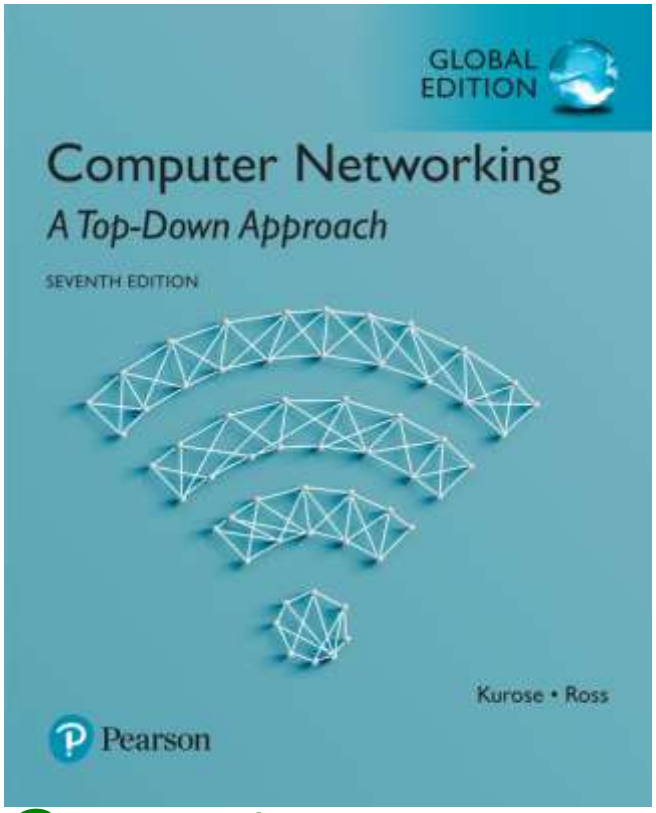
Chapter 1 Introduction



A note on the use of these Powerpoint slides:

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

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

Thanks and enjoy! JFK/KWR

Computer Networking: A Top Down Approach

7th Edition, Global Edition

Jim Kurose, Keith Ross
Pearson

April 2016

Chapter 1: introduction

our goal:

- get "feel" and terminology
- more depth, detail *later* in course
- approach:
  - use Internet as example

overview:

- what's the Internet?
- what's a protocol?
- network edge; hosts, access net, physical media
- network core: packet/circuit switching, Internet structure
- performance: loss, delay, throughput
- security
- protocol layers, service models
- history

Chapter 1: roadmap

1.1 what *is* the Internet?

1.2 network edge

- end systems, access networks, links

1.3 network core

- packet switching, circuit switching, network structure

1.4 delay, loss, throughput in networks

1.5 protocol layers, service models

1.6 networks under attack: security

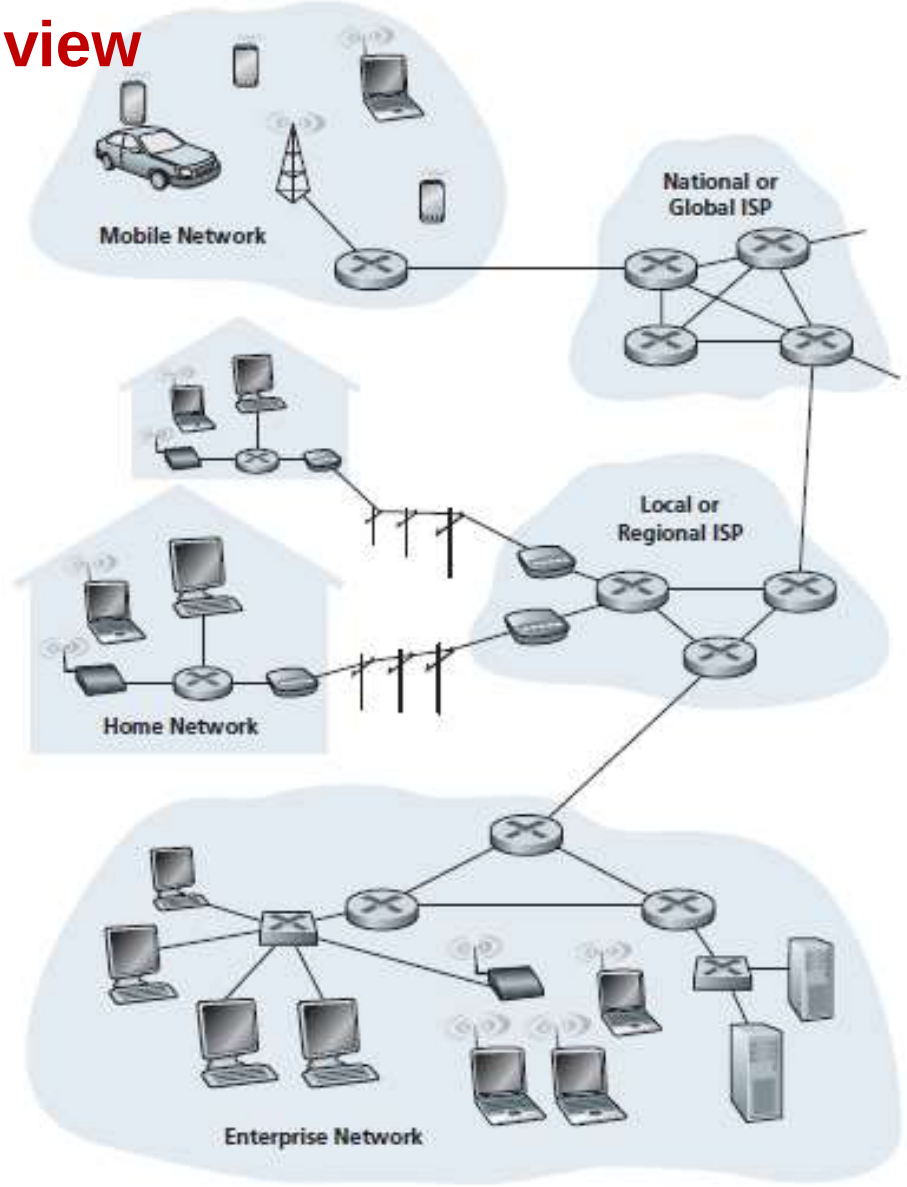
1.7 history

What Is the Internet?

- Internet can be defined in two ways
  - The basic components – i.e. the nuts and bolt
    - the basic hardware and software components that make up the Internet
  - The services it provide
    - a networking infrastructure that provides services to distributed applications

Internet: the network of networks

The 'nuts and bolts' view

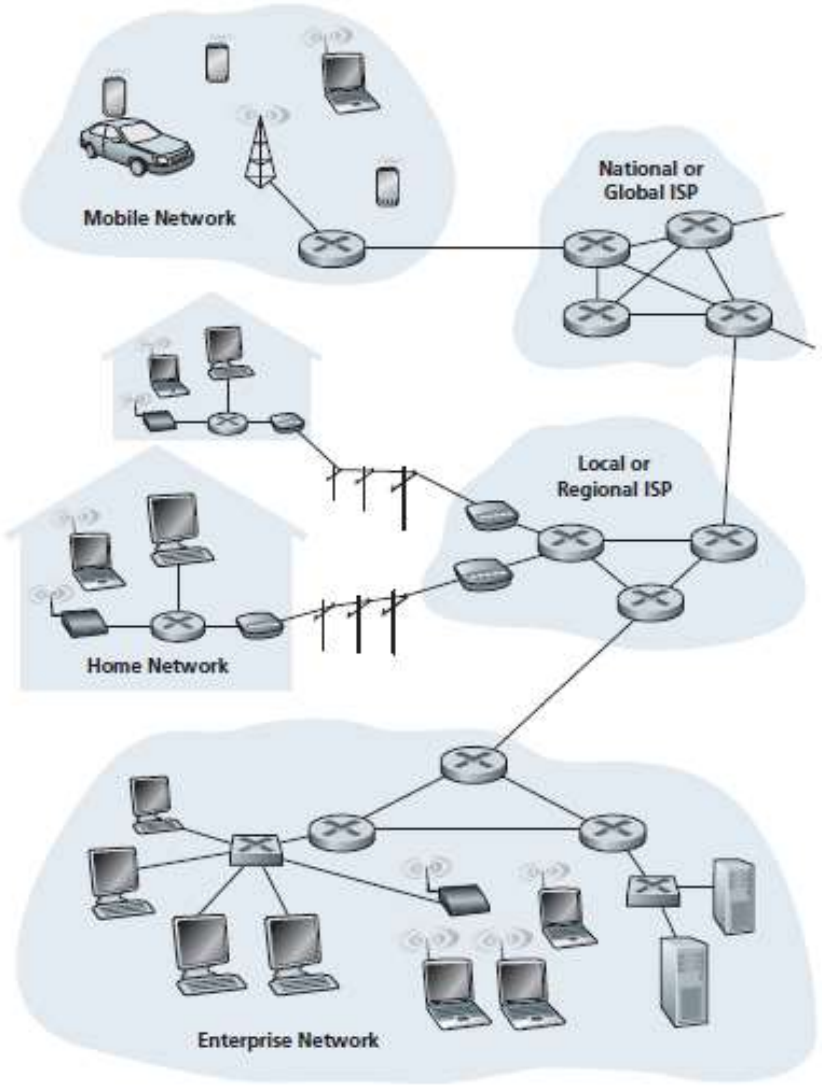


- billions of connected computing devices:
  - *hosts* = *end systems*
  - running *network apps*
- *communication links*
  - fiber, copper, radio, satellite
  - transmission rate: *bandwidth*
- *Packet switches:* forward packets (chunks of data)
  - *routers* and *switches*

• All the pieces of the Internet run *protocols* that controls the sending and receiving of information

• TCP/IP are two of the most important protocols in the Internet.

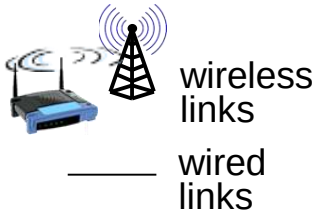
The 'service' view



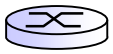
- These applications include
  - *email, Web surfing, social networks, instant messaging, Voiceover-IP (VoIP), video streaming, distributed games, peer-to-peer (P2P) file sharing*, *etc*
  - *Distributed applications* involve multiple end systems exchanging data
- Have *Application Programming Interface (API)*
  - to communicate between different platforms through the Internet
- Internet is a network of networks
  - Interconnected ISPs

What's the Internet: "nuts and bolts" view

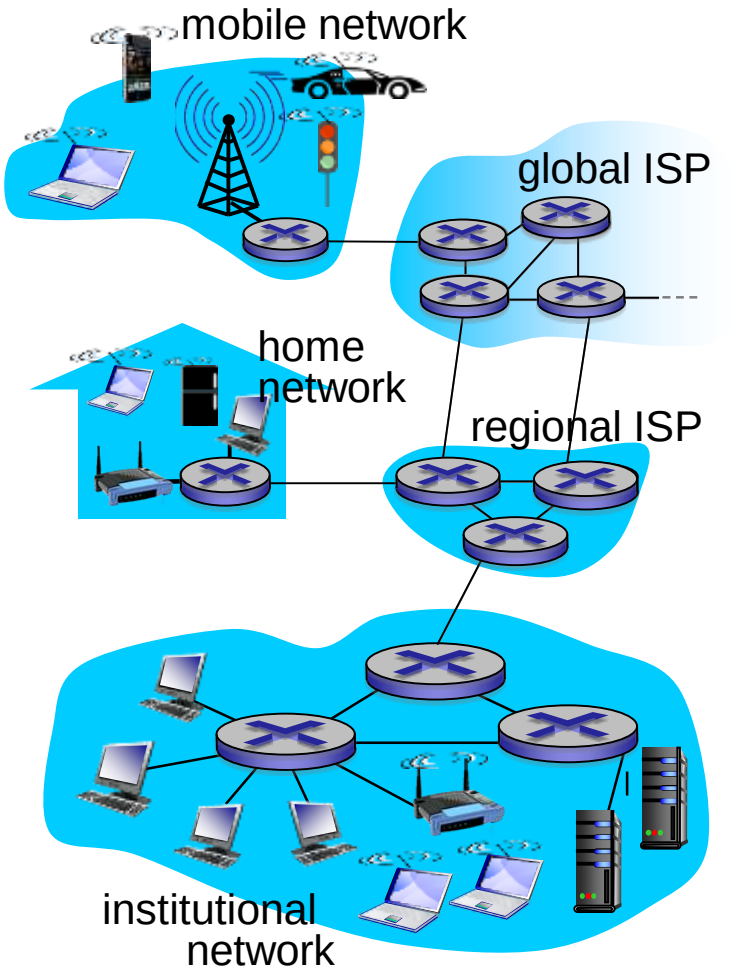
- billions of connected computing devices:
  - *hosts = end systems*
  - running *network apps*



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

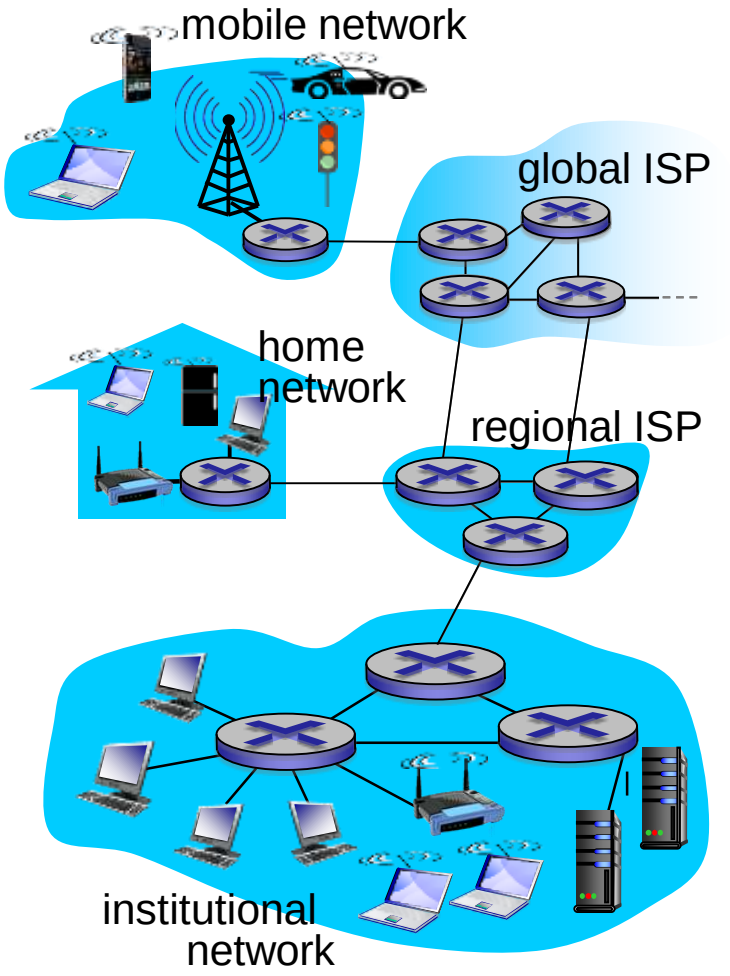


- *packet switches:* forward packets (chunks of data)
  - *routers* and *switches*



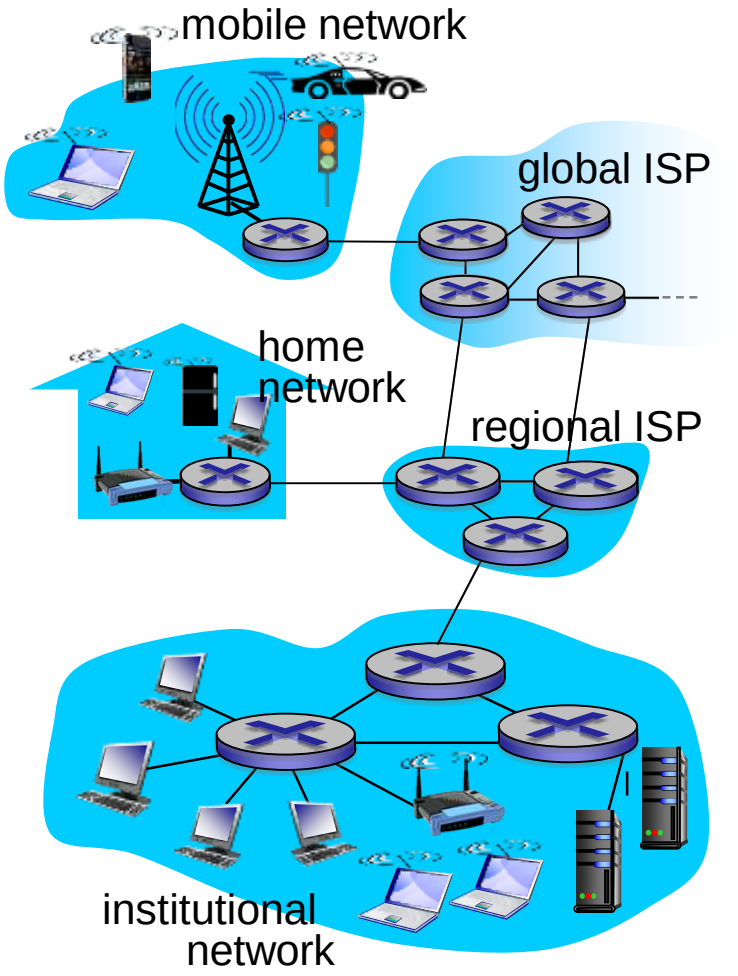
What's the Internet: "nuts and bolts" view

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



What's 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



“Fun” Internet-connected devices

IP picture frame
http://www.ceiva.com/

Web-enabled toaster + weather forecaster

Smart Lock

Tweet-a-watt:
monitor energy use

Internet
refrigerator

Slingbox: watch,
control cable TV remotely

sensorized,
bed
mattress

Internet phones

What's a protocol?

human protocols:

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

… specific messages sent

… specific actions taken when messages received, or other events

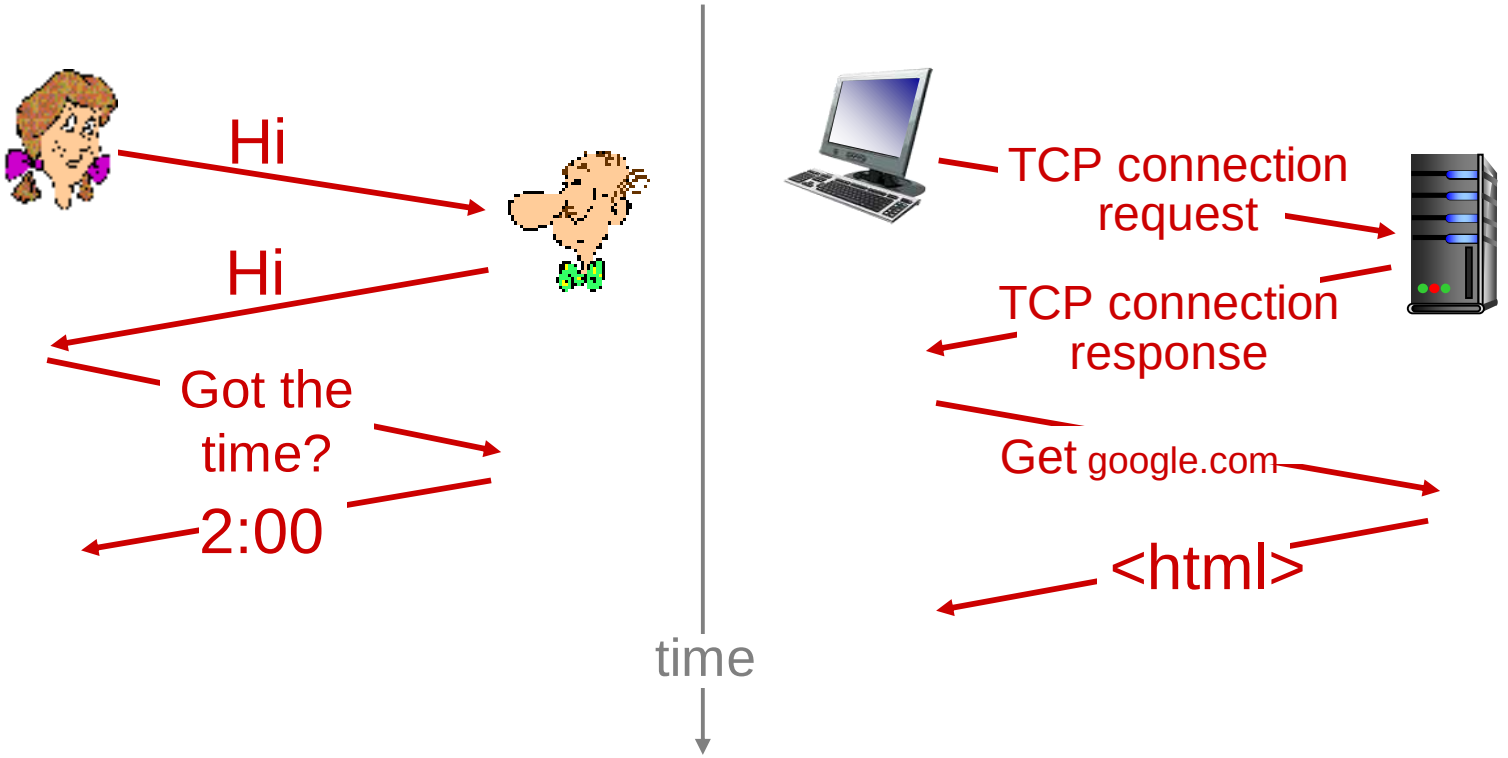
network protocols:

- 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 message transmission, receipt

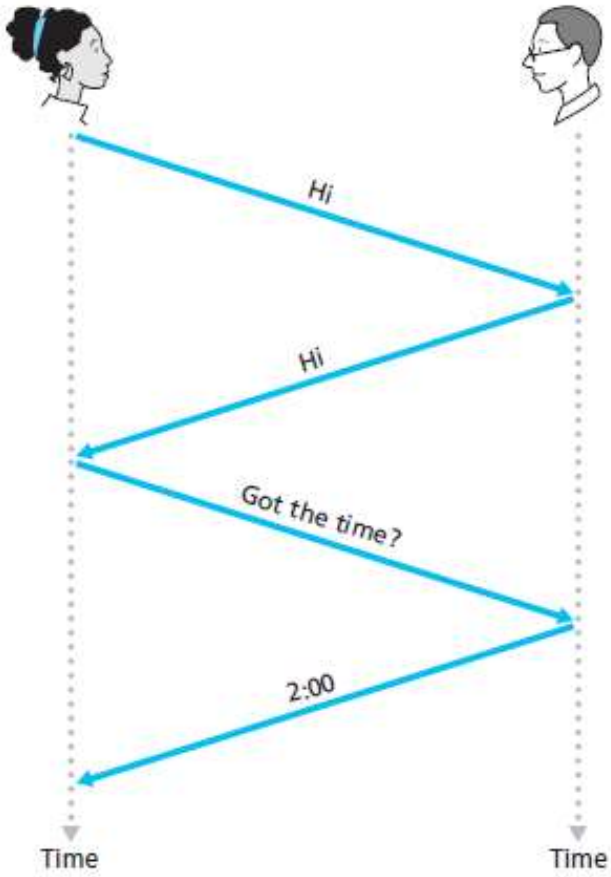
What's a protocol?

a human protocol and a computer network protocol:

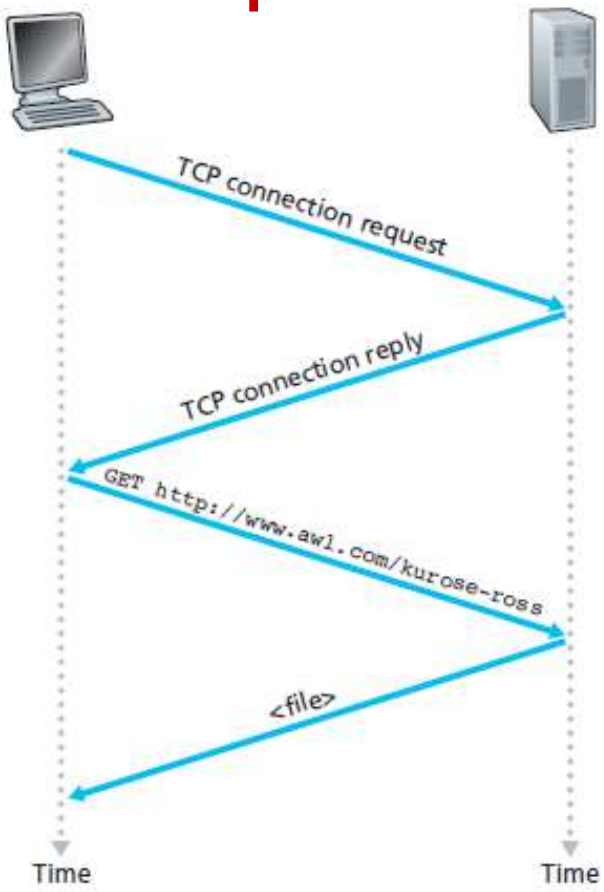


Q: other human protocols?

A human protocol



A computer network protocol



Chapter 1: roadmap

1.1 what *is* the Internet?

1.2 network edge

- end systems, access networks, links

1.3 network core

- packet switching, circuit switching, network structure

1.4 delay, loss, throughput in networks

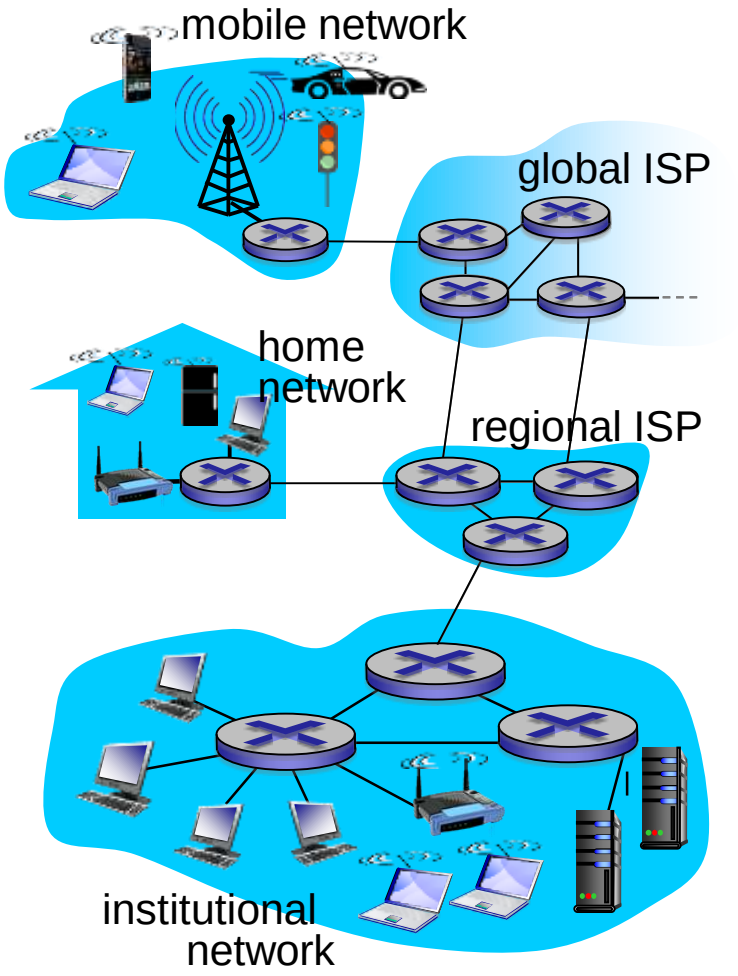
1.5 protocol layers, service models

1.6 networks under attack: security

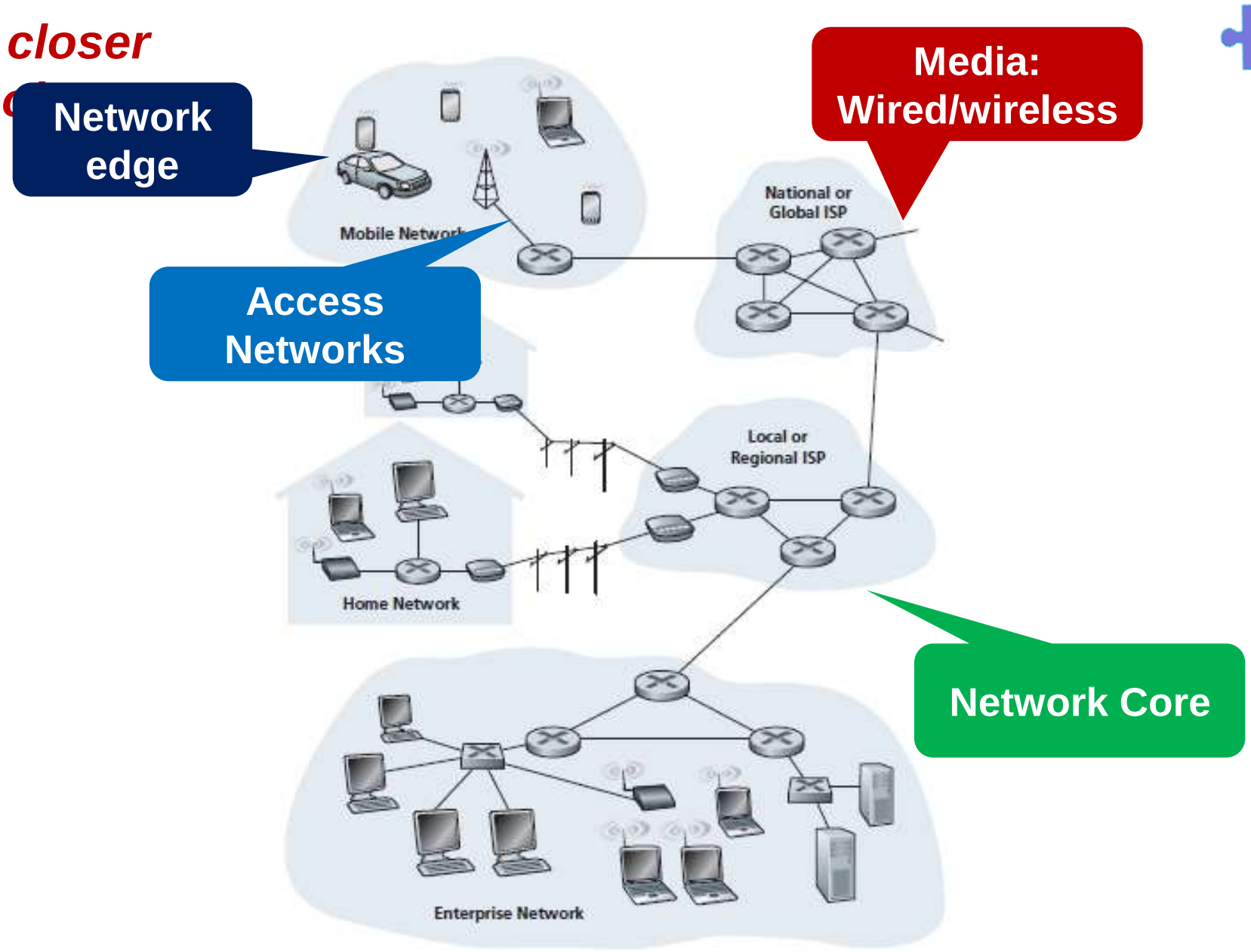
1.7 history

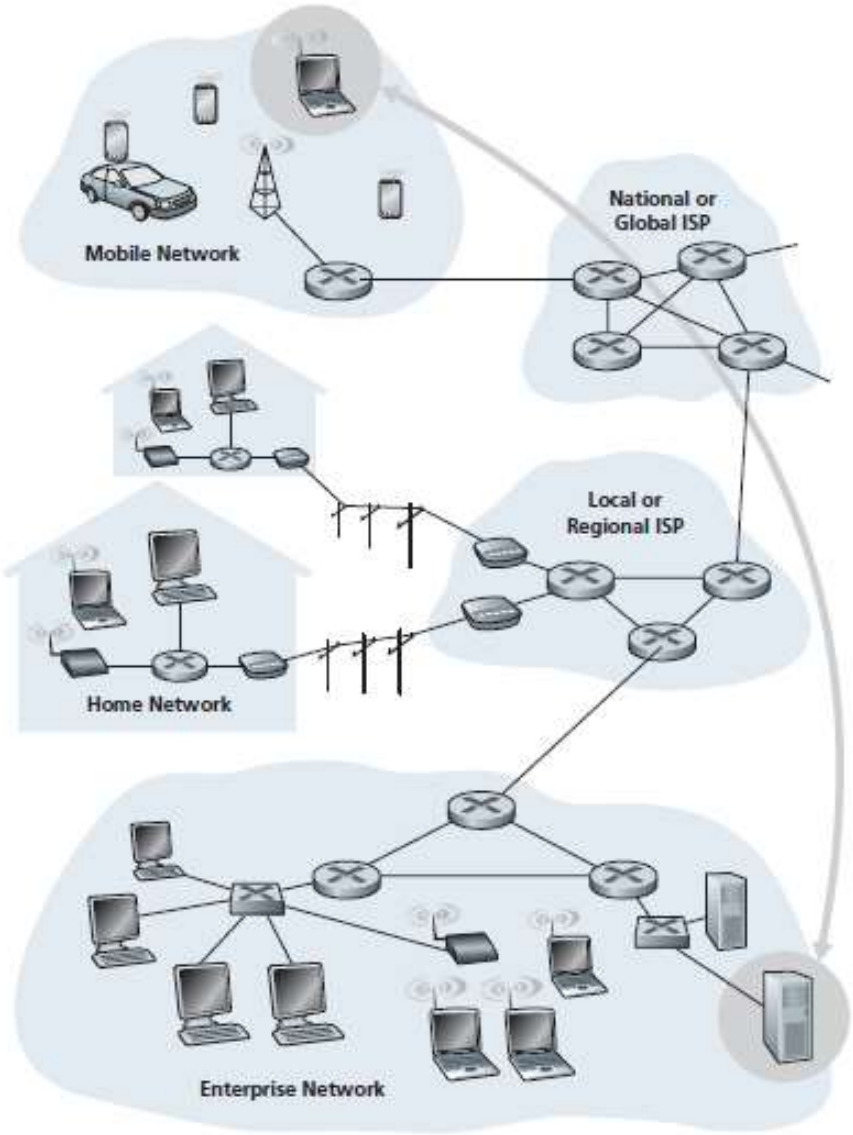
A closer look at network structure:

- *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

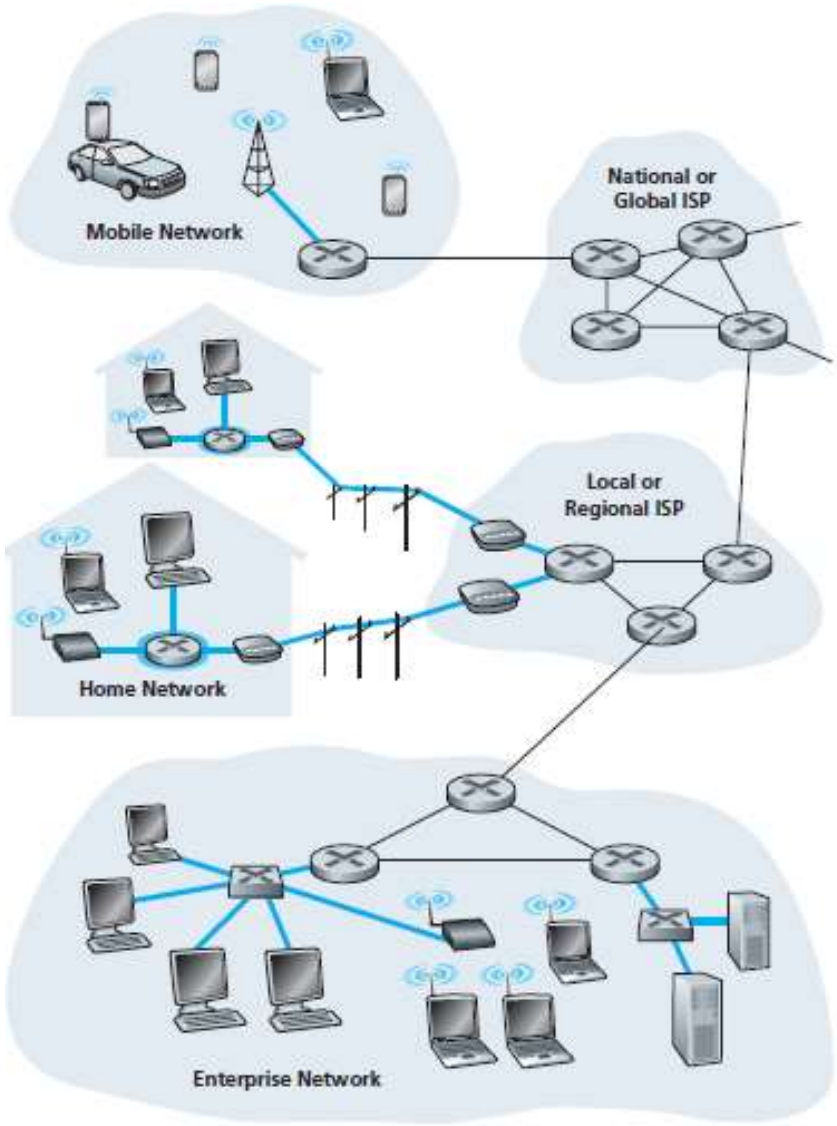


A closer look





End system interaction



Access networks

The network that physically connects an end system to the first router (also known as the "*edge router*") on a path from the end system to any other distant end system.

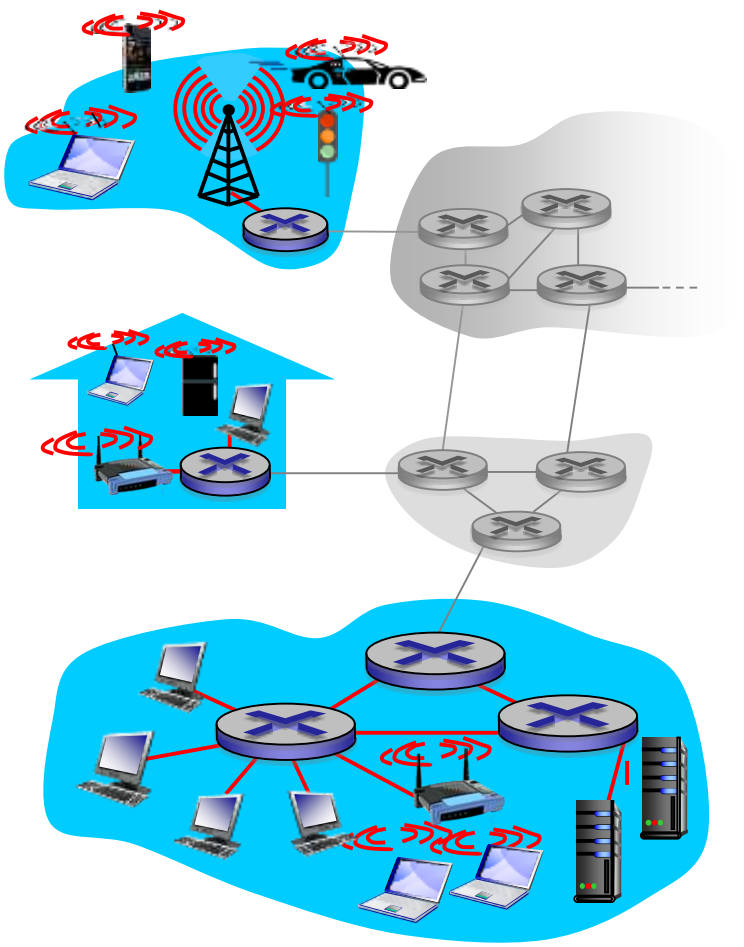
Access networks and physical media

Q: How to connect end systems to edge router?

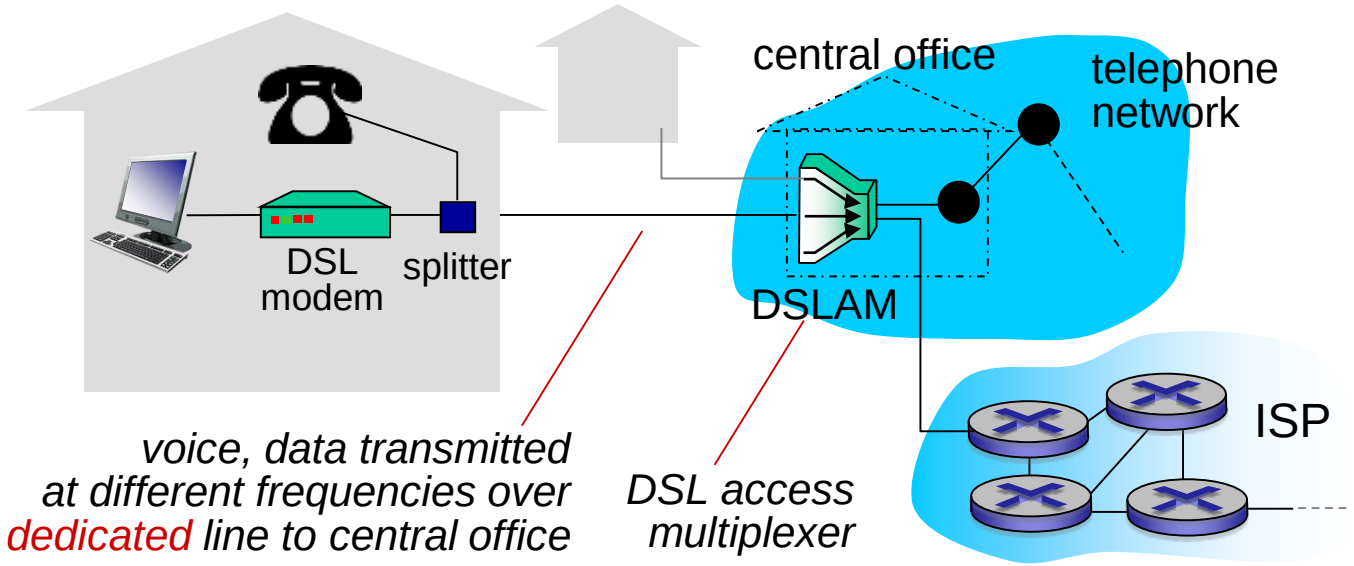
- residential access nets
- institutional access networks (school, company)
- mobile access networks

keep in mind:

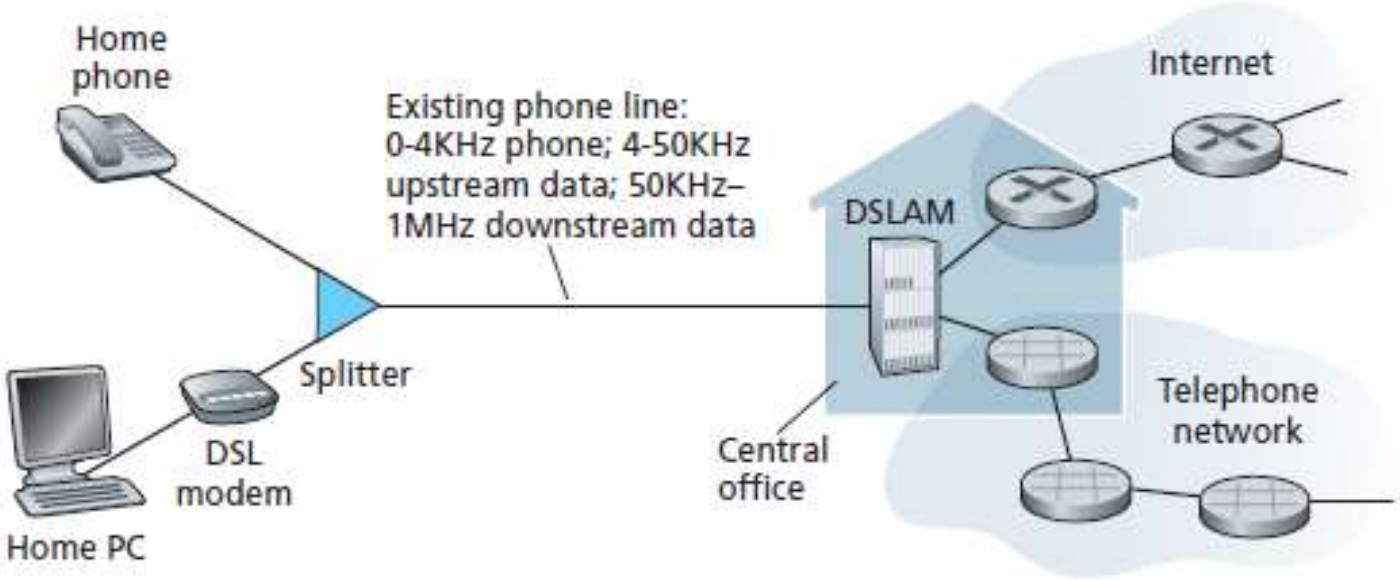
- bandwidth (bits per second) of access network?
- shared or dedicated?



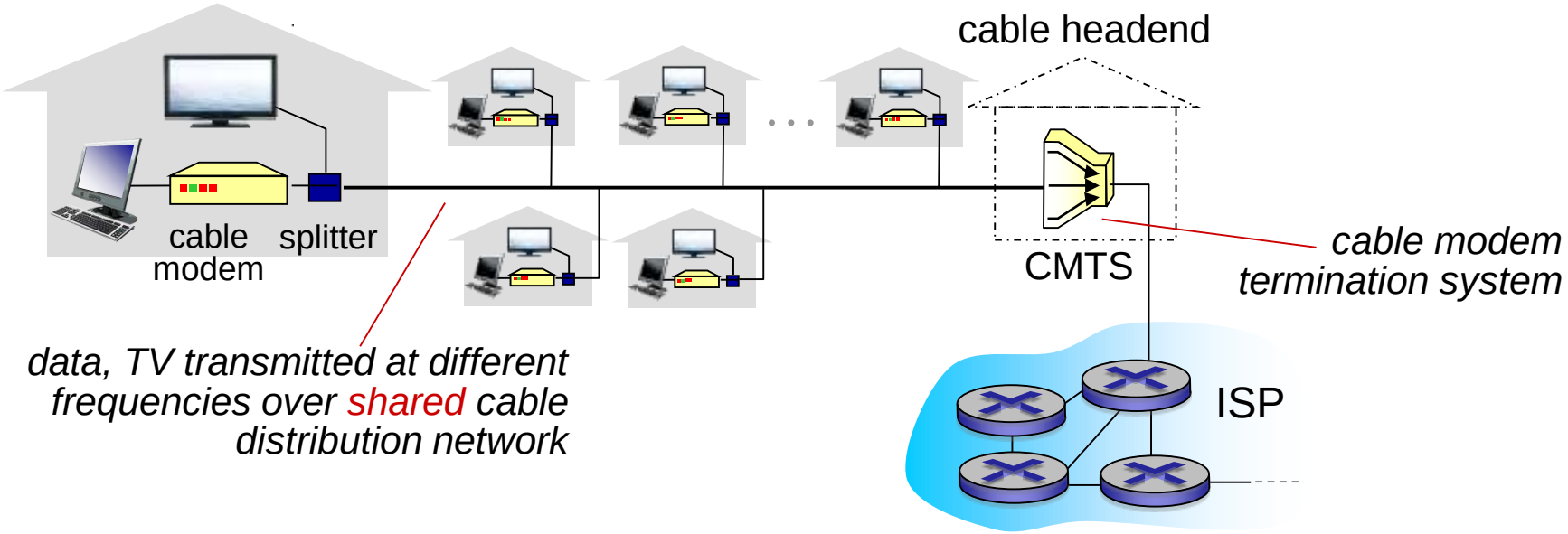
Access network: digital subscriber line (DSL)



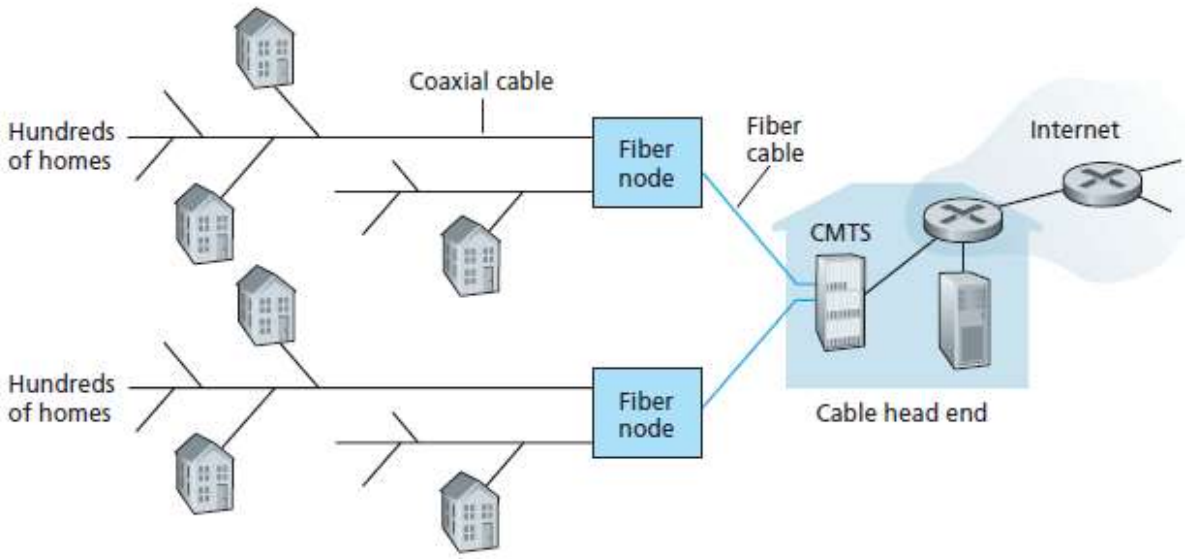
- use *existing* telephone line to central office DSLAM
  - data over DSL phone line goes to Internet
  - voice over DSL phone line goes to telephone net
- < 2.5 Mbps upstream transmission rate (typically < 1 Mbps)
- < 24 Mbps downstream transmission rate (typically < 10 Mbps)


Home phone
Existing phone line:
0-4KHz phone; 4-50KHz
upstream data; 50KHz–
1MHz downstream data
Internet
DSLAM
Splitter
DSL modem
Home PC
Central office
Telephone network

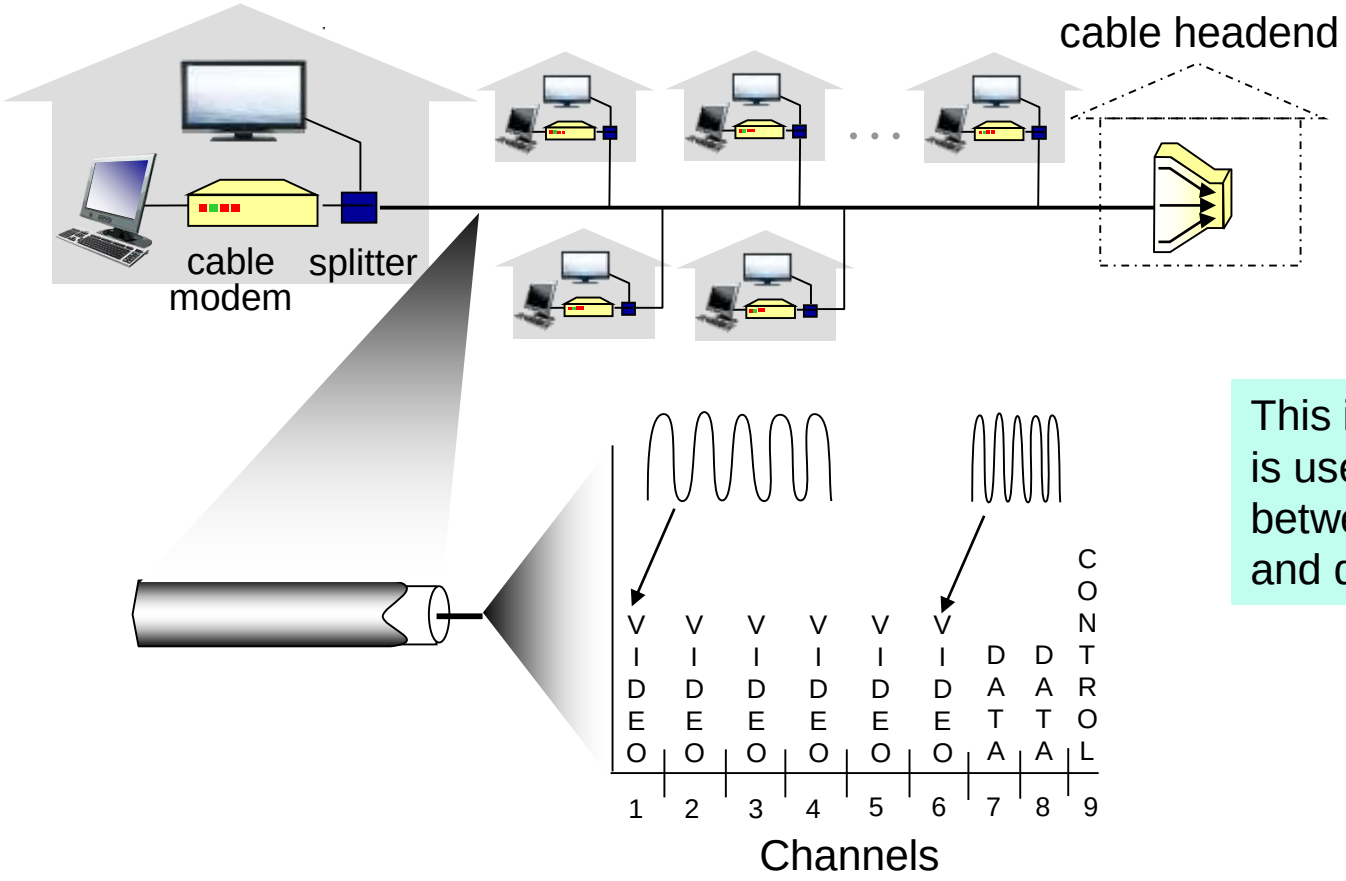
Access network: cable network



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

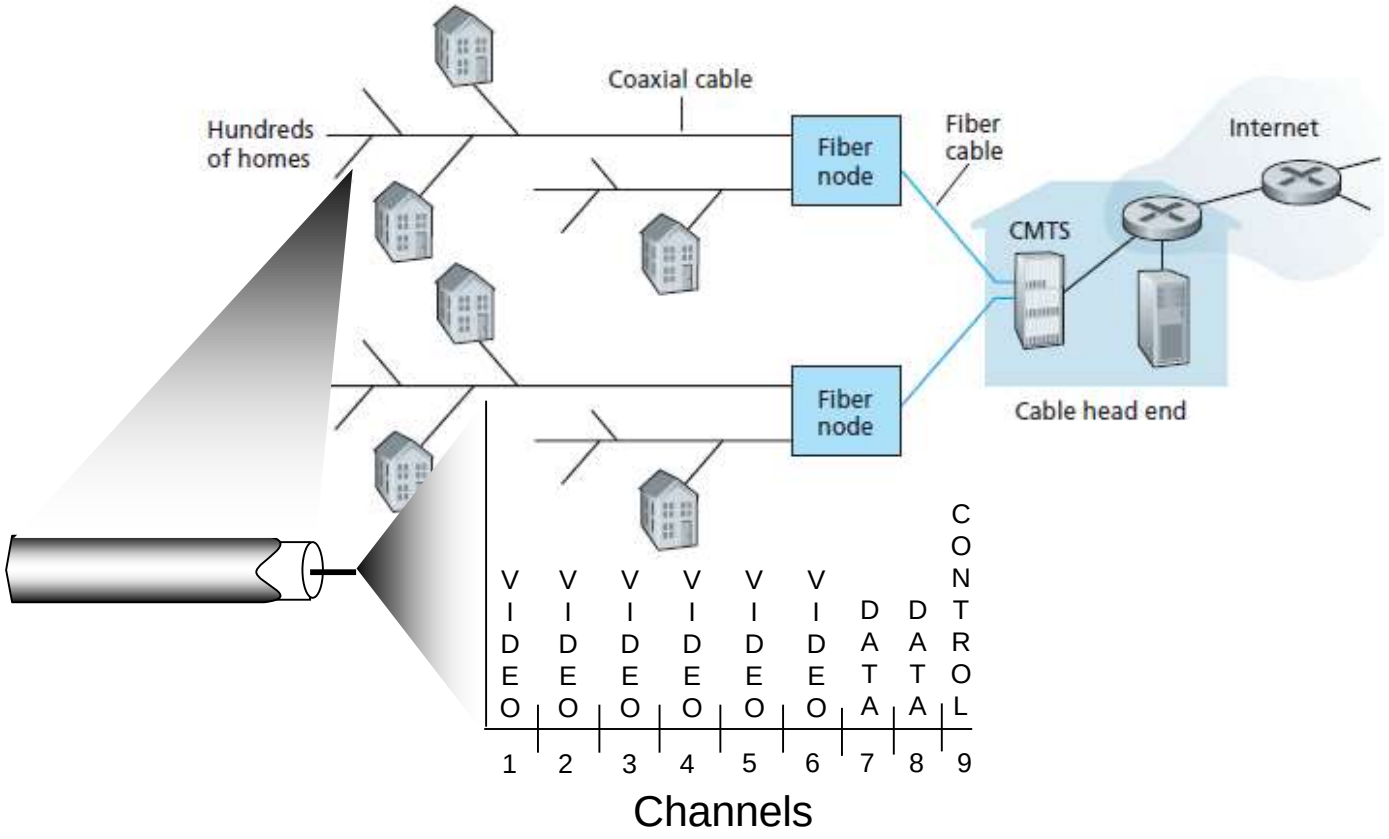

Coaxial cable
Hundreds of homes
Fiber node
Fiber cable
Internet
CMTS
Hundreds of homes
Fiber node
Cable head end

Access network: cable network

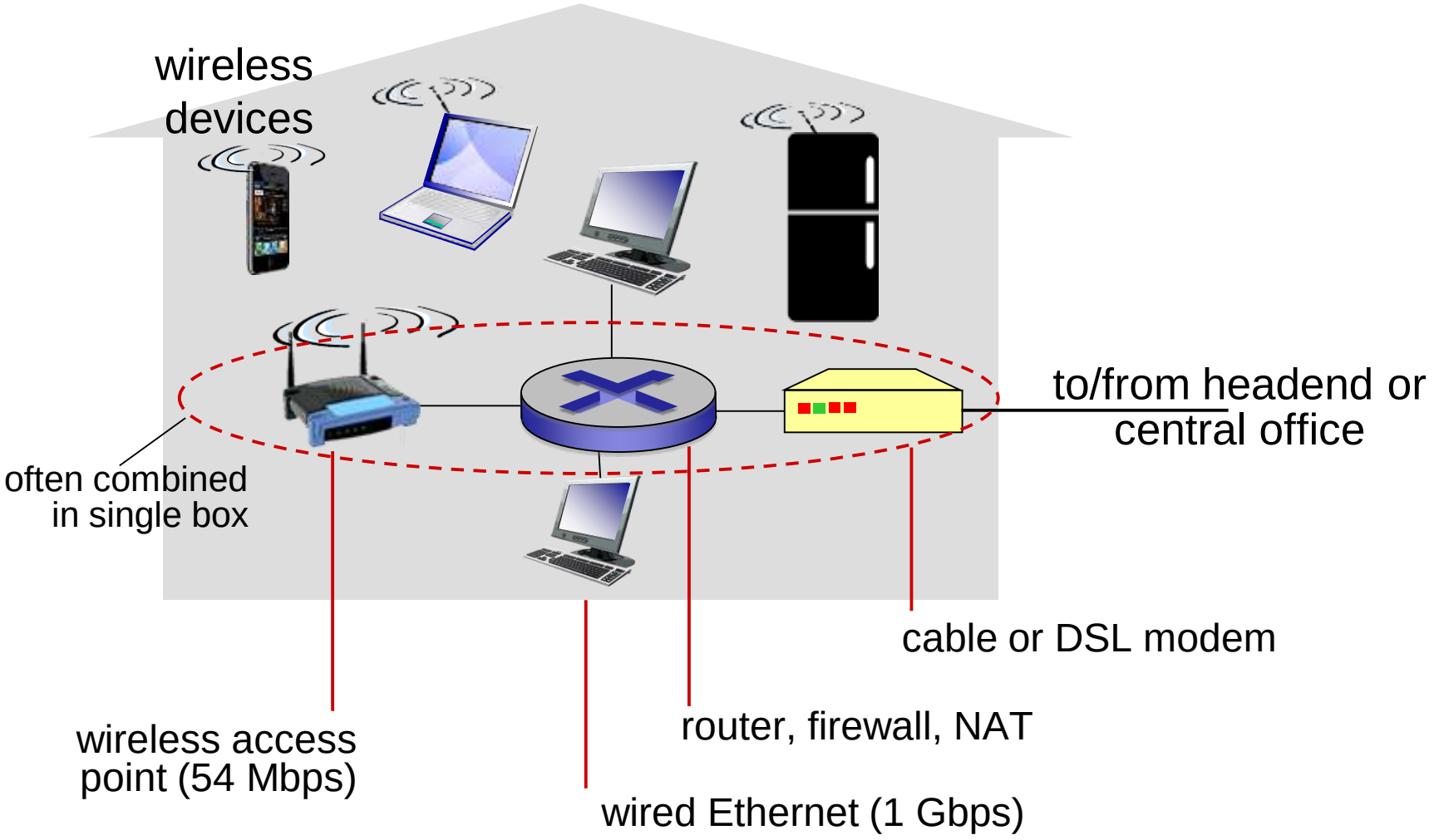


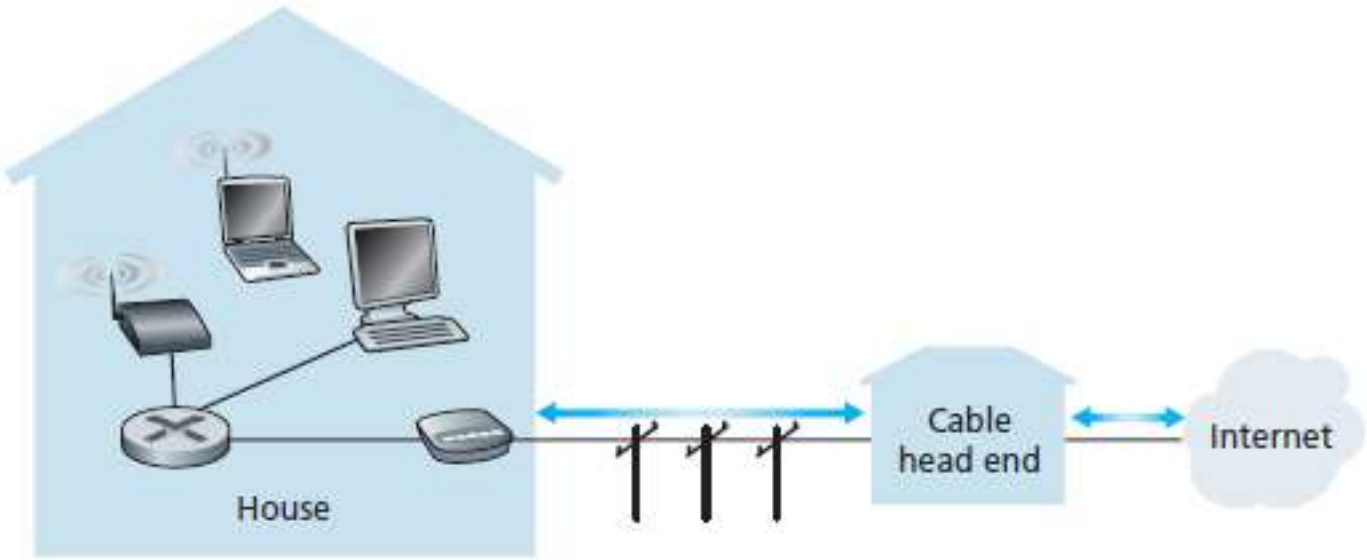
This is where the splitter is useful – to split between voice (phone) and data (computer)

frequency division multiplexing: different channels transmitted in different frequency bands

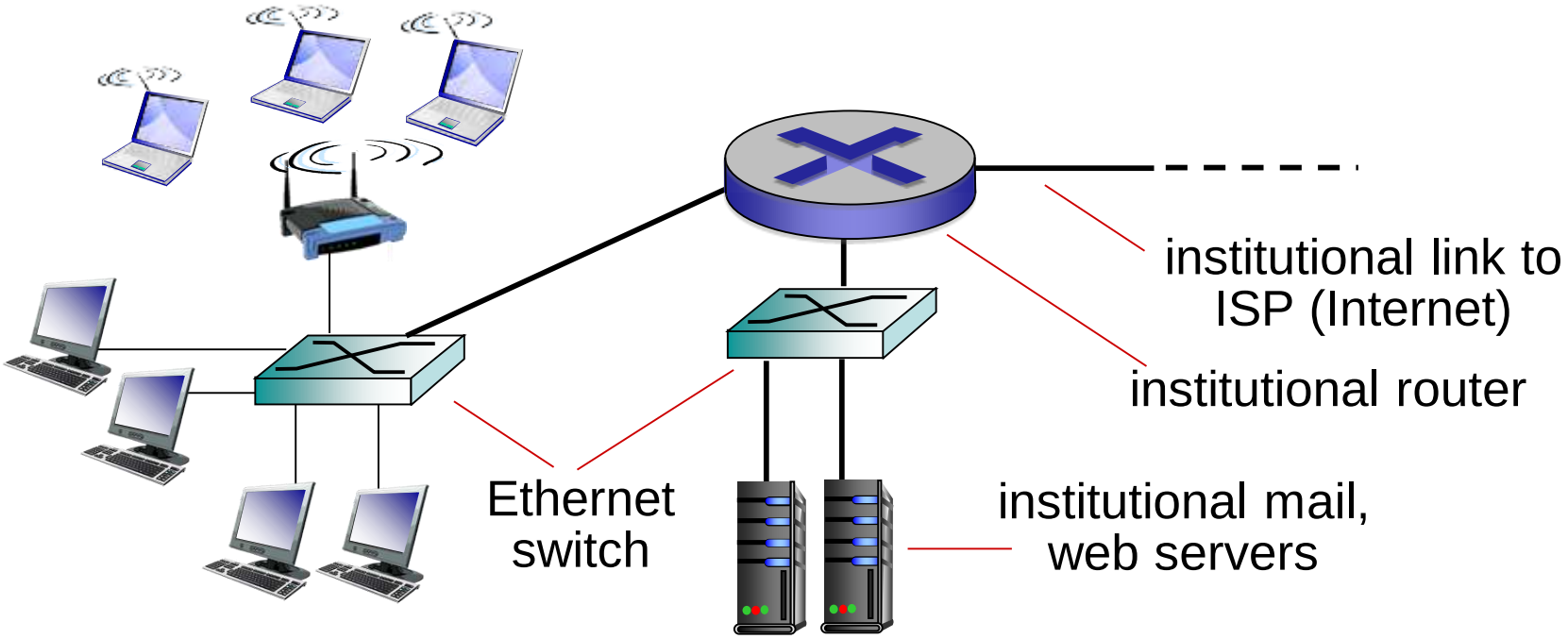

Hundreds of homes
Coaxial cable
Fiber node
Fiber node
Fiber cable
Internet
CMTS
Cable head end
VIDEO
VIDEO
VIDEO
VIDEO
VIDEO
VIDEO
DATA
DATA
CONTROL
1
2
3
4
5
6
7
8
9
Channels

Access network: home network

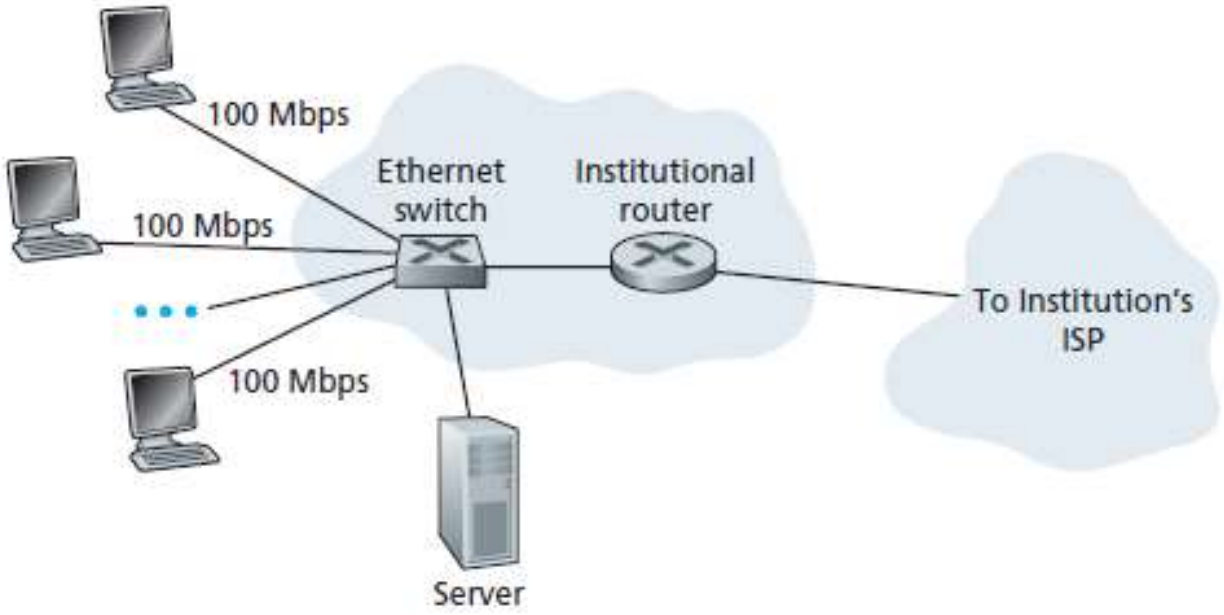



House
Cable
head end
Internet

Enterprise access networks (Ethernet)



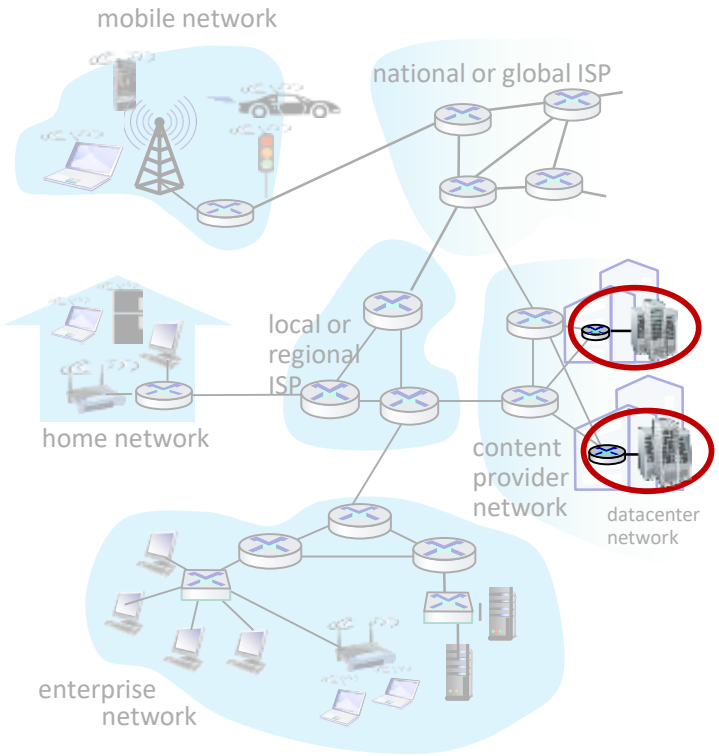
- typically used in companies, universities, etc.
- 10 Mbps, 100Mbps, 1Gbps, 10Gbps transmission rates
- today, end systems typically connect into Ethernet switch


100 Mbps
100 Mbps
100 Mbps
Ethernet switch
Institutional router
To Institution's ISP
Server

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)



Wireless access networks

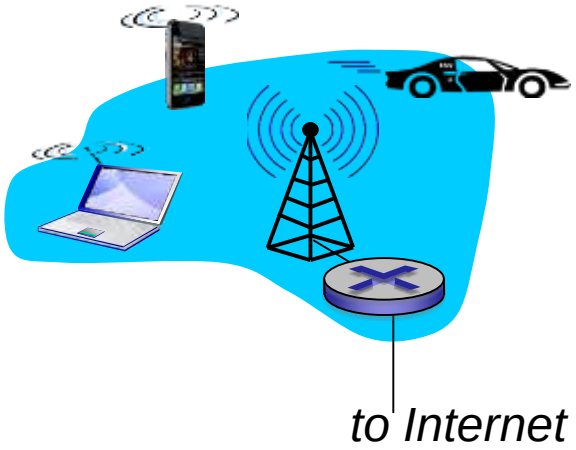
- shared *wireless* access network connects end system to router
  - via base station aka "access point"

wireless LANs:

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

wide-area wireless access

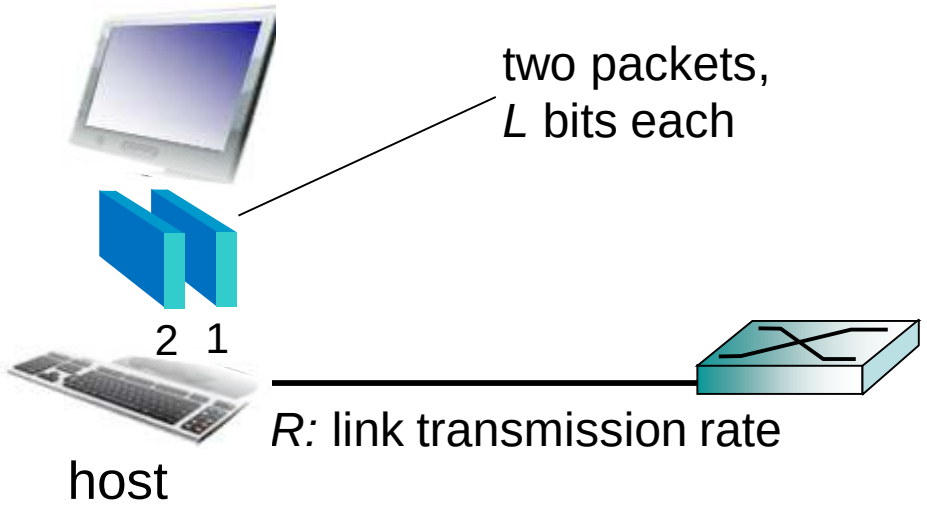
- provided by telco (cellular) operator, 10's km
- between 1 and 10 Mbps
- 3G, 4G: LTE



Host: sends *packets* of data

host sending function:

- takes application message
- breaks into smaller chunks, known as *packets*, of length *L* bits (e.g. 1KB=1024 bit)
- transmits packet into access network at *transmission rate R*
  - link transmission rate, aka link *capacity, aka link bandwidth*

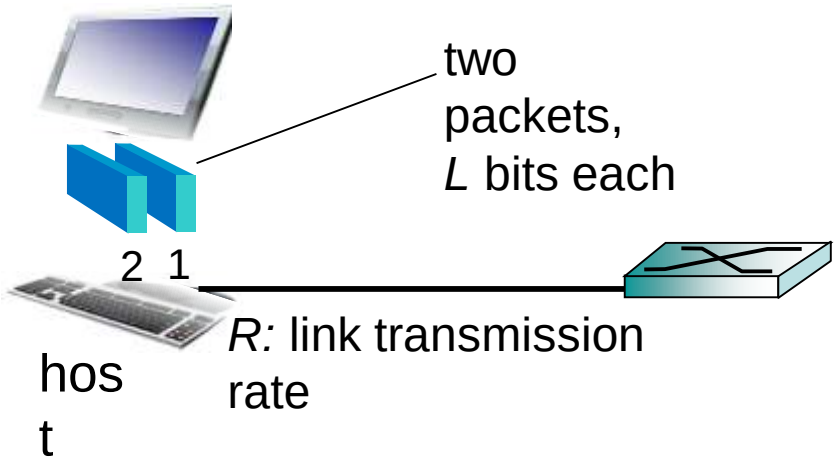


$$\text{packet transmission delay} = \text{time needed to transmit } L\text{-bit packet into link} = \frac{L \text{ (bits)}}{R \text{ (bits/sec)}}$$

Host: sends *packets* of data

packet transmission delay = time needed to transmit *L*-bit packet into link

$$= \frac{L \text{ (bits)}}{R \text{ (bits/sec)}}$$

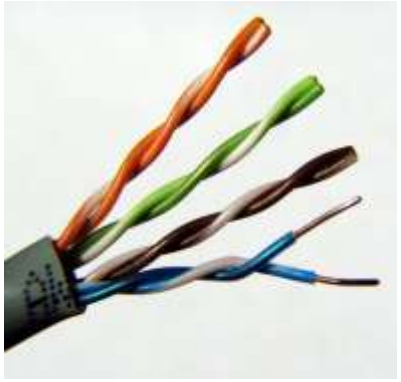


Example: L=30Mbit, R=10Mbps

packet transmission delay
= L/R
= 30Mbit/10Mbps
= 3s

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



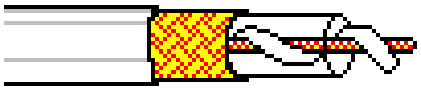
twisted pair (TP)

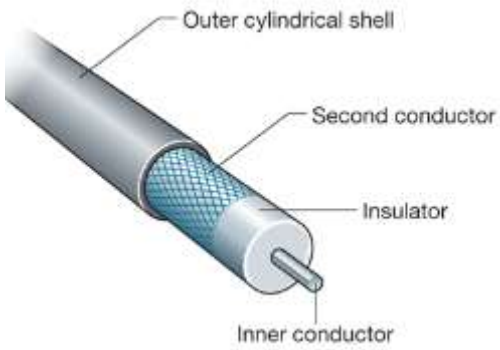
- two insulated copper wires
  - Category 5: 100 Mbps, 1 Gbps Ethernet
  - Category 6: 10Gbps

Physical media: coax, fiber

coaxial cable:

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





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:
  - repeaters spaced far apart
  - immune to electromagnetic noise

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
  - e.g. up to 45 Mbps channels
- LAN (e.g., WiFi)
  - 54 Mbps
- wide-area (e.g., cellular)
  - 4G cellular: ~ 10 Mbps
- satellite
  - Kbps to 45Mbps channel (or multiple smaller channels)
  - 270 msec end-end delay
  - geosynchronous versus low altitude

Chapter 1: roadmap

1.1 what *is* the Internet?

1.2 network edge

- end systems, access networks, links

1.3 network core

- packet switching, circuit switching, network structure

1.4 delay, loss, throughput in networks

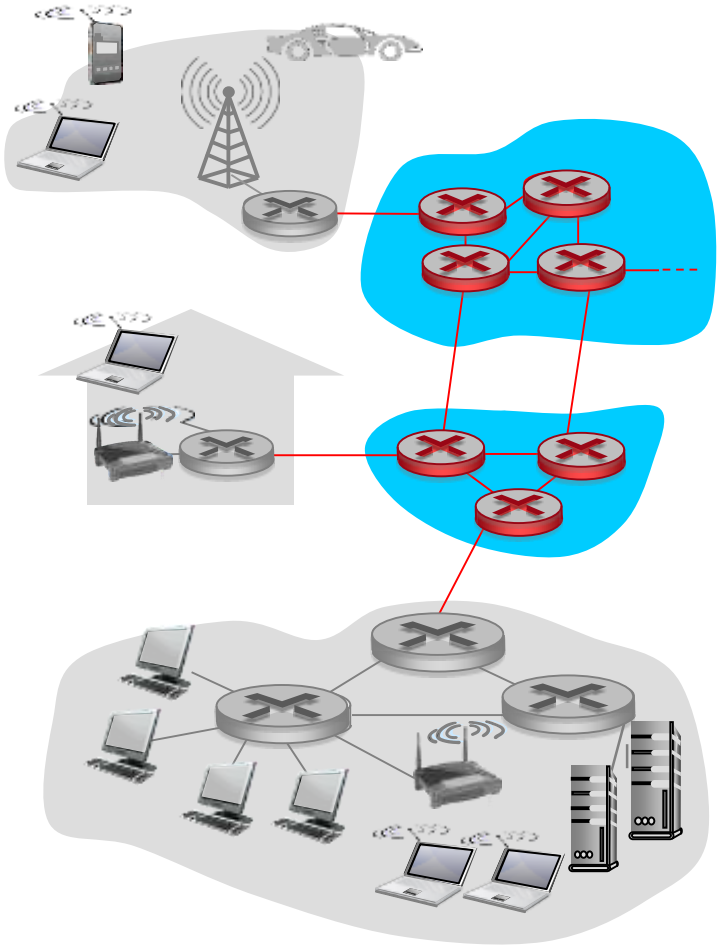
1.5 protocol layers, service models

1.6 networks under attack: security

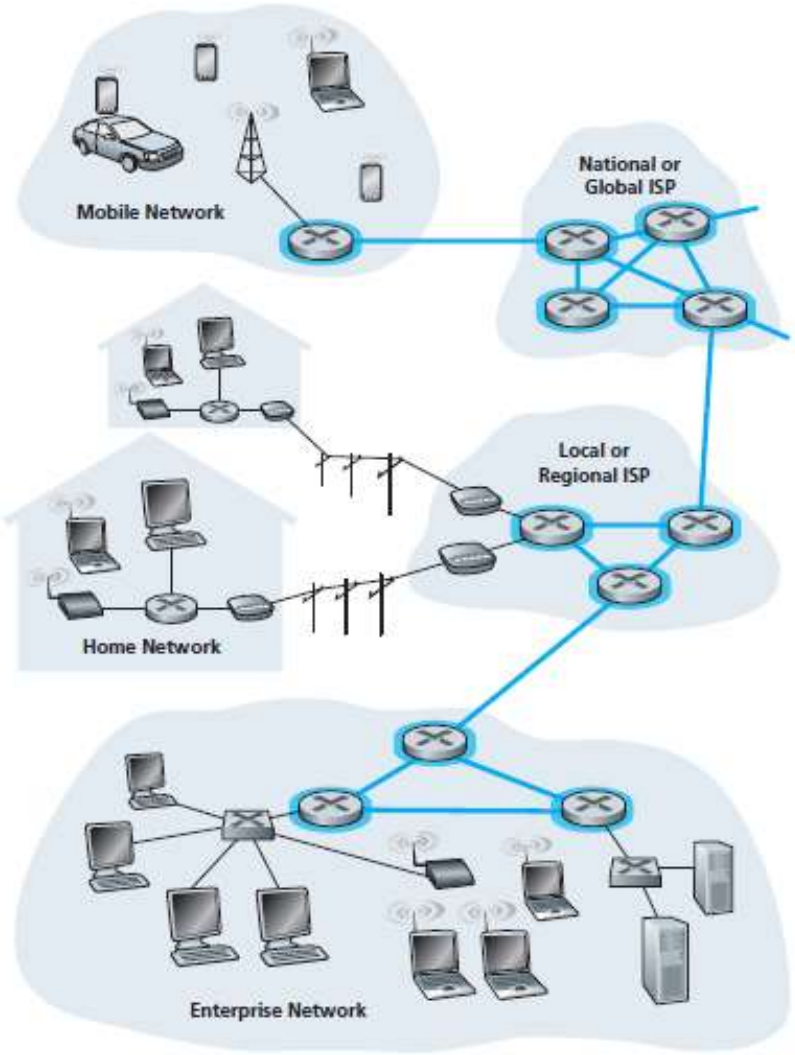
1.7 history

The network core

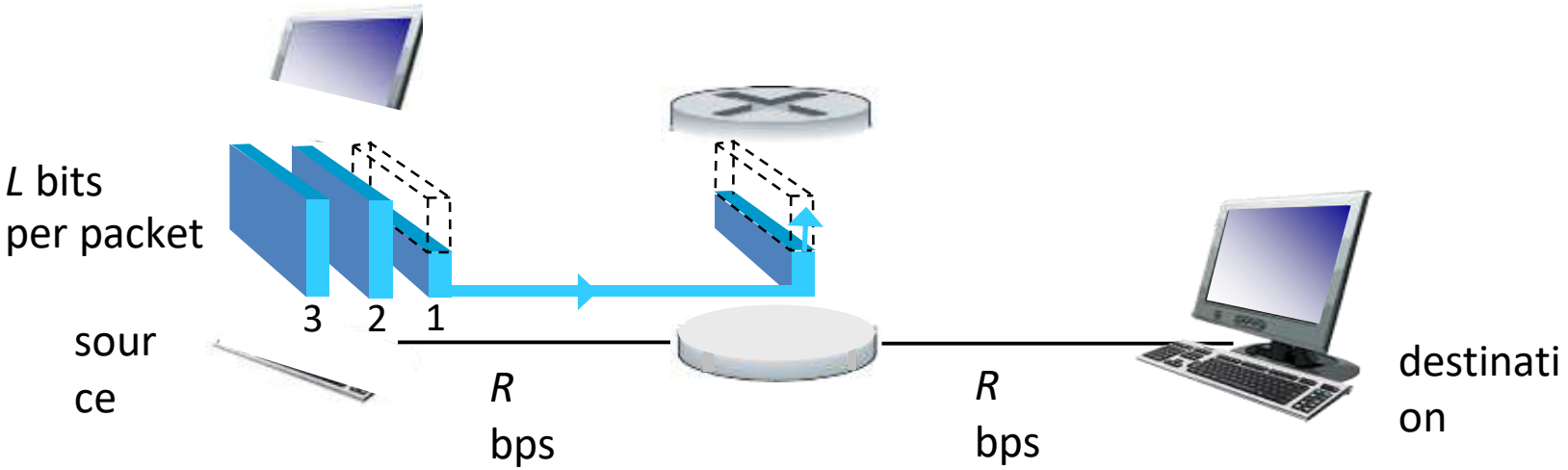
- mesh of interconnected routers
- packet-switching: hosts break application-layer messages into *packets*
  - forward packets from one router to the next, across links on path from source to destination
  - each packet transmitted at full link capacity



The network core



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

one-hop numerical example:

- L = 7.5 Mbits
- R = 1.5 Mbps
- one-hop transmission delay = 5 sec

more on delay shortly …

one-hop numerical example:

- *L* = 7.5 Mbits
- *R* = 1.5 Mbps
- one-hop transmission delay = L/R = 5 sec

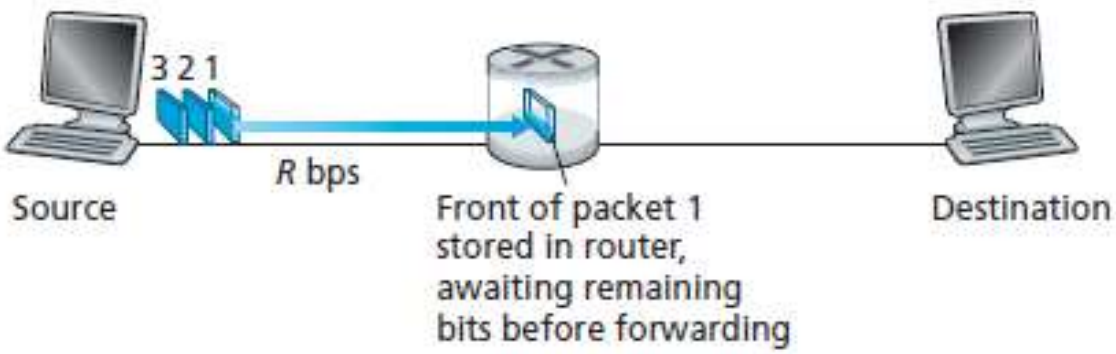
One-hop numerical example:

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

transmitting some of packet 1

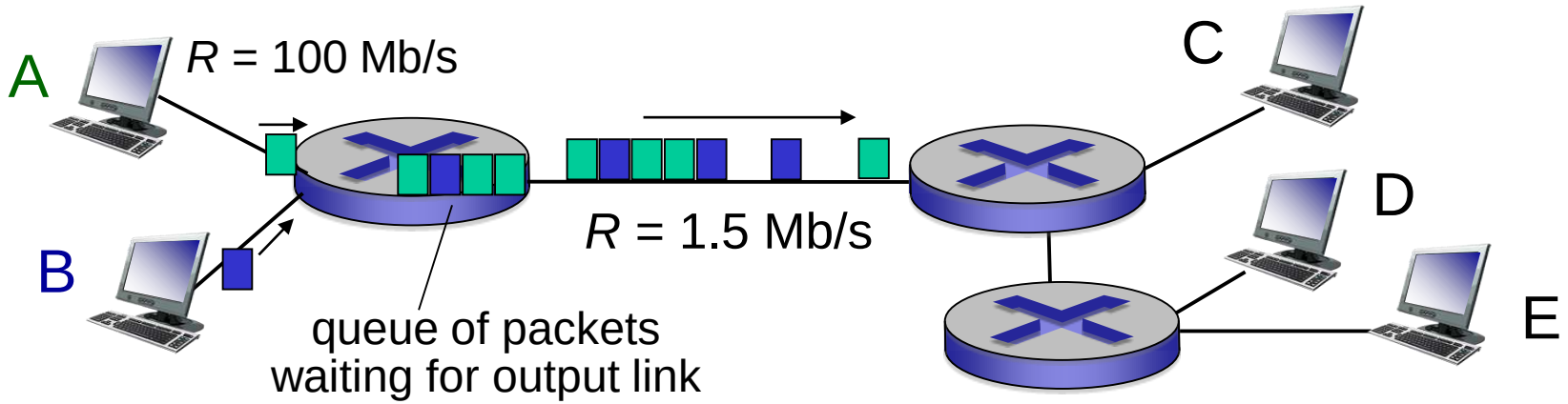
the source has three packets, each consisting of *L bits, to send to the destination.*

When the router has received ***all*** *of the packet's bits can it begin to transmit the packet onto the* outbound link.



The router employs store-and-forwarding so must first buffer ("store") the packet's bits

Packet Switching: queueing delay, loss



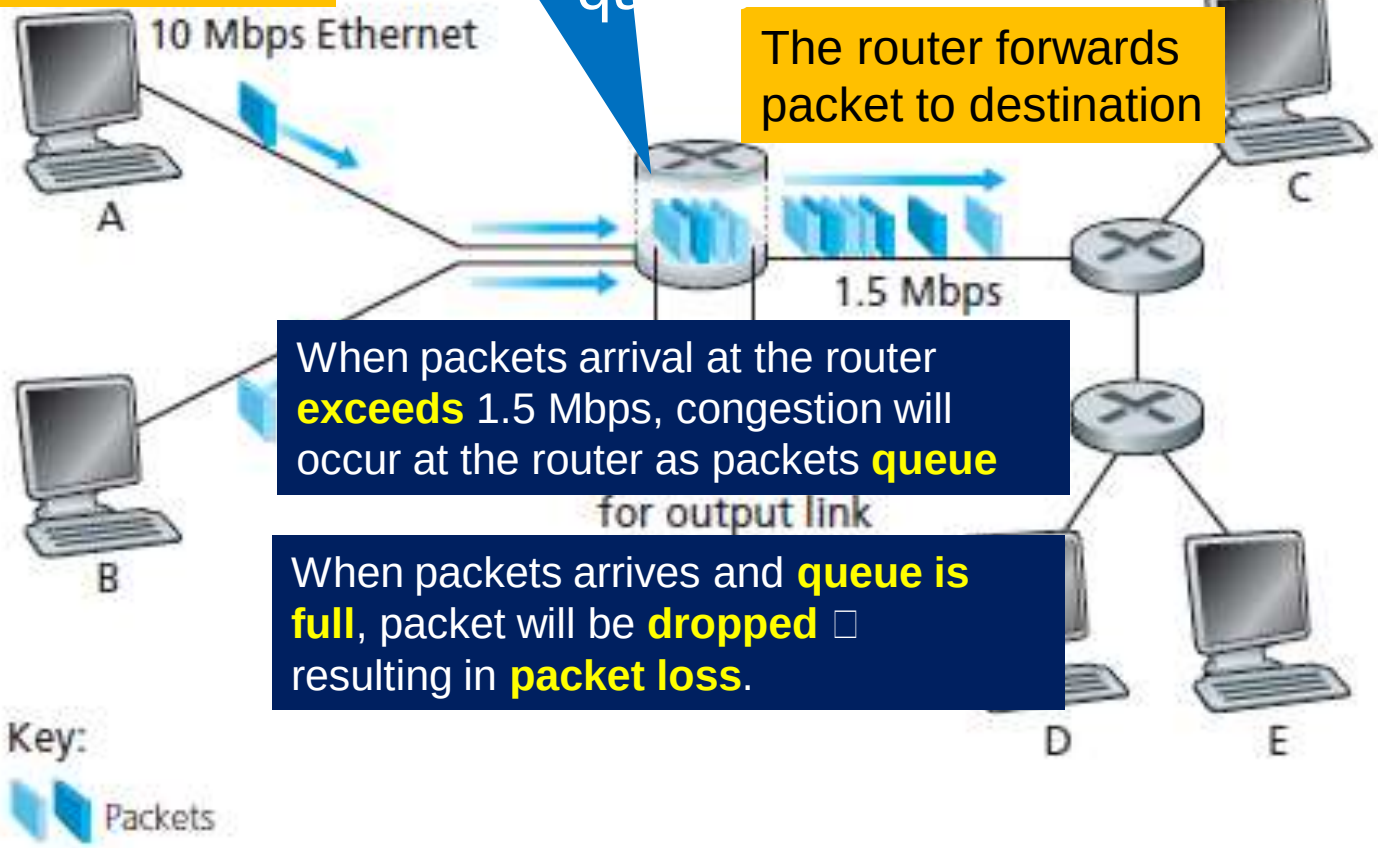
queuing and loss:

- 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

Packet Switching: Queuing Delays and Packet Loss

A and B sends packets to the router

/output



The router forwards packet to destination

When packets arrival at the router **exceeds** 1.5 Mbps, congestion will occur at the router as packets **queue**

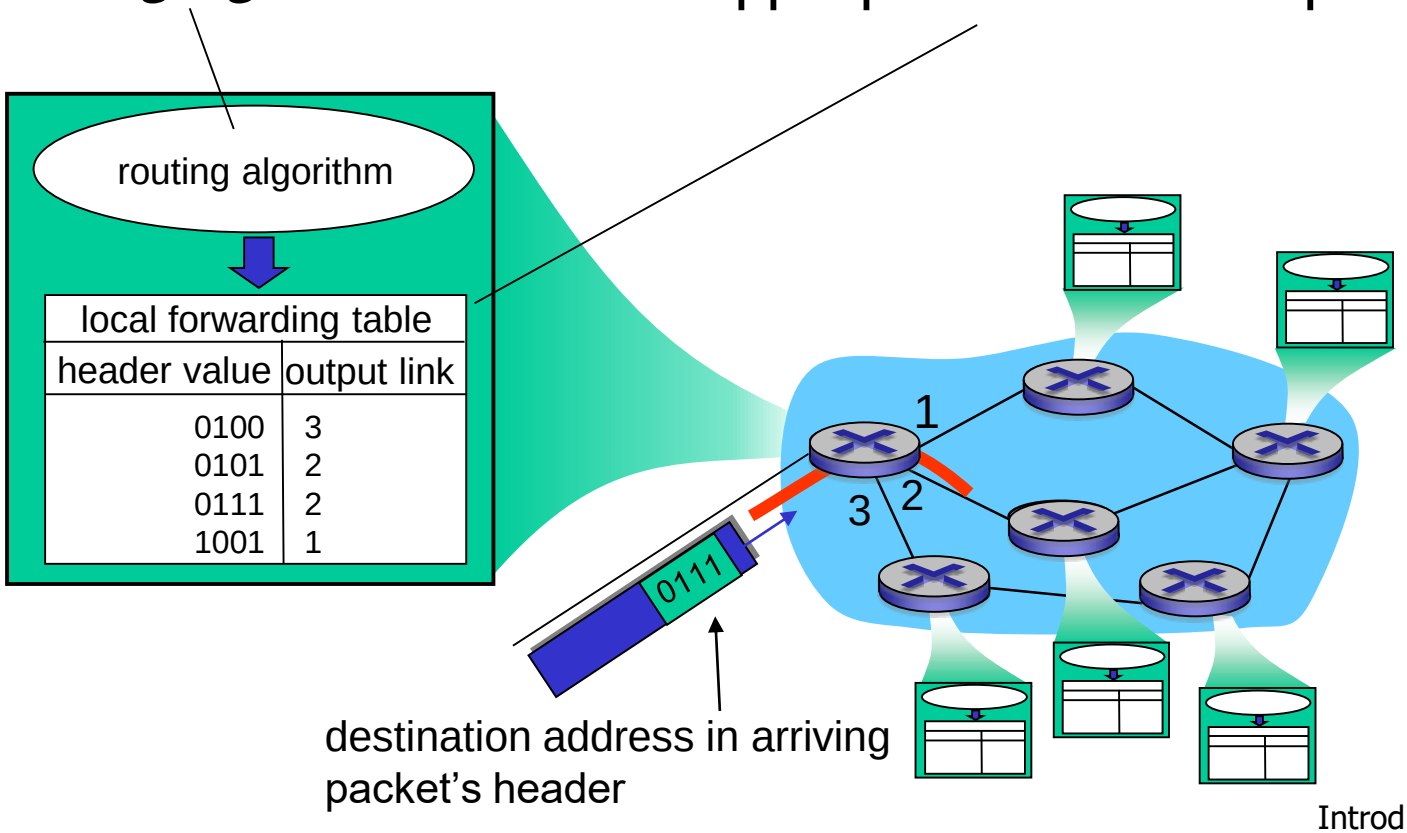
When packets arrives and **queue is full**, packet will be **dropped** □ resulting in **packet loss**.

Two key network-core functions

routing: determines source-destination route taken by packets

- *routing algorithms*

forwarding: move packets from router's input to appropriate router output



Two key network-core functions

ROUTING

- determines source to destination route taken by packets
  - routing algorithms
- Global action

FORWARDING

- move packets from router input interface to appropriate router output interface
- Local Action

1 Router R1 receives a packet

2 R1 checks the destination address

3 R1 finds the best path to destination

4 R1 forwards the packet (or drops it)

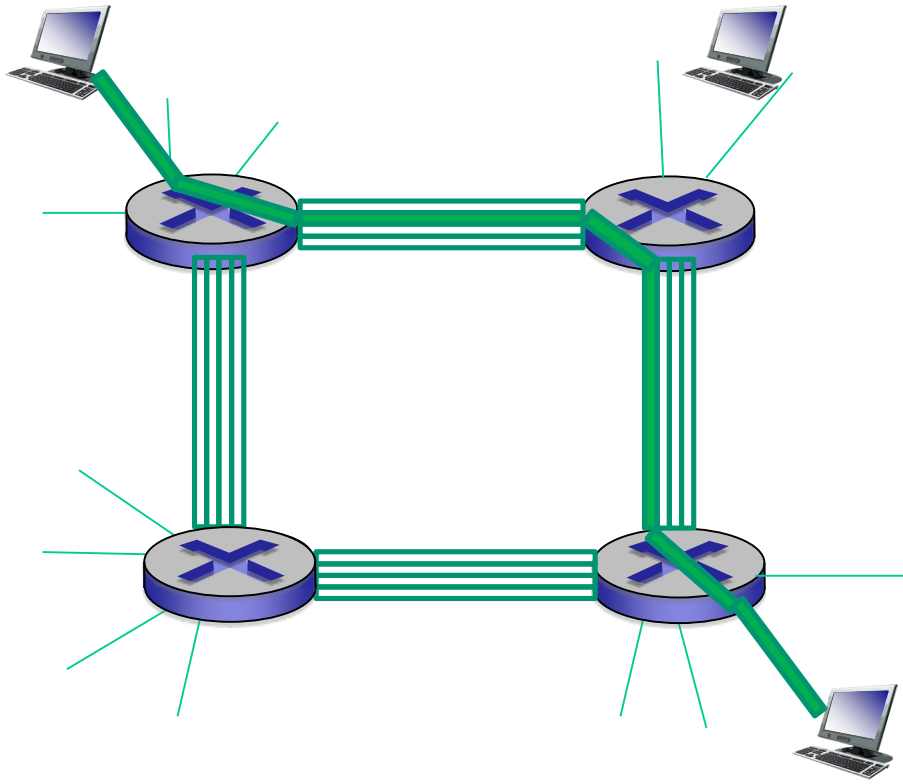
Packet destination: 192.168.1.1

```
R1# show ip route
     172.16.0.0/24 is subnetted, 3 subnets
C       172.16.1.0 is directly connected, FastEthernet0/0
C       172.16.2.0 is directly connected, Serial0/0/0
R       172.16.3.0 [120/1] via 172.16.2.2, 00:00:25, Serial0/0/0
R    192.168.1.0/24 [120/1] via 172.16.2.2, 00:00:25, Serial0/0/0
```

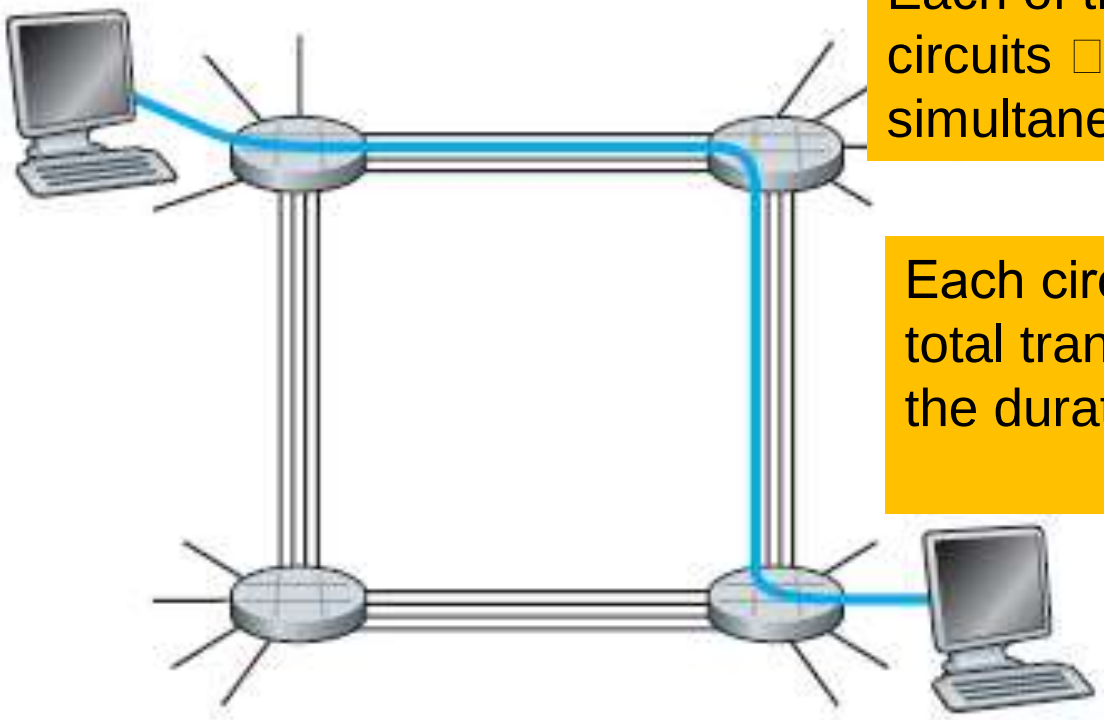
Alternative core: circuit switching

end-end resources allocated to, reserved for "call" between source & dest:

- 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



The 4 circuit switches are interconnected by 4 links.



Each of these links has four circuits can support 4 simultaneous connections.

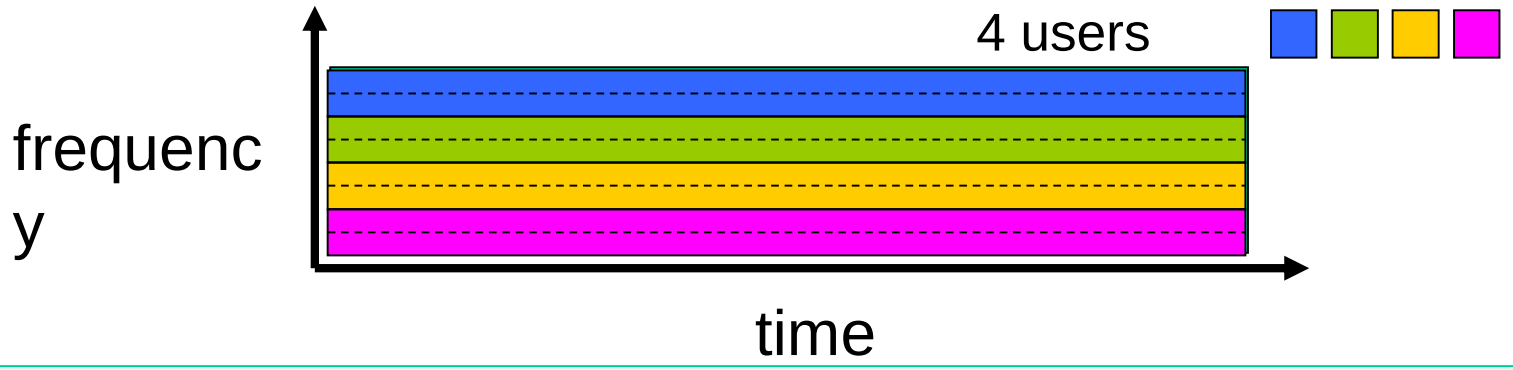
Each circuit ¼ of the link's total transmission capacity for the duration of the connection.

Example:
If a link has a transmission rate of **1 Mbps**
 Each connection gets **250 kbps** of dedicated transmission rate.

Circuit switching: FDM versus TDM

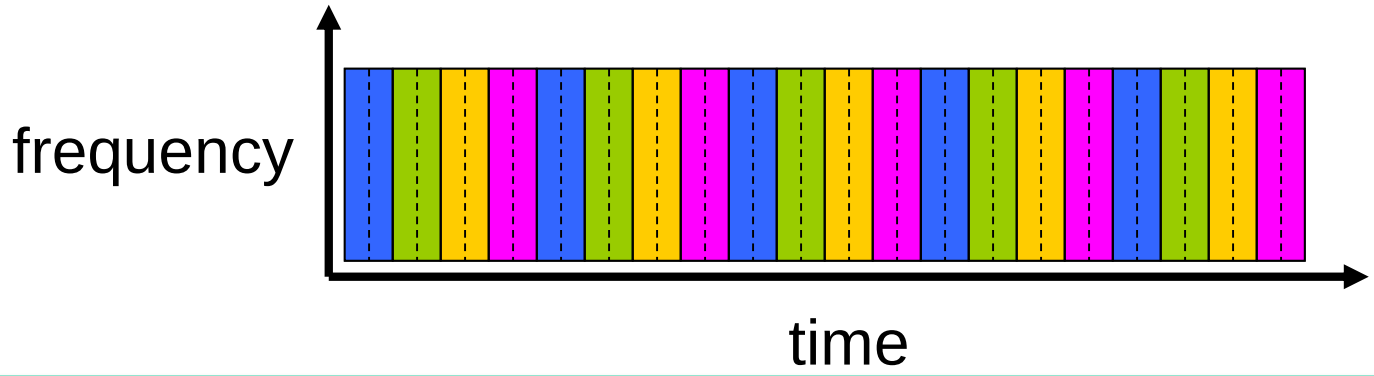
FDM

Example:

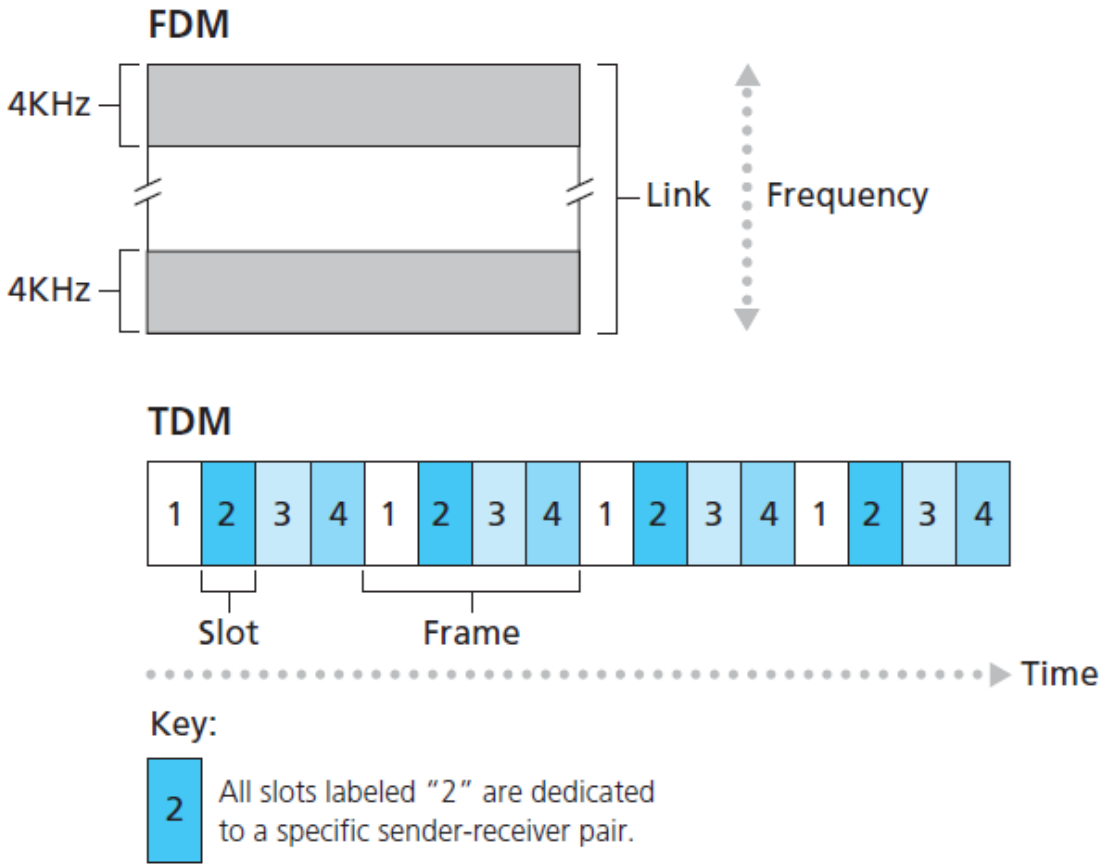


Users will access media (e.g. cable) based on DIFFERENT frequency (Hz) allocation

TDM



Users will access media (e.g. cable) based on DIFFERENT time slot (second) allocation



With FDM, each circuit continuously gets a fraction of the bandwidth.

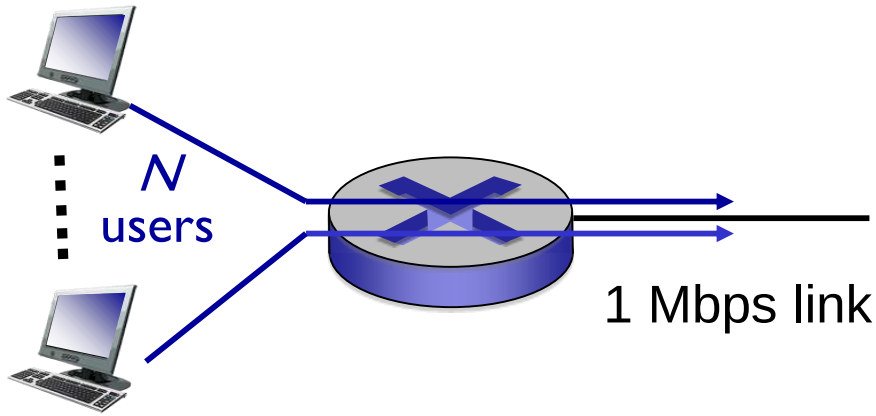
With TDM, each circuit gets all of the bandwidth periodically during brief intervals of time (that is, during slots)

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 > 10 active at same time is less than .0004 *

Q: how did we get value 0.0004?

Q: what happens if > 35 users ?

Answer: *Since we can safely assume that each user is active independently from the others, we can model our problem by a binomial distribution* $X \sim B(N,p)$ *where* $N = 35$ *is the total number of users and* $p = 0.1$ *is the probability of being active. The probability* $P(X = k)$ *of having* k *active users simultaneously is defined as* $\binom{N}{k} \cdot p^k \cdot (1-p)^{N-k}$ *where* $\binom{N}{k} = \frac{N!}{k! \cdot (N-k)!}$. *Thus the probability of having more than 10 active users is* $\sum_{i=11}^{35} P(X=i) = 1 - \sum_{i=0}^{10} P(X=i) = 1 - \sum_{i=0}^{10} \frac{35!}{i! \cdot (35-i)!} \cdot p^i \cdot (1-p)^{35-i} \approx 0.0004$

http://mosi.cs.uni-saarland.de/data/DataNetworks13/Homework1-solution.pdf

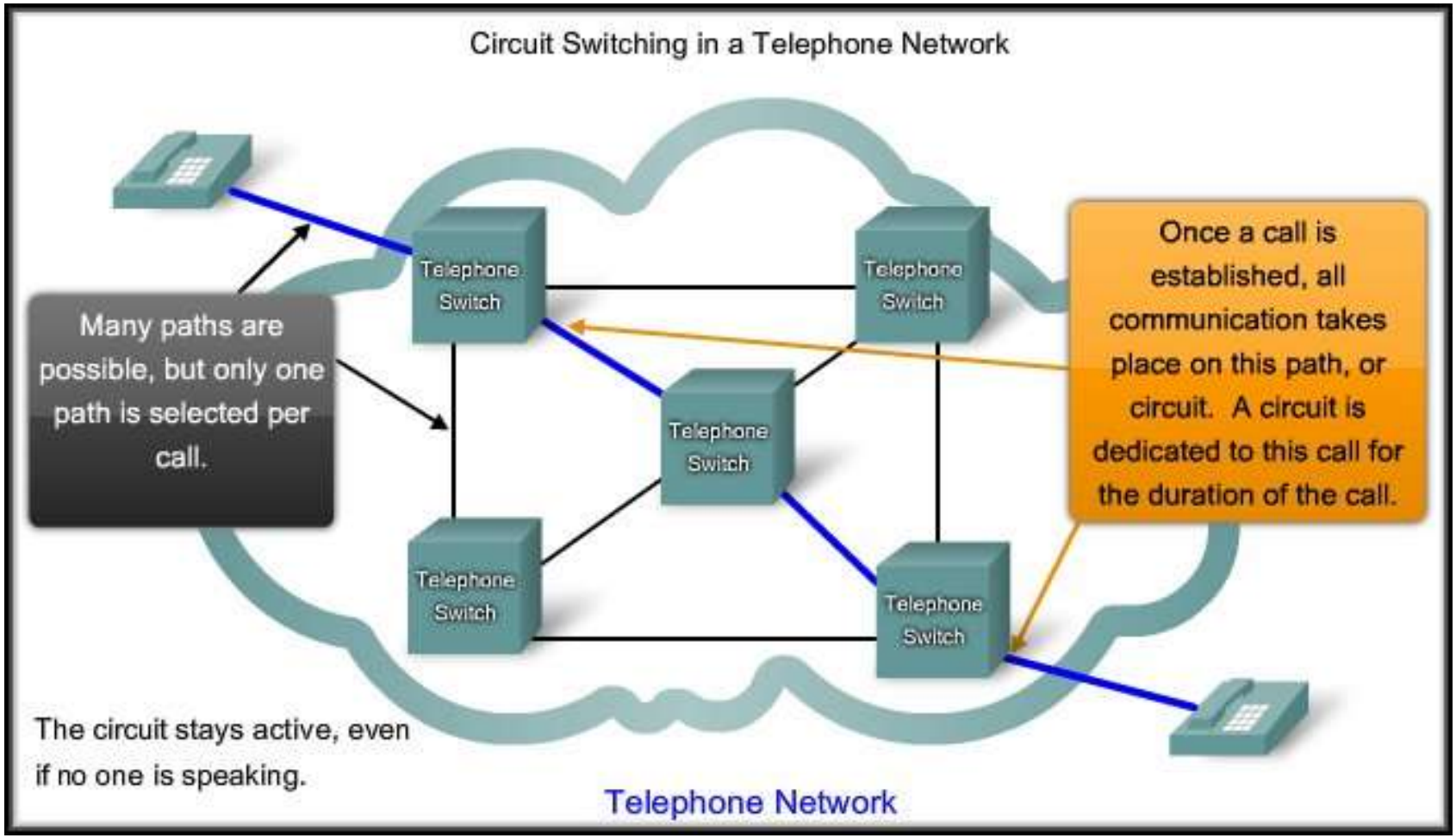
Packet switching versus circuit switching

is packet switching a "slam dunk winner?"

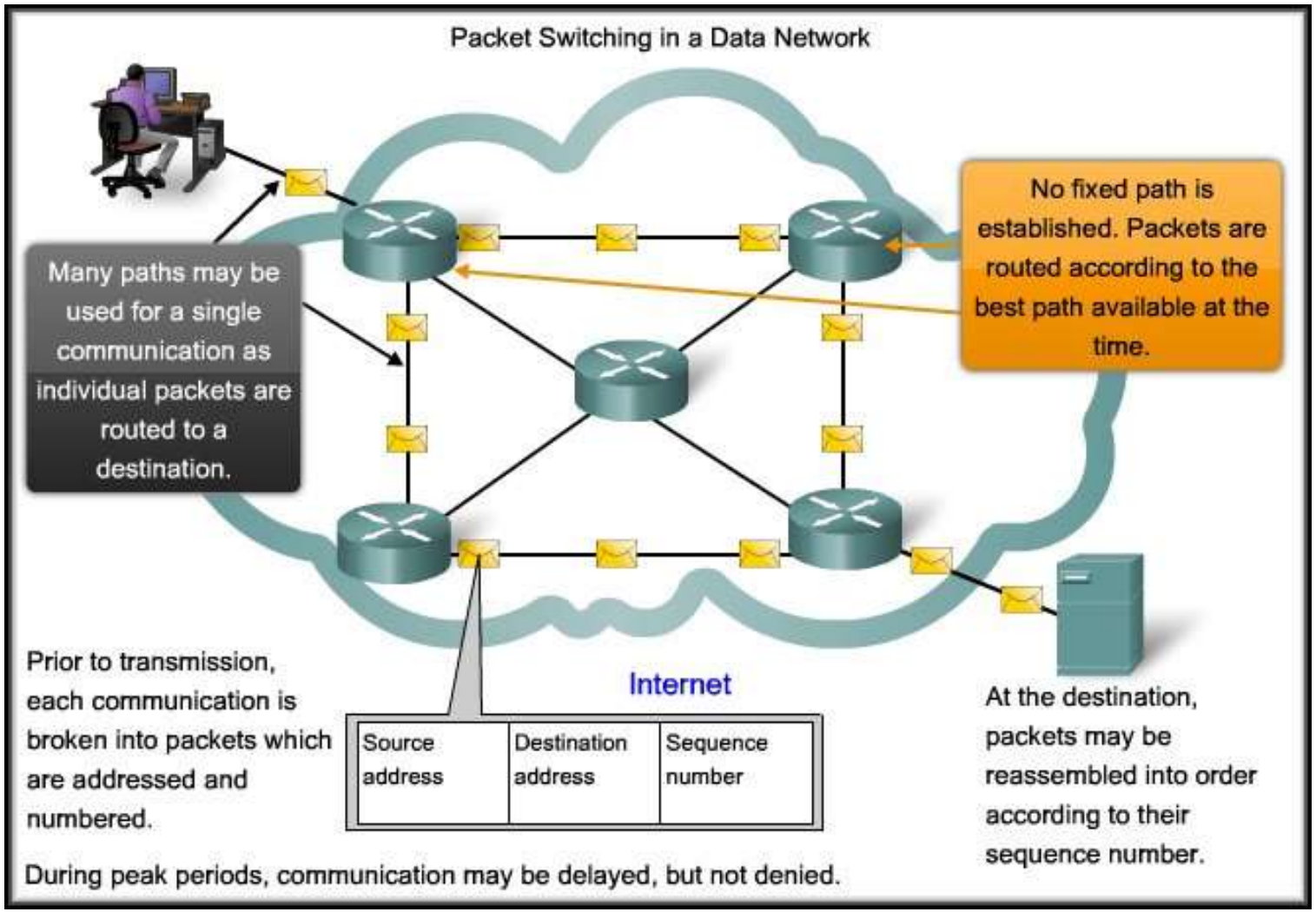
- great for bursty data
  - resource sharing
  - simpler, no call setup
- excessive congestion possible: packet delay and loss
  - protocols needed for reliable data transfer, congestion control
- *Q:* How to provide circuit-like behavior?
  - bandwidth guarantees needed for audio/video apps
  - still an unsolved problem (chapter 7)

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

Circuit Switched – Connection-Oriented Networks



Packet Switched – Connectionless Networks



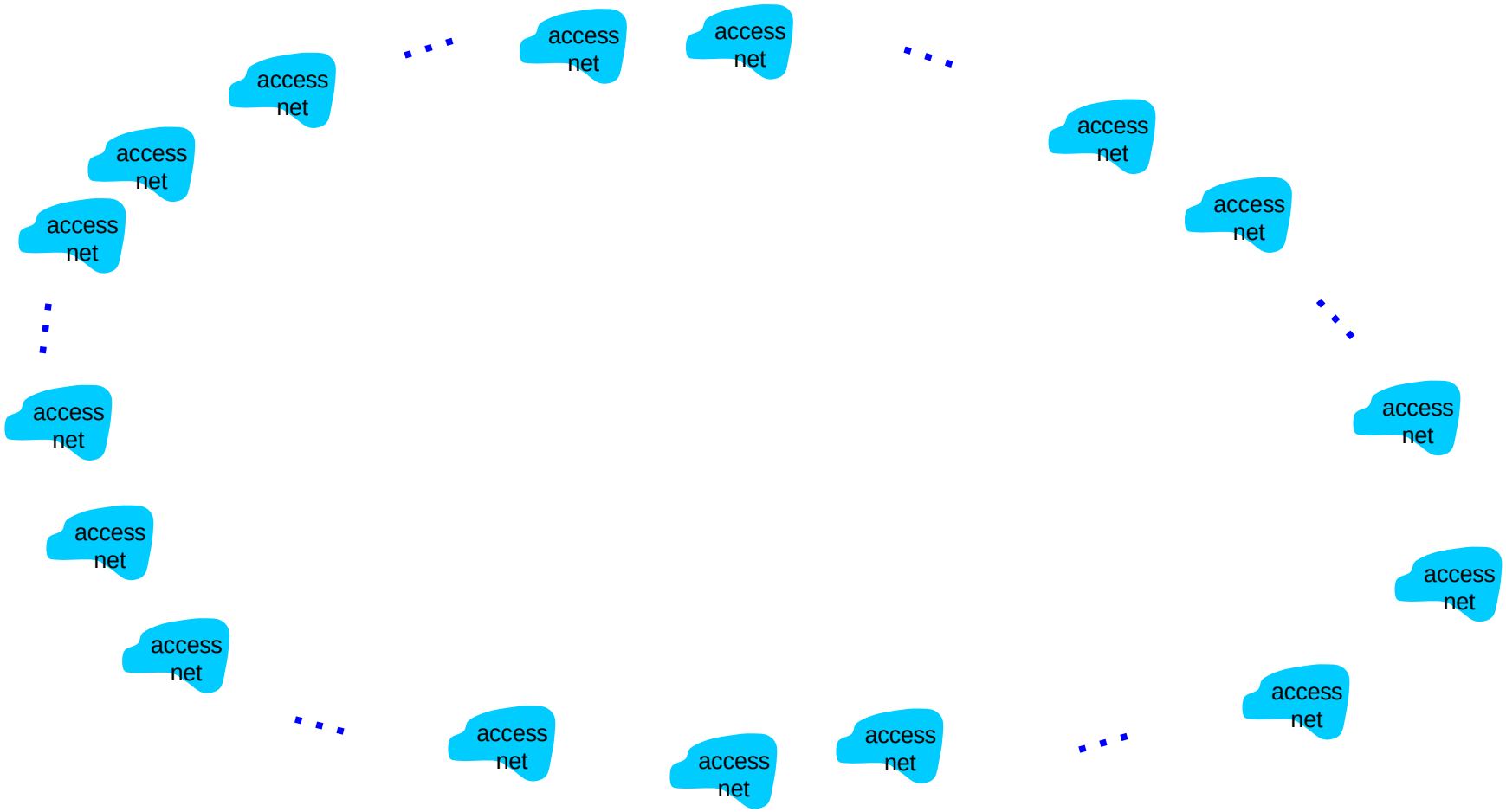
Circuit Switched	Packet Switched
Connection-oriented	Connectionless
Dedicated Circuit	Shared Circuit
Guaranteed level of service (Bandwidth, QoS)	Messages divided into packets
Inefficient use of Medium	Efficient use of Medium
Single path, no redundancy	**Fault Tolerant**, multiple possible paths

Internet structure: network of networks

- End systems connect to Internet via access ISPs (Internet Service Providers)
  - residential, company and 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
- Let's take a stepwise approach to describe current Internet structure

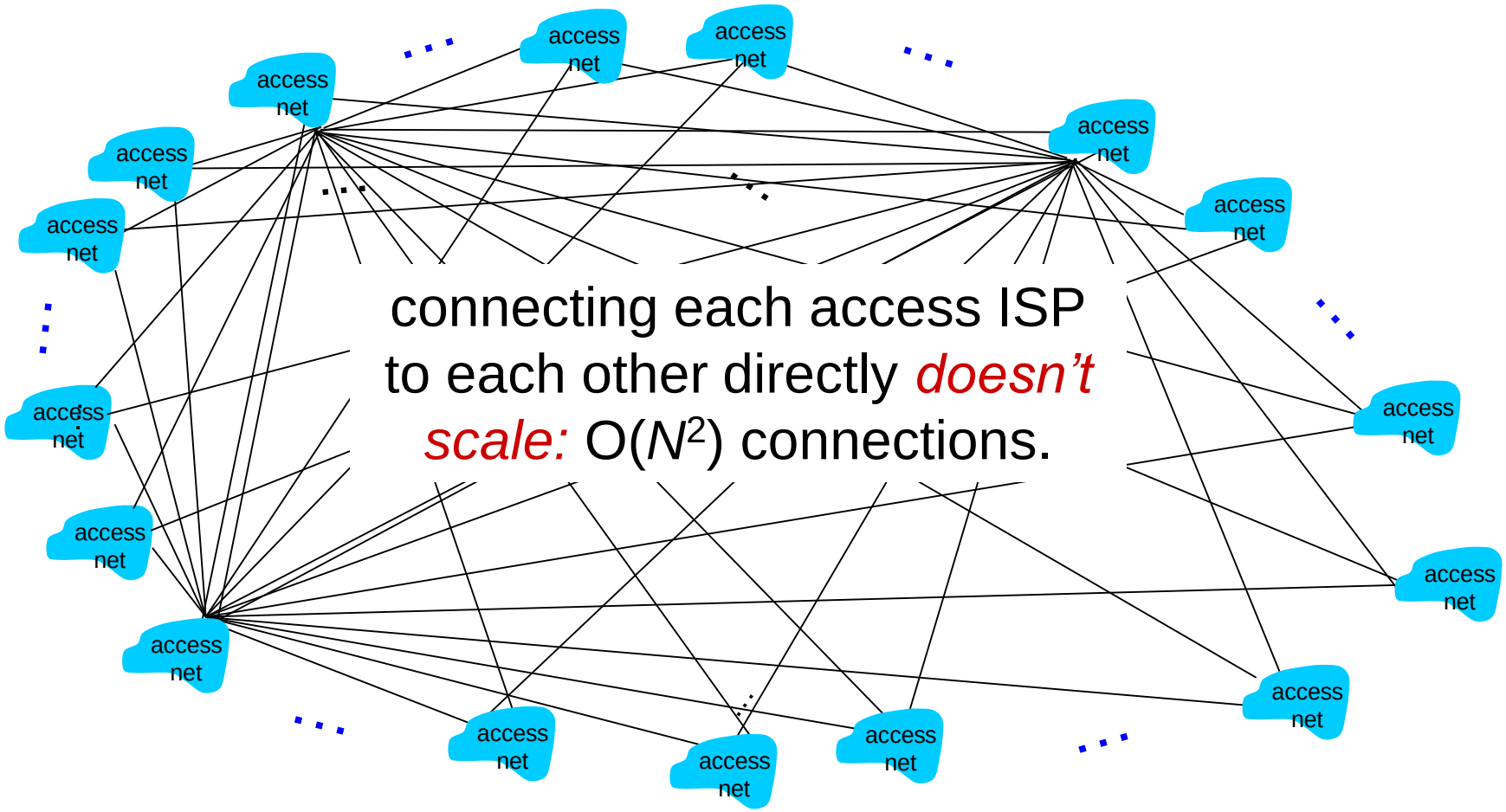
Internet structure: network of networks

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



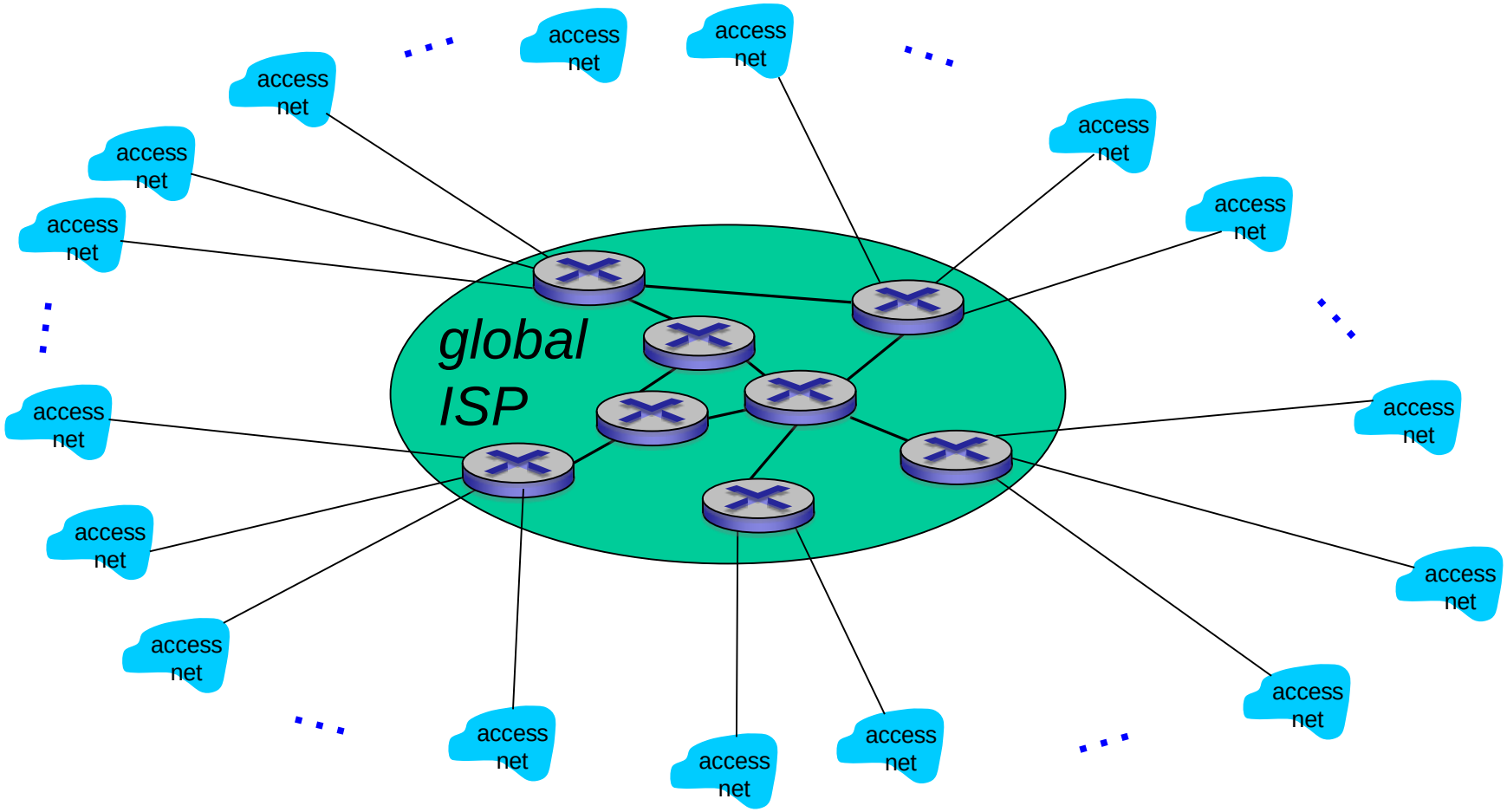
Internet structure: network of networks

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



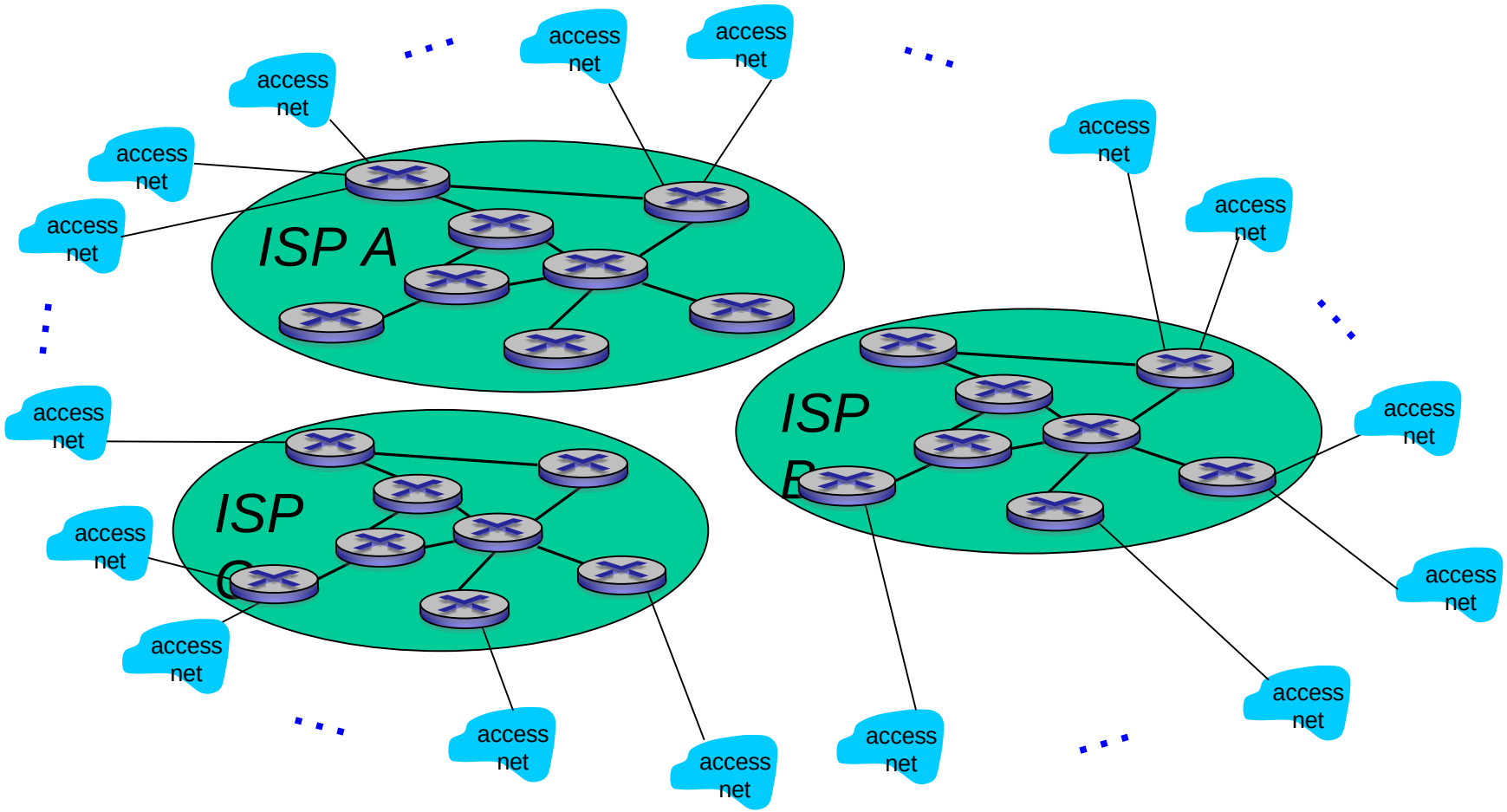
Internet structure: network of networks

Option: connect each access ISP to one global transit ISP?
Customer and provider ISPs have economic agreement.



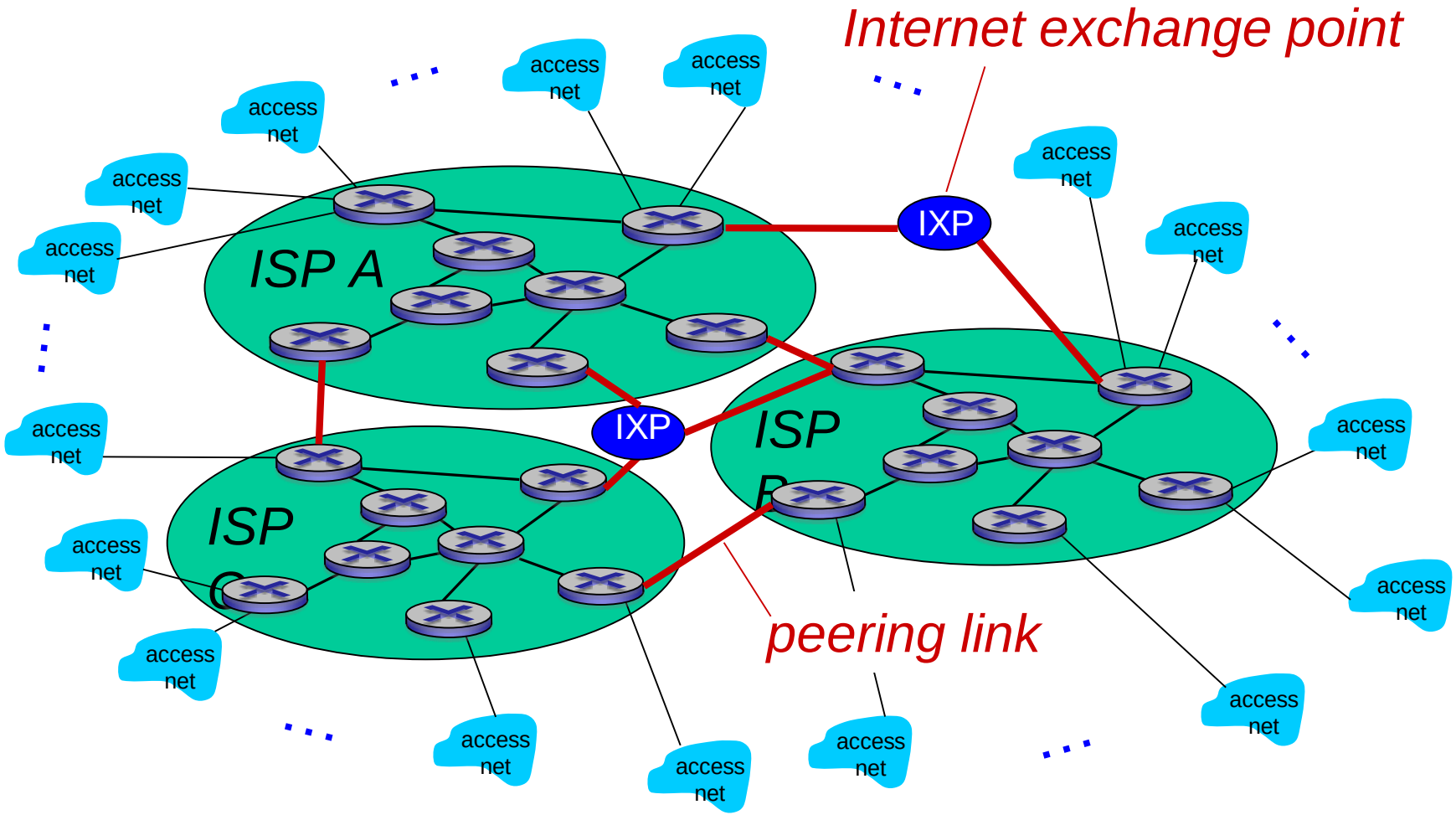
Internet structure: network of networks

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



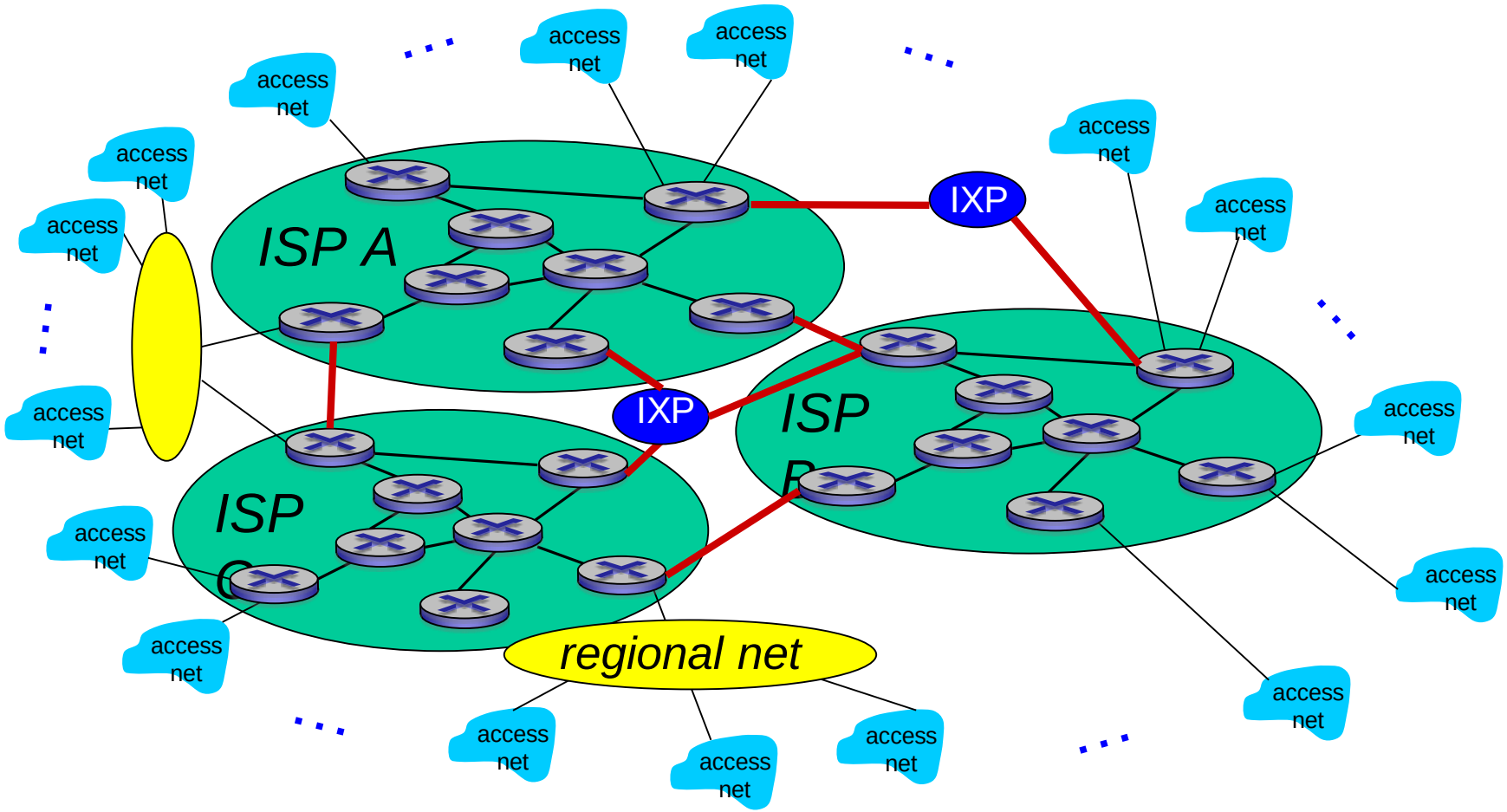
Internet structure: network of networks

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



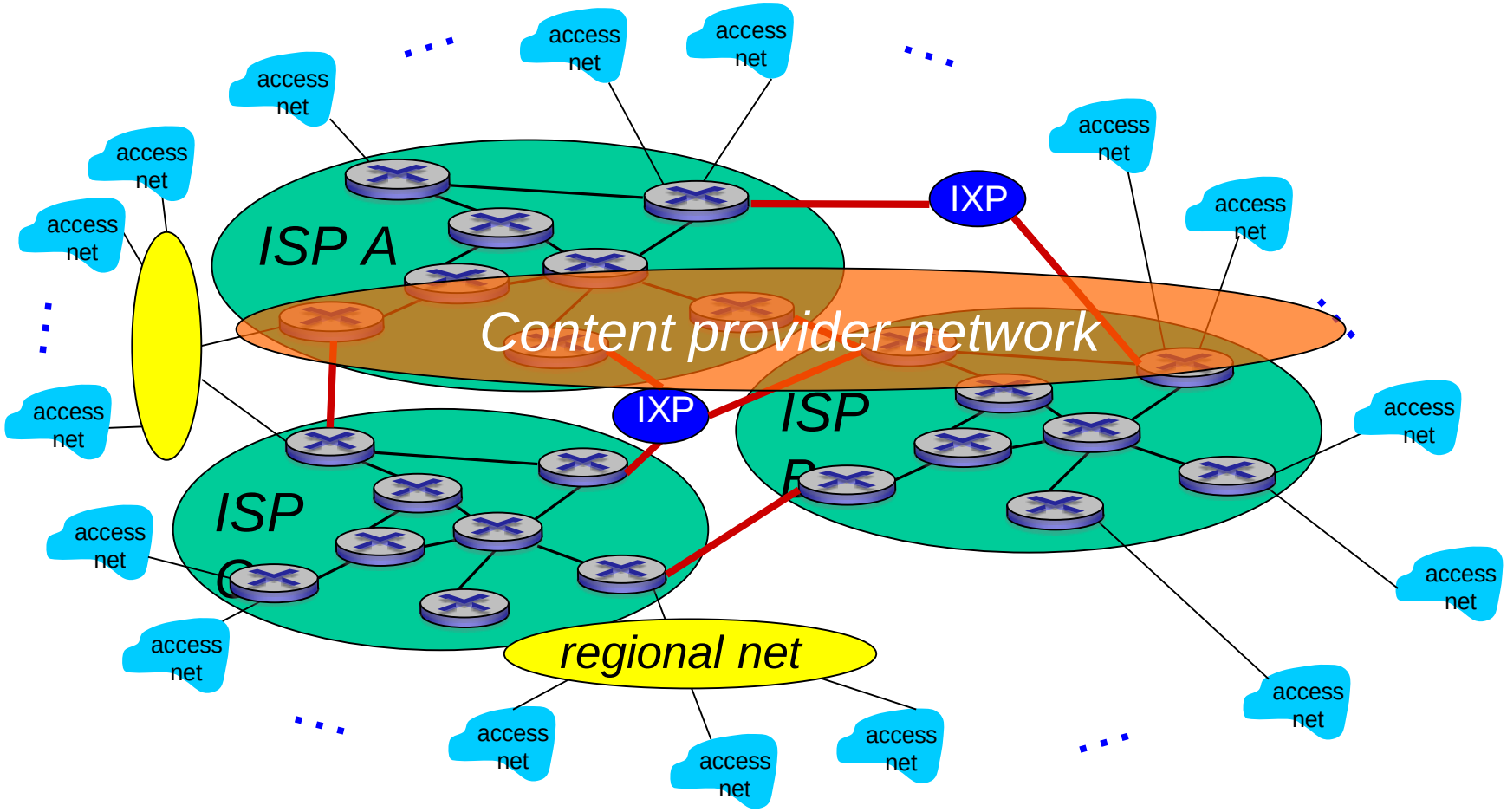
Internet structure: network of networks

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

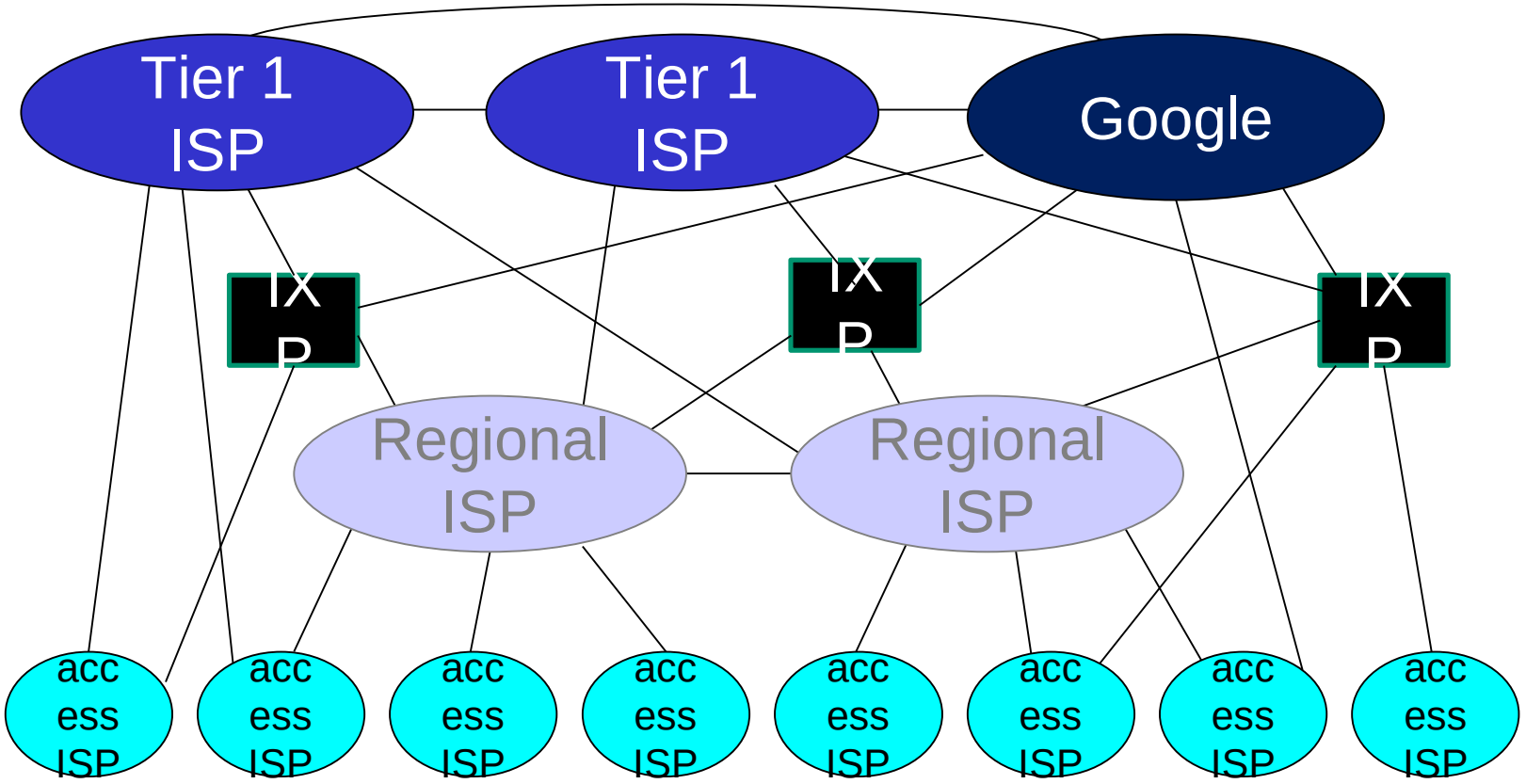


Internet structure: network of networks

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

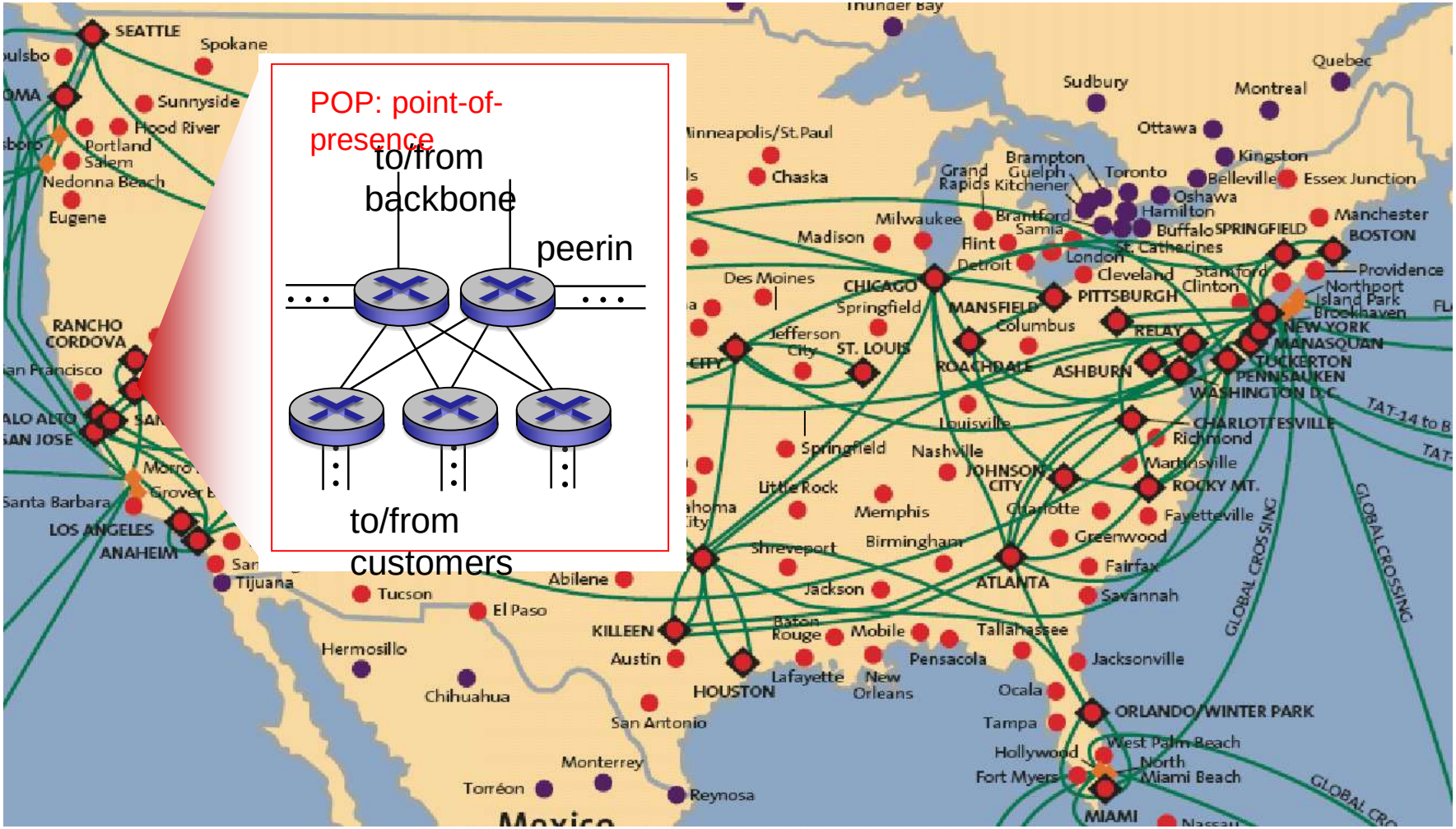


Internet structure: network of networks



- at center: small number of well-connected large networks
  - "tier-1" commercial ISPs (e.g., Level 3, Sprint, AT&T, NTT), national & international coverage
  - content provider network (e.g., Google): private network that connects it data centers to Internet, often bypassing tier-1, regional ISPs
  - Example ISP list in Malaysia: List , Service Offer , Service Cost

Tier-1 ISP: e.g., Sprint



Chapter 1: roadmap

1.1 what *is* the Internet?

1.2 network edge

- end systems, access networks, links

1.3 network core

- packet switching, circuit switching, network structure

1.4 delay, loss, throughput in networks

1.5 protocol layers, service models

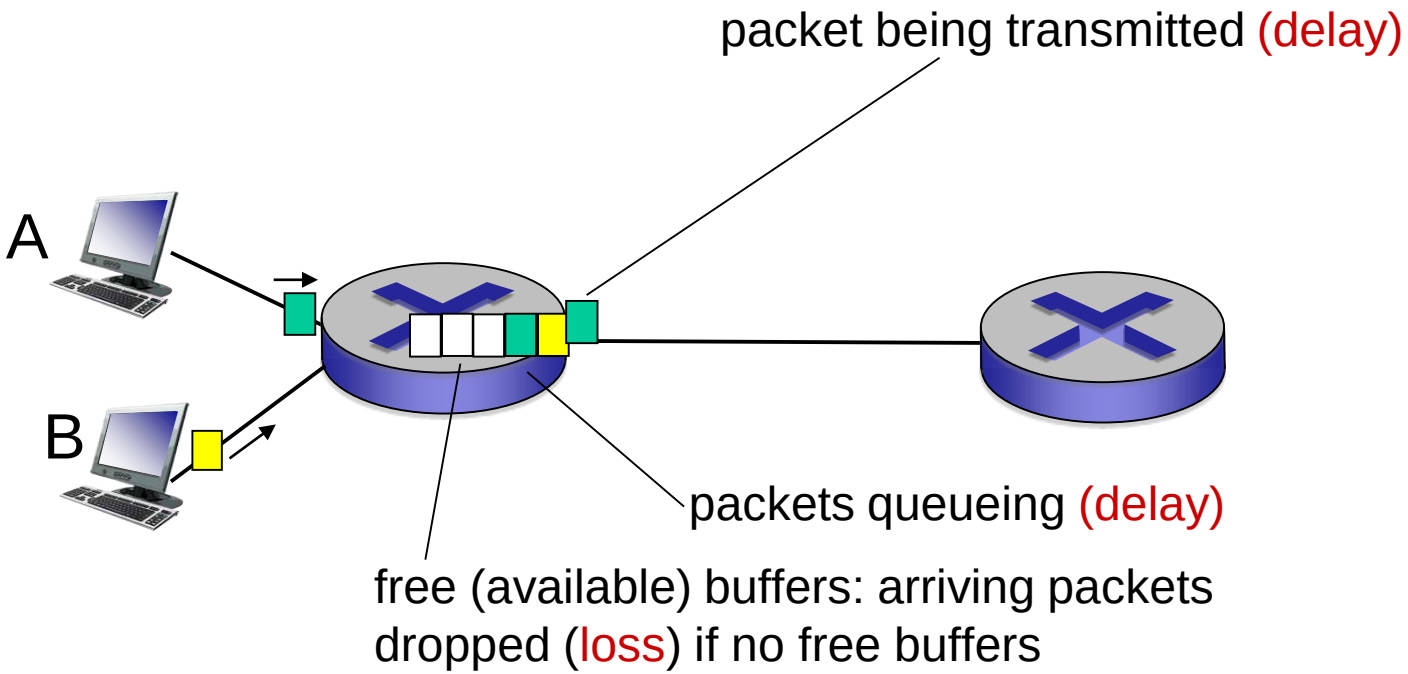
1.6 networks under attack: security

1.7 history

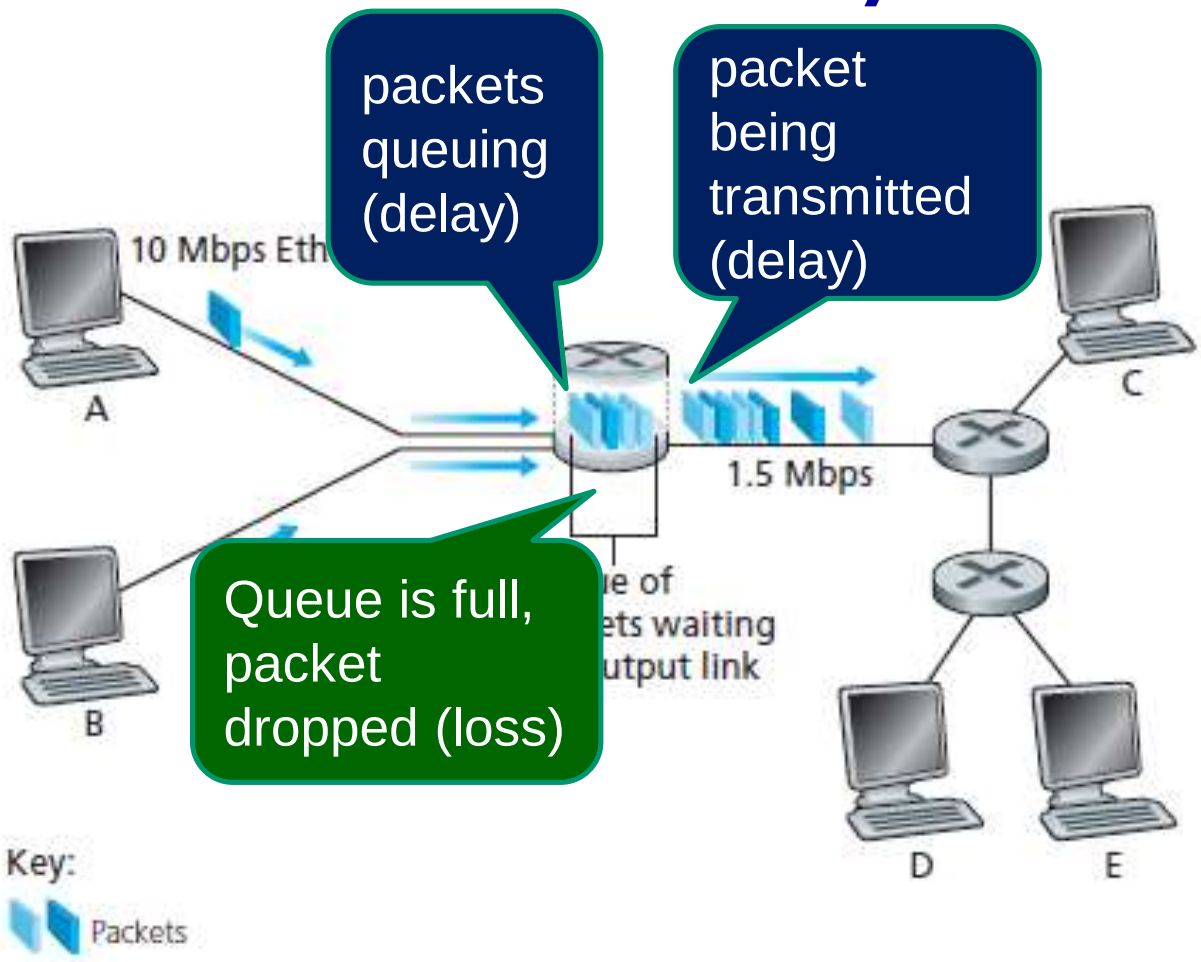
How do loss and delay occur?

packets *queue* in router buffers

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

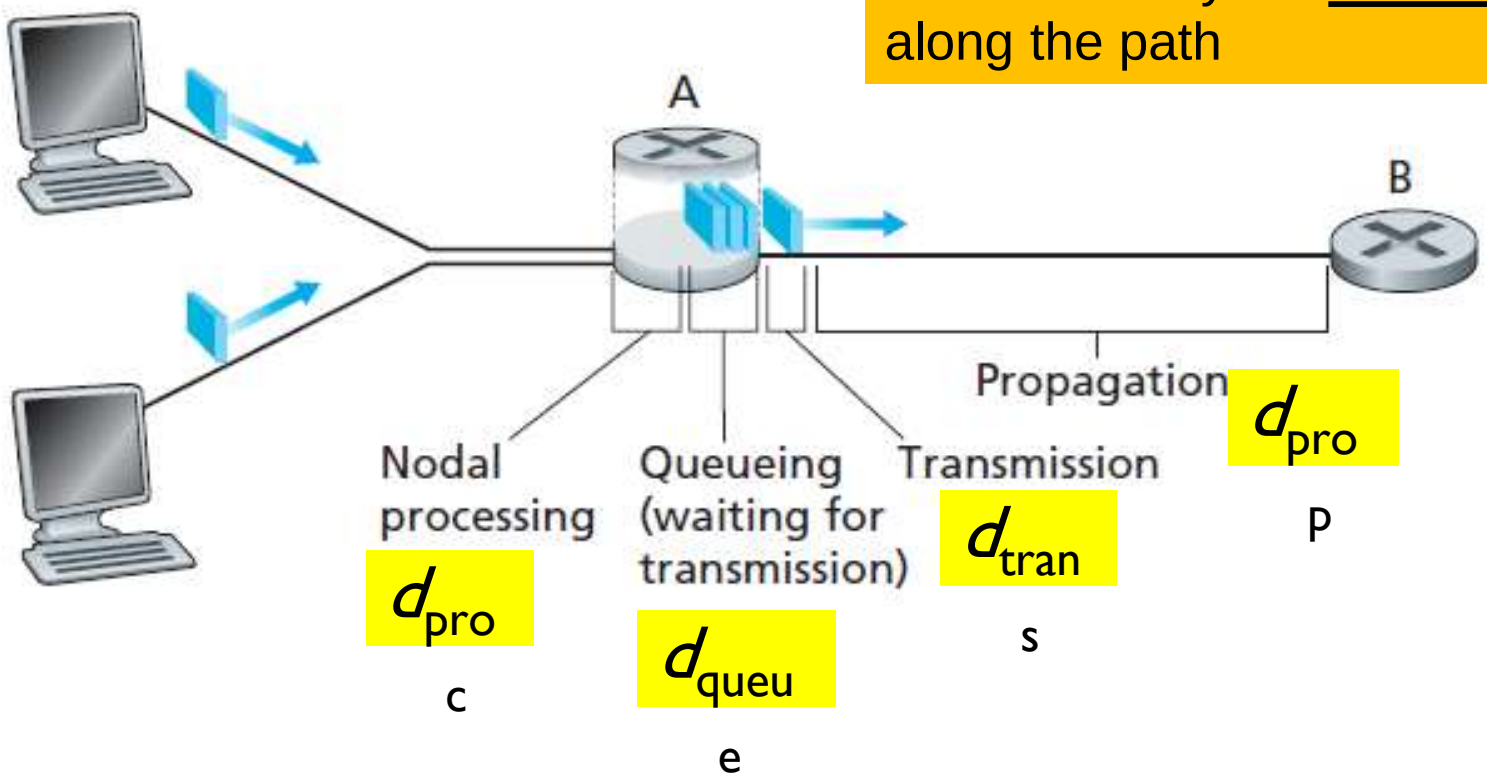


How do loss and delay occur?



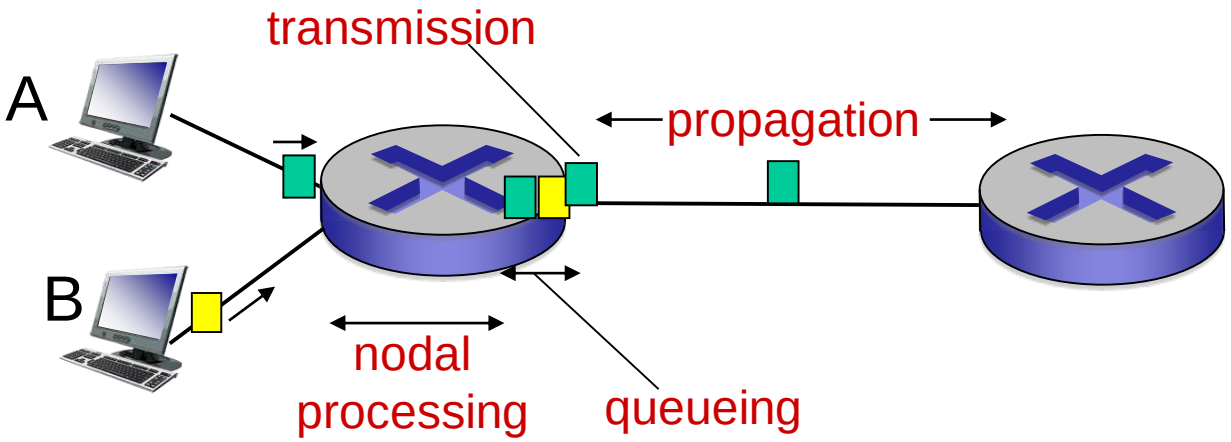
The nodal delay at router A

A packet goes through a series of delays at **each node** along the path



$$\textit{Total nodal delay} = d_{nodal} = d_{proc} + d_{queue} + d_{trans} + d_{prop}$$

Four sources of packet delay



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

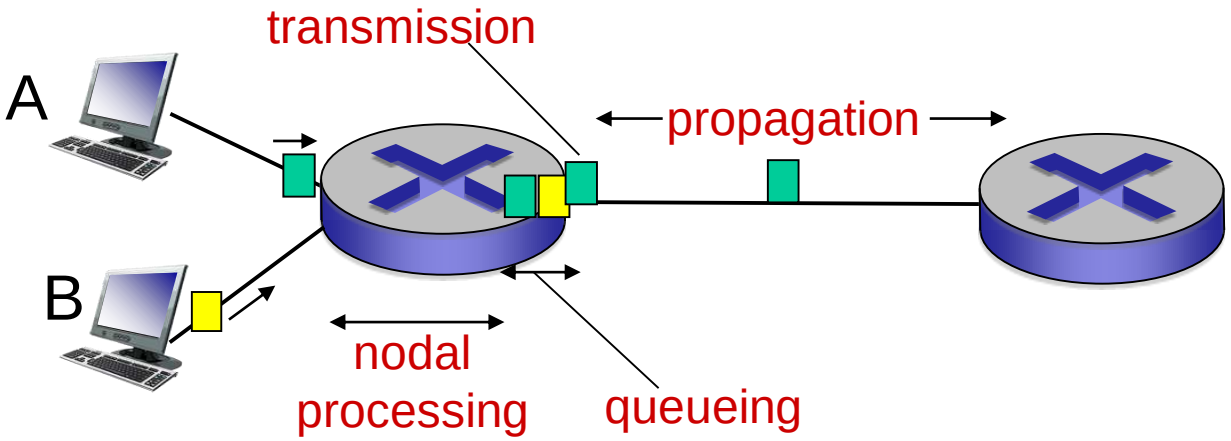
d_{proc}: nodal processing

- 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:

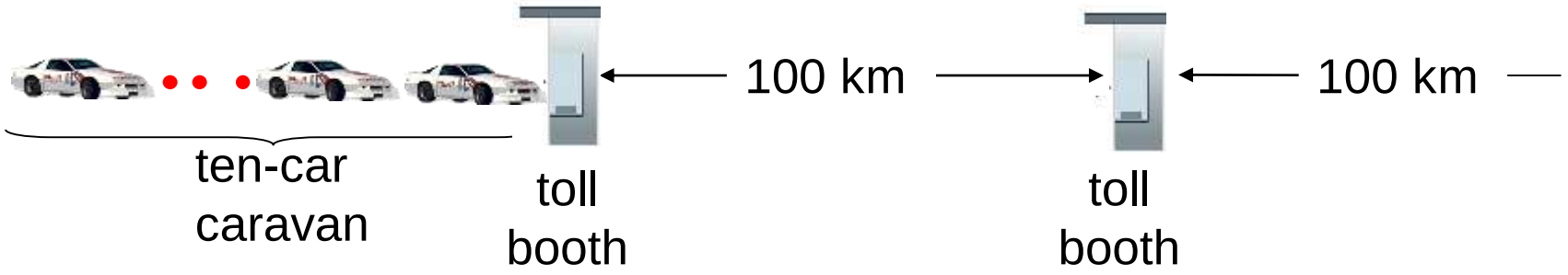
- L: packet length (bits)
- R: link *bandwidth (bps)*
- $d_{trans} = L/R$

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

d_{prop}: propagation delay:

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

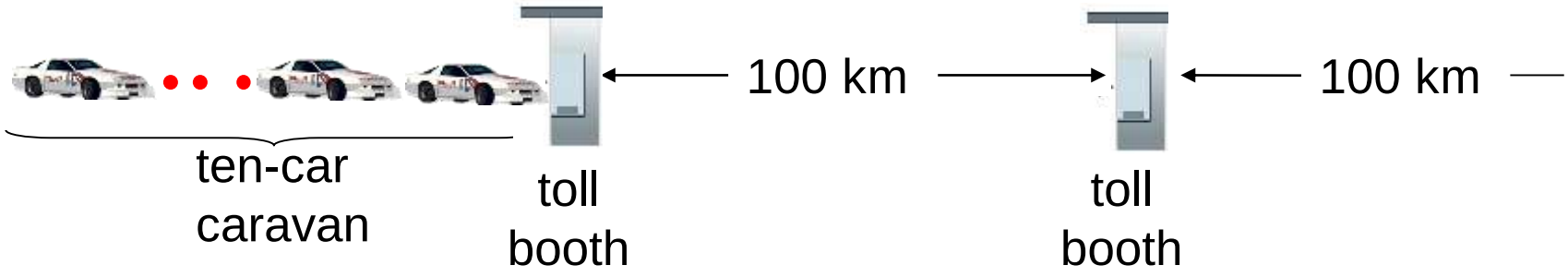
Caravan analogy



- cars "propagate" at 100 km/hr
- toll booth takes 12 sec to service car (bit transmission time)
- car ~ bit; caravan ~ packet
- *Q:* How long until caravan is lined up before 2nd toll booth?

- time to "push" entire caravan through toll booth onto highway = 12*10 = 120 sec
- time for last car to propagate from 1st to 2nd toll both: 100km/(100km/hr)= 1 hr
- *A:* 62 minutes

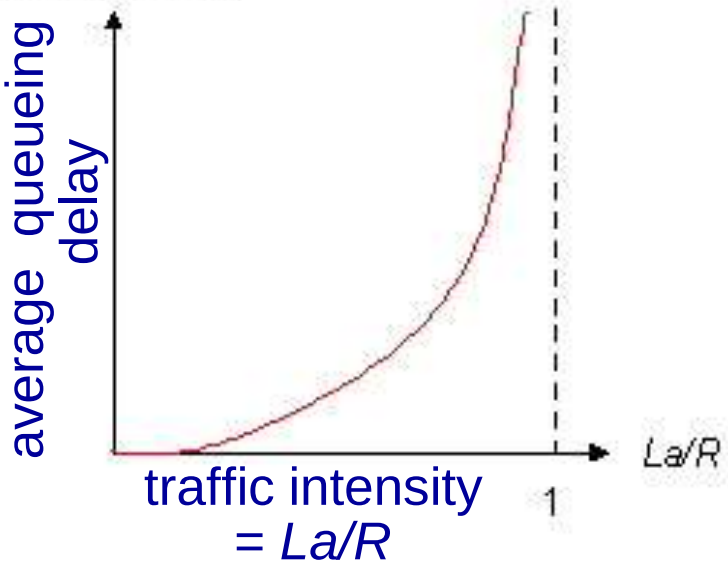
Caravan analogy (more)



- suppose cars now "propagate" at 1000 km/hr
- and suppose toll booth now takes one (1) min to service a car
- *Q:* Will cars arrive to 2nd booth before all cars serviced at first booth?
  - *A: Yes!* after 7 min, first car arrives at second booth; three cars still at first booth

Queueing delay (revisited)

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



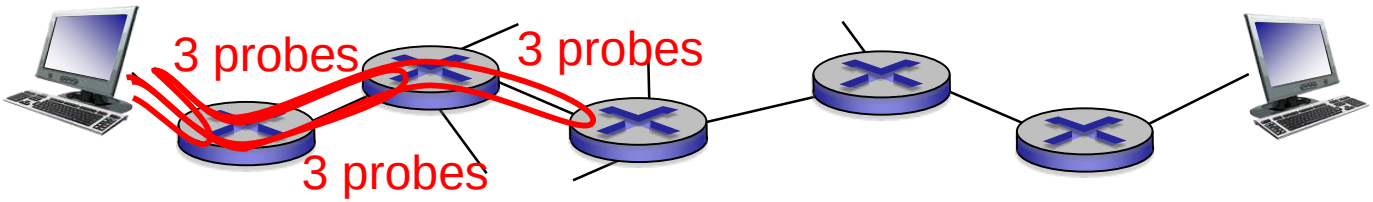
- *La/R* ~ 0: avg. queueing delay small
- *La/R* -> 1: avg. queueing delay large
- *La/R* > 1: more "work" arriving than can be serviced, average delay infinite!

La/R ~ 0

La/R -> 1

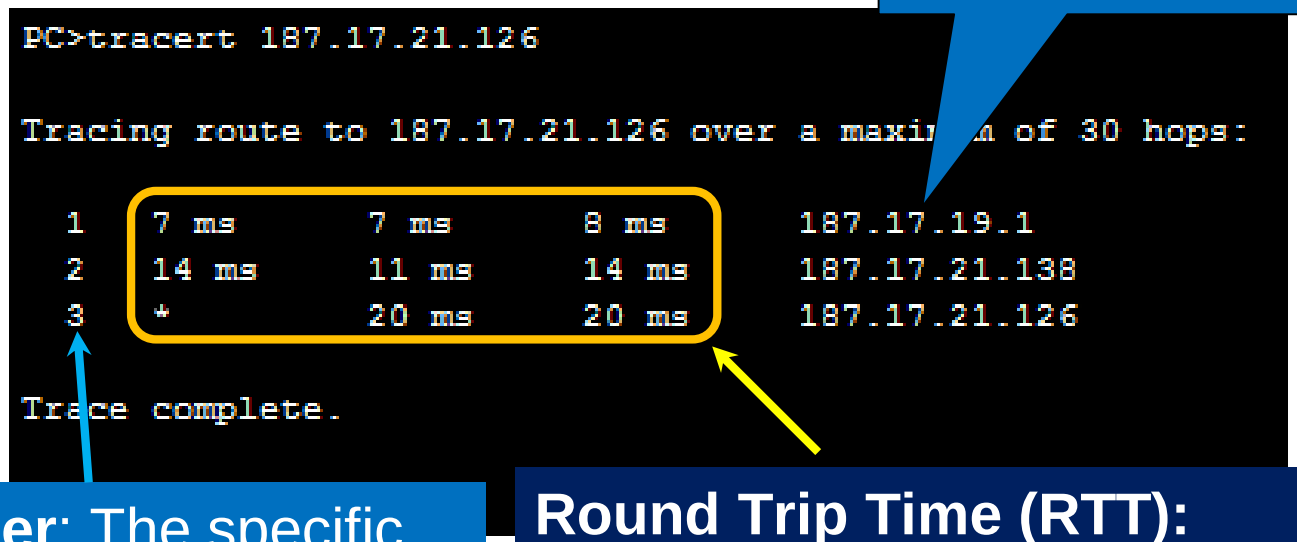
"Real" Internet delays and routes

- what do "real" Internet delay & loss look like?
- `traceroute/tracert` 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.



the destination system

The IP address of the hop



Hop number: The specific hop number in the path from the sender to the destination.

Round Trip Time (RTT):
The time it takes for a packet to get to a hop and back, displayed in milliseconds (ms).

- By default, **tracert** sends 3 packets to each hop, so the output lists three roundtrip times per hop.

"Real" Internet delays, routes

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

3 delay measurements from gaia.cs.umass.edu to cs-gw.cs.umass.edu

```
1  cs-gw (128.119.240.254)  1 ms  1 ms  2 ms
2  border1-rt-fa5-1-0.gw.umass.edu (128.119.3.145)  1 ms  1 ms  2 ms
3  cht-vbns.gw.umass.edu (128.119.3.130)  6 ms 5 ms 5 ms
4  jn1-at1-0-0-19.wor.vbns.net (204.147.132.129)  16 ms 11 ms 13 ms
5  jn1-so7-0-0-0.wae.vbns.net (204.147.136.136)  21 ms 18 ms 18 ms
6  abilene-vbns.abilene.ucaid.edu (198.32.11.9)  22 ms  18 ms  22 ms
7  nycm-wash.abilene.ucaid.edu (198.32.8.46)  22 ms  22 ms  22 ms
8  62.40.103.253 (62.40.103.253)  104 ms 109 ms 106 ms
9  de2-1.de1.de.geant.net (62.40.96.129)  109 ms 102 ms 104 ms
10  de.fr1.fr.geant.net (62.40.96.50)  113 ms 121 ms 114 ms
11  renater-gw.fr1.fr.geant.net (62.40.103.54)  112 ms  114 ms  112 ms
12  nio-n2.cssi.renater.fr (193.51.206.13)  111 ms  114 ms  116 ms
13  nice.cssi.renater.fr (195.220.98.102)  123 ms  125 ms  124 ms
14  r3t2-nice.cssi.renater.fr (195.220.98.110)  126 ms  126 ms  124 ms
15  eurecom-valbonne.r3t2.ft.net (193.48.50.54)  135 ms  128 ms  133 ms
16  194.214.211.25 (194.214.211.25)  126 ms  128 ms  126 ms
17  * * *
18  * * *
19  fantasia.eurecom.fr (193.55.113.142)  132 ms  128 ms  136 ms
```

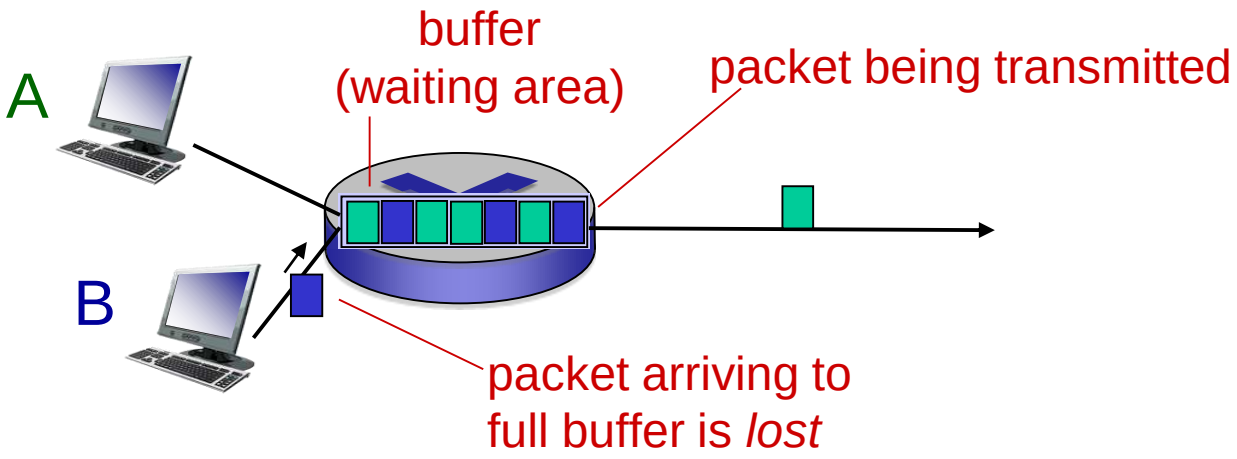
trans-oceanic link

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

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

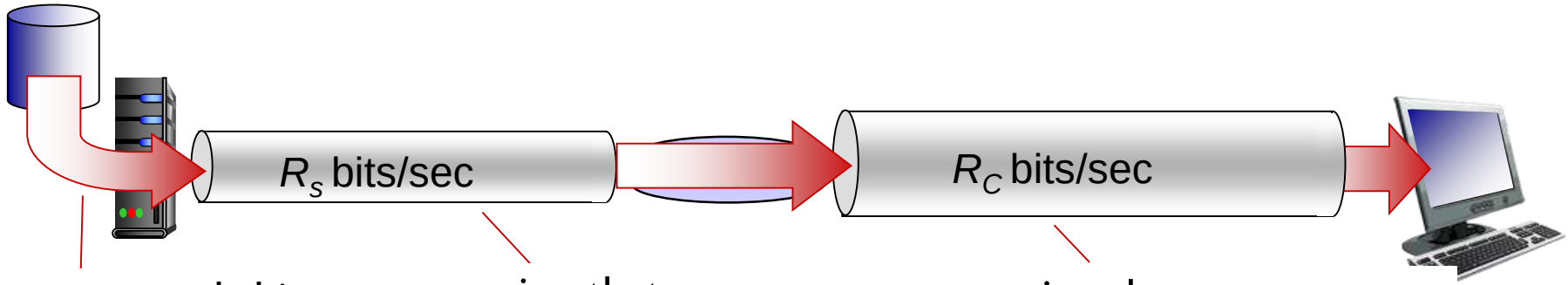
Packet loss

- 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



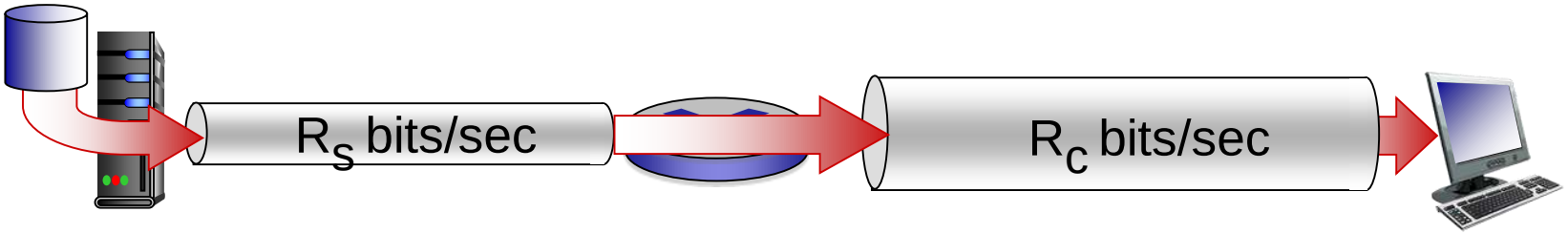
Throughput

- *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

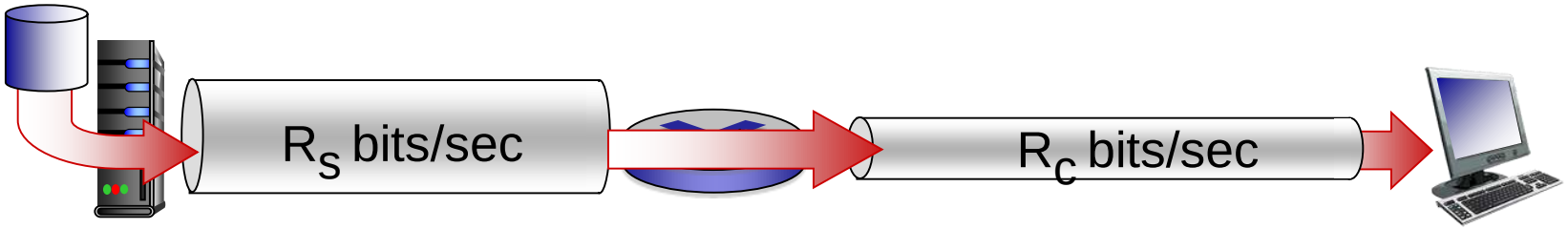


Throughput (more)

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

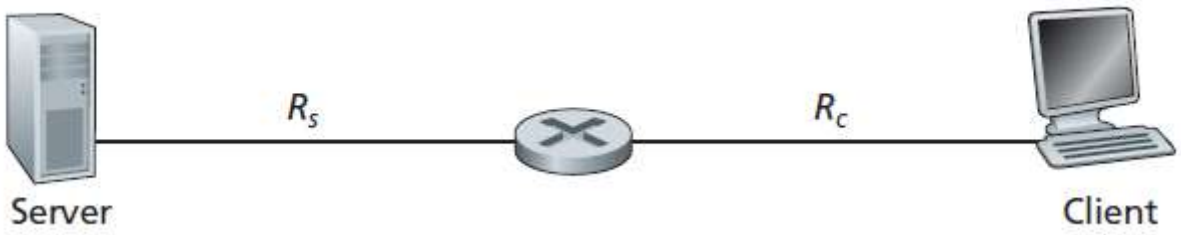


- $R_s > R_c$ What is average end-end throughput?



bottleneck link

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

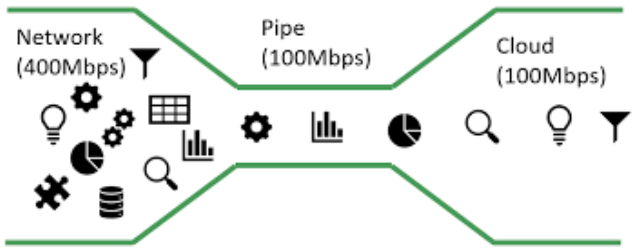


- If $R_s < R_c$
  - Flow from Server to Client not interrupted
  - Throughput = R_s

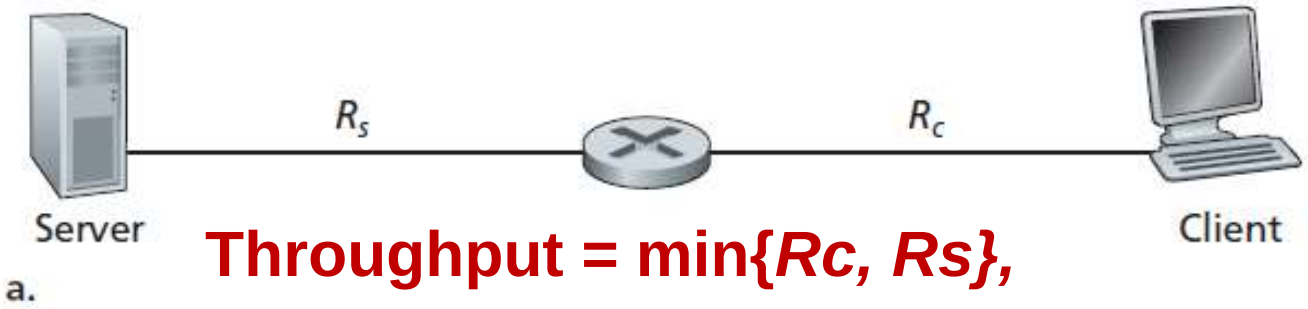
- If $R_s > R_c$
  - Flow from Server to Client is interrupted, queue may form
  - Throughput = R_c

Throughput = min{*Rc, Rs*},

- the transmission rate of the **bottleneck link**

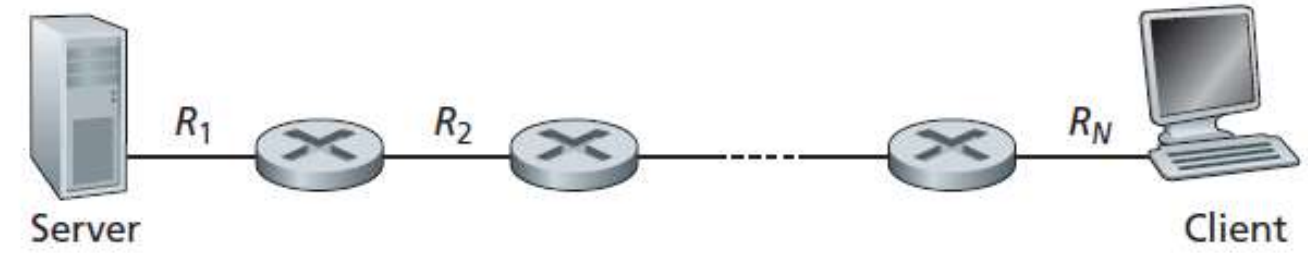


Throughput for a file transfer from server to client



a.

Throughput = min{*Rc, Rs*},



b.

Throughput = min{*R1, R2,..., RN*},

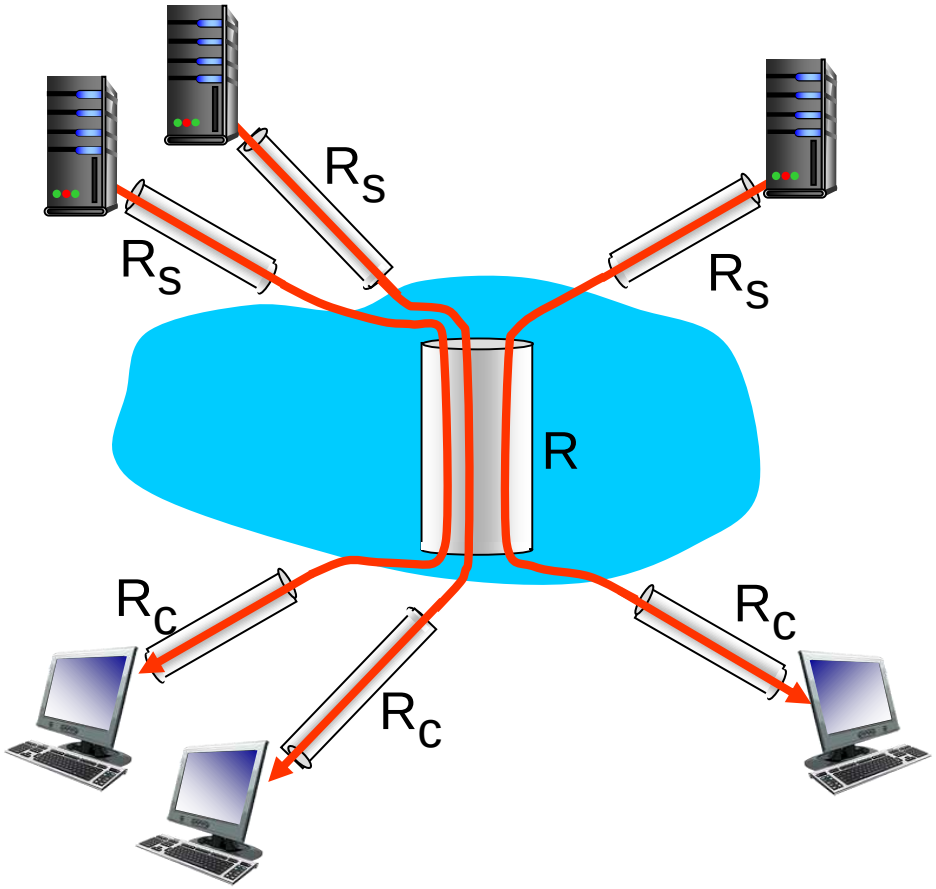
Example A

- The time it takes to transfer a large file of *F bits from server to* client as *F/min{Rs, Rc}*
- *Say, you are downloading a* file of *F = 32 million bits, the server has a transmission rate of Rs = 2 Mbps,* and you have an access link of *Rc = 1 Mbps.*
- *The time needed to transfer the file is*

Transfer time = (32 x 10^6) bits/1Mbps
= 32 seconds

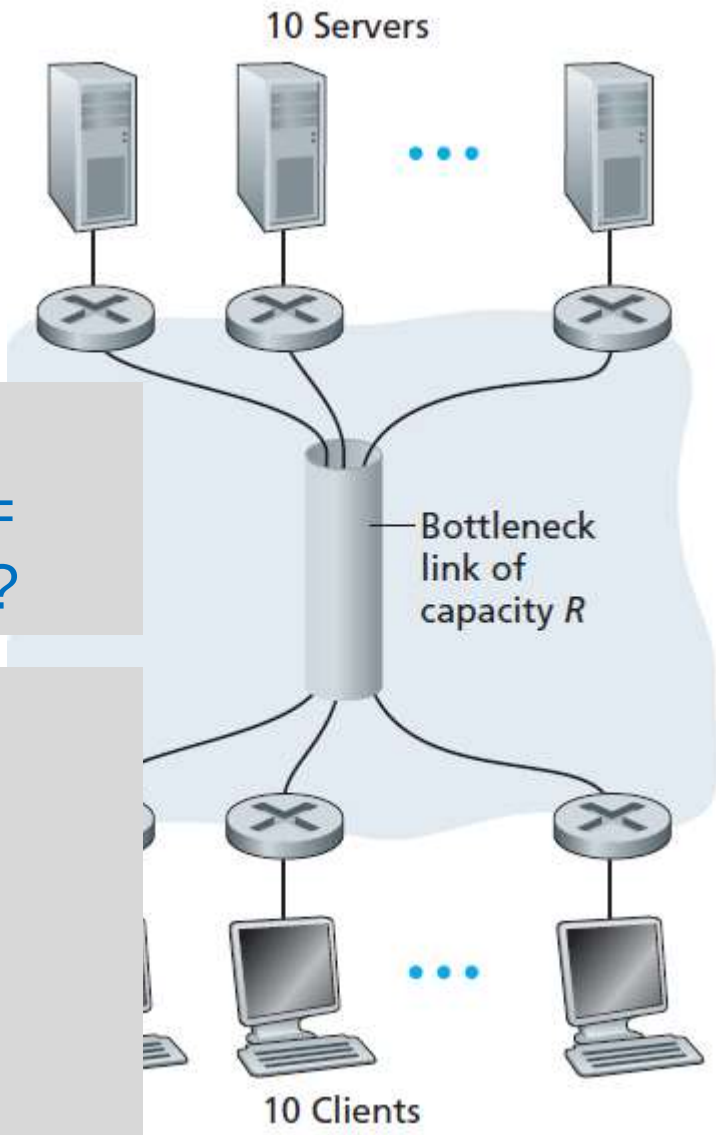
Throughput: Internet scenario

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



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

End-to-end throughput: Internet scenario



Example B:
If R_s = 2Mbps, R_c = 1Mbps, R = 5Mbps, what is the throughput?

Answer:
Because R is shared, what is available for each link is (5Mbps/10) = 500kbps

Throughput = 500kbps

Calculation: Example 1

Suppose Host A wants to send a large file to Host B. The path from Host A to Host B has 3-links, of rate R_1=150kbps, R_2=2Mbps, and R_3=1Mbps.

i) Assuming no other traffic in the network, what is the throughput for the file transfer? Justify your answer. [2 mark]

Throughput = min(R)=150 kbps

ii) Suppose the file is 4 million bytes. Roughly, how long will it take to transfer the file to Host B? [5 marks]

Time transfer=$4*10^6*8/(150*10^3)$=213.3 seconds

Calculation: Example 2

Host A wants to send a 30-Mbit MP3 file to host B. All the links in the path between source and destination have a transmission rate of 10Mbps. Assume that the propagation speed is $2*10^8$ meters/sec, and the distance between source and destination is 10km. Initially suppose there is only one link between the source and the destination. Also suppose that the entire MP3 file is sent as one packet (Ignore processing delay and queuing delay). Show your workings.

i) Calculate the transmission delay?

transmission delay = L/R = 30M/10Mbps = 3s

ii) What will be the end-to-end delay?

D_{prop} = d/s = $10000/2*10^8$ = 0.00005s

D_{total} = 3+0.00005 = 3. 00005s

iii) How many bits will the source have transmitted when the first bit of the MP3 file arrives at the destination?

R* D_{prop} = 10Mbps*0.00005s=500bit

Chapter 1: roadmap

1.1 what *is* the Internet?

1.2 network edge

- end systems, access networks, links

1.3 network core

- packet switching, circuit switching, network structure

1.4 delay, loss, throughput in networks

1.5 protocol layers, service models

1.6 networks under attack: security

1.7 history

Protocol "layers"

Networks are complex, with many "pieces":

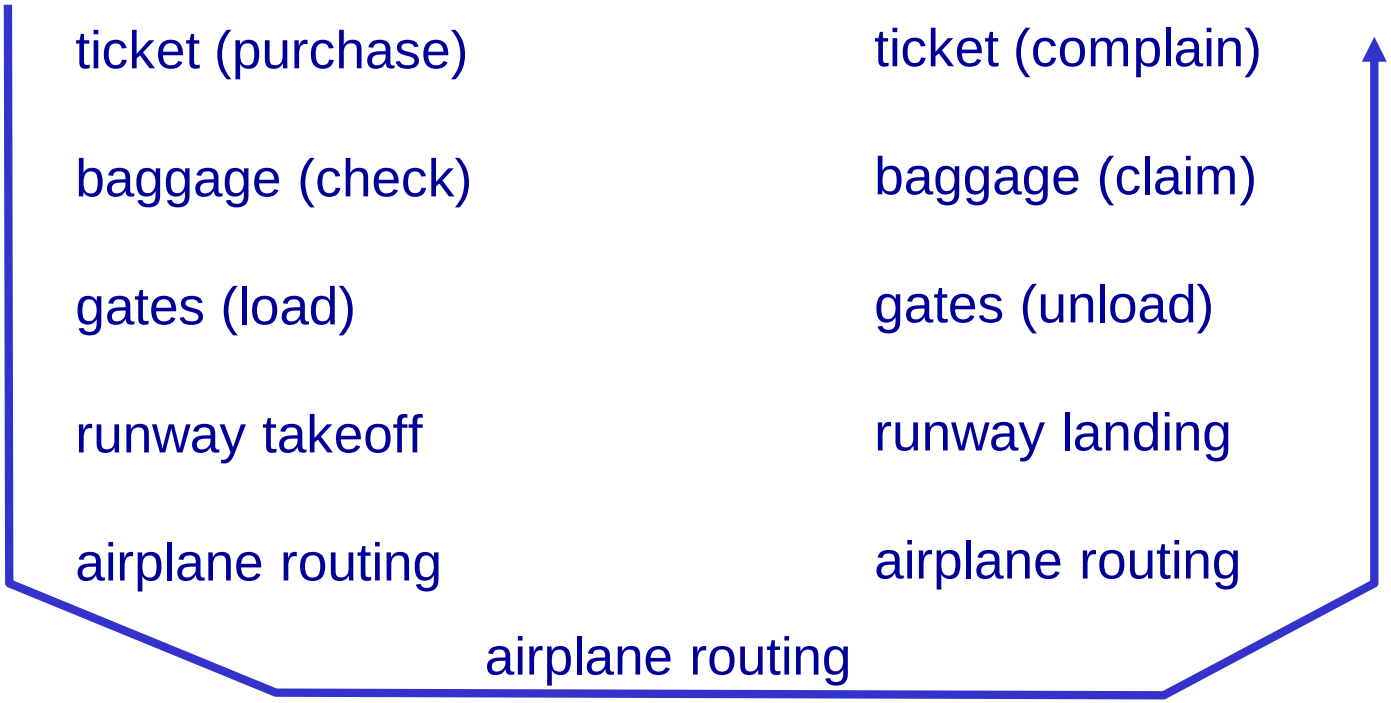
- hosts
- routers
- links of various media
- applications
- protocols
- hardware, software

Question:

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

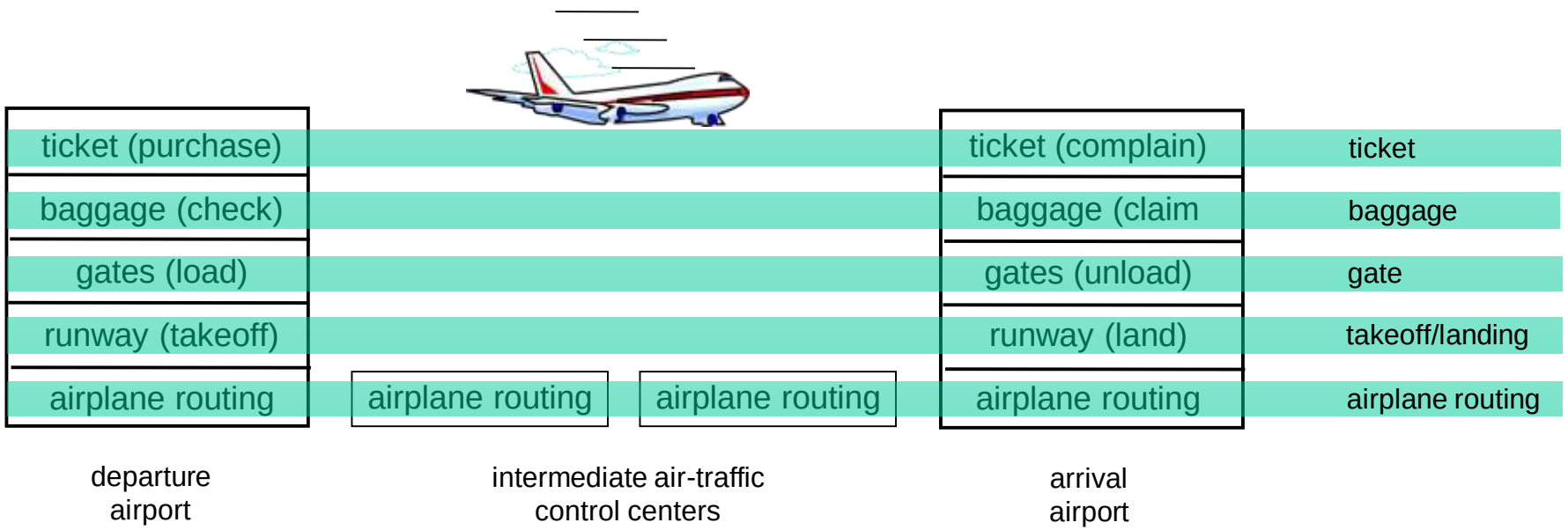
…. or at least our discussion of networks?

Organization of air travel



- a series of steps

Layering of airline functionality



layers: each layer implements a service

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

Why layering?

dealing with complex systems:

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

ISO/OSI reference model

***O**pen **S**ystem **I**nterconnection (OSI)* model defines a **generic** networking framework to implement protocols in seven (7) layers

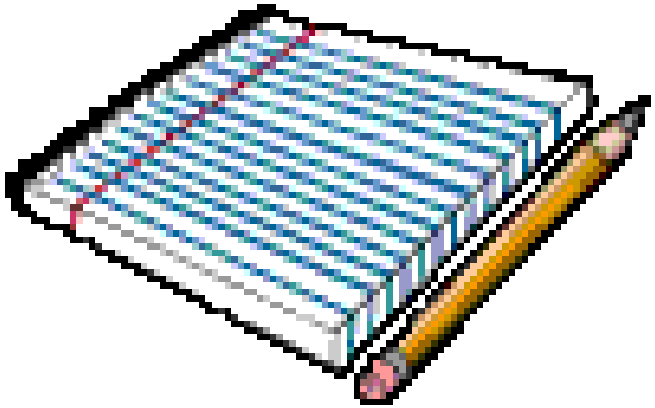
Generic OSI Model

	Data unit	Layer	Function
Host layers	Data	7. Application	Network process to application
		6. Presentation	allow applications to interpret meaning of data, e.g. data representation, encryption and decryption, convert machine dependent data to machine independent data
		5. Session	Interhost communication, managing sessions between applications, synchronization, data recovery
	Segments	4. Transport	Reliable delivery of packets between points on a network.
Media layers	Packet/Datagram	3. Network	Addressing, routing and (not necessarily reliable) delivery of datagrams between points on a network.
	Bit/Frame	2. Data link	A reliable direct point-to-point data connection.
	Bit	1. Physical	A (not necessarily reliable) direct point-to-point data connection.

OSI Model: AN Analogy: Application Layer - Source Host

After riding your new bicycle a few times in Tokyo, you decide that you want to give it to a friend who studies in UTM, JB.

Presentation & Session Layer - Source Host



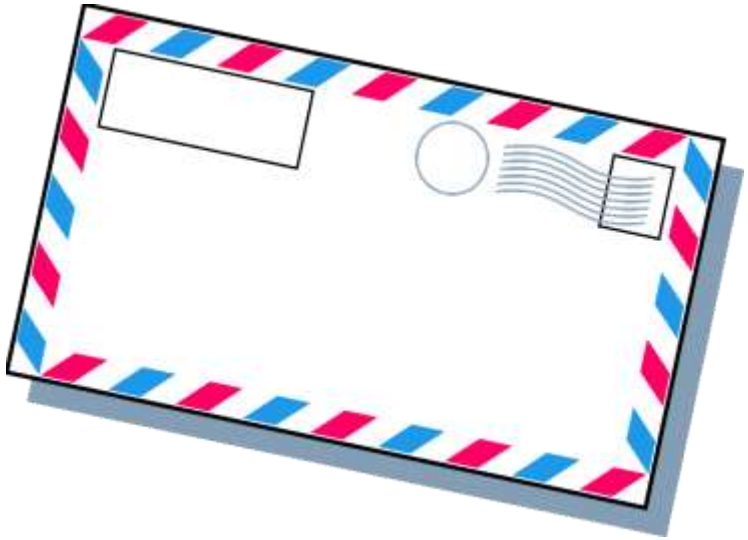
PRESENTATION:
Make sure you have the proper directions to disassemble and reassemble the bicycle.

SESSION:
Call your friend and make sure you have his correct address.

OSI Model Analogy :Transport Layer - Source Host

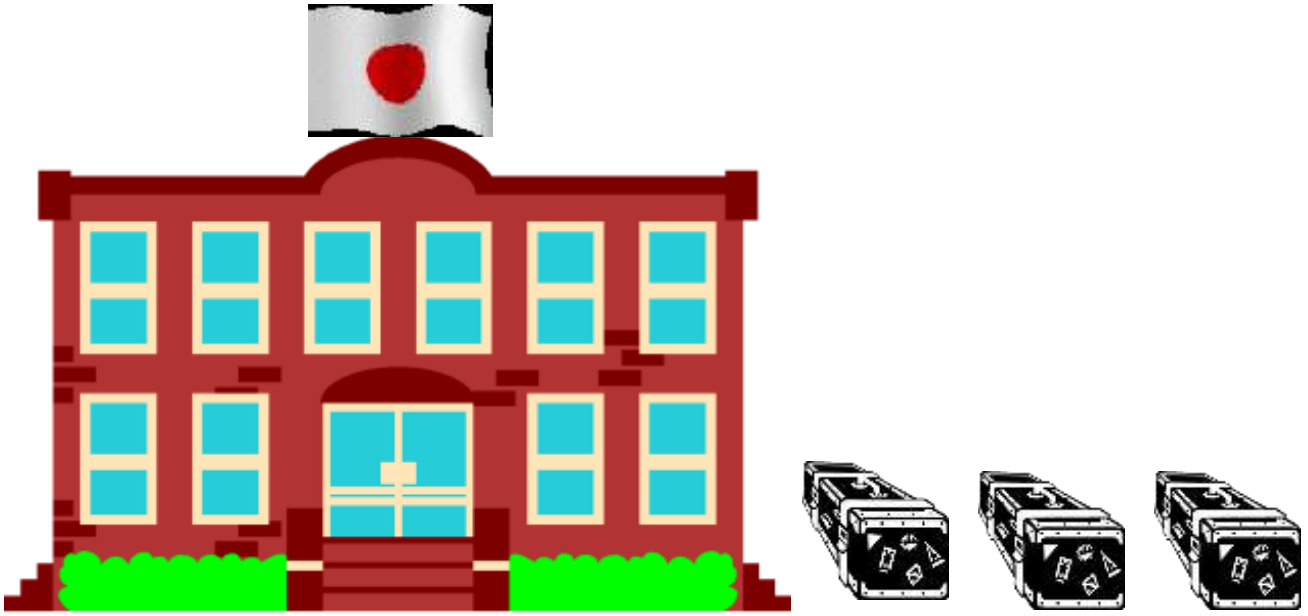
Disassemble the bicycle and put different pieces in different boxes. The boxes are labeled "1 of 3", "2 of 3", and "3 of 3".

OSI Model Analogy: Network Layer - Source Host



Put your friend's complete mailing address (and yours) on each box. Since the packages are too big for your mailbox (and since you don't have enough stamps) you determine that you need to go to the post office.

OSI Model Analogy: Data Link Layer – Source Host



Tokyo post office takes possession of the boxes.

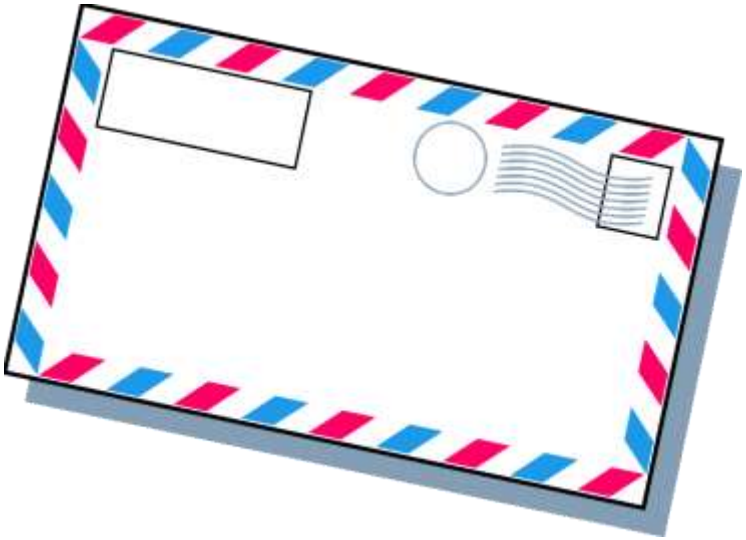
OSI Model Analogy: Physical Layer - Media

The boxes are flown from Tokyo to JB

OSI Model Analogy : Data Link Layer - Destination

UTM post office receives your boxes.

OSI Model Analogy: Network Layer - Destination



Upon examining the destination address, UTM post office determines that your boxes should be delivered to the written home address.

OSI Model Analogy : Transport Layer - Destination

Your friend calls you and tells you he got all 3 boxes and he is having another friend named FARIS reassemble the bicycle.

Session & Presentation Layer - Destination

SESSION:
Your friend hangs up because he is done talking to you.

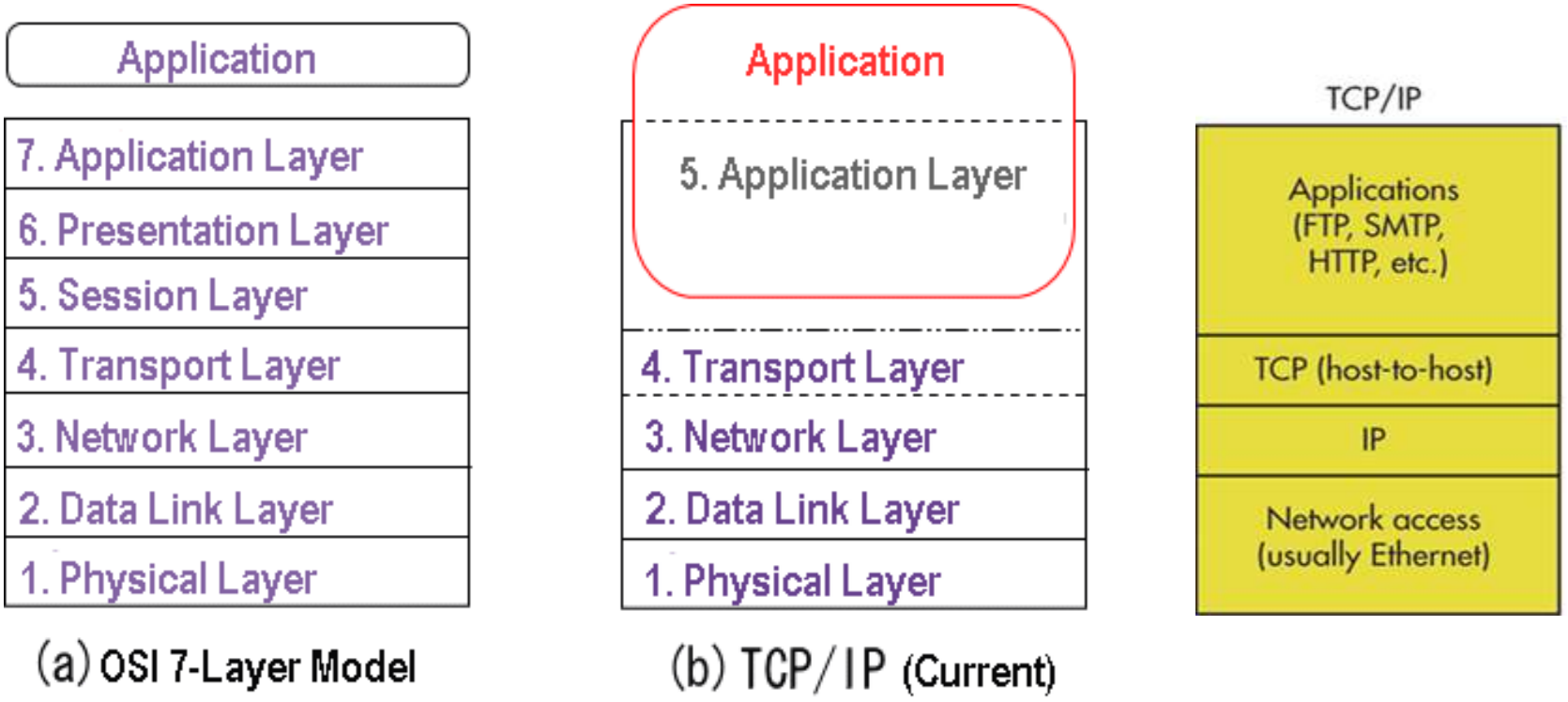
PRESENTATION:
FARIS is finished and "presents" the bicycle to your friend. Another way to say it is that your friend is finally getting his "present".

OSI Model Analogy : Application Layer - Destination

Your friend enjoys riding his new bicycle in UTM.

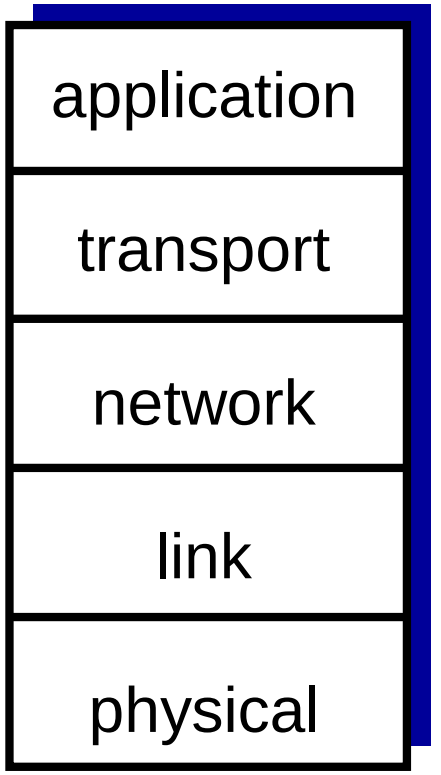
Internet protocol stack (TCP/IP model)

TCP/IP was developed by Advanced Research Projects Agency (ARPA) to build a nationwide packet data network in 1960s. It was first used in UNIX-based computers in universities and government installations. Today, it is the main protocol used in all Internet operations.



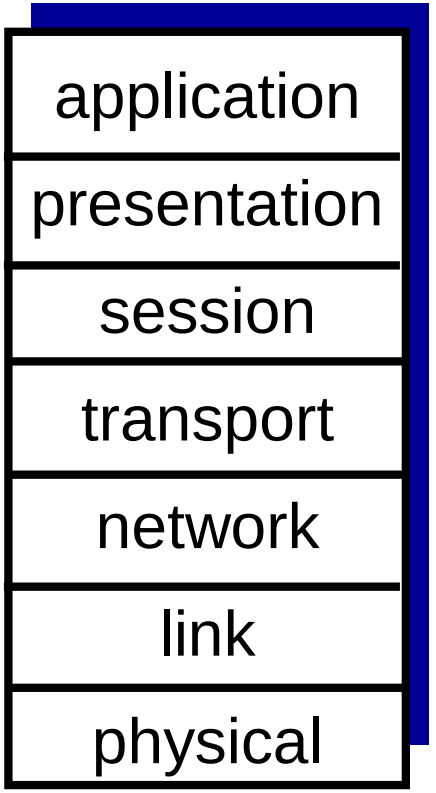
Internet protocol stack

- *application:* supporting network applications
  - FTP, SMTP, HTTP
- *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, 802.111 (WiFi), PPP
- *physical:* bits "on the wire"

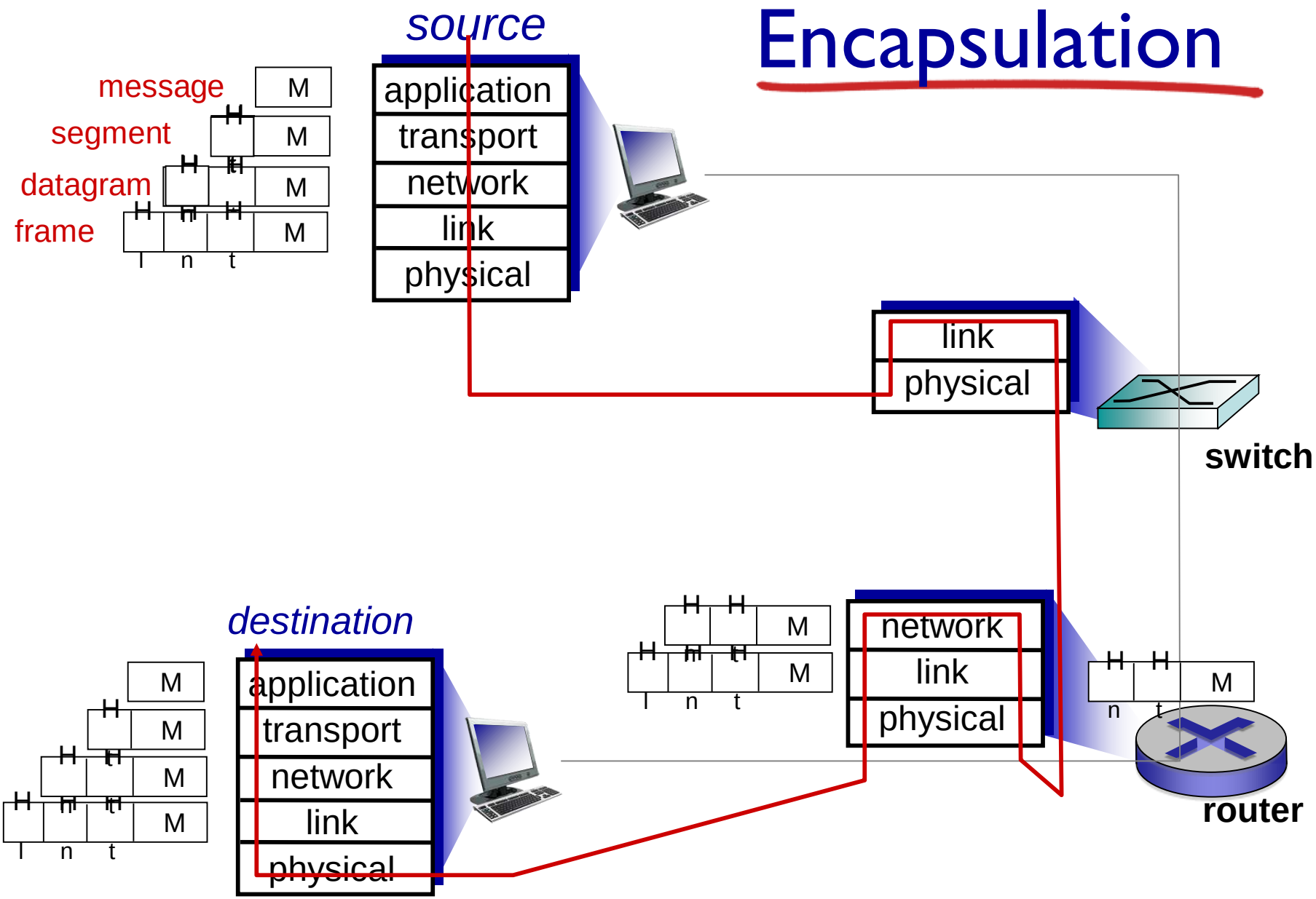


ISO/OSI reference model

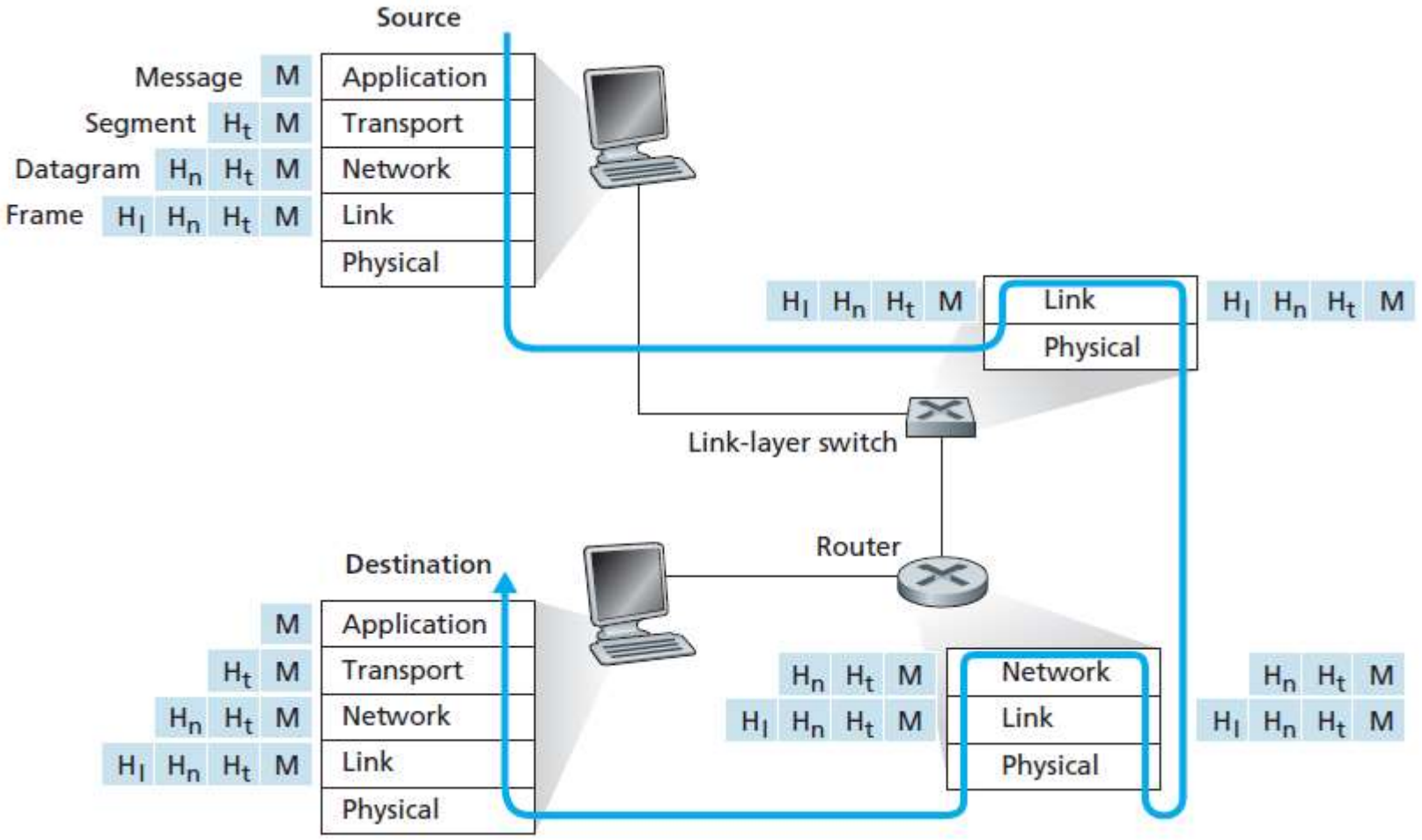
- *presentation:* allow applications to interpret meaning of data, e.g., encryption, compression, machine-specific conventions
- *session:* synchronization, checkpointing, recovery of data exchange
- Internet stack "missing" these layers!
  - these services, *if needed,* must be implemented in application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



Encapsulation



Encapsulation



Chapter 1: roadmap

1.1 what *is* the Internet?

1.2 network edge

- end systems, access networks, links

1.3 network core

- packet switching, circuit switching, network structure

1.4 delay, loss, throughput in networks

1.5 protocol layers, service models

1.6 networks under attack: security

1.7 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 a transparent network" ☺
  - Internet protocol designers playing "catch-up"
  - security considerations in all layers!

Bad guys: put malware into hosts via Internet

- malware can get in host from:
  - *virus:* self-replicating infection by receiving/executing object (e.g., e-mail attachment)
  - *worm:* self-replicating infection by passively receiving object that gets itself executed
- spyware malware can record keystrokes, web sites visited, upload info to collection site
- infected host can be enrolled in botnet, used for spam, DDoS attacks

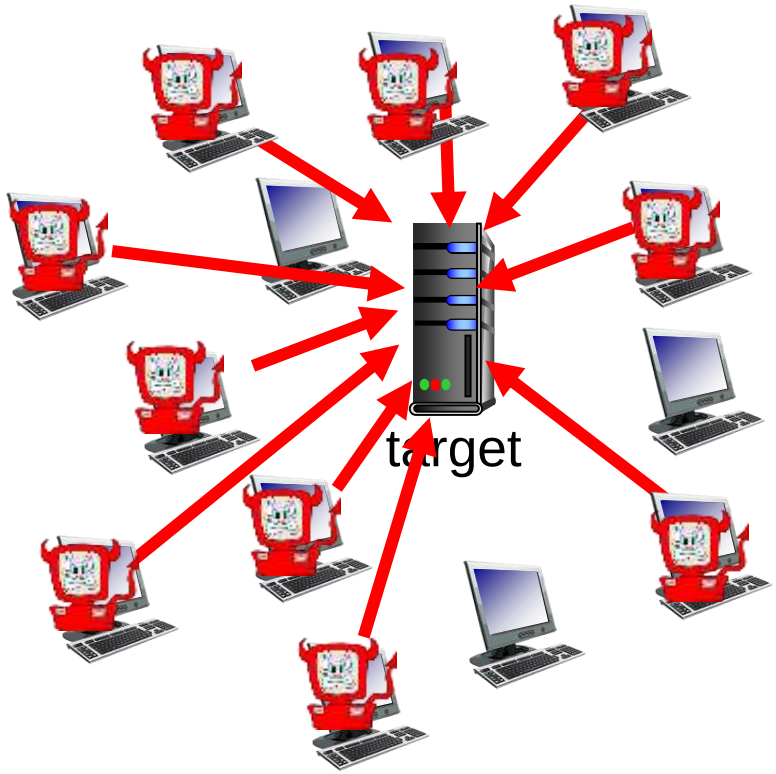
Bad guys: put malware into hosts via Internet

- Malware = malicious software
  - Examples: Worm, Virus, Trojans, spyware, botnet, etc.
- Malware can be
  - Self-replicating ☐ once it infects one host, from that host it seeks entry into other hosts over the Internet, and just keep going
  - User activated (click on an email attachment) or
  - Program activated (a vulnerable app that activates malware)
  - Recruiting your device to be part of a botnet, to be use for any purpose later

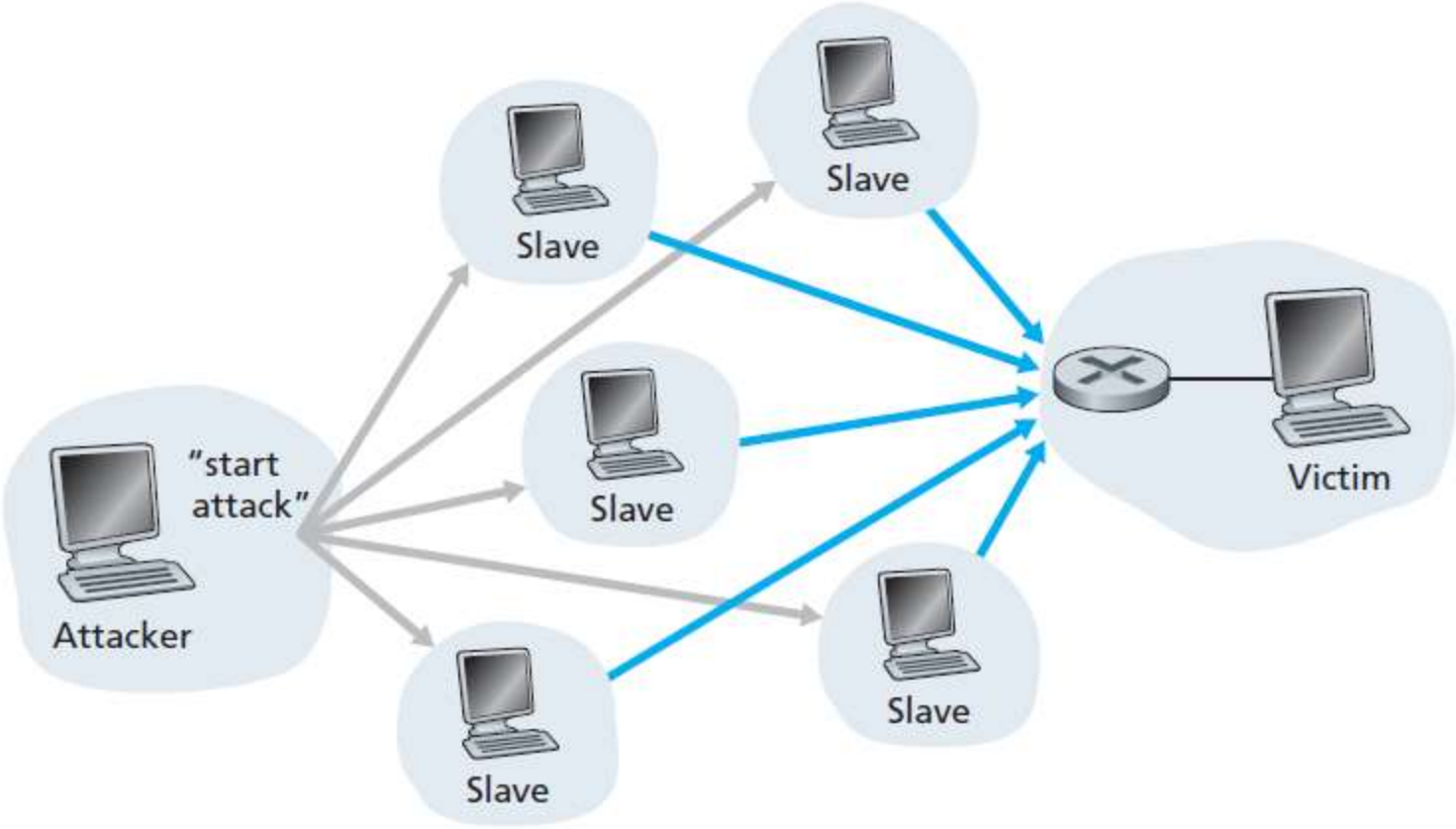
Bad guys: attack server, network infrastructure

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



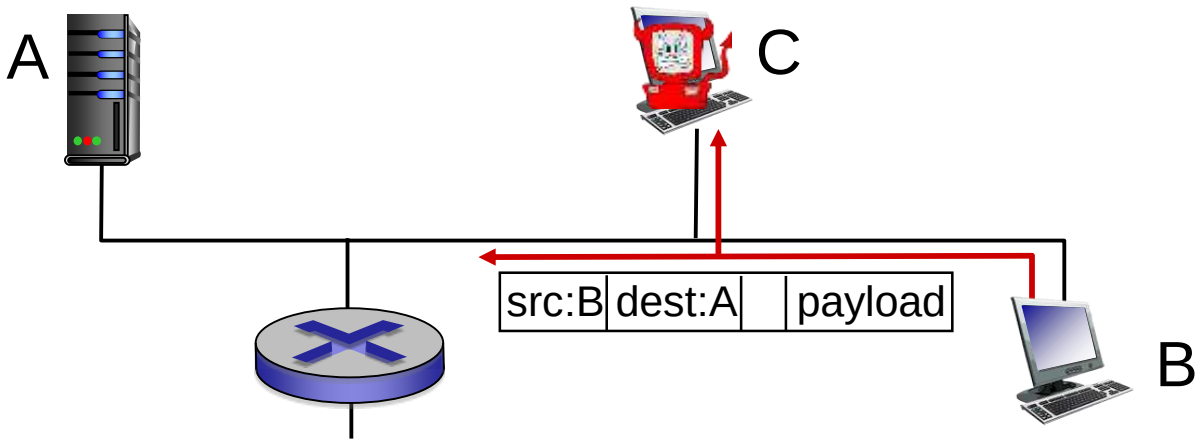
Denial of Service (DoS):



Bad guys can sniff packets

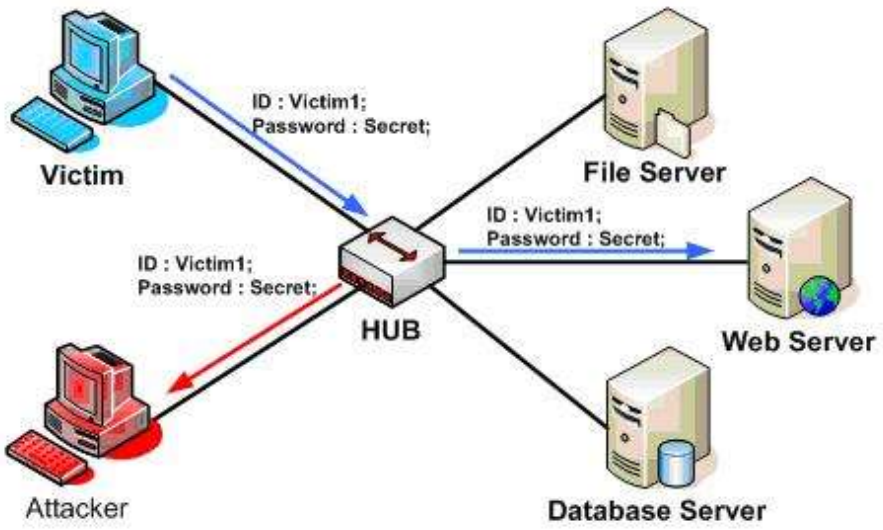
packet "sniffing":

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



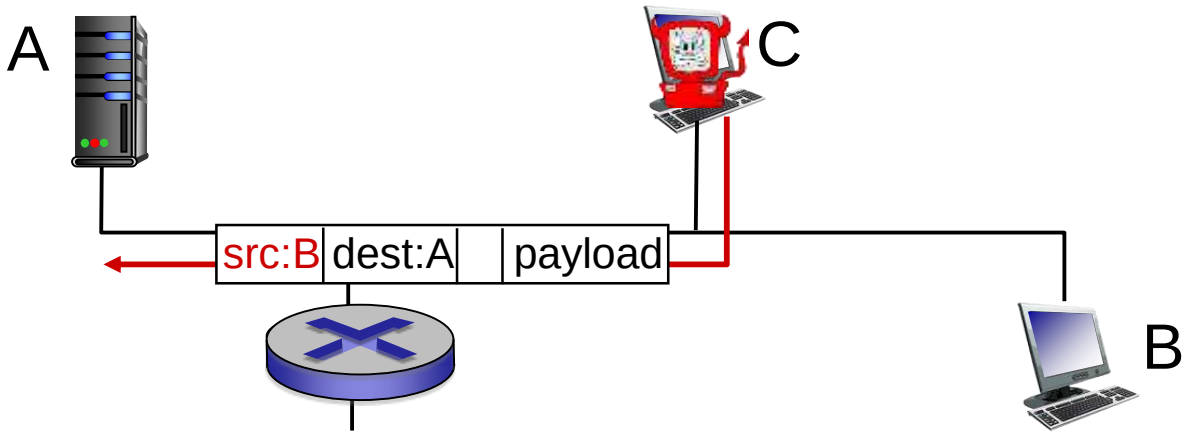
- wireshark software used for end-of-chapter labs is a (free) packet-sniffer

Packet "sniffing":



Bad guys can use fake addresses

IP spoofing: send packet with false source address



IP spoofing:

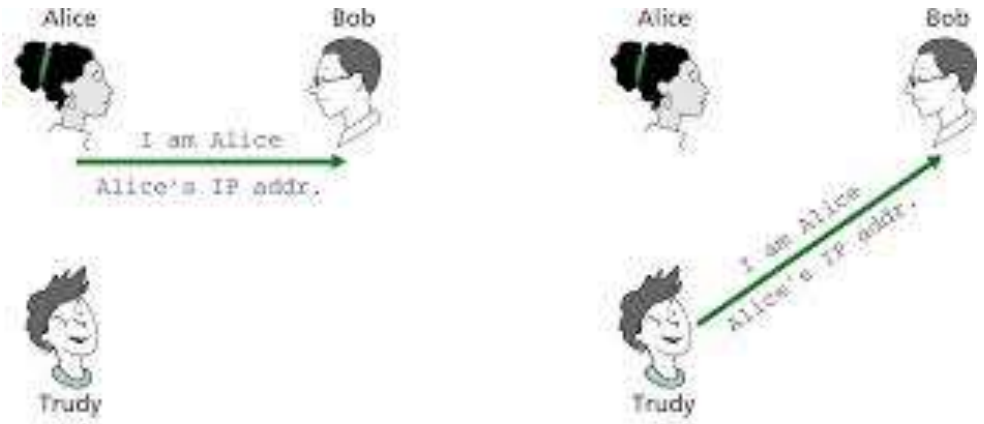


Figure 8.8 ♦ Protocol *ap2.0* and a failure scenario

Chapter 1: roadmap

1.1 what *is* the Internet?

1.2 network edge

- end systems, access networks, links

1.3 network core

- packet switching, circuit switching, network structure

1.4 delay, loss, throughput in networks

1.5 protocol layers, service models

1.6 networks under attack: security

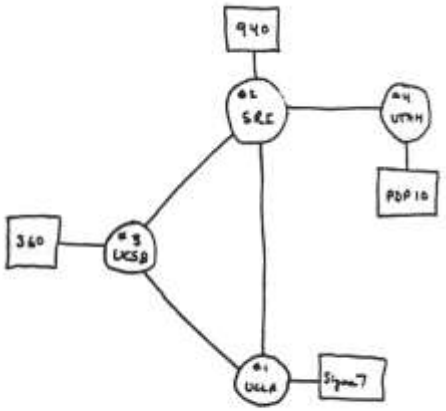
1.7 history

Internet history

1961-1972: Early packet-switching principles

- 1961: Kleinrock - queueing theory shows effectiveness of packet-switching
- 1964: Baran - packet-switching in military nets
- 1967: ARPAnet conceived by Advanced Research Projects Agency
- 1969: first ARPAnet node operational

- 1972:
  - ARPAnet public demo
  - NCP (Network Control Protocol) first host-host protocol
  - first e-mail program
  - ARPAnet has 15 nodes



THE ARPA NETWORK

Internet history

1972-1980: Internetworking, new and proprietary nets

- 1970: ALOHAnet satellite network in Hawaii
- 1974: Cerf and Kahn - architecture for interconnecting networks
- 1976: Ethernet at Xerox PARC
- late70's: proprietary architectures: DECnet, SNA, XNA
- late 70's: switching fixed length packets (ATM precursor)
- 1979: ARPAnet has 200 nodes

Cerf and Kahn's internetworking principles:

- minimalism, autonomy - no internal changes required to interconnect networks
- best effort service model
- stateless routers
- decentralized control

define today's Internet architecture

Internet history

1980-1990: new protocols, a proliferation of networks

- 1983: deployment of TCP/IP
- 1982: smtp e-mail protocol defined
- 1983: DNS defined for name-to-IP-address translation
- 1985: ftp protocol defined
- 1988: TCP congestion control

- new national networks: CSnet, BITnet, NSFnet, Minitel
- 100,000 hosts connected to confederation of networks

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
  - hypertext [Bush 1945, Nelson 1960's]
  - HTML, HTTP: Berners-Lee
  - 1994: Mosaic, later Netscape
  - late 1990's: commercialization of the Web

late 1990's – 2000's:

- more killer apps: instant messaging, P2P file sharing
- network security to forefront
- est. 50 million host, 100 million+ users
- backbone links running at Gbps

Internet history

2005-present

- ~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)

Introduction: summary

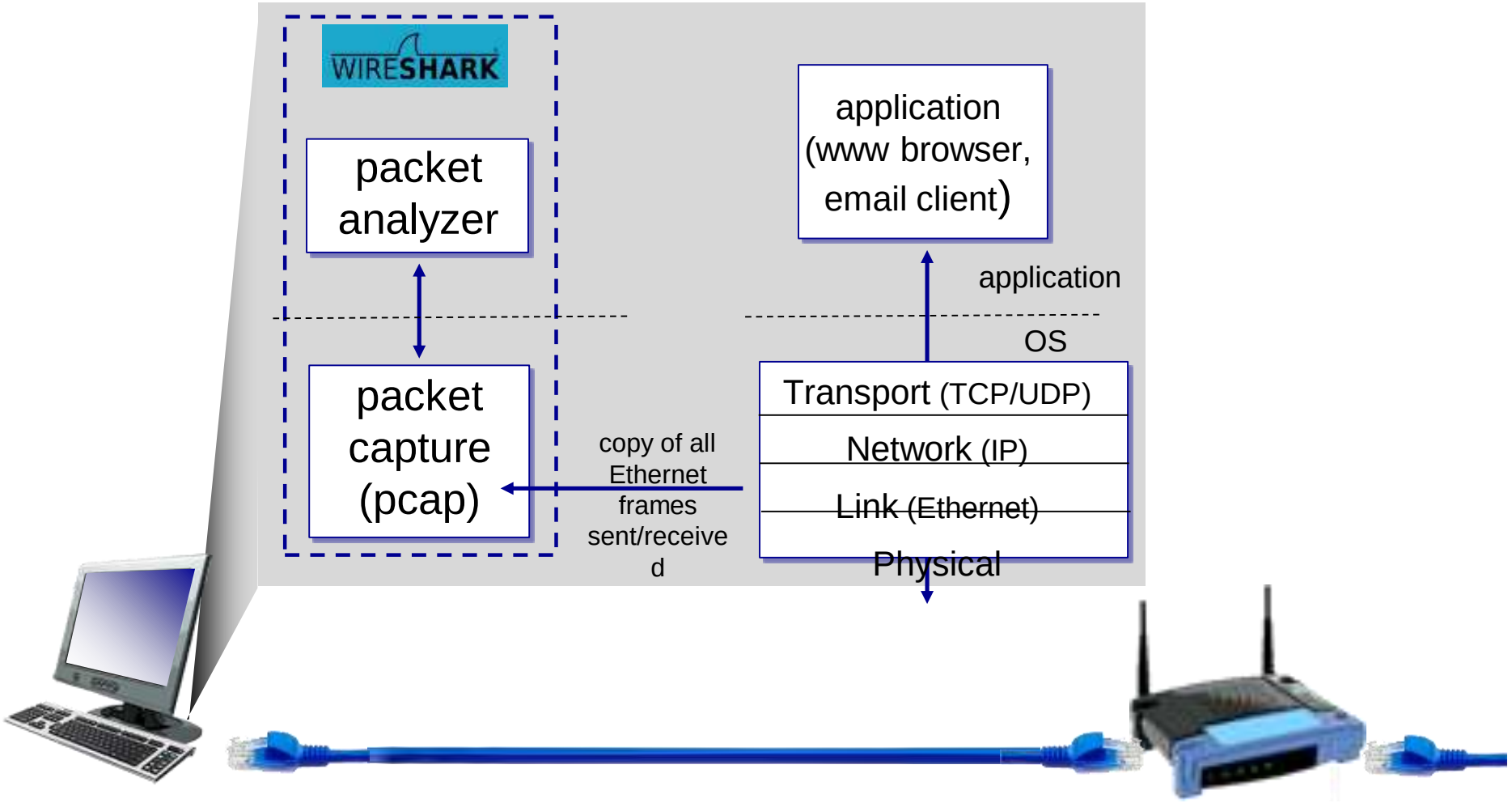
covered a "ton" of material!

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

you now have:

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

Chapter 1
Additional Slides


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