (peer) entities.
  - Must speak the same language
- Key elements of a protocol
  - Syntax :
    - format, size and contents of protocol messages or packets
  - Semantics (Control):
    - meaning of messages
    - actions to take in
    - response to reception of different messages
  - Timing :
    - when to discard a message, retransmit, give up, etc.
- Protocols become very complicated: layered architecture

Roadmap

1. what *is* the Internet?
2. network edge
   - end systems, access networks, links
3. network core
   - packet switching, circuit switching, network structure
4. delay, loss, throughput in networks
5. protocol layers, service models
6. networks under attack: security
7. Internet history

A closer look at network structure

- *network edge*
  - host (end system):
    - clients and servers
    - most of servers often reside in data centers
  - *physical media*
    - wired, wireless communication links
  - *access networks*
- *network core*
  - interconnected routers
  - network of networks



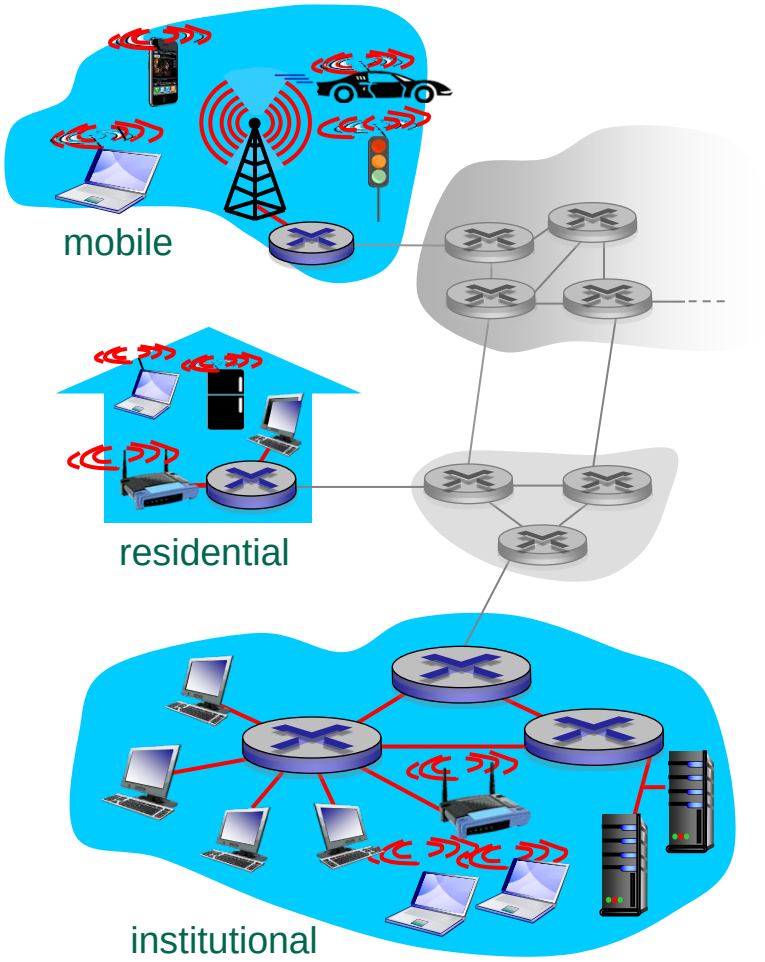
Access networks

How to connect end systems to edge router?

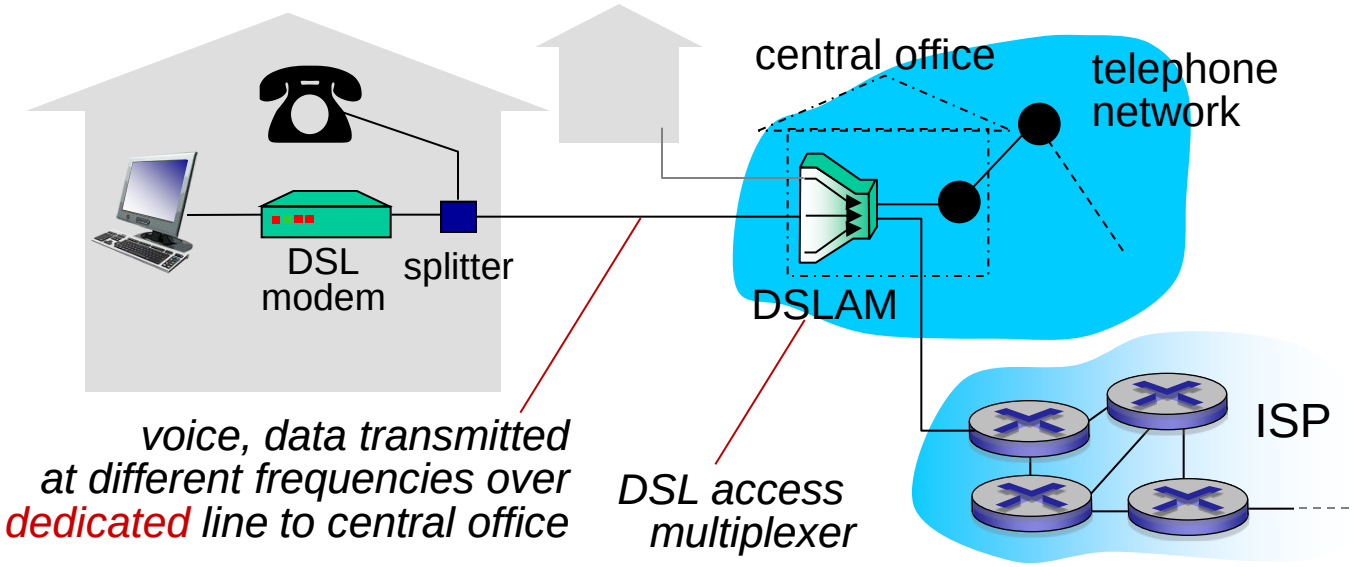
- residential access networks
- institutional access networks (school, company)
- mobile access networks

keep in mind:

- bandwidth (bits per second) of access network?
- shared or dedicated?

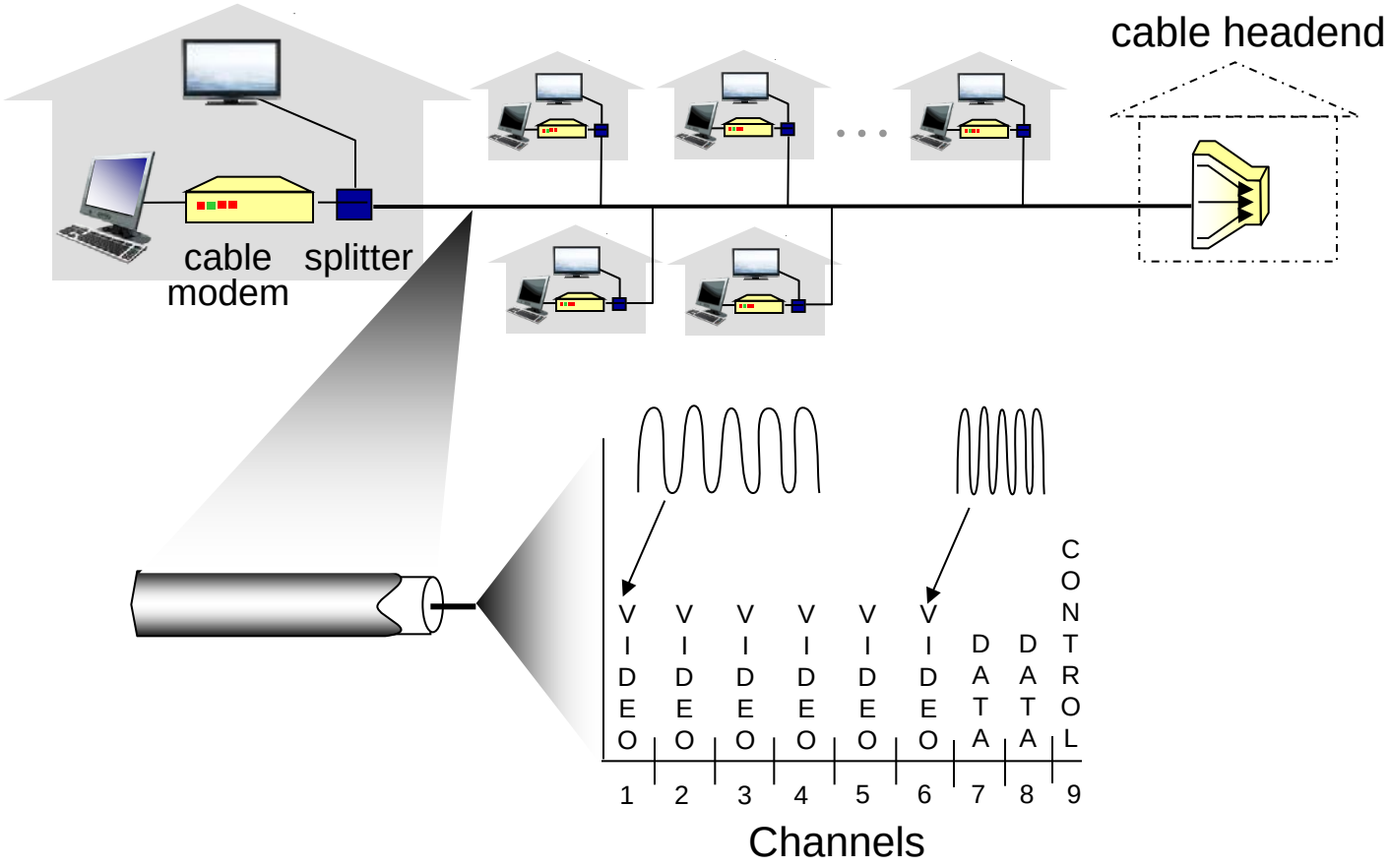


Access network: digital subscriber line (DSL)



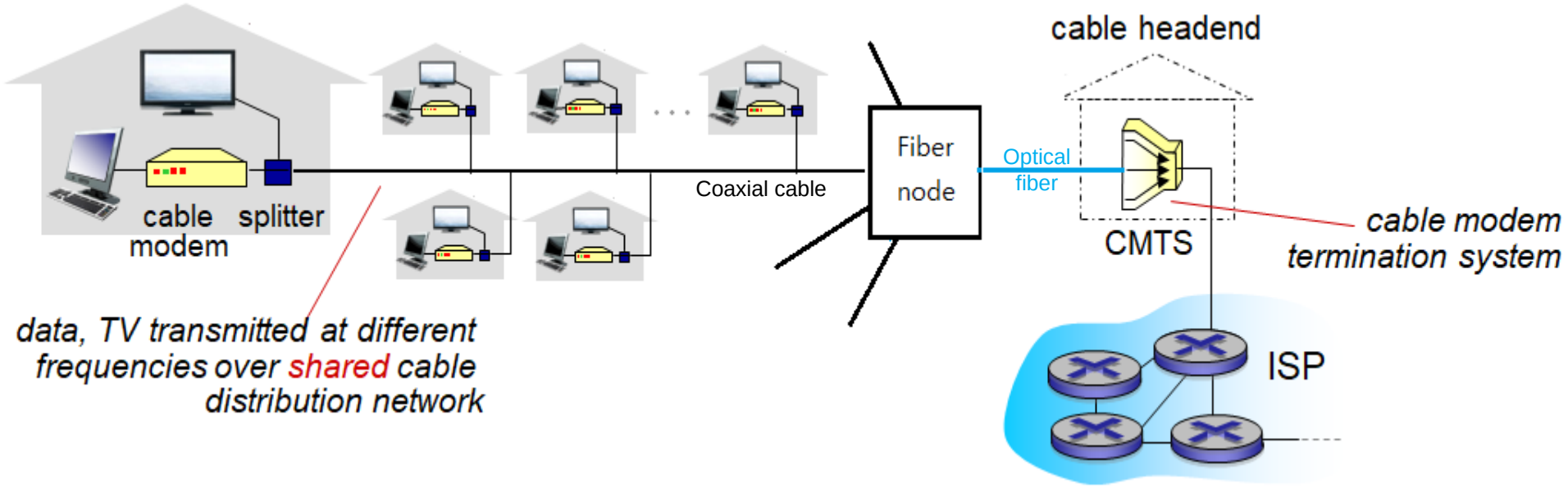
- use *existing* telephone line
  - DSLAM in central office
    - data over DSL phone line goes to Internet
    - voice over DSL phone line goes to telephone net
- < 2.5 Mbps upstream transmission rate (typically < 1 Mbps)
- < 24 Mbps downstream transmission rate (typically < 10 Mbps)

Access network: cable network



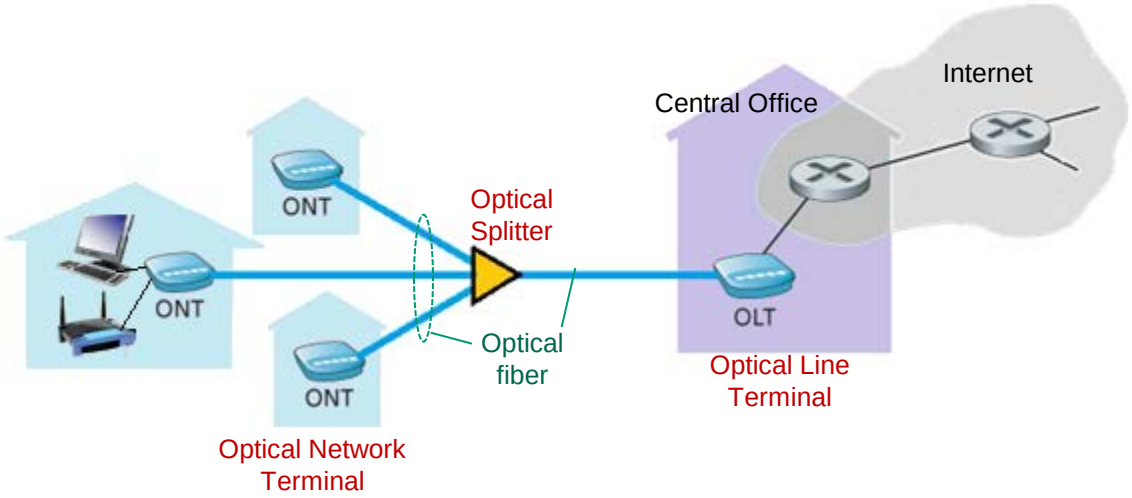
frequency division multiplexing: different channels transmitted in different frequency bands

Access network: cable network

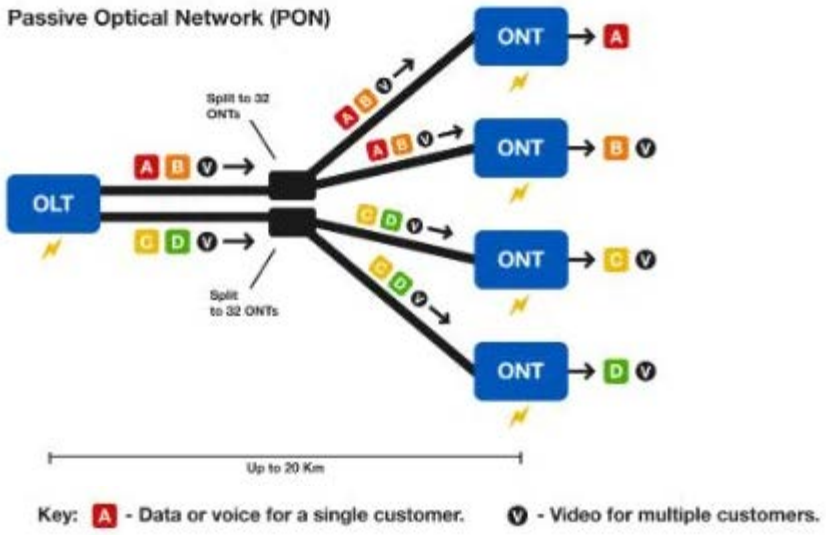


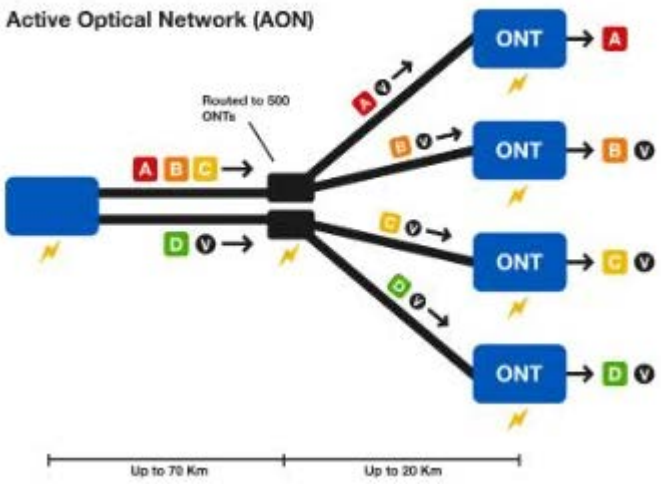
- HFC: hybrid fiber coaxial
  - asymmetric: up to 30Mbps downstream transmission rate, 2 Mbps upstream transmission rate
- Cable network attaches homes to ISP router
  - homes *share access network* to cable headend
  - unlike DSL which has dedicated access to central office

Access network: FTTH (Fiber to the Home)



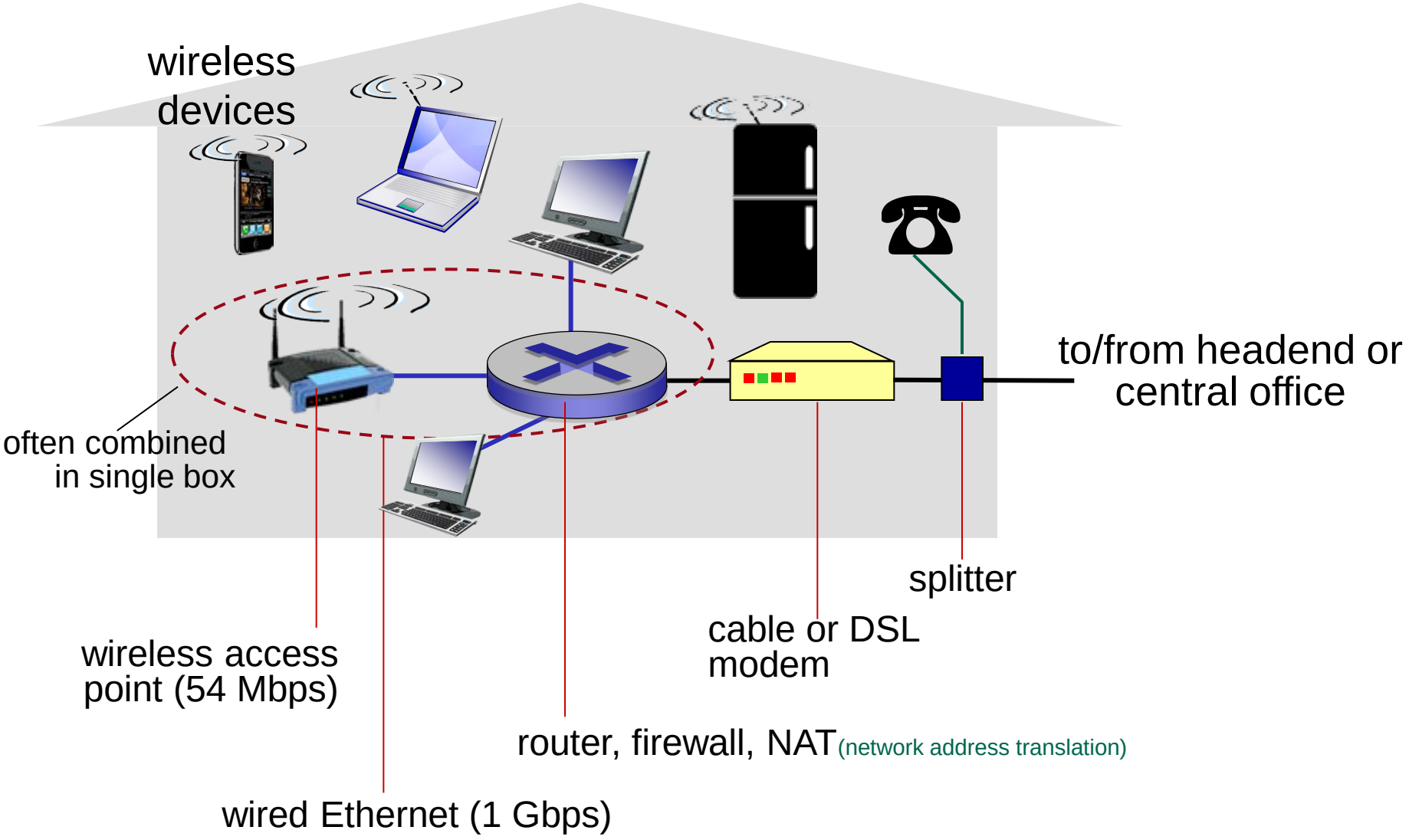
- Passive Optical Network vs Active Optical Network



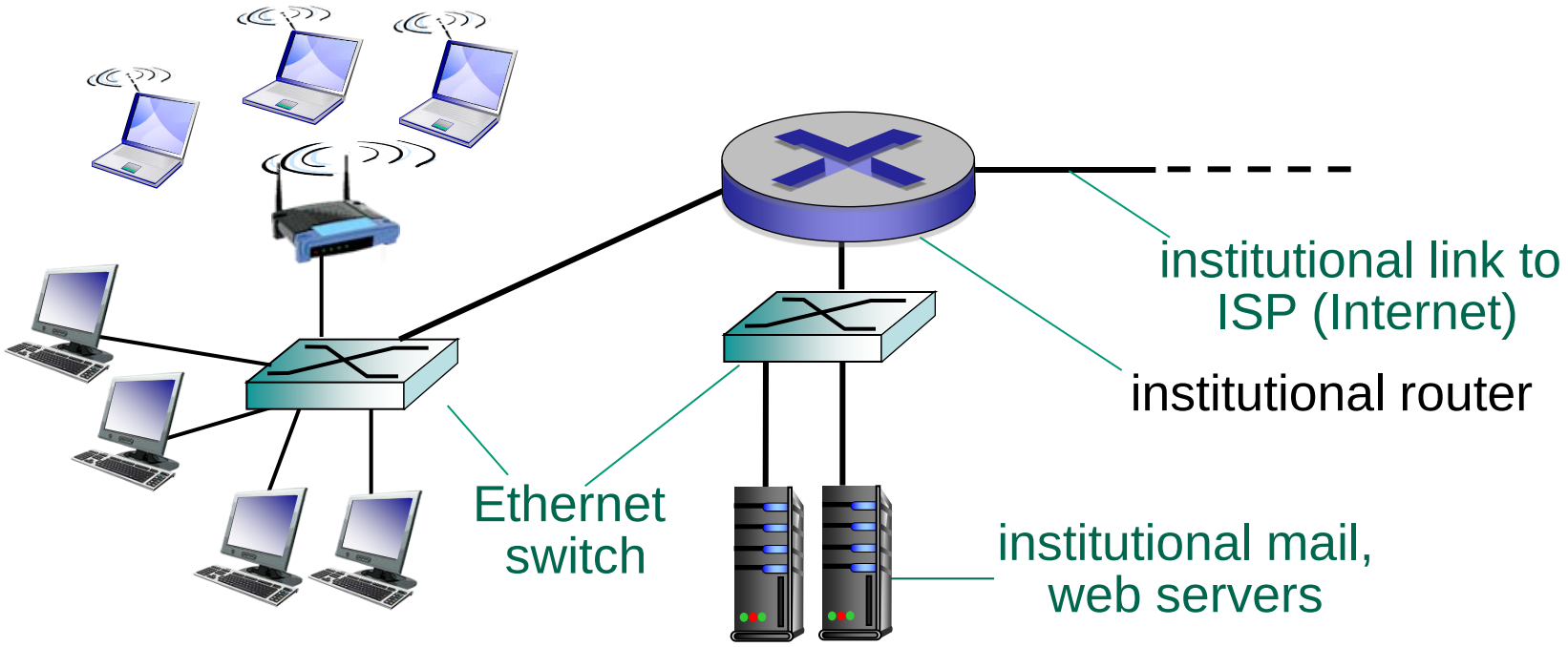


https://community.element14.com/products/manufacturers/bulgin/b/blog/posts/what-s-the-difference-between-passive-pon-and-active-aon-optical-networks

Access network: home network



Enterprise access networks (Ethernet)



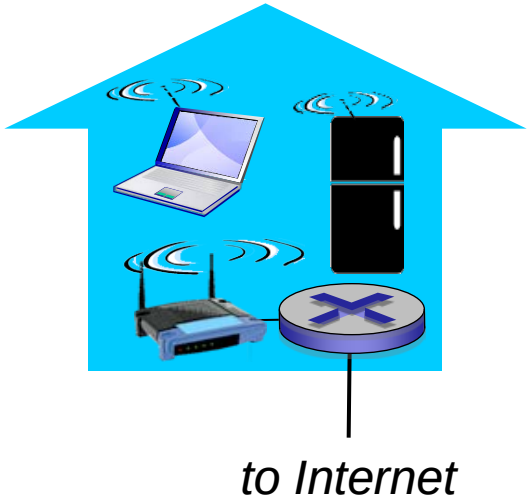
- typically used in companies, universities, etc.
- 10 Mbps, 100Mbps, 1Gbps, 10, 100Gbps transmission rates
- today, end systems typically connect into Ethernet switch

Wireless access networks

- shared *wireless* access network connects end system to router
  - via "access point" or "base station"

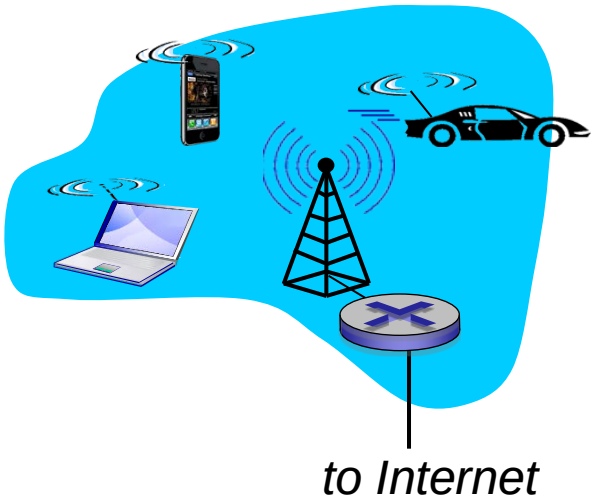
wireless LANs:

- within building /office/home
- 802.11b/g/n (WiFi): 11, 54, 450 Mbps
- 802.11ay: 20 Gbps



wide-area wireless access

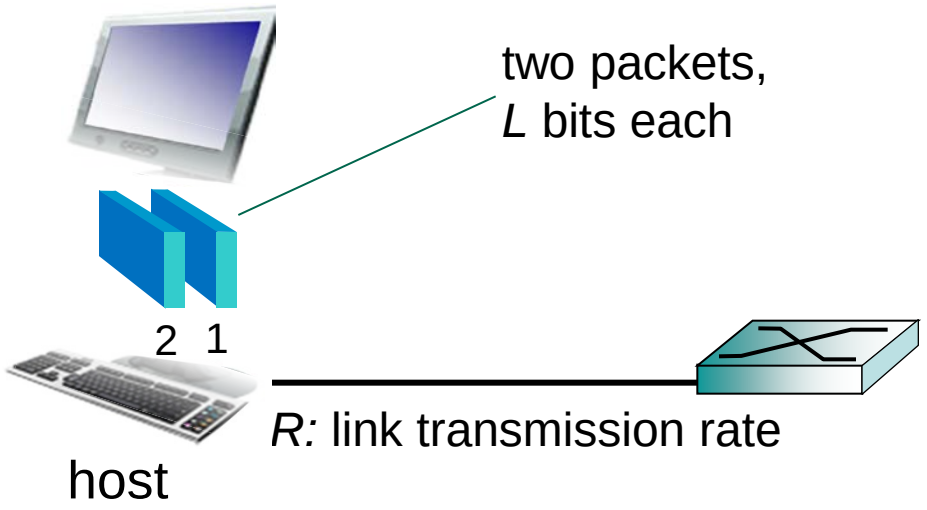
- provided by telco (cellular) operator, (several tens of kms)
- between 1Mbps and 1 Gbps
- 3G, 4G: LTE-A, 5G



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: link *capacity, bandwidth*



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

Physical media

- Bits encoded to the signal propagate between transmitter/receiver pairs
- physical media: transmission channel that lies between transmitter & receiver
- guided media:
  - signals propagate in solid media: fiber, coax., twisted pair
- unguided media:
  - signals propagate freely, e.g., radio

twisted pair (TP)

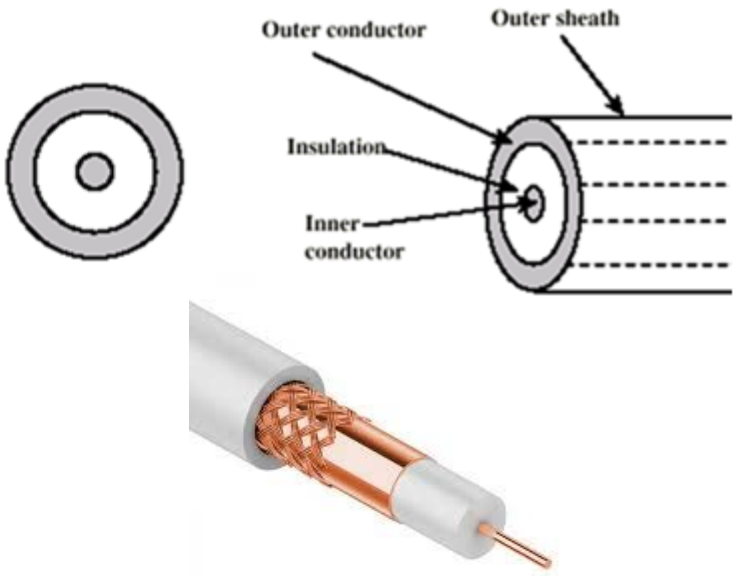
- two insulated copper wires

  - Category 5: 100 Mbps, 1 Gbps Ethernet
  - Category 6: 10 Gbps

Physical media: coax, fiber

coaxial cable:

- two concentric copper conductors
- bidirectional
- broadband:
  - multiple channels on cable



fiber optic cable:

- glass fiber carrying light pulses, each pulse a bit
- high-speed operation:
  - high-speed point-to-point transmission (e.g., 10's-100's Gbps transmission rate)
- low error rate:
  - immune to electromagnetic noise
  - repeaters spaced far apart

Physical media: radio

- signal carried in electromagnetic spectrum
- no physical "wire"
- bidirectional
- propagation environment effects:
  - reflection
  - obstruction by objects
  - interference

radio link types:

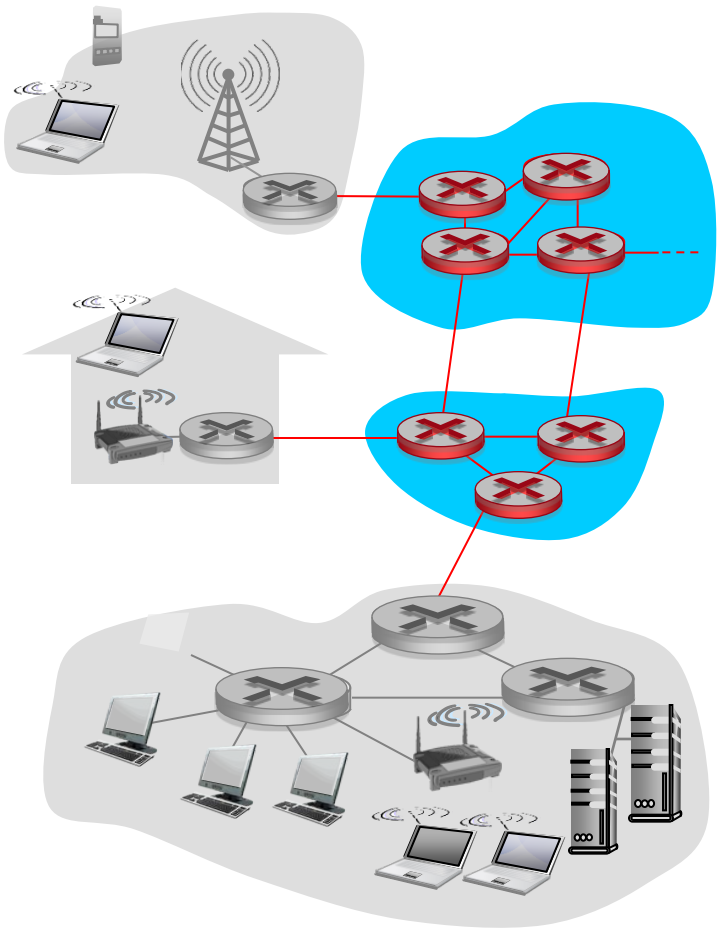
- terrestrial microwave
- LAN (e.g., WiFi)
- wide-area (e.g., cellular)
- satellite
  - geostationary (36,000 km)
    - 270 msec end-to-end delay
  - Low earth orbiting (LEO)

Roadmap

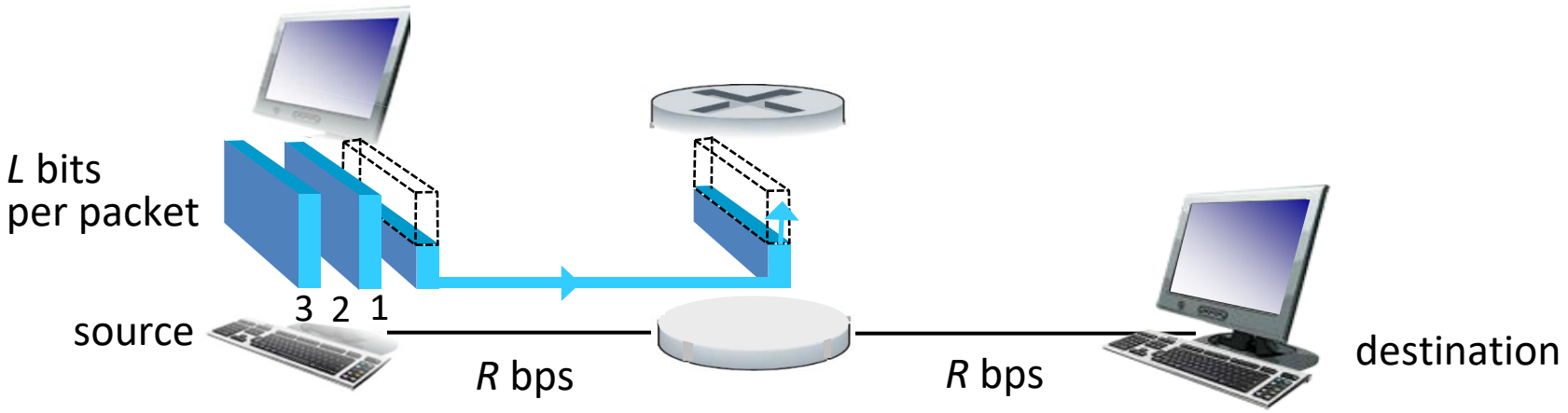
1. what *is* the Internet?
2. network edge
   - end systems, access networks, links
3. network core
   - packet switching, circuit switching, network structure
4. delay, loss, throughput in networks
5. protocol layers, service models
6. networks under attack: security
7. Internet history

Network core

- mesh of interconnected routers
- packet-switching:
  - hosts break application-layer messages into *packets*
  - forward packets from one router to the next, across links on path from source to destination
  - each packet transmitted at full link capacity



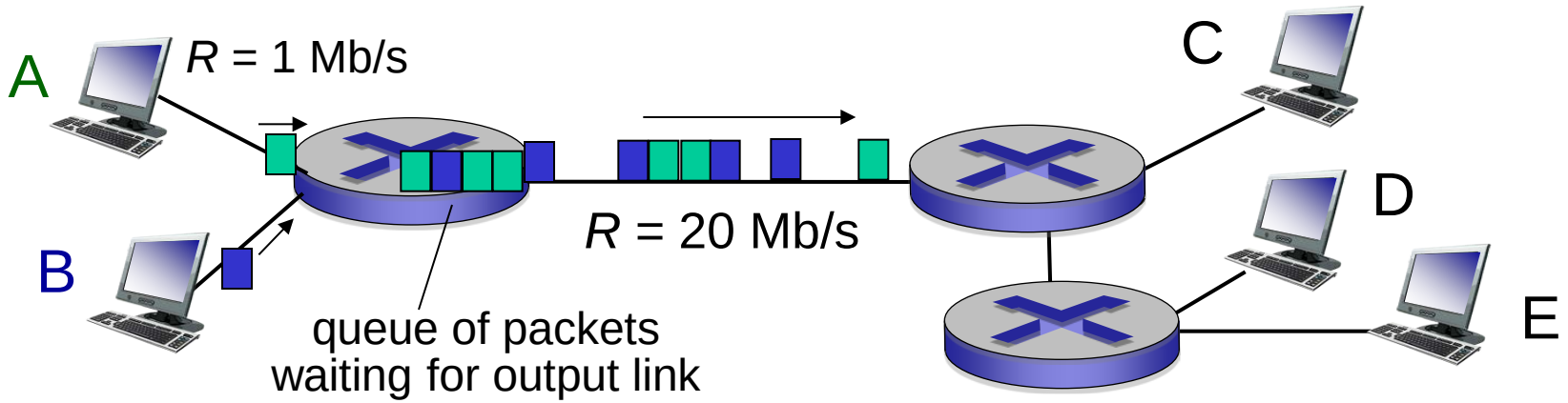
Packet-switching: store-and-forward



- 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
- In the above example,

  end-end delay = 2**L*/*R* (assuming zero propagation delay)

Packet Switching: queueing delay, loss



queueing and loss:

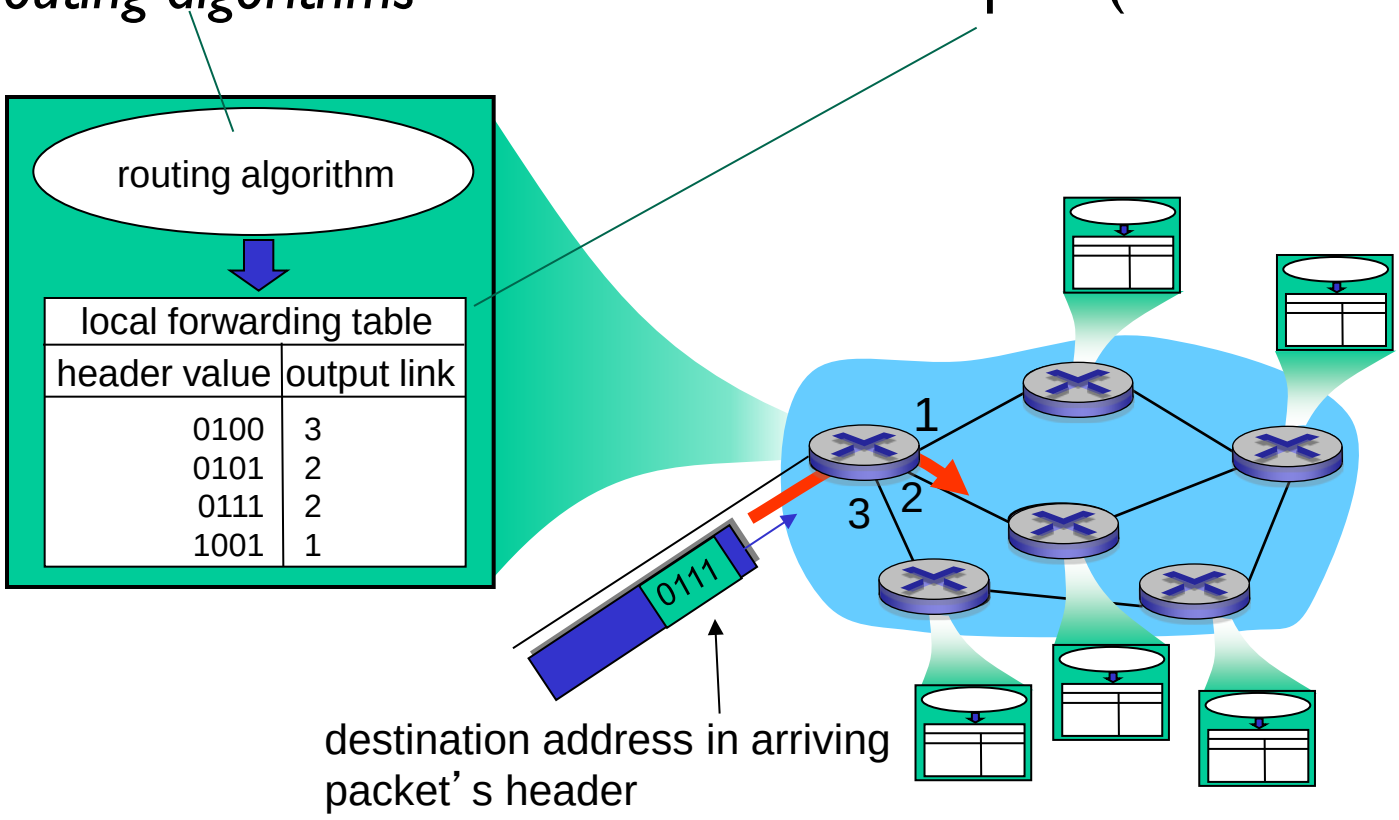
- if arrival rate (in bits) to link exceeds transmission rate of link for a period of time:
  - packets will queue, wait to be transmitted on link
  - packets can be dropped (lost) if memory (buffer) fills up

Two key network-core functions

routing: determines source-destination route taken by packets (Control Plane)

- *routing algorithms*

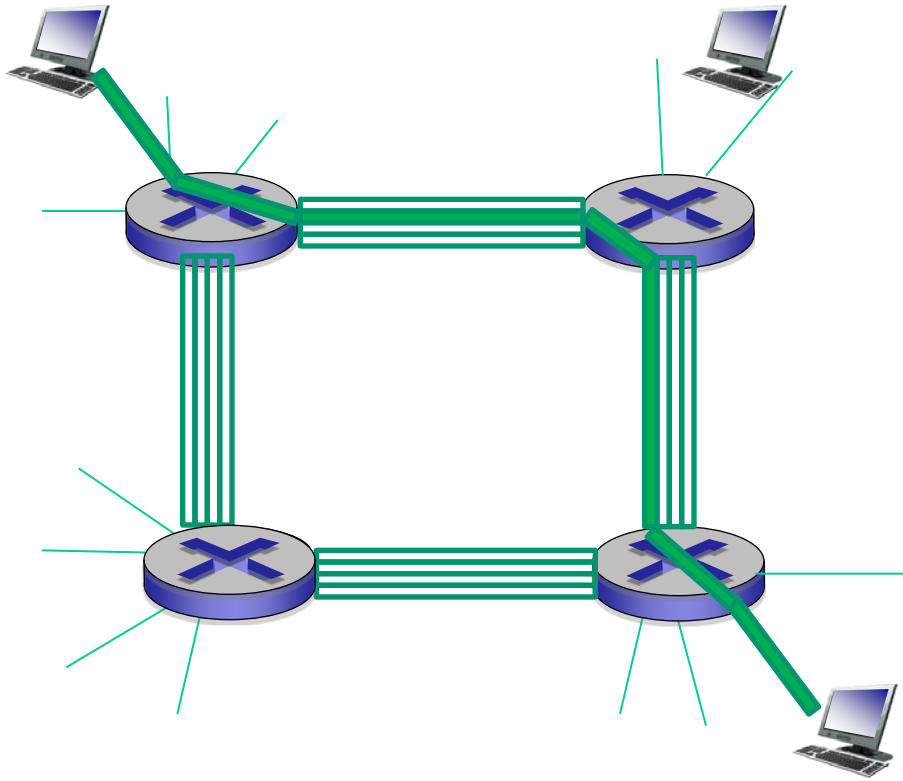
forwarding: move packets from router's input to appropriate router output (Data Plane)



Alternative core: circuit switching

End-end resources allocated (reserved) to "call" between source & dest:

- dedicated resources
  - circuit-like (guaranteed) performance
- circuit segment is idle if not used by call *(no sharing)*
- commonly used in traditional telephone networks

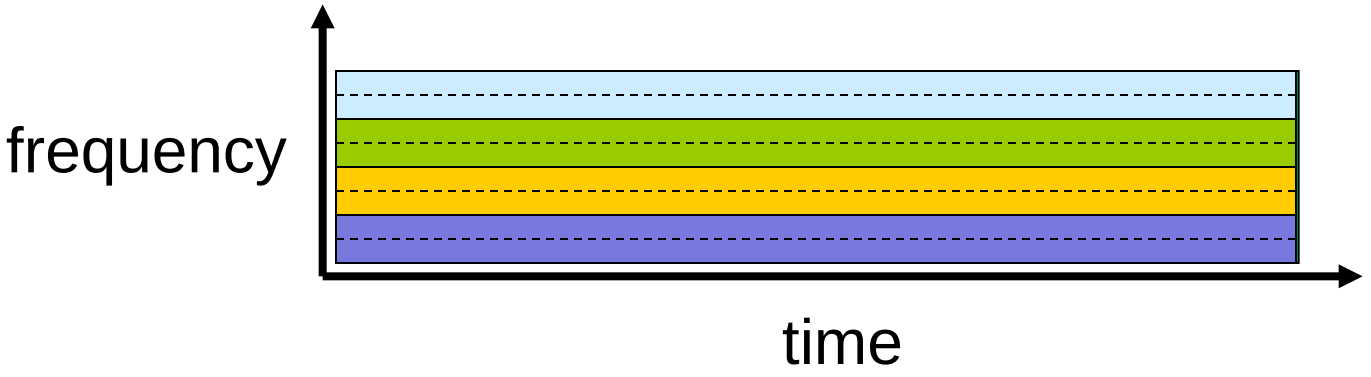


Resource Allocation in Circuit Switching

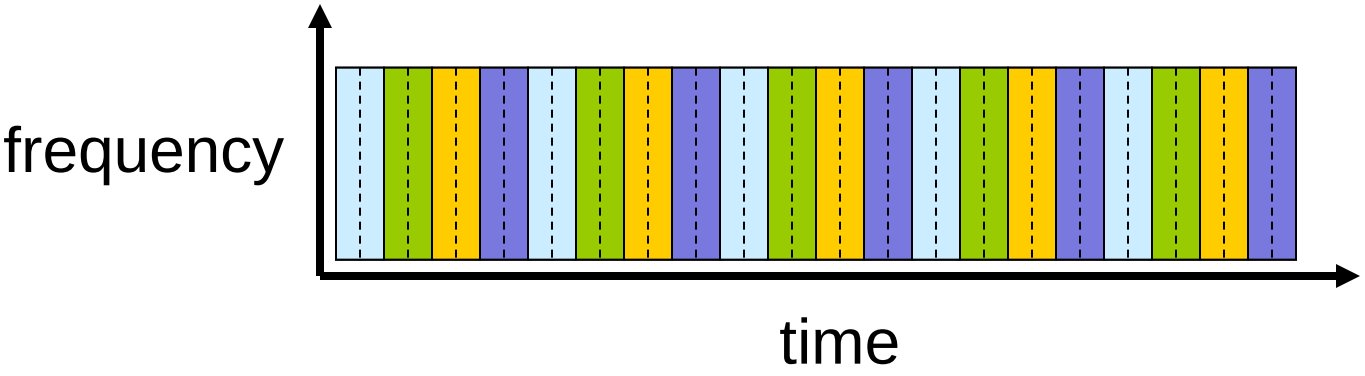
Example:

4 users

Frequency Division



Time Division

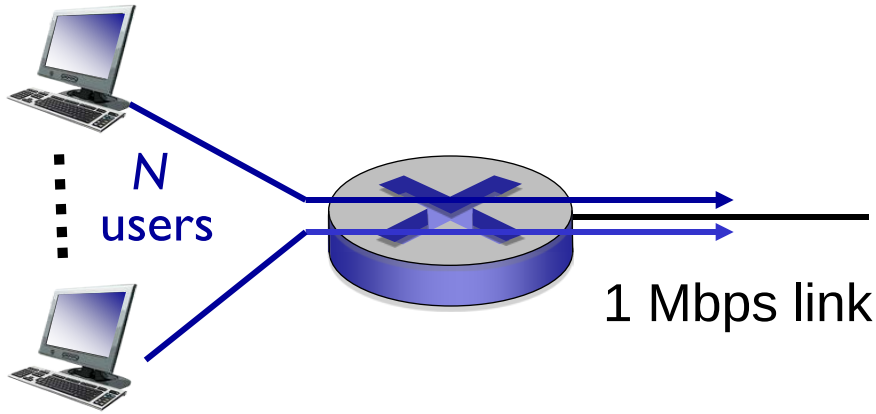


Packet switching versus Circuit switching

packet switching allows more users to use network!

example:

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



- *circuit-switching:*
  - 10 users
- *packet switching:*
  - with 35 users,
  - probability of more than 10 active users at same time is less than 0.0004

How did we get value 0.0004?

$$35 \quad \sum_{i=11}^{N} \binom{N}{i} (0.1)^i (0.9)^{N-i}$$

What happens if > 35 users ?

Packet switching versus Circuit switching

Is packet switching always "the best"?

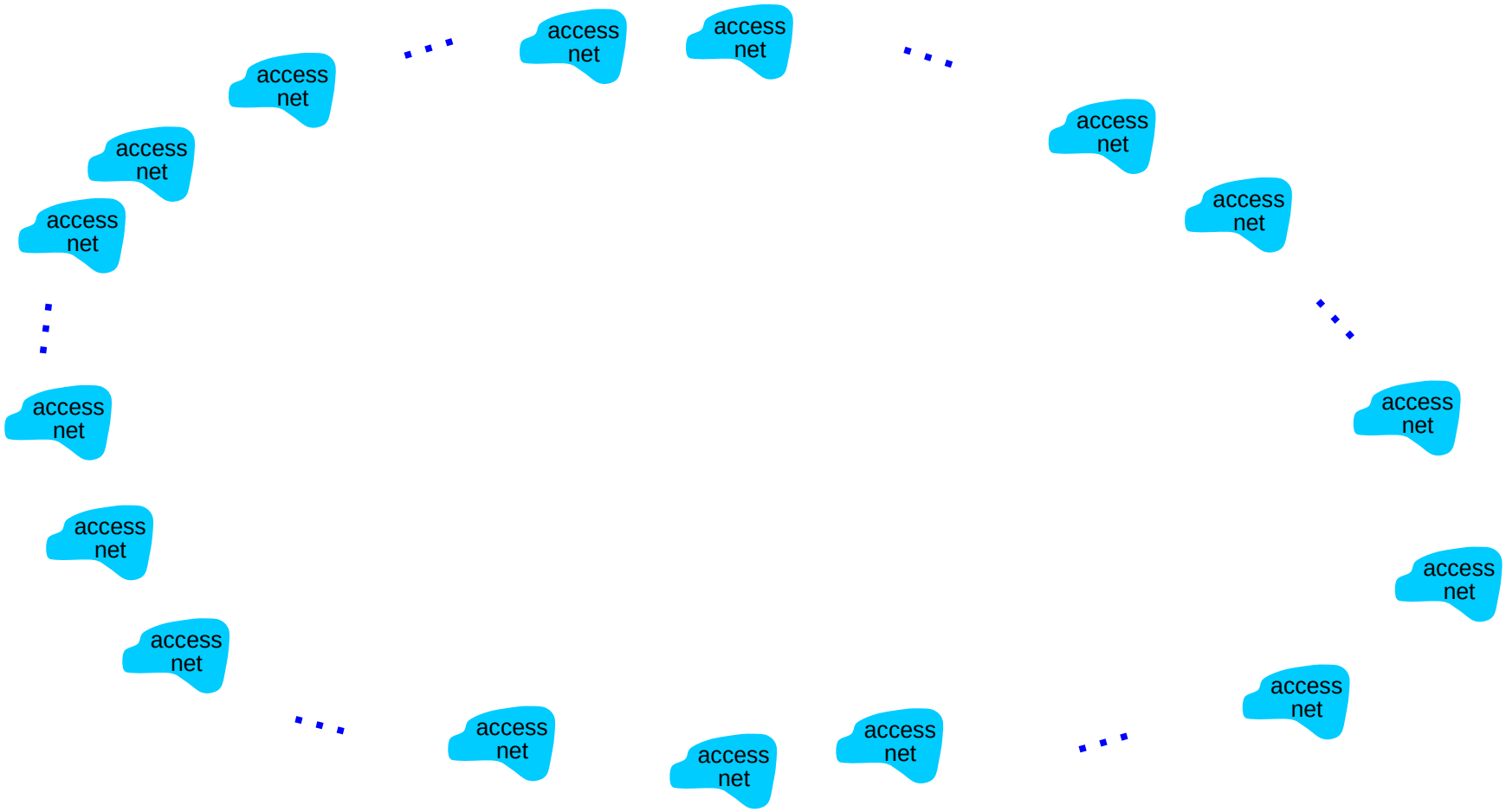
- great for bursty data
  - resource sharing
  - simpler, no call setup
- excessive congestion possible: packet delay and loss
  - protocols needed for reliable data transfer, congestion control
- How to provide circuit-like behavior? (challenge)
  - bandwidth guarantees needed for audio/video apps
  - still an unsolved problem

Internet structure: network of networks

- End systems connect to Internet via access ISPs (Internet Service Providers)
  - residential, company, university ISPs
- Access ISPs in turn must be interconnected.
  - so that any two hosts can send packets to each other
- Resulting network of networks is very complex
  - evolution was driven by economics and national policies
  - Hierarchical (see slide 1-41)

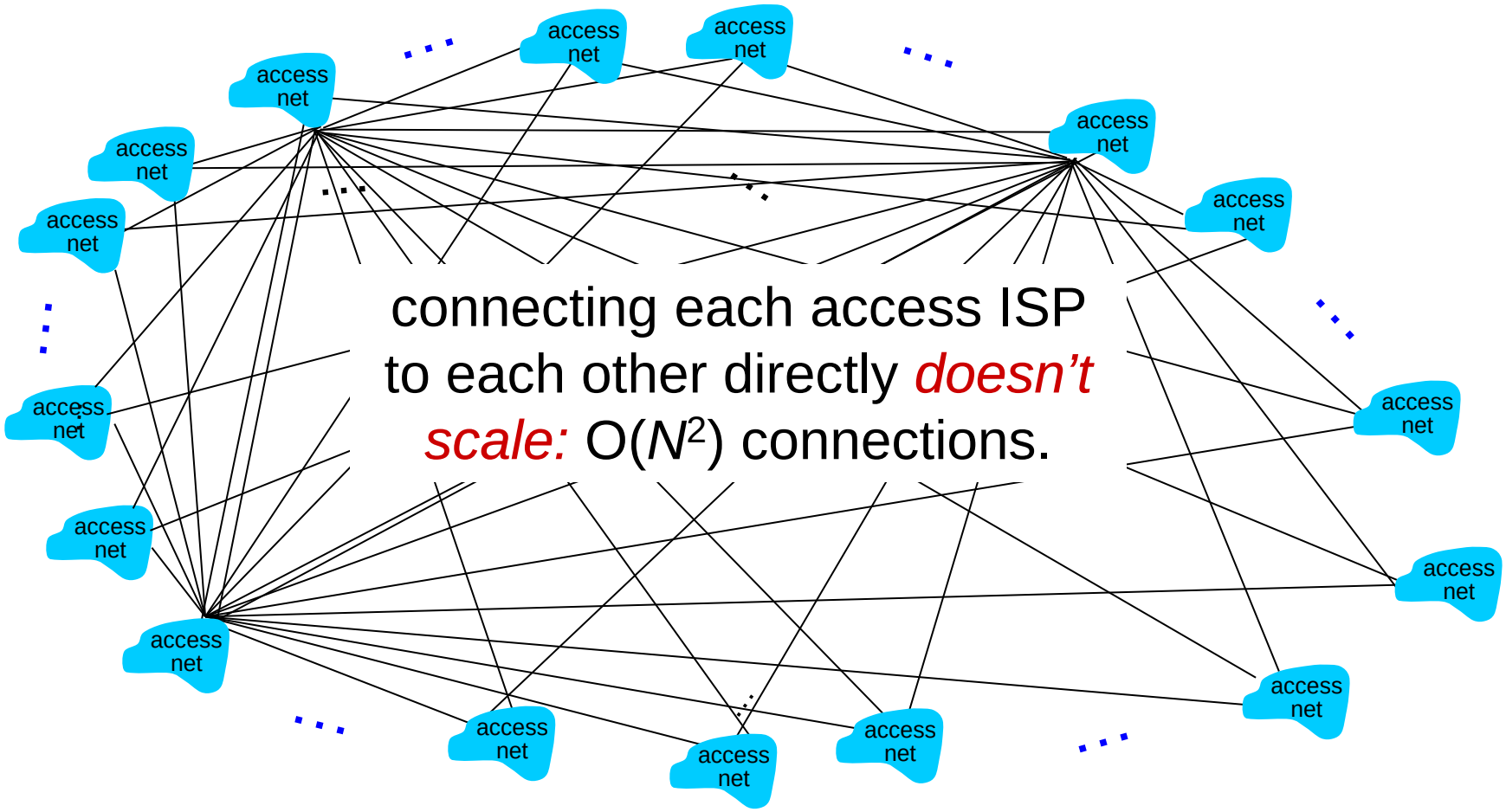
Internet structure: network of networks

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



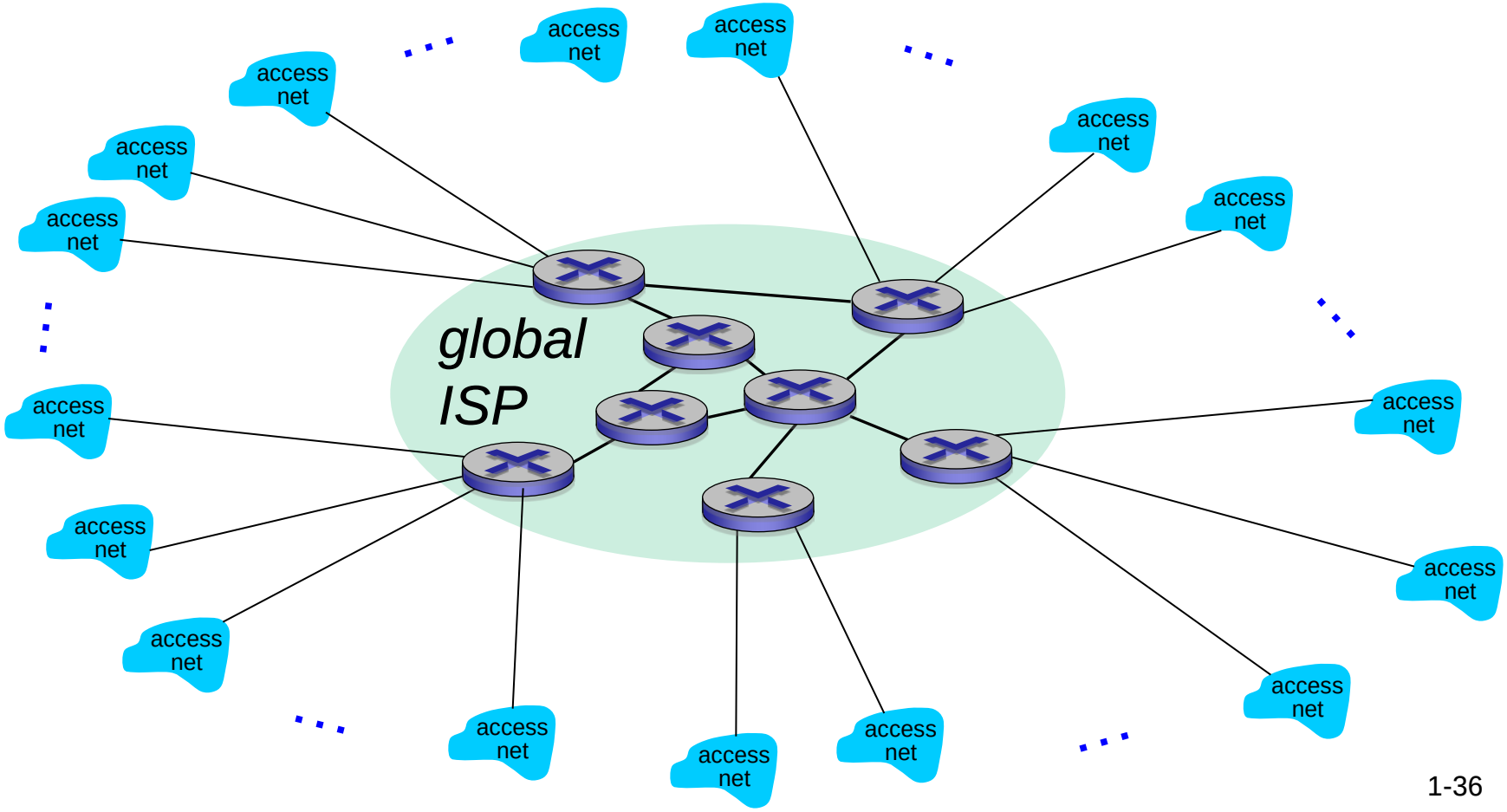
Internet structure: network of networks

Option: connect each access ISP to every other access ISP?



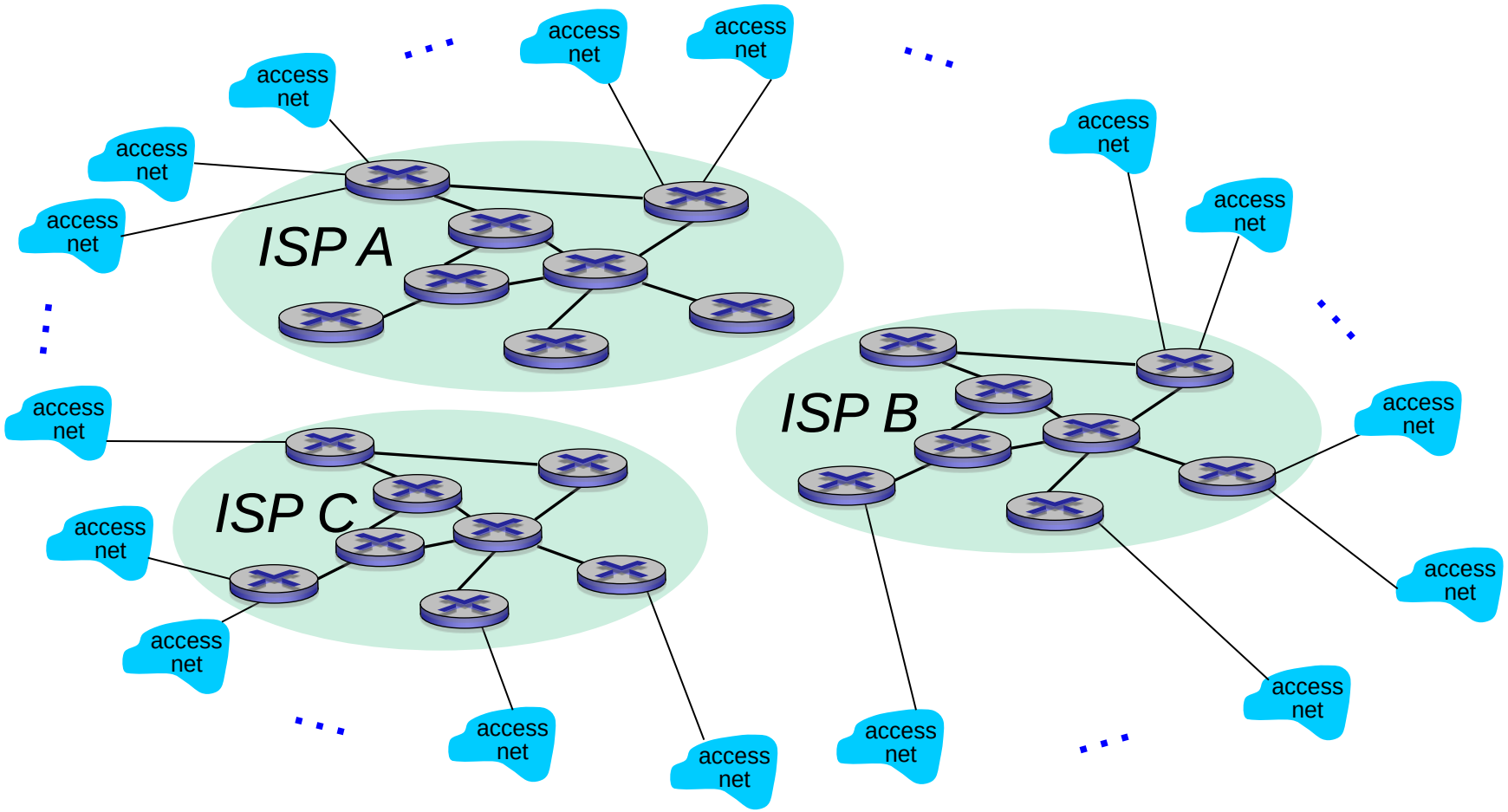
Internet structure: network of networks

- *Option: connect each access ISP to one global transit ISP?*
- Access ISP (customer) and global ISP (provider) have economic agreement (customer ISP pays the global transit ISP, based on the amount of traffic)



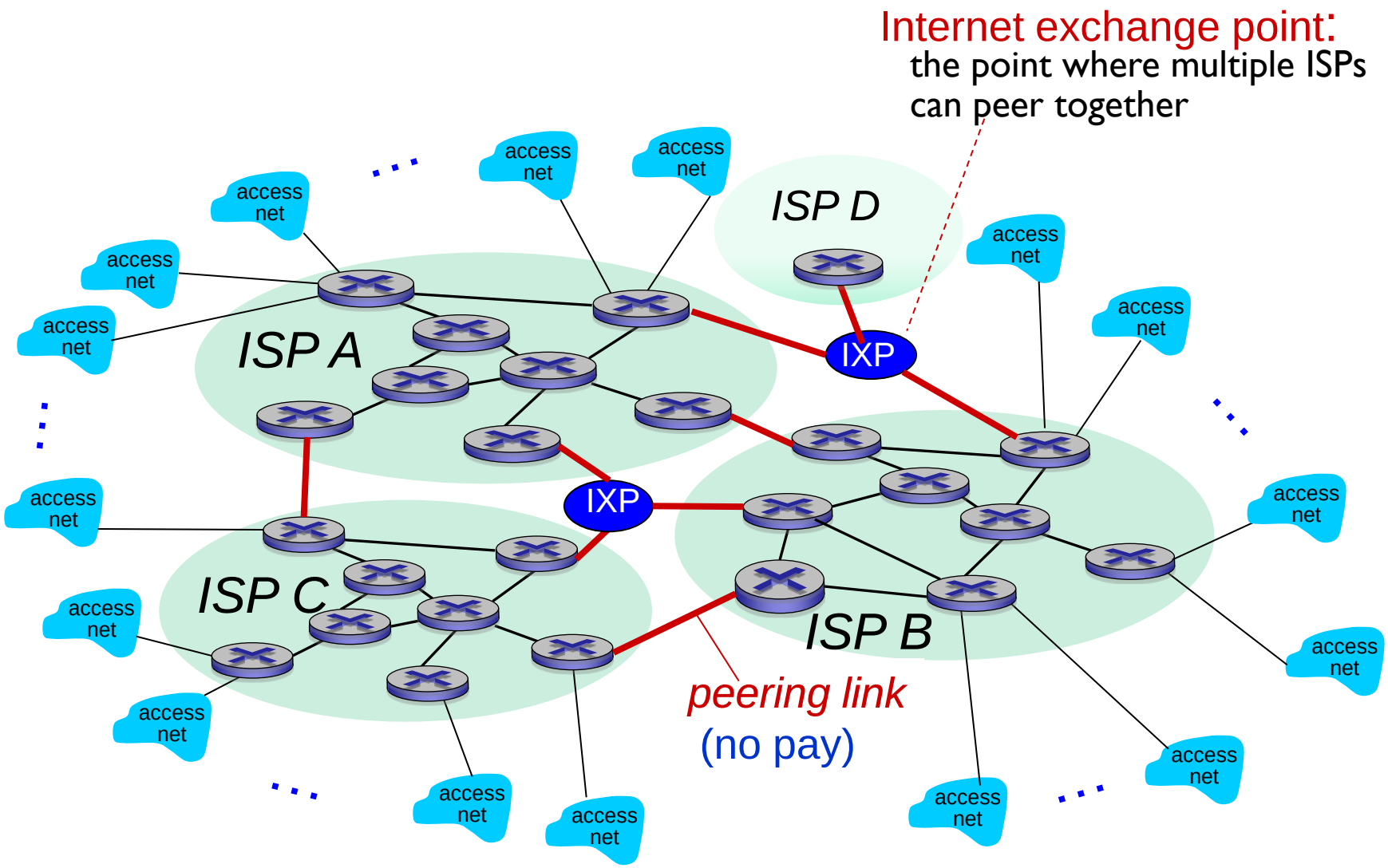
Internet structure: network of networks

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



Internet structure: network of networks

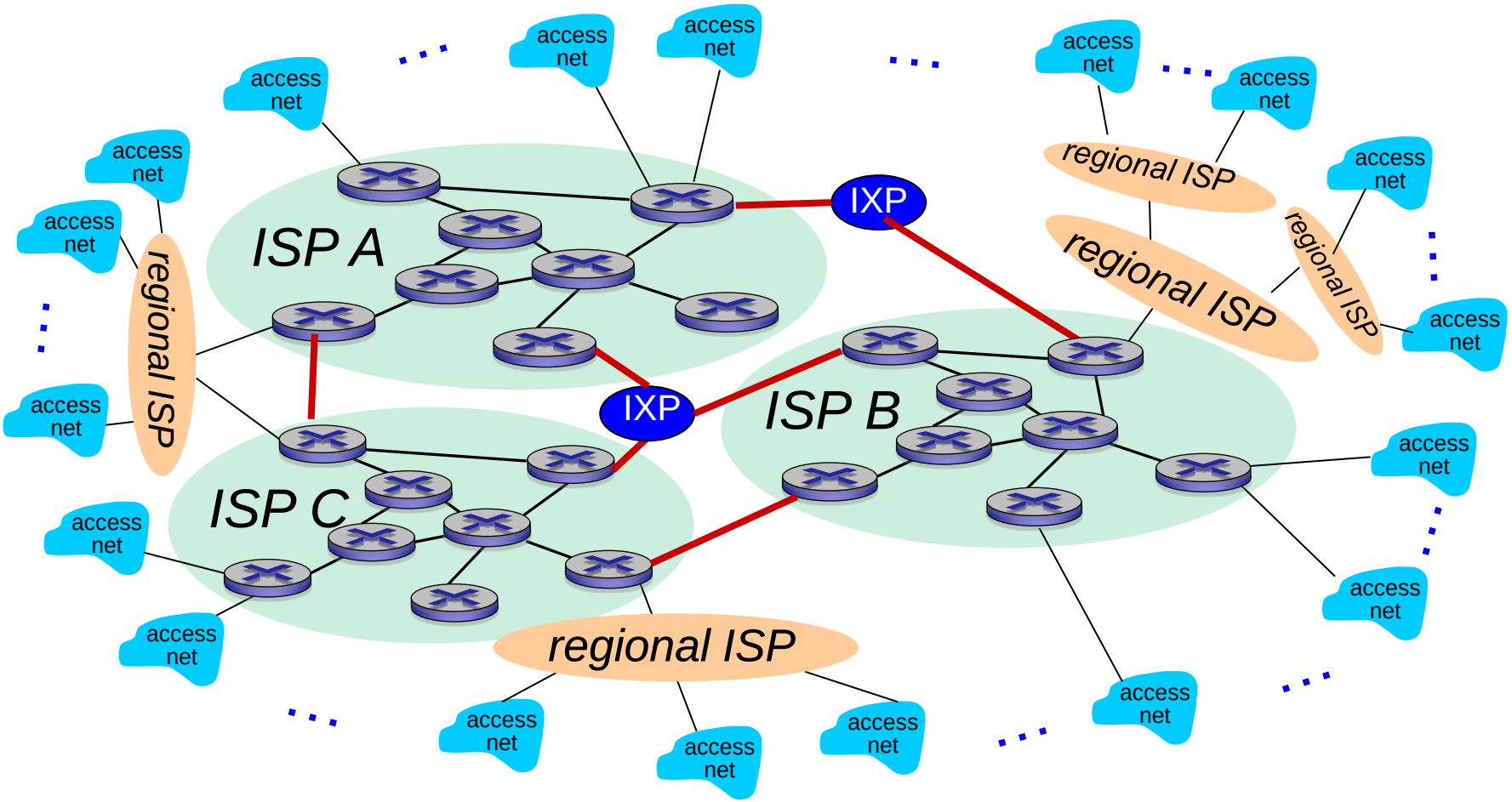
- ISPs must be interconnected



- a pair of nearby ISPs at the same level of the hierarchy can **peer**

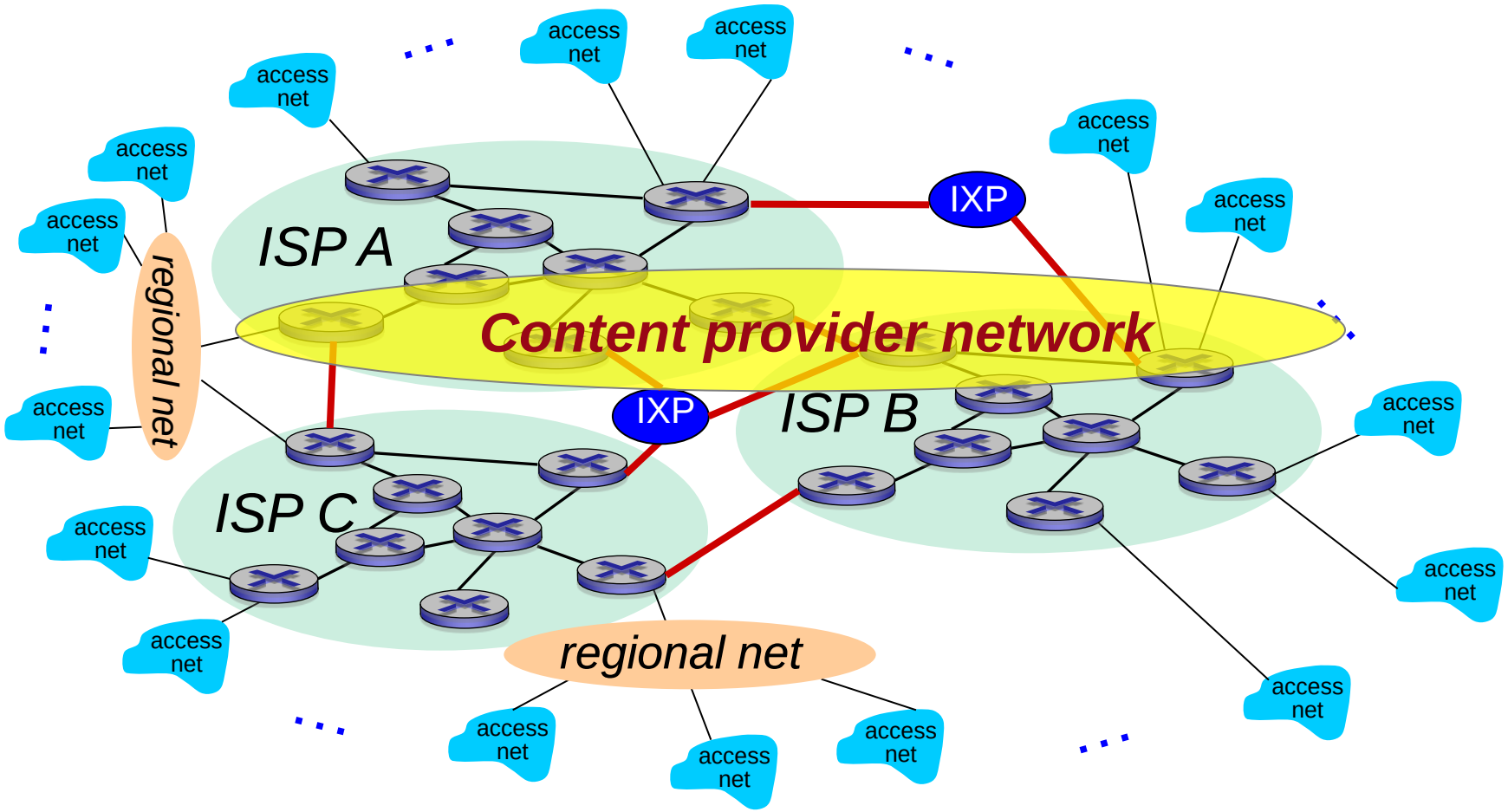
Internet structure: network of networks

- Regional networks may arise to connect access nets to ISPs
- Multi-tier hierarchy

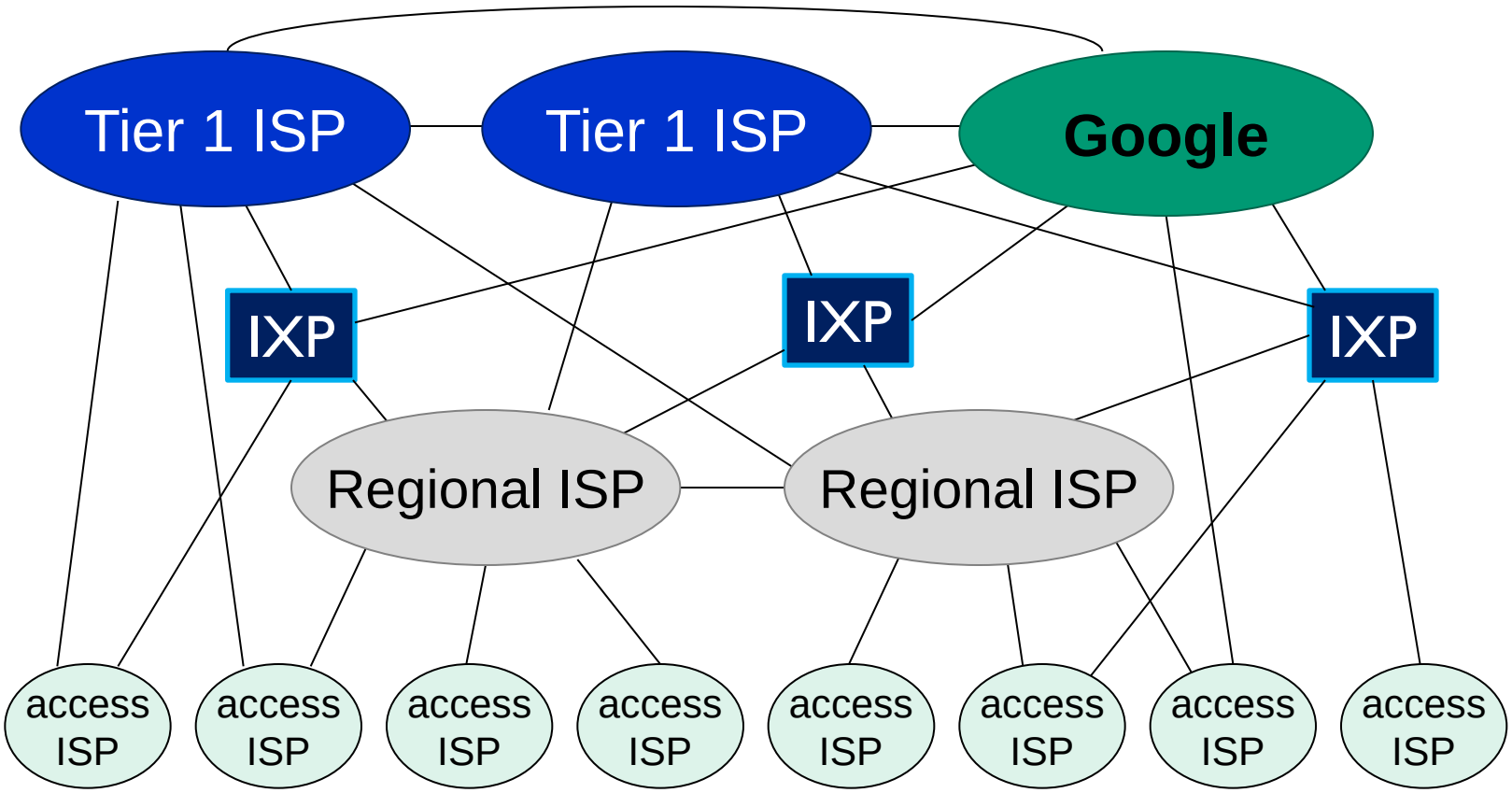


Internet structure: network of networks

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



Internet structure: network of networks



- at center: small # of well-connected large networks
  - "tier-1" commercial ISPs (e.g., Level 3 Communications, Sprint, AT&T, NTT), national & international coverage
  - content provider network (e.g., Google): private network that connects it's data centers to Internet, often bypassing tier-1, regional ISPs

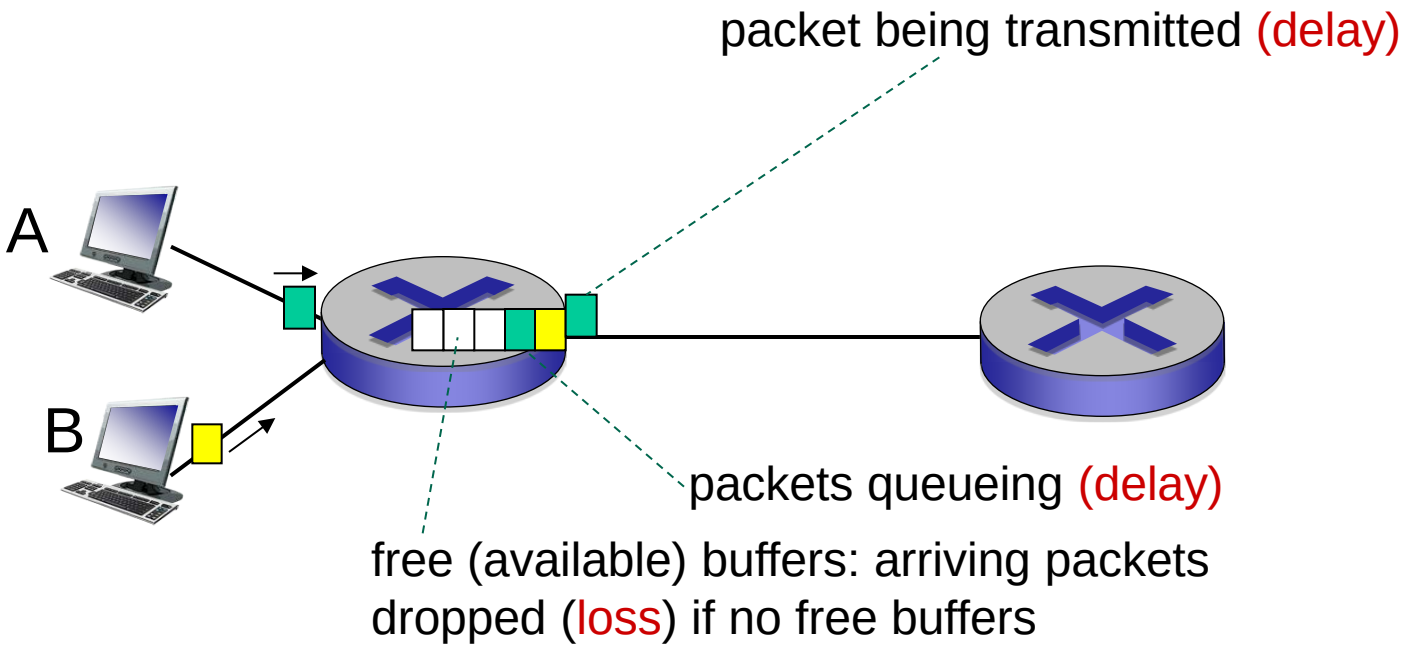
Roadmap

1. what *is* the Internet?
2. network edge
   - end systems, access networks, links
3. network core
   - packet switching, circuit switching, network structure
4. delay, loss, throughput in networks
5. protocol layers, service models
6. networks under attack: security
7. Internet history

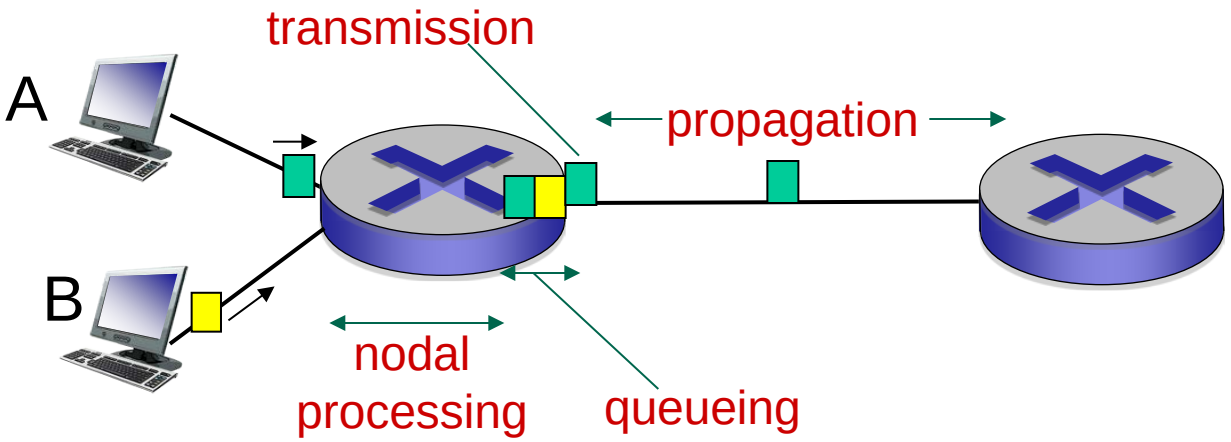
How do loss and delay occur?

packets *queue* in router buffers

- packet arrival rate to link (temporarily) exceeds output link capacity
- packets queue, wait for turn



Four sources of packet delay



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

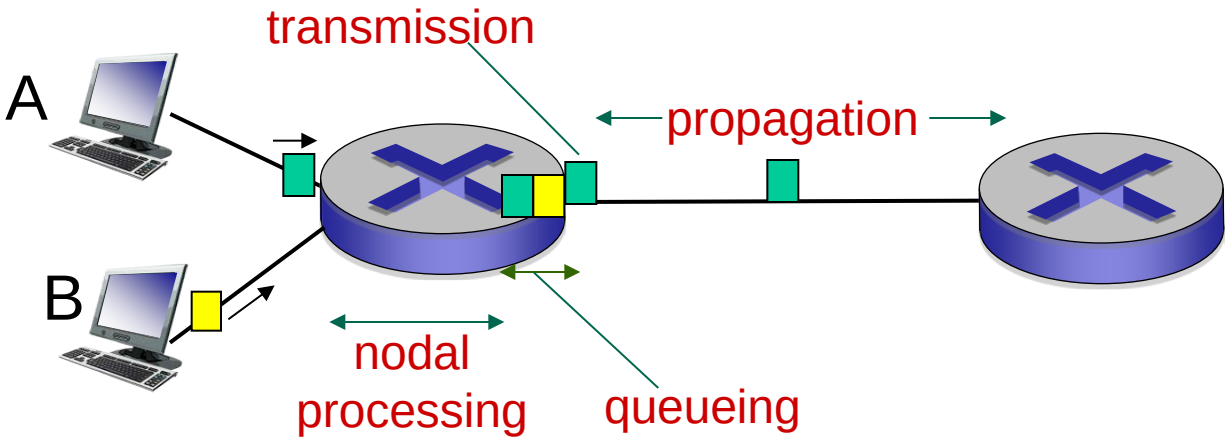
d_{proc}: nodal processing

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

d_{queue}: queueing delay

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

Four sources of packet delay



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

d_{trans}: transmission delay:

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

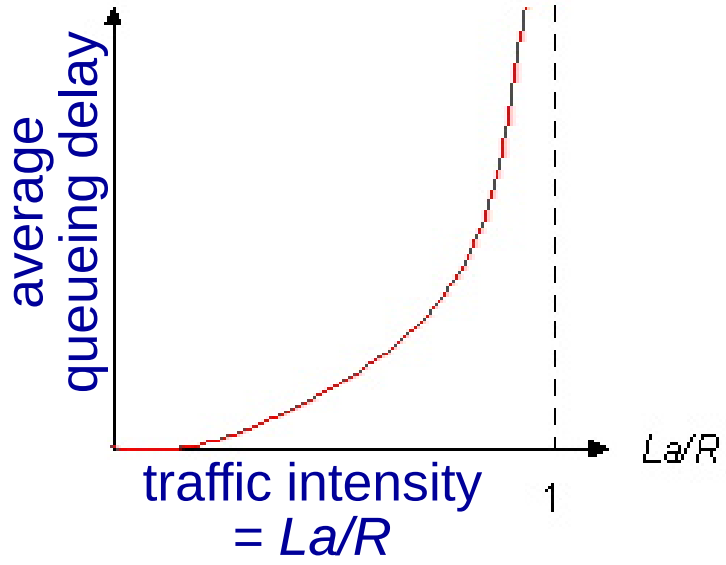
d_{trans} and d_{prop} *very* different

d_{prop}: propagation delay:

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

Queueing delay (revisited)

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



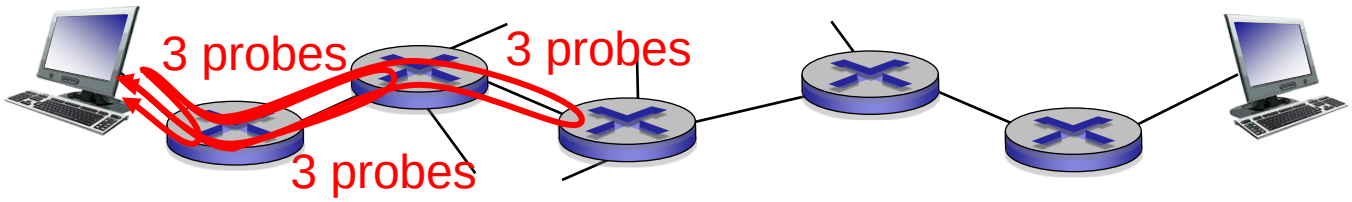
La/R ~ 0

La/R⟶1

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

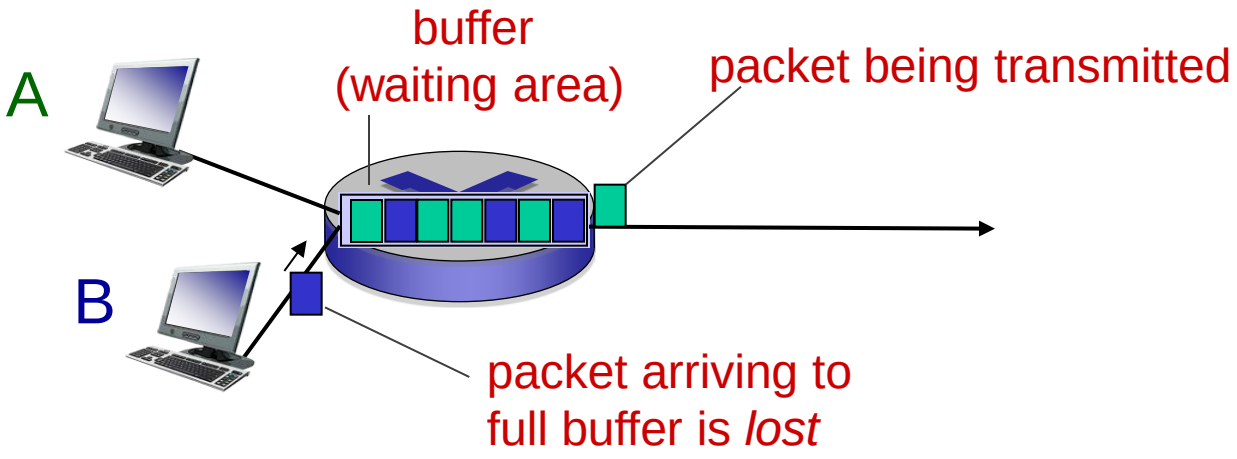
"Real" Internet delays and routes

- 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
  - router *i* will return packets to sender
  - sender times interval between transmission and reply.



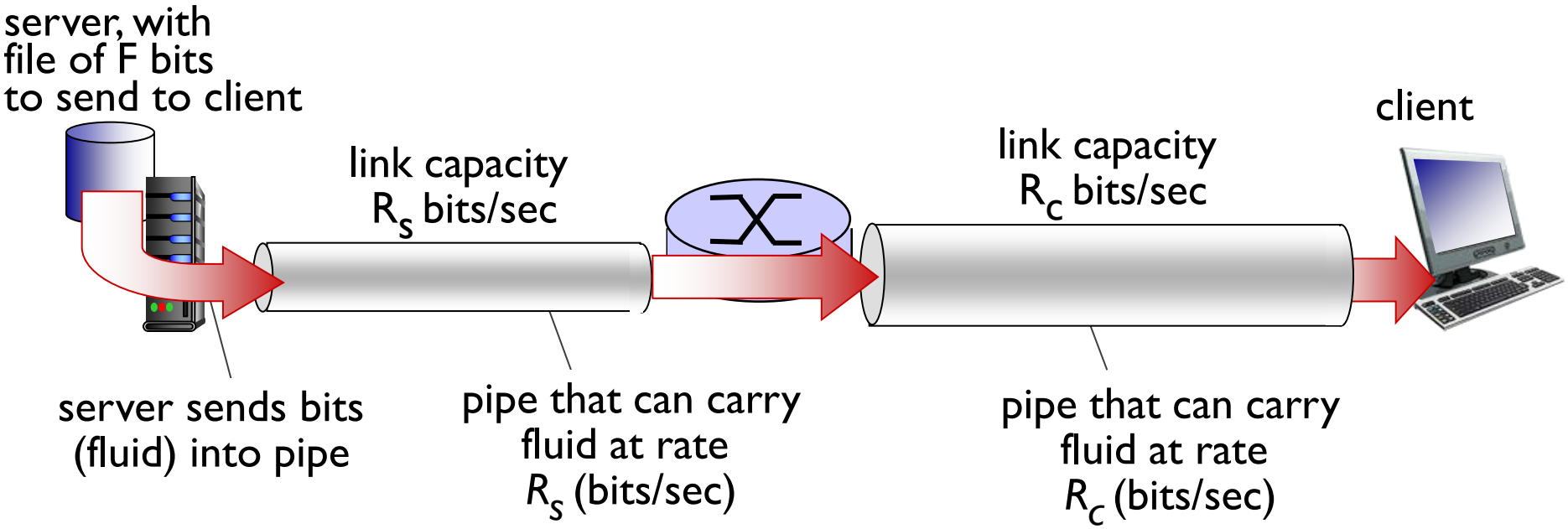
Packet loss

- Transmission buffer (queue) has finite capacity
- packet arriving to full queue dropped (i.e., lost)
- lost packet may be retransmitted by previous node, by source end system, or not at all



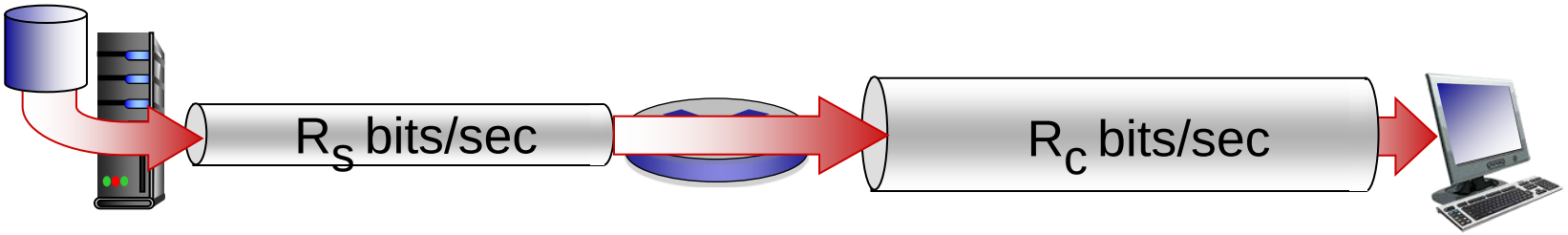
Throughput

- *throughput:* rate (bits/time unit) at which bits transferred between sender/receiver
  - *instantaneous:* rate at given point in time
  - *average:* rate over longer period of time

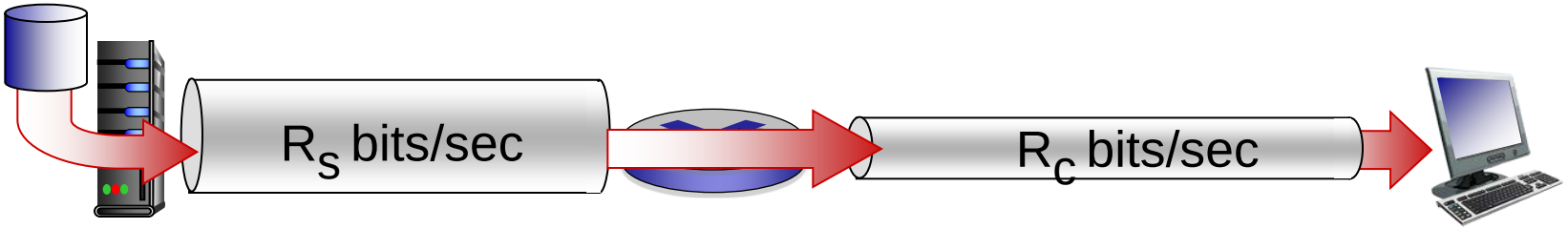


Throughput (more)

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



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

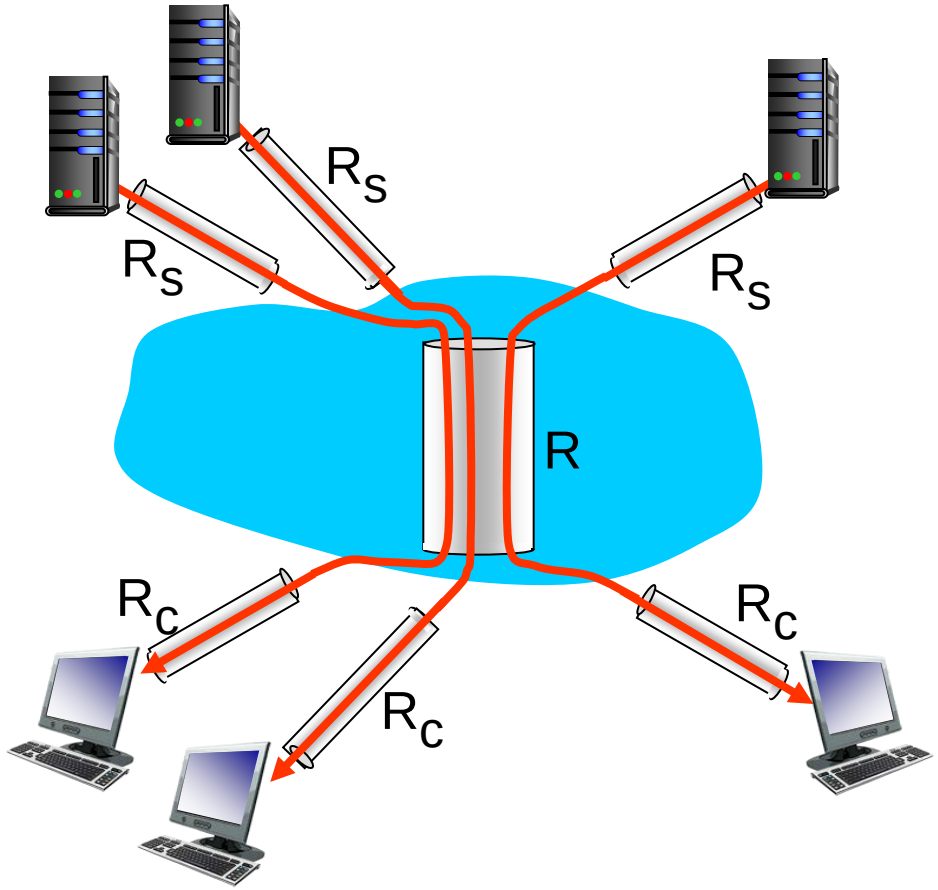


bottleneck link

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

Throughput: Internet scenario

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



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

Roadmap

1. what *is* the Internet?
2. network edge
   - end systems, access networks, links
3. network core
   - packet switching, circuit switching, network structure
4. delay, loss, throughput in networks
5. protocol layers, service models
6. networks under attack: security
7. Internet history

Protocol "layers"

Networks are complex, with many "pieces":

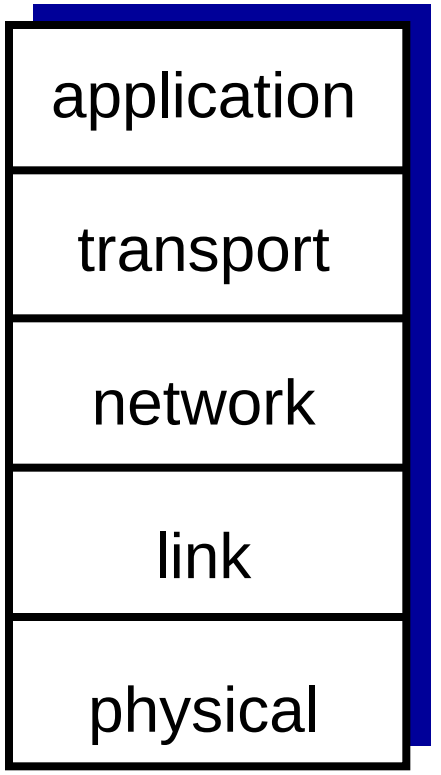
- hosts
- routers
- links of various media
- applications
- protocols
- hardware, software

Why layering?

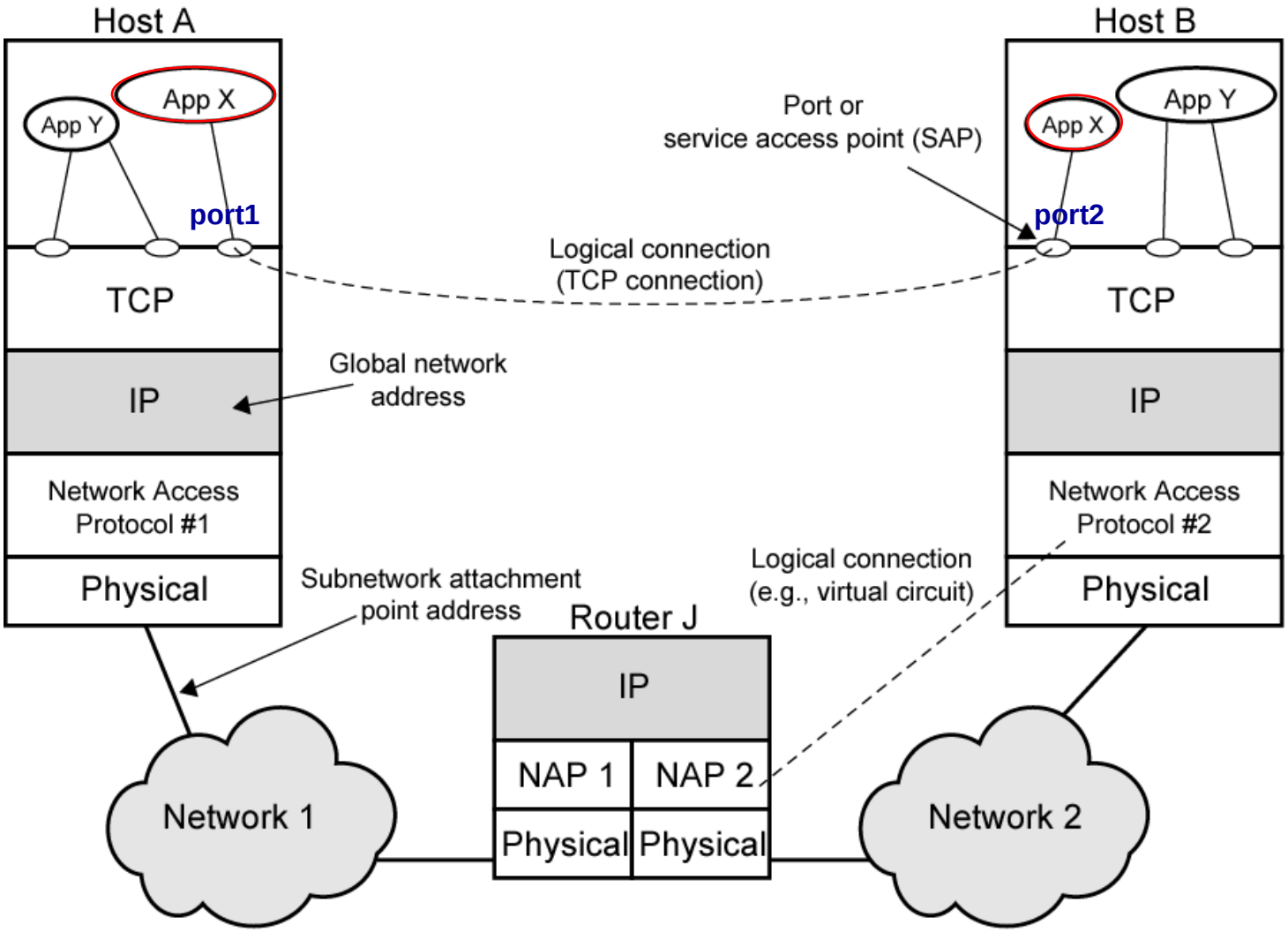
- Modularization: task of a large and complex system broken up to modules
  - allow us to efficiently handle a large and complex system
  - eases maintenance, updating of system
  - change of implementation of layer's service transparent to rest of system
- Layering is a popular way of structuring such a family of network protocols: Layered Architecture
- Number of Layers?
  - Internet: five layers
  - OSI model: seven layers

Internet protocol stack

- *application:* supporting network applications
  - FTP, SMTP, HTTP
- *transport:* data transfer between applications (peer process-process)
  - TCP, UDP
- *network:* routing of datagrams from source host to destination host
  - IP, routing protocols
- *link:* data transfer between neighboring network elements
  - Ethernet, 802.11 (WiFi), Cellular
- *physical:* bits "on the wire"



TCP/IP *Concept* (Internet)



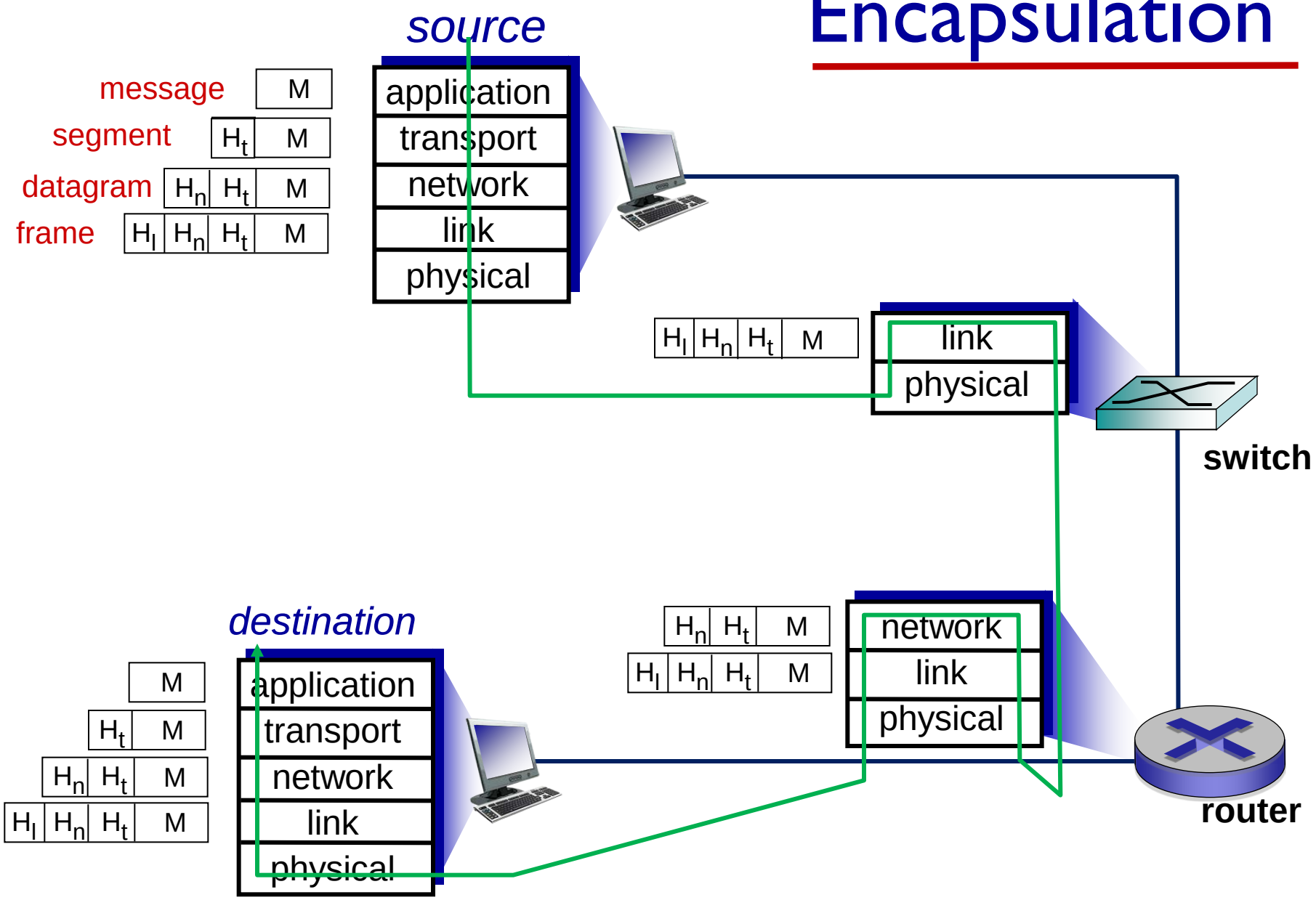
Two level addressing

- Each computer needs a unique network address
  - IP or internet address
- Each application on a (multi-tasking) host needs a unique address within the computer
  - Port number

[Example]

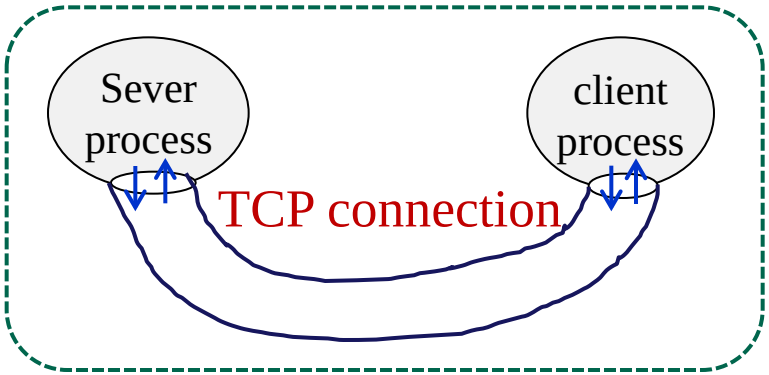
- Process associated with port 1 in host A sends message to port 2 in host B
  - Process at A hands down message to TCP to send to port 2
  - TCP hands down to IP to send to host B
  - IP hands down to network access layer (e.g. Ethernet) to send to router J
- Generates a set of encapsulated PDUs

Encapsulation



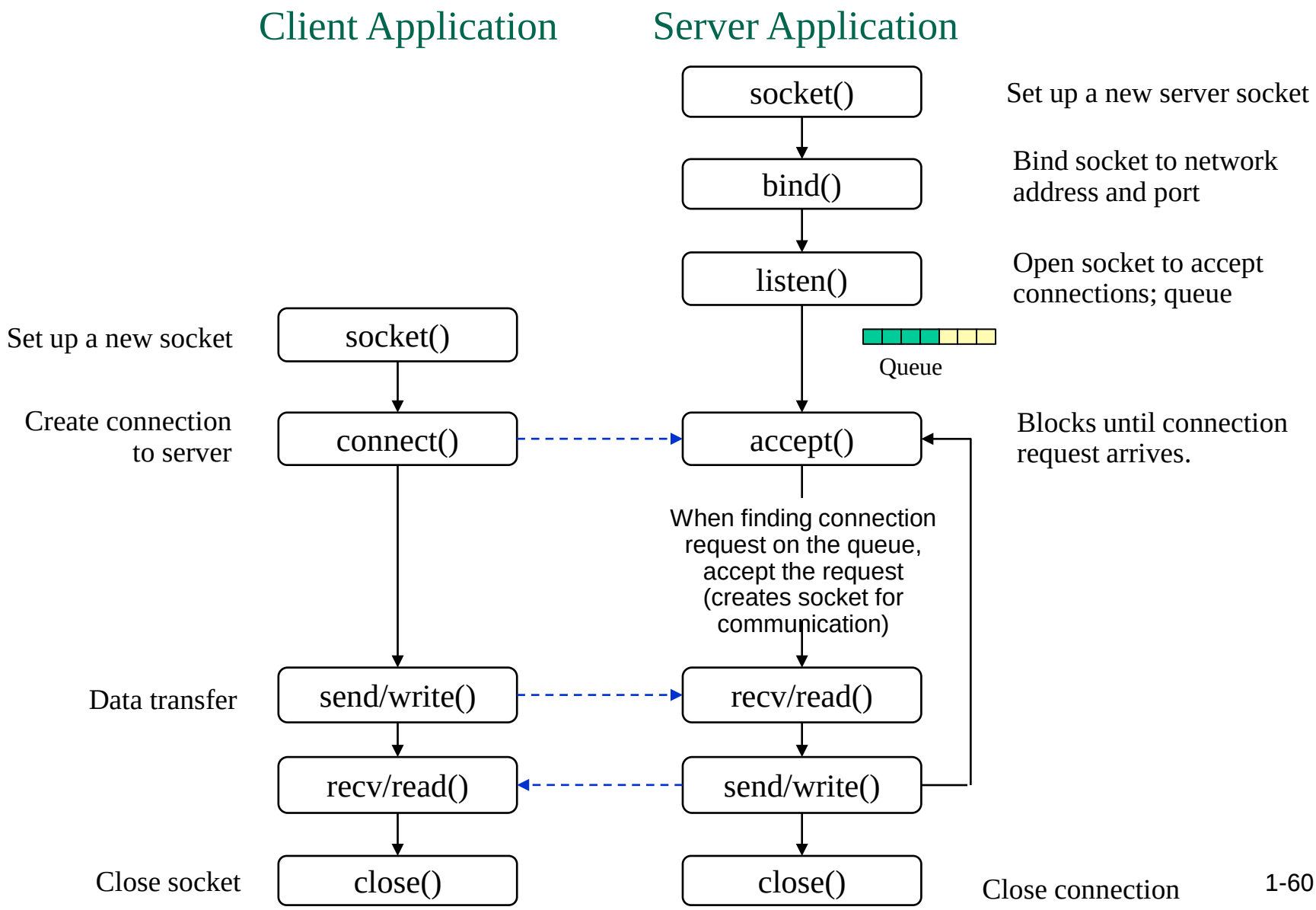
Socket

- Socket
  - Concatenation of a port value and an IP address
  - may be connection-oriented or connectionless
  - Generic communication interface for writing application programs that use TCP or UDP



- Socket Programming
  - enables communication between a client process and a server process (server-client model), without profound knowledge of network architecture

Stream(TCP) Socket Communication



Problems with Layering

- Inefficiency
  - each layer introduces overhead
- Restrictive
  - layer N may need access to lower layers than N-1
- Redundancy
  - of functions such as flow control, error handling, addressing, packetizing and encapsulation between layers

Standards

- Required to allow for interoperability between equipment
- Advantages
  - Ensures a large market for equipment and software
  - Allows products from different vendors to communicate
- Disadvantages
  - Freeze technology
  - May be multiple standards for the same thing

Standards Organizations

- Internet Society (IETF) : Internet
- ISO
- ITU
- IEEE 802: LAN/PAN
- OCF, OneM2M, Matter : IoT Platform
- LoRa Alliance, ZigBee Alliance, Bluetooth SIG : IoT
- 3GPP: Cellular (LTE, LTE-A. 5G, 6G)

Roadmap

1. what *is* the Internet?
2. network edge
   - end systems, access networks, links
3. network core
   - packet switching, circuit switching, network structure
4. delay, loss, throughput in networks
5. protocol layers, service models
6. networks under attack: security
7. Internet history

Network security

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

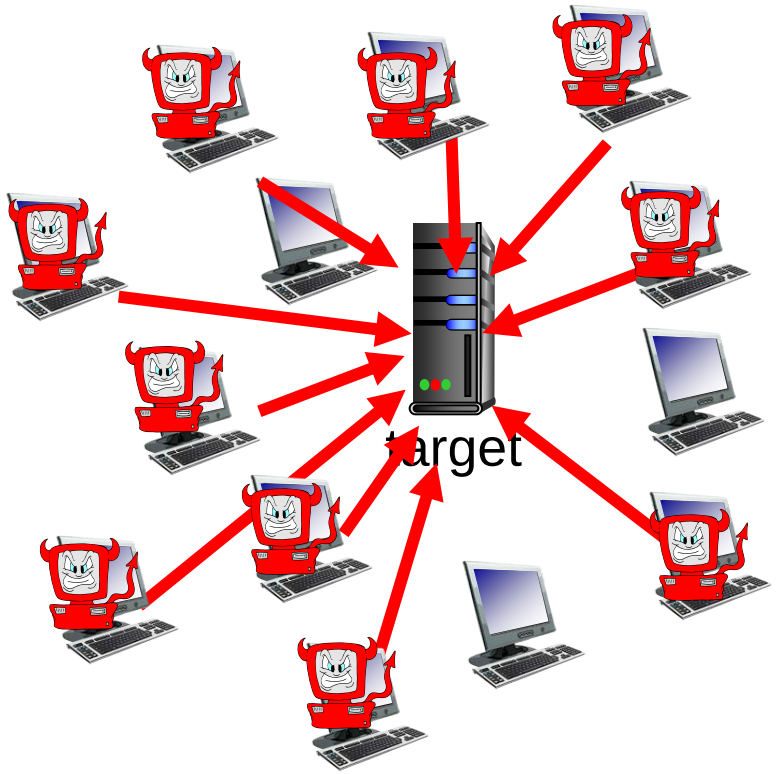
Bad guys: put malware into hosts via Internet

- malware can get in host from:
  - *virus:*
    - self-replicating infection by receiving/executing object (malware that require some form of user interaction to infect)
    - e.g.: e-mail attachment
  - *worm:*
    - self-replicating infection by passively receiving object that gets itself executed (malware that can enter a device without any explicit user interaction)
- spyware malware collects user's private information (e.g., keystrokes, password, web sits visited) and uploads the info to collection site
- infected host can be enrolled in botnet, used for spam E-mail distribution or DDoS attacks

Bad guys: attack server, network infrastructure

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

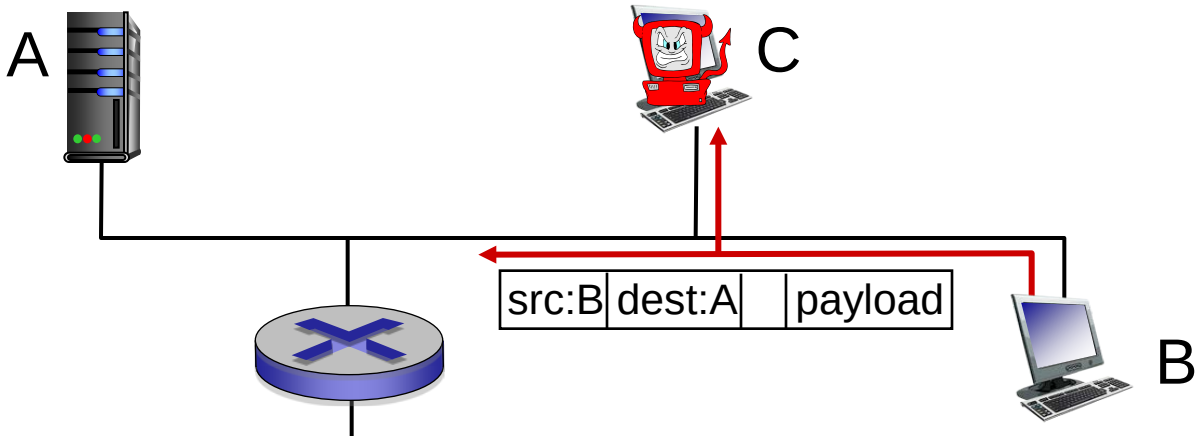
1. select target
2. break into hosts around the network (see botnet)
3. send packets to target from compromised hosts



Bad guys can sniff packets

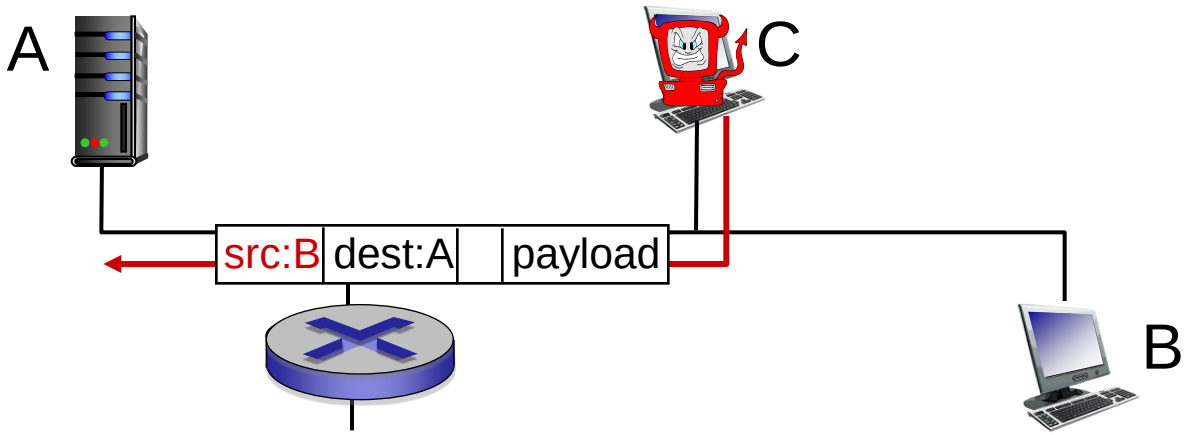
packet "sniffing":

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



Bad guys can use fake addresses

IP spoofing: send packet with false source address



Roadmap

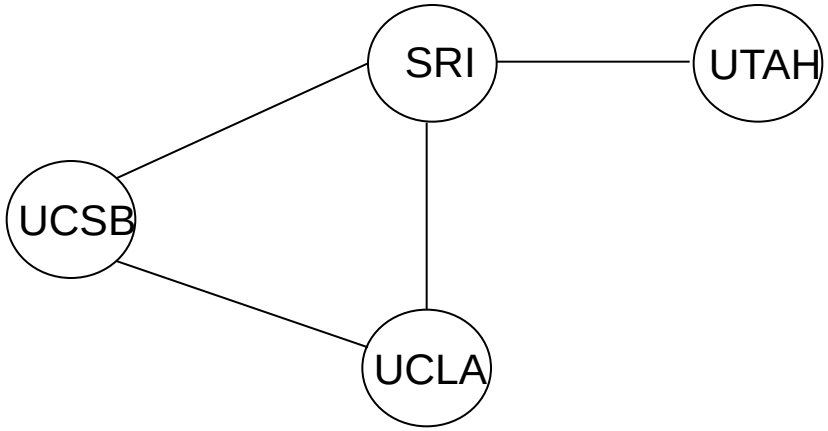
1. what *is* the Internet?
2. network edge
   - end systems, access networks, links
3. network core
   - packet switching, circuit switching, network structure
4. delay, loss, throughput in networks
5. protocol layers, service models
6. networks under attack: security
7. Internet history

Internet history

1961-1972: *Development of packet-switching principles*

- 1961: Kleinrock, using queueing theory, demonstrated effectiveness of packet-switching for burst traffic
- 1964: Baran : packet-switching for secure voice in military net
- 1967: Overall plan for ARPAnet by Advanced Research Projects Agency
- 1969: first ARPAnet node (UCLA). Three nodes are added UCSB, Utah, Standard RI

- 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 nets*

- 1970: ALOHAnet (packet-based radio network in Hawaii)
- 1974: Cerf and Kahn: architecture for interconnecting networks
  - minimalism-no internal changes required to interconnect networks
  - best effort service model
  - decentralized control
- 1976: Ethernet at Xerox PARC
- late 70's: proprietary architectures: DECnet, SNA, etc.
- 1979: ARPAnet has 200 nodes

Internet history

1980-1990: *new protocols, a proliferation of networks*

- 1982: smtp e-mail protocol defined
- 1983: deployment of TCP/IP (replacing the NCP protocol).
- 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, 2000's: *commercialization, the Web, new apps*

- early 1990's: ARPAnet decommissioned
- 1991: NSF lifts restrictions on commercial use of NSFnet (decommissioned, 1995)
- early 1990s: Web
  - 1989~1991: Berners-Lee developed initial version of HTML, HTTP, a Web server, Web browser
  - 1994: Andressen developed a Web browser with GUI interface, Mosaic, later Netscape
  - late 1990's: commercialization of the Web
- late 1990's – 2000's:
  - more killer apps: P2P file sharing
  - network security to forefront
  - est. 50 million host, 100 million+ users
  - backbone links running at Gbps

Internet history

2005-present

- ~5B devices attached to Internet (2016)
  - smartphones and tablets
- aggressive deployment of broadband access
- increasing ubiquity of high-speed wireless access
- emergence of online social networks:
  - Facebook: ~ one billion users
- service providers (Google, Microsoft) create their own networks
  - bypass Internet, providing "instantaneous" access to search, video content, email, etc.
- e-commerce, universities, enterprises running their services in "cloud" (e.g., Amazon EC2)