Day 2: The Internet, Protocols, and Packet Switching

CSEE 4119
Computer Networks
Ethan Katz-Bassett

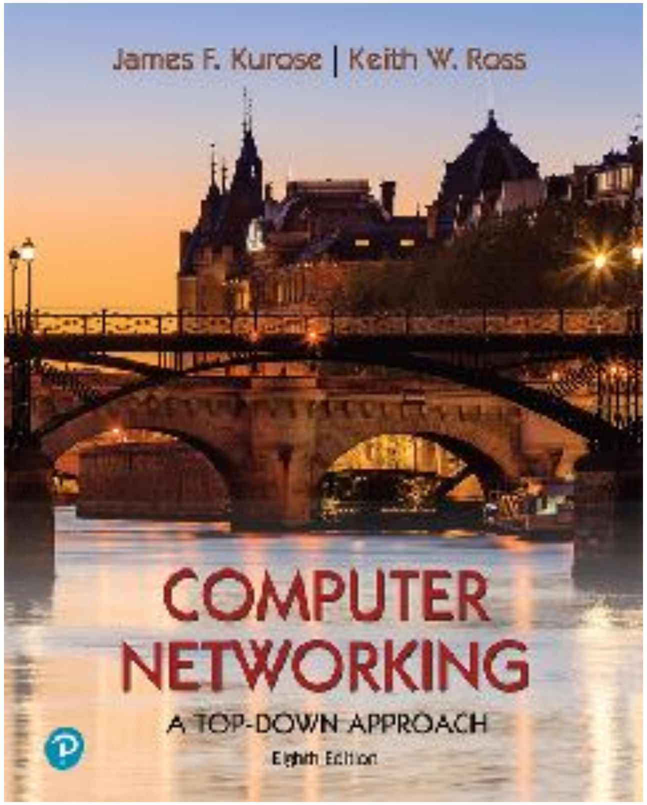
COLUMBIA UNIVERSITY
IN THE CITY OF NEW YORK

Sept 8 admin

- Masks ***required***, over nose and mouth
- Attendance & participation ***not*** required
- If you are not feeling well or were exposed to COVID, please stay home
- Videos of lectures available
  (live Zoom potentially available, but no live Q&A)

- Get book: Kurose/Ross 8th edition
  - See syllabus for some options
- Get & read syllabus and course policies: definitive version linked from Courseworks
  - Please follow syllabus instructions for contacting us
- Reminder: Sign up for Ed Discussion & CourseWorks (if you weren't automatically added)

Chapter 1
Introduction



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

Chapter 1: introduction

our goal:

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

overview: *Next*

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

Chapter 1: roadmap

Today's class

1.1 what *is* the Internet?
1.2 network edge
- end systems, access networks, links

1.3 network core
- network structure, packet switching, circuit switching

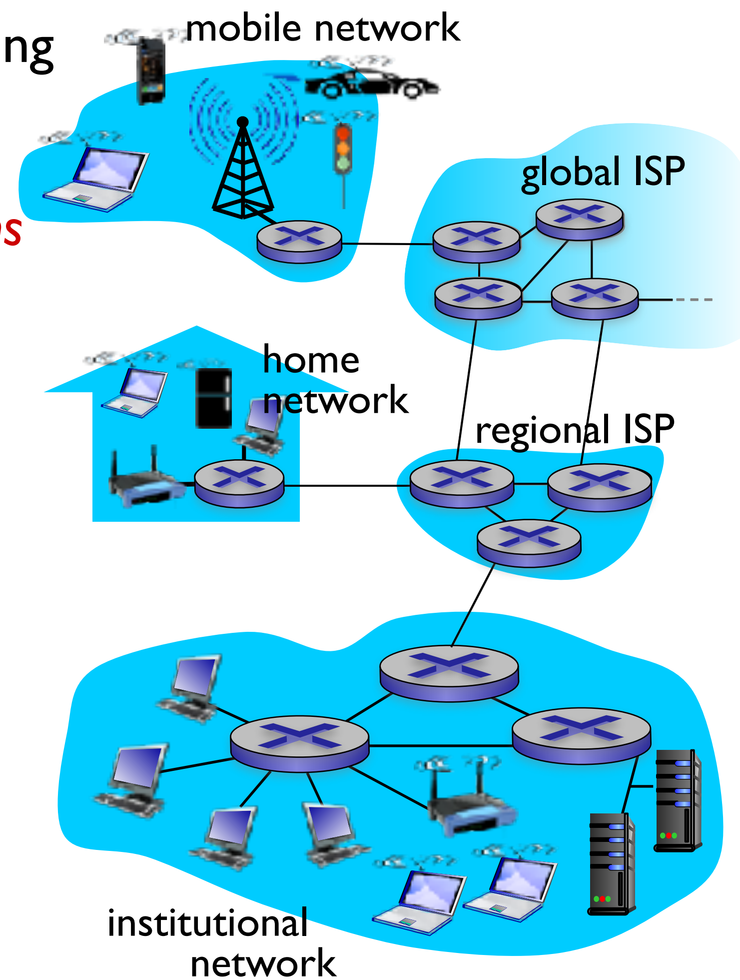
1.4 delay, loss, throughput in networks
1.5 protocol layers, service models
1.6 networks under attack: security
1.7 history

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

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

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

- *packet switches:* forward packets (chunks of data). Two types:
  - *(network-layer) routers*
  - *link-layer switches*



Other Internet-connected devices

Amazon Echo

IP picture frame

Web-enabled toaster +
weather forecaster

Pacemaker & Monitor

bikes

cars

scooters

Internet
refrigerator

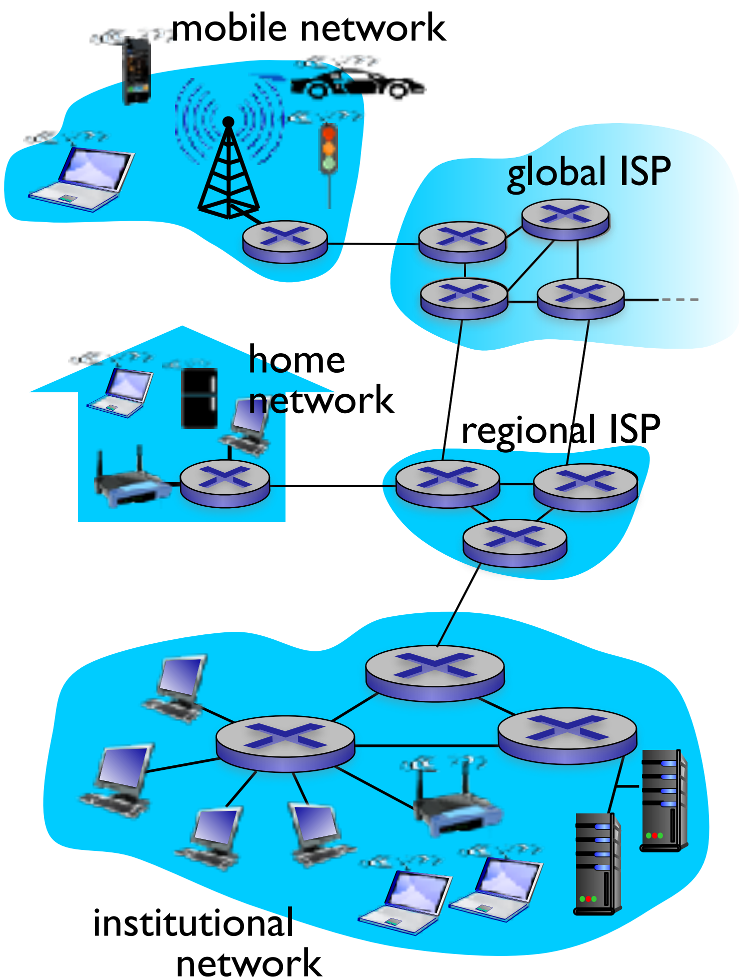
Slingbox: watch,
control cable TV remotely

Security Camera

Internet phones

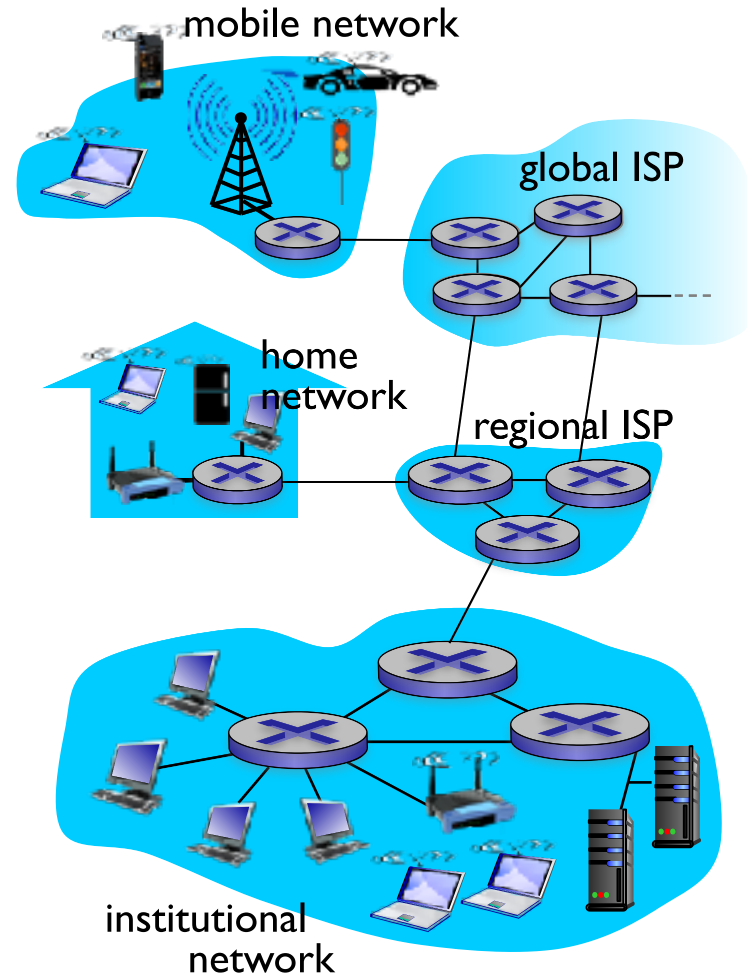
What's the Internet: a service view

- *infrastructure that provides services to applications:*
  - Web, VoIP, email, games, e-commerce, social nets, …
- *provides programming interface to applications*
  - hooks that allow sending and receiving applications to "connect" to Internet
  - provides service options, analogous to postal service
    - reliable data delivery from source to destination
    - "best effort" (unreliable) data delivery



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



What's a protocol?

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

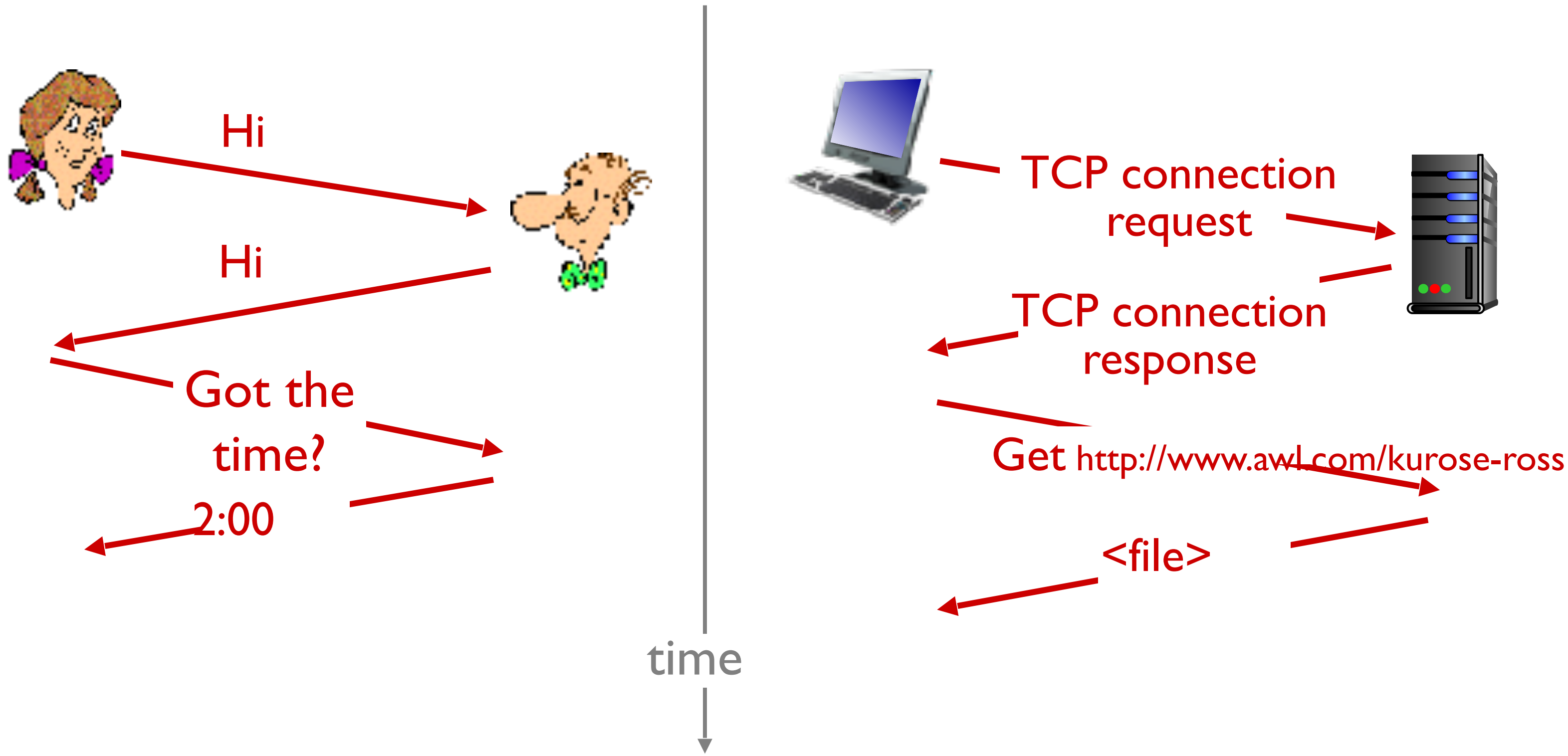
network protocols:

- 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:



How are protocols defined?

- *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

10s of 1000s of independent networks, many directly competing with each other…

that need to send compatible messages for the Internet to work:
"Be conservative in what you send and liberal in what you accept"
— Jon Postel

IETF defines these protocols:
"We reject: kings, presidents and voting. We believe in: rough consensus and running code."
— Dave Clark

Chapter 1: roadmap

1.1 what *is* the Internet?

1.2 network edge
- end systems, access networks, links

1.3 network core
- network structure, packet switching, circuit switching

1.4 delay, loss, throughput in networks

1.5 protocol layers, service models

1.6 networks under attack: security

1.7 history

Chapter 1: roadmap

1.1 what *is* the Internet? *Next*

1.2 network edge
- end systems, access networks, links

1.3 network core
- network structure, packet switching, circuit switching

1.4 delay, loss, throughput in networks

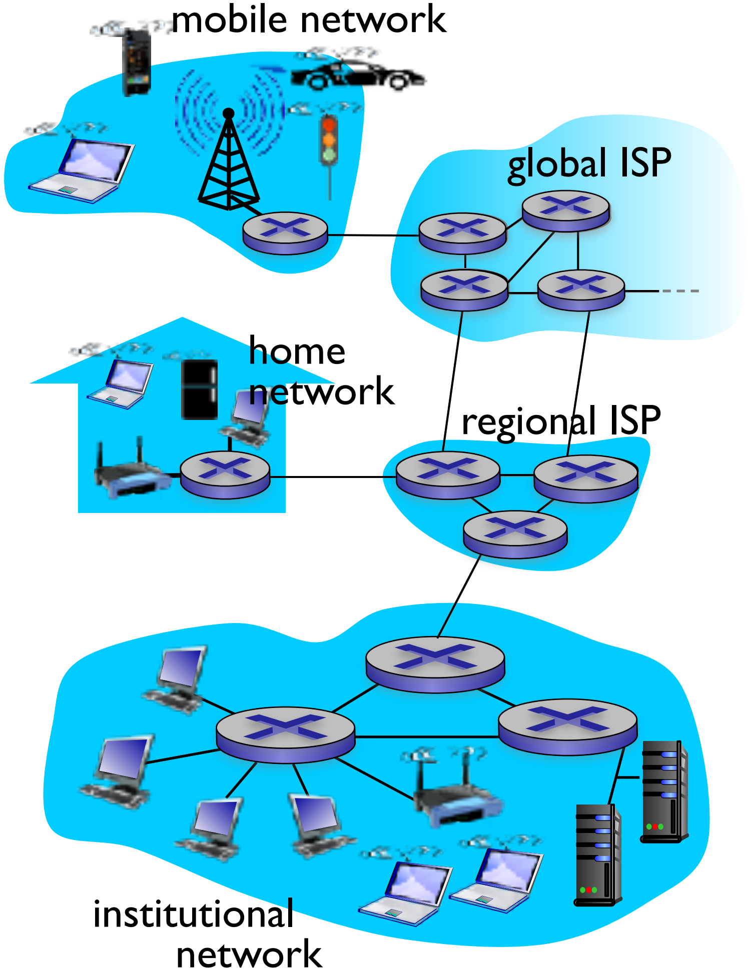
1.5 protocol layers, service models

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:*

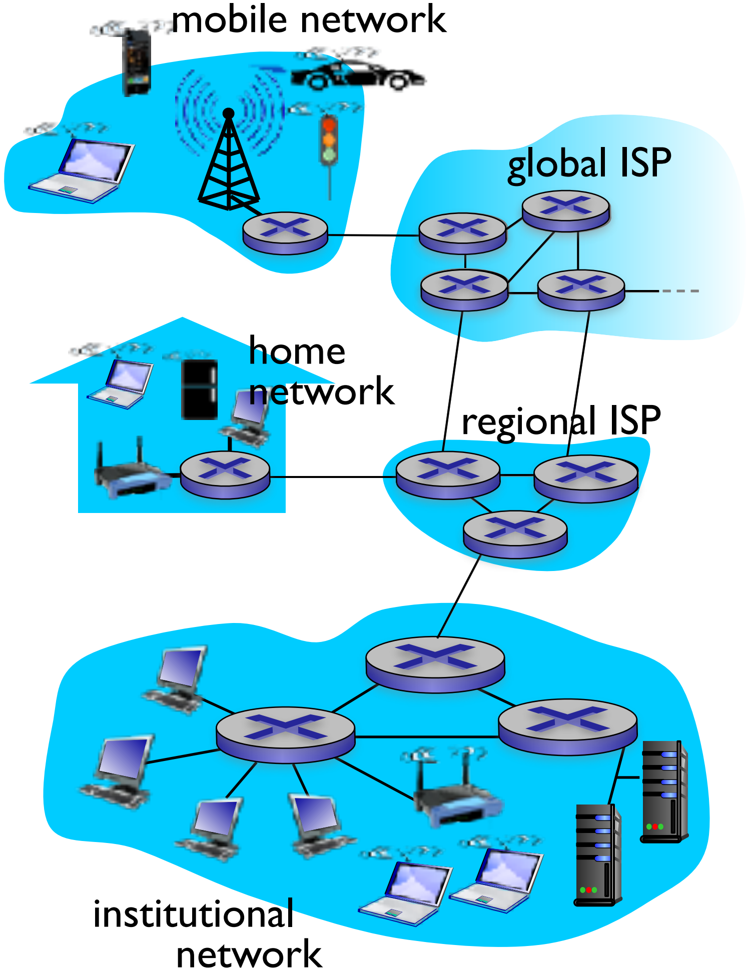
- *network core:*

Data center

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



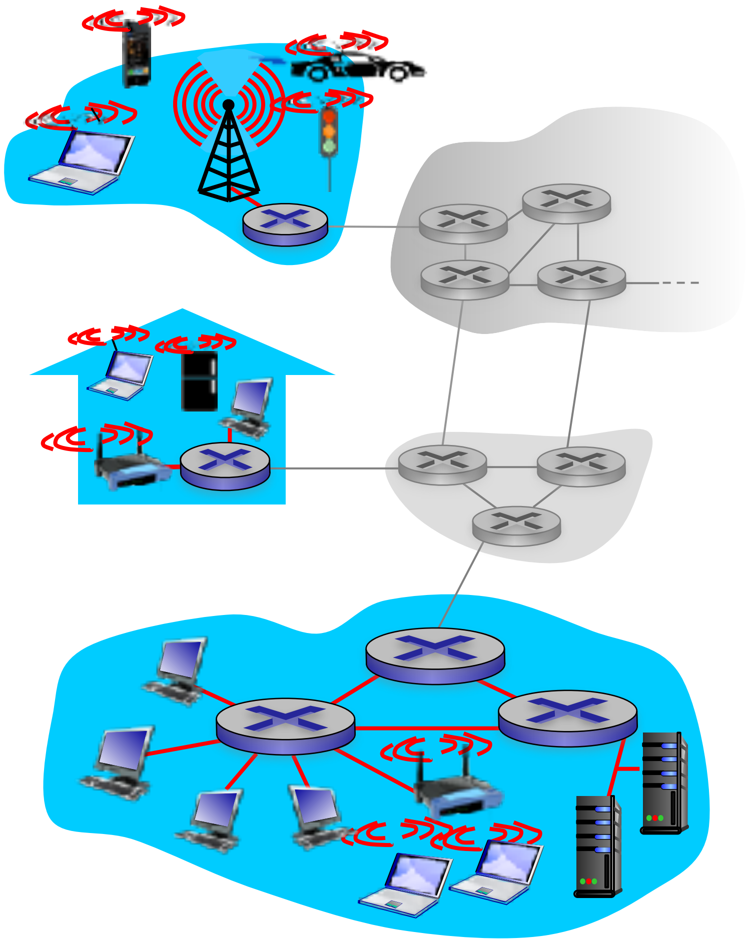
Access networks and physical media

Q: How to connect end systems to edge router?

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

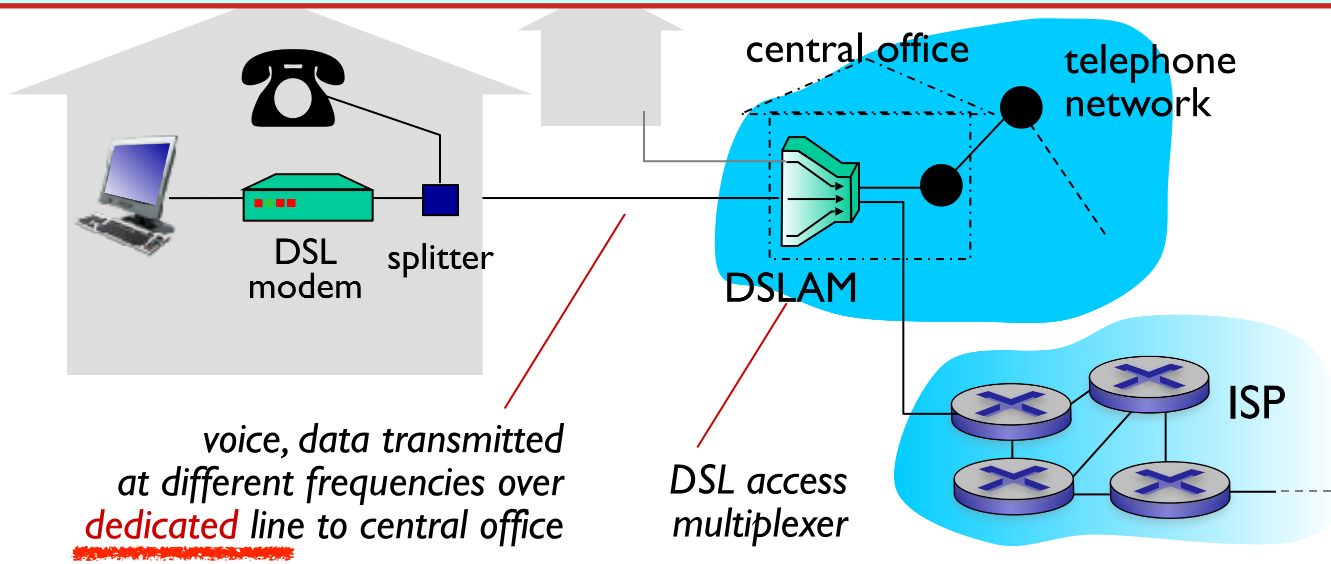
keep in mind:

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



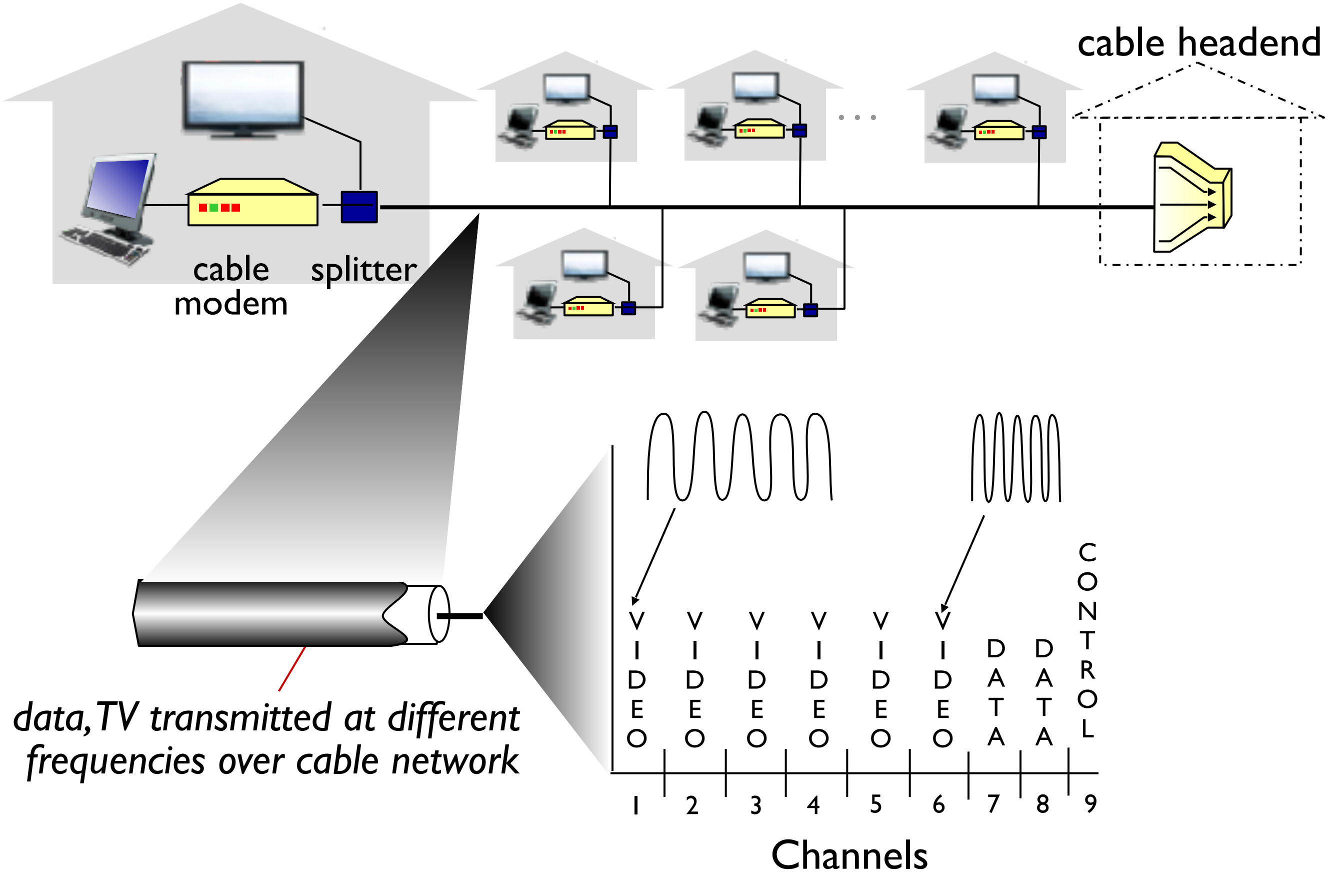
Access network: digital subscriber line (DSL)

Key networking challenge (course will come back to this):
How to support new applications on top of deployed technologies?



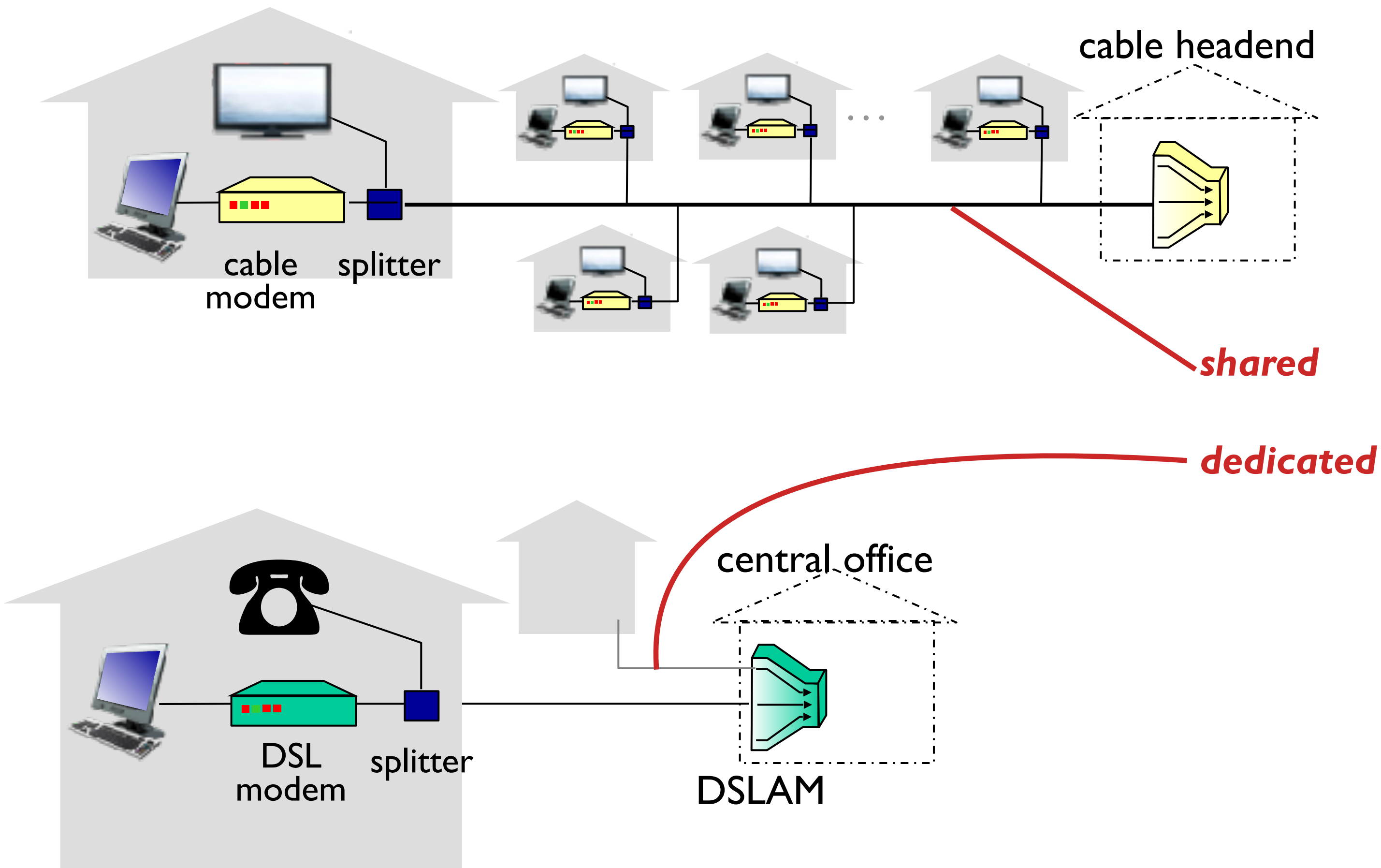
- use *existing* telephone line to central office DSLAM
  - data over DSL phone line goes to Internet
  - voice over DSL phone line goes to telephone net
- < 16 Mbps upstream transmission rate (typically < 1 Mbps)
- < 52 Mbps downstream transmission rate (typically < 10 Mbps)
- Also a two-way telephone channel
- Upstream/downstream/telephone channels use different frequencies

Access network: cable network



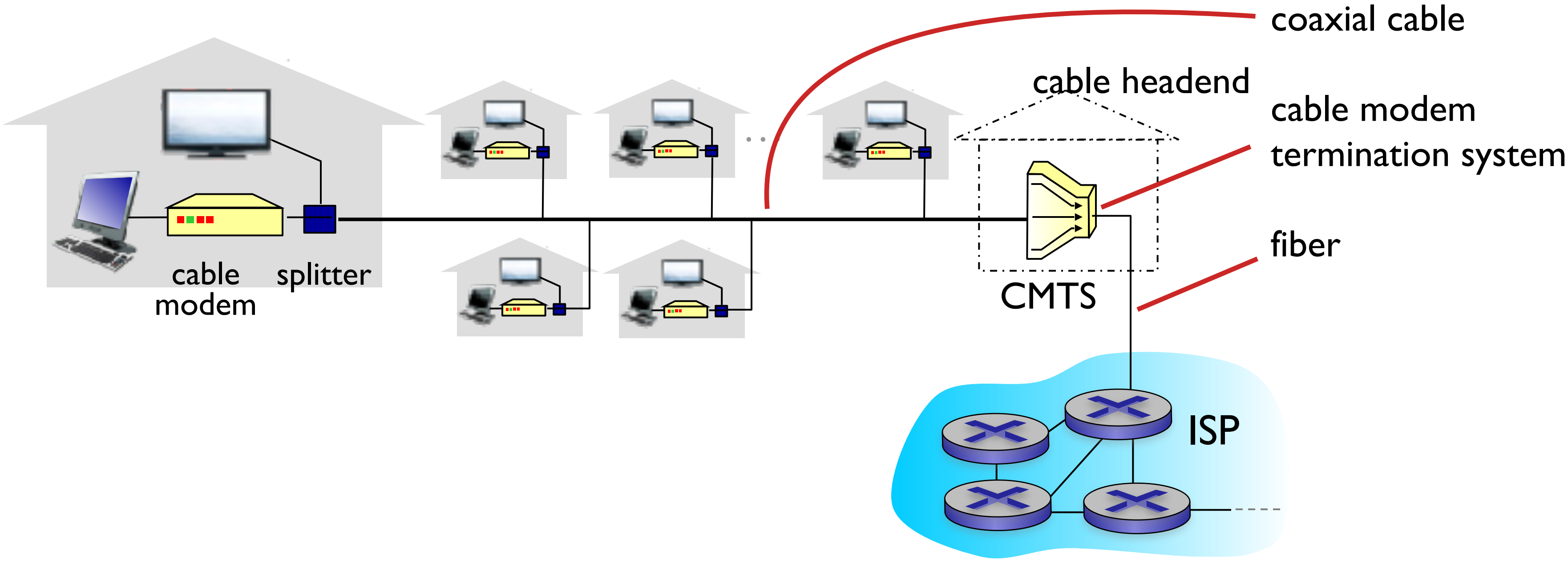
frequency division multiplexing: different channels transmitted in different frequency bands

Cable vs DSL: what's the major difference?



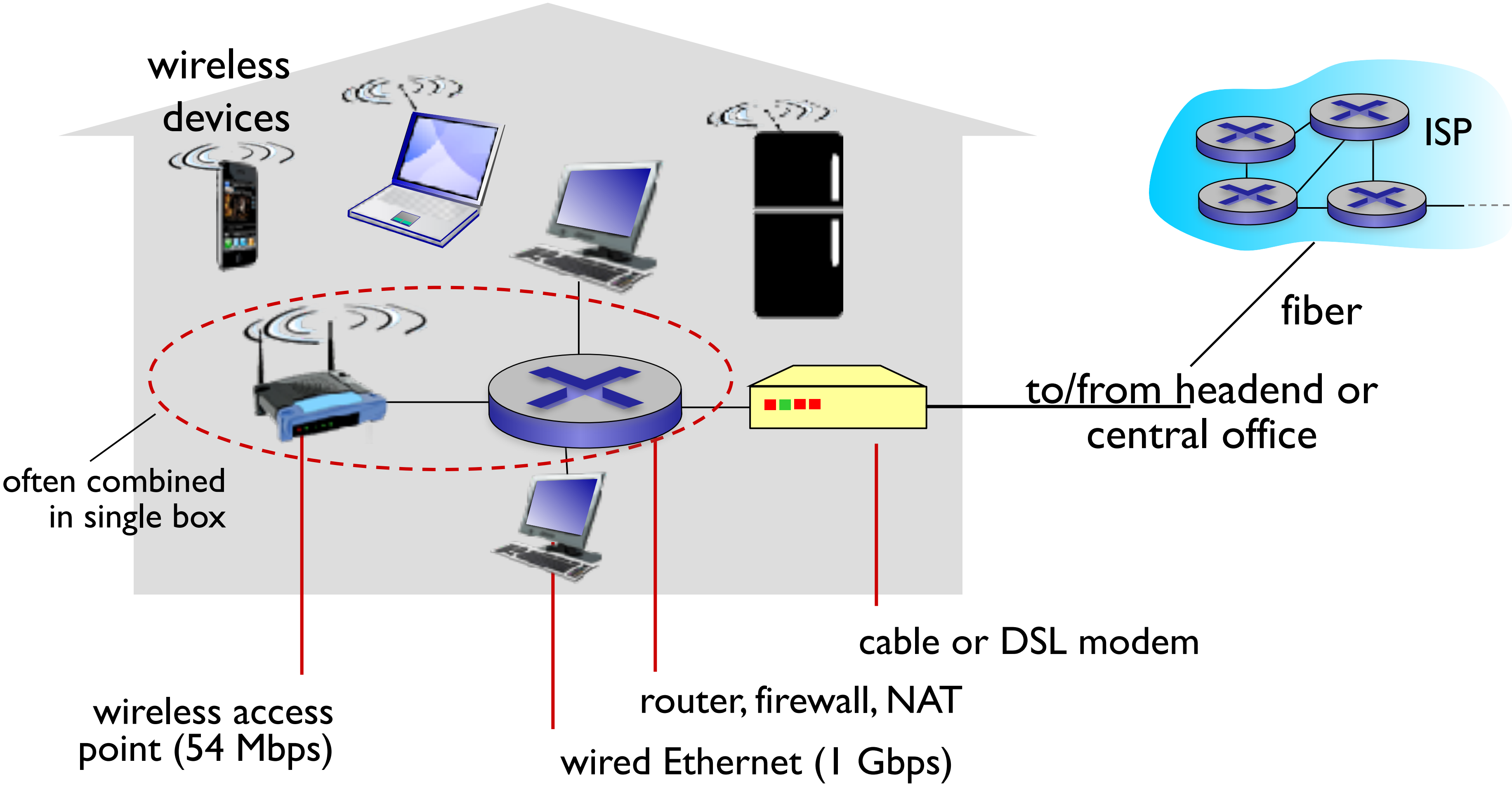
Key networking challenge (class will come back to this):
How to efficiently share a common medium?

Access network: cable network



- 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
- HFC: hybrid fiber coax
  - CMTS controls access of cable modems to shared coax cable segment
  - asymmetric: up to ~~42.8 Mbps~~ 1.2 Gbps downstream transmission rate, 100 ~~30.7~~ Mbps upstream transmission rate

Access network: home network



Key networking challenge (class will come back to this):
How to connect different types of networks?

Wireless access networks

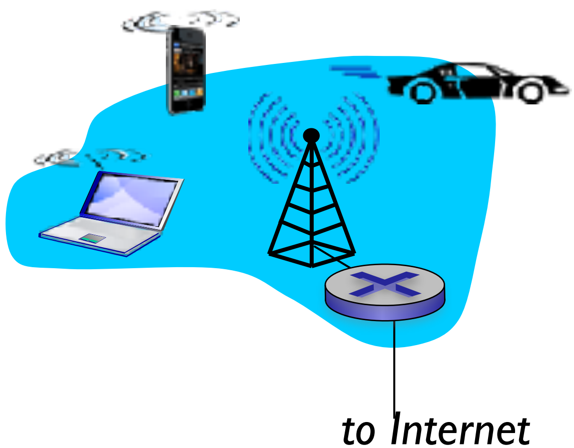
- shared *wireless* access network connects end system to router
  - via base station aka "access point"

wireless LANs:

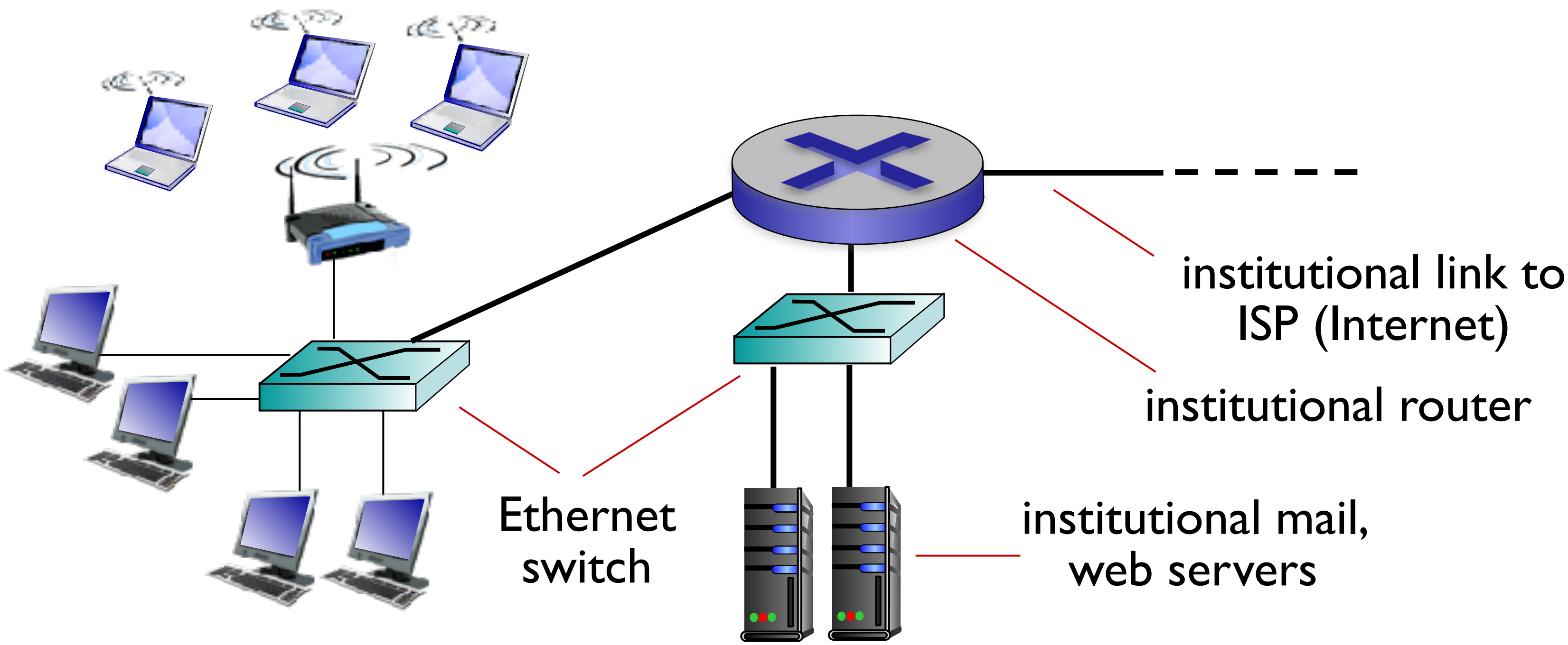
- within building (100 ft.)
- 802.11b/g/n (WiFi): 11, 54, 450 Mbps transmission rate

wide-area wireless access

- provided by telco (cellular) operator, 10's km
- up to100s of Mbps
- 3G, 4G: LTE, 5G



Enterprise access networks

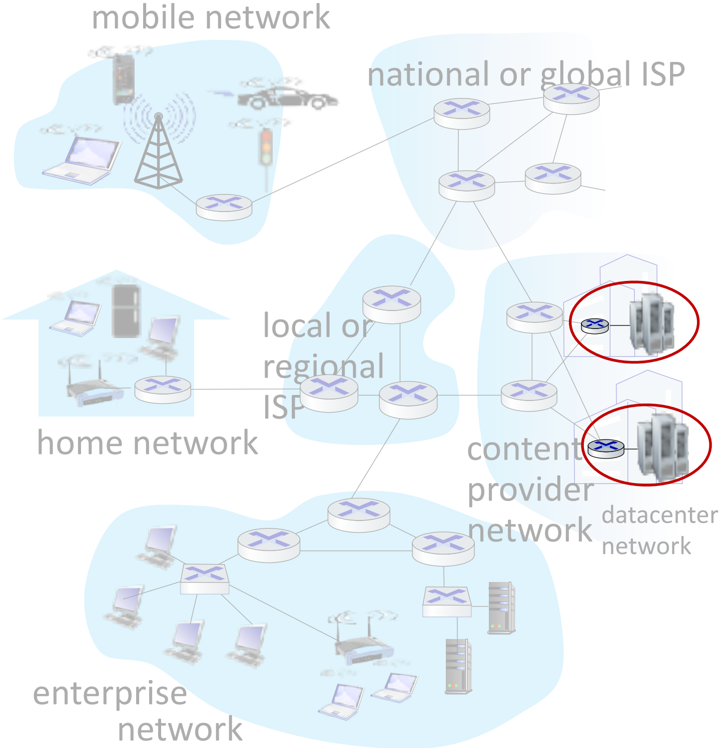


- companies, universities, etc.
- mix of wired, wireless link technologies, connecting a mix of switches and routers (we'll cover differences shortly)
  - Ethernet: wired access at 100Mbps, 1Gbps, 10Gbps
  - WiFi: wireless access points at 11, 54, 450 Mbps

Access networks: data center networks

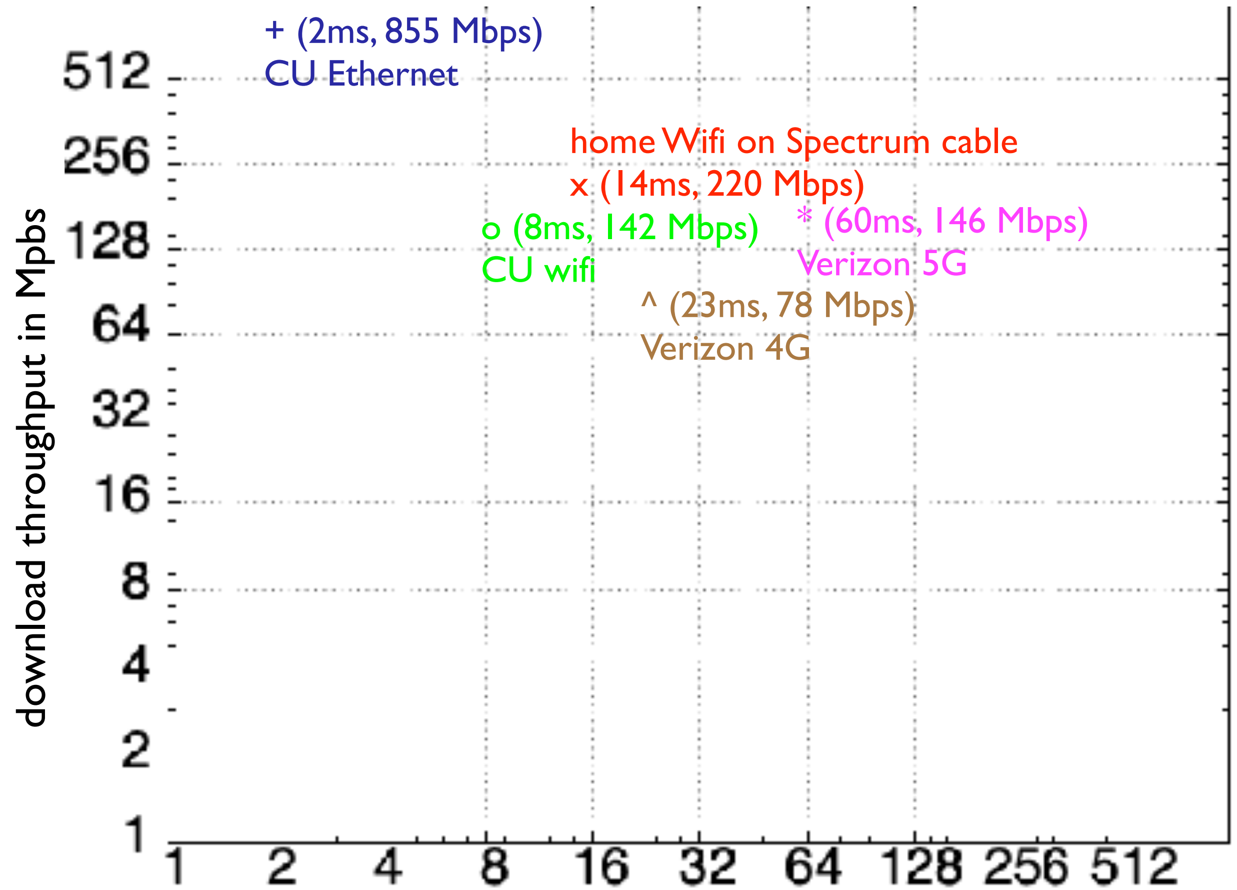
- high-bandwidth links (10s to 100s Gbps) connect hundreds to thousands of servers together, and to Internet

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



Ethan's Internet speeds?

(all except Ethernet measured from phone)

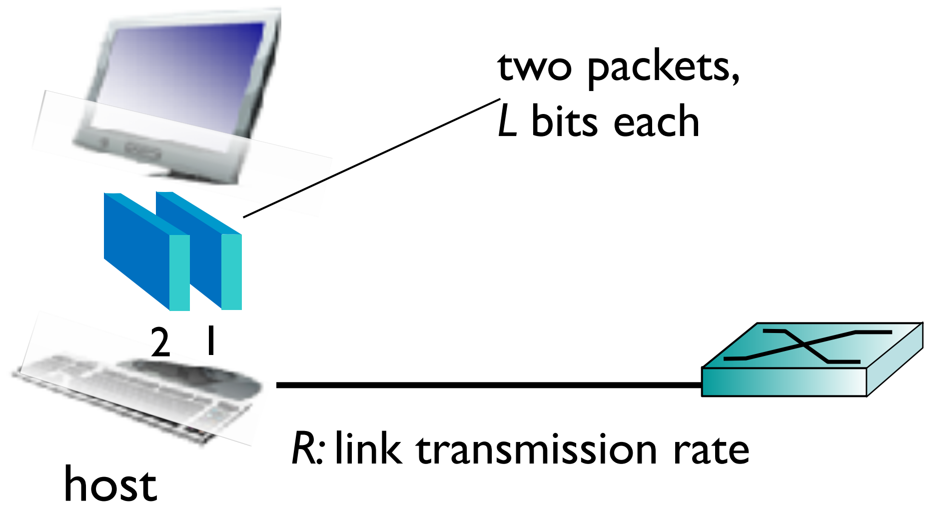


ping latency in ms

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*
  - aka link *capacity*, aka link *bandwidth*



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

one-hop numerical example:

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

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

Physical media: twisted pair, coax, fiber

twisted pair (TP)

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

coaxial cable:

- two concentric copper conductors
- bidirectional
- broadband:
  - multiple frequency 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 (10's-100's Gbps)
- low error rate:
  - repeaters spaced far apart
  - immune to electromagnetic noise

Physical media: radio

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

radio link types:

- Wireless LAN (e.g.,WiFi)
  - 10-100's Mbps; 10's of meters
- Wide-area (e.g., 4G cellular)
  - 10's Mbps over ~10 Km
- Bluetooth: cable replacement
  - short distances, limited rate
- terrestrial microwave
  - point-to-point; 45 Mbps channels
- satellite
  - Up to 45Mbps channel
  - 270 msec end-end delay
  - geostationary versus low altitude

Key networking challenge (class will come back to this):
How to reliably transmit data across unreliable network?

Chapter 1: roadmap

1.1 what *is* the Internet?

1.2 network edge

- end systems, access networks, links

Next

1.3 network core

- network structure, packet switching, circuit switching

1.4 delay, loss, throughput in networks

1.5 protocol layers, service models

1.6 networks under attack: security

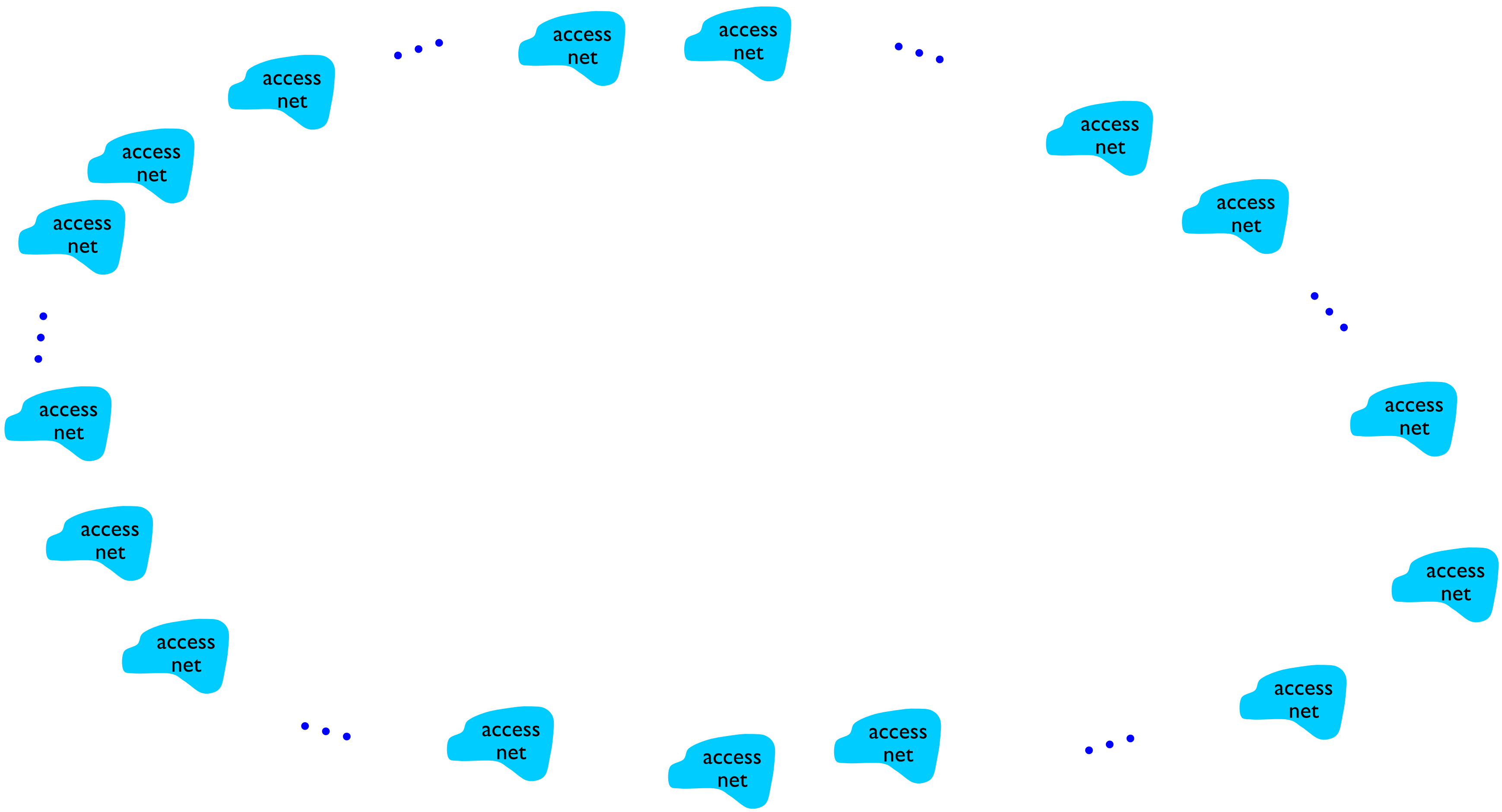
1.7 history

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

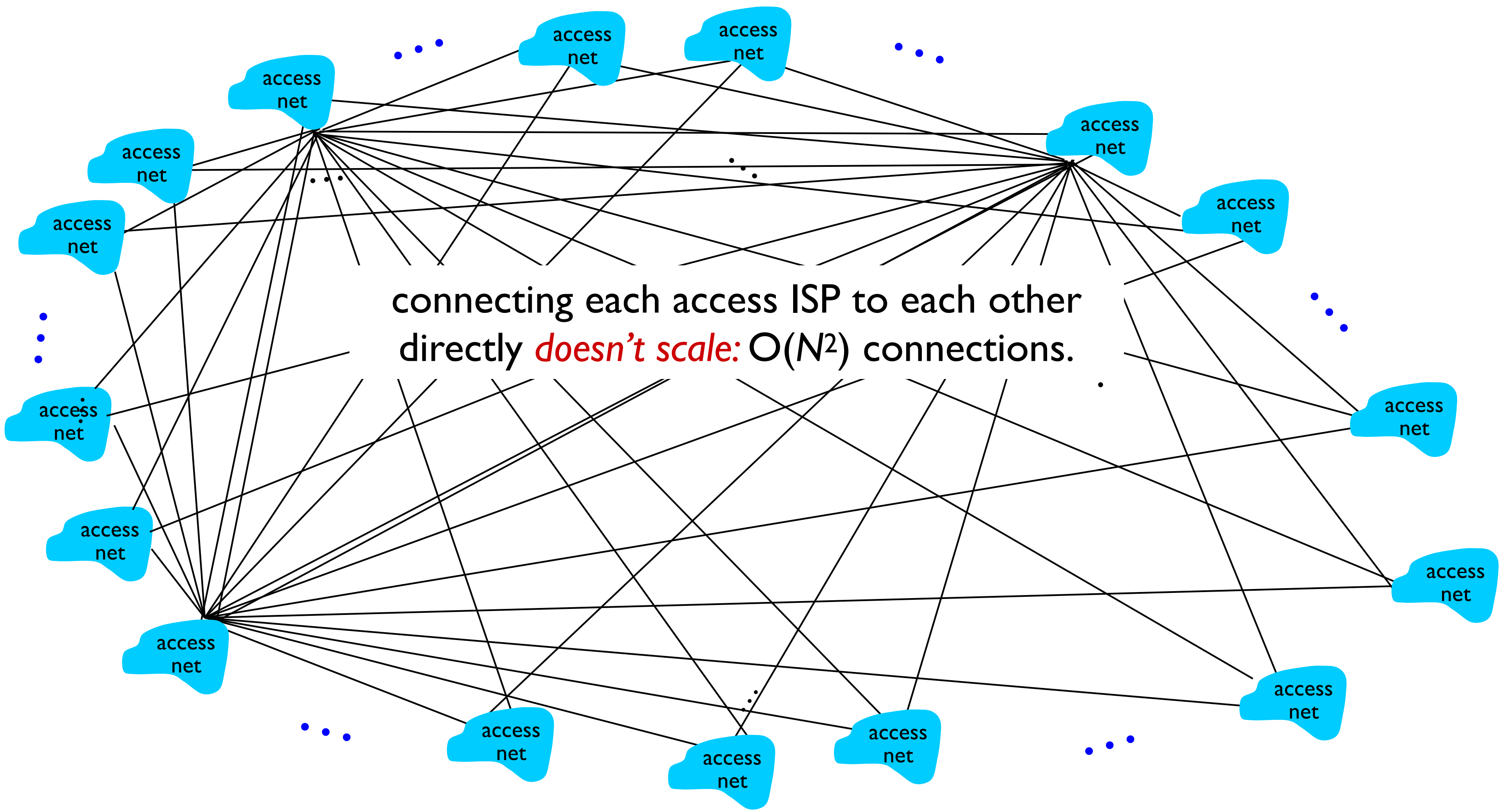
Internet structure: network of networks

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



Internet structure: network of networks

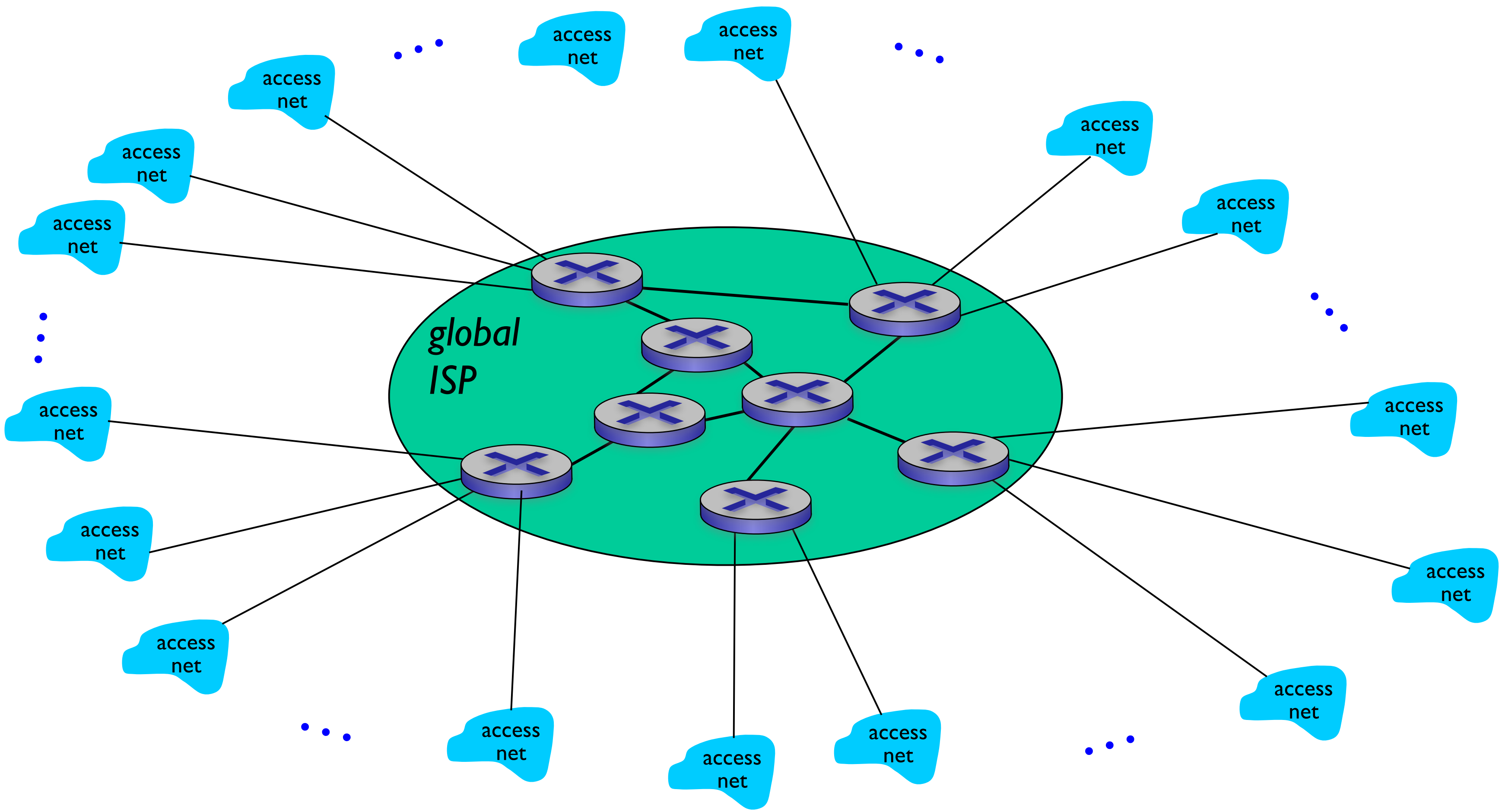
Option: connect each access ISP to every other access ISP?



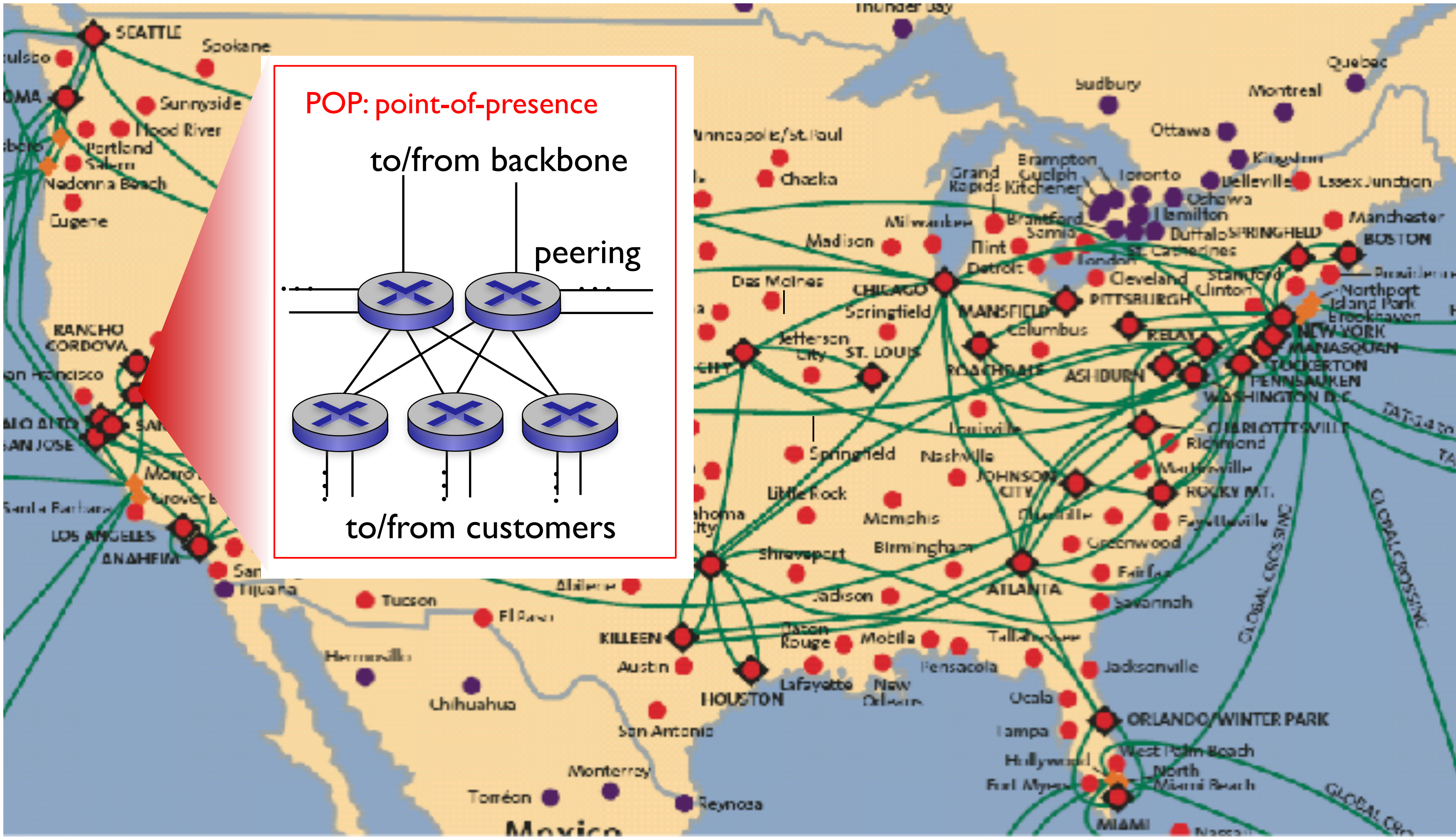
connecting each access ISP to each other directly *doesn't scale:* $O(N^2)$ connections.

Internet structure: network of networks

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

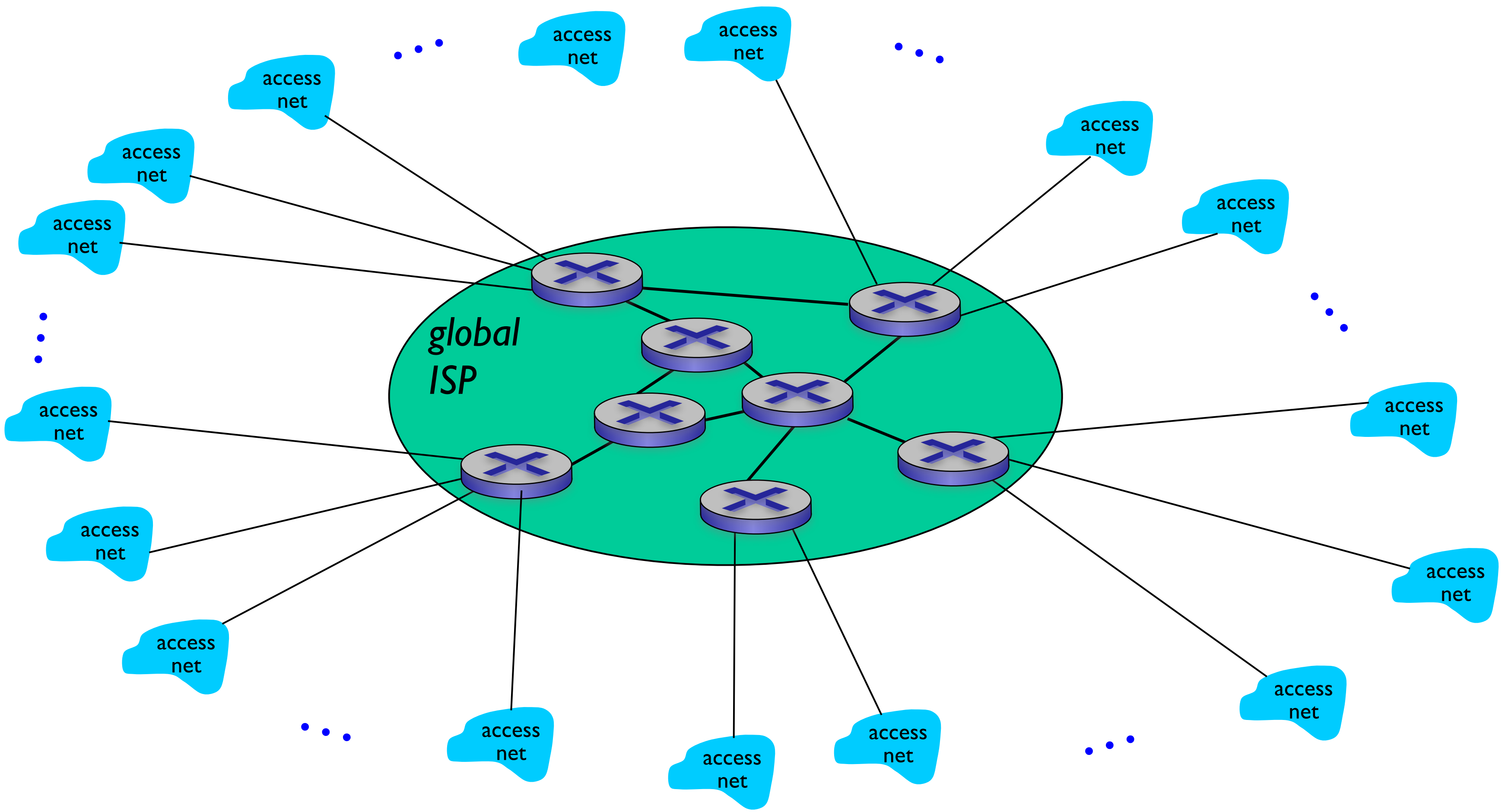


Tier-1 ISP: e.g., Sprint



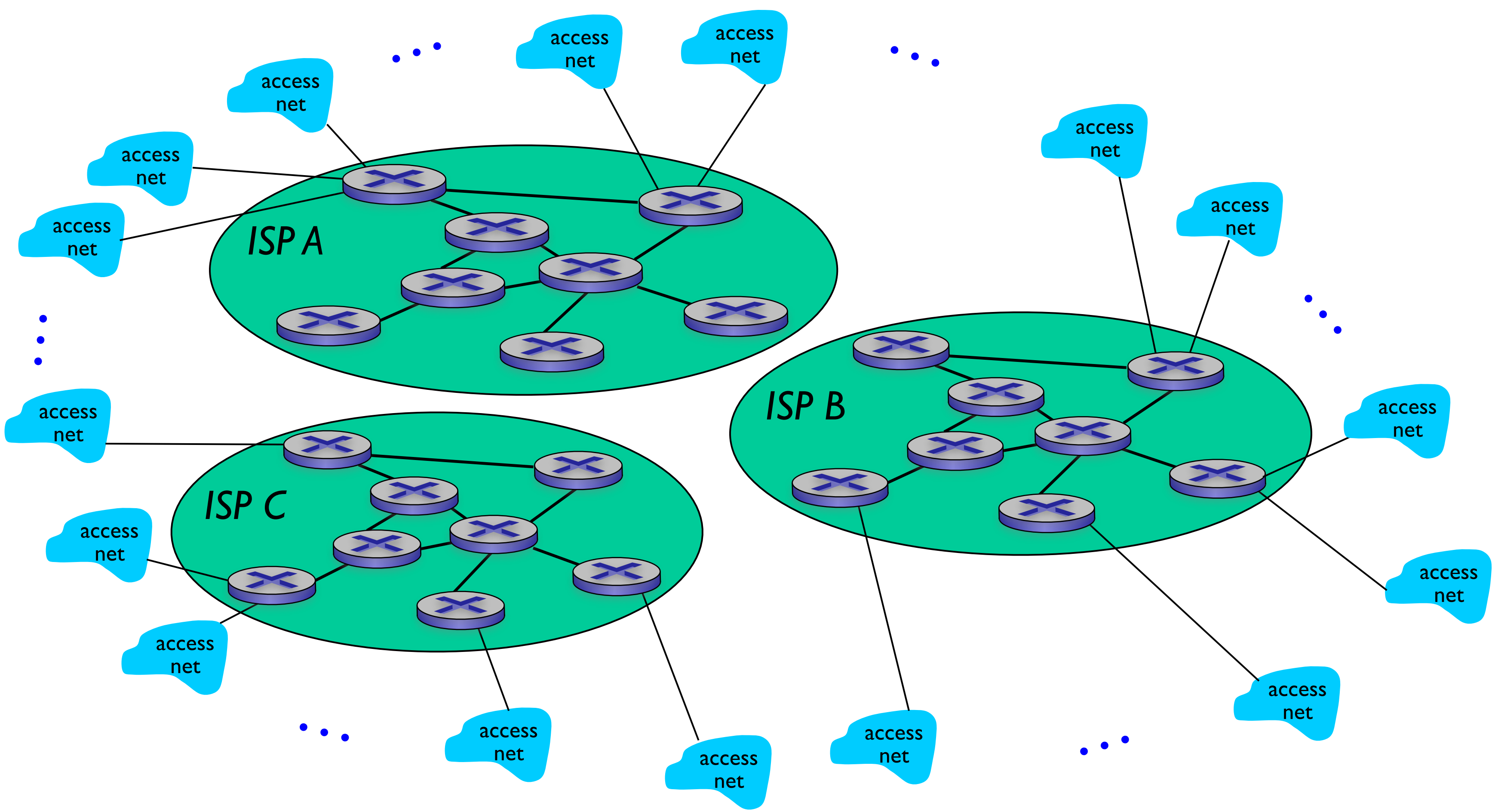
Internet structure: network of networks

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



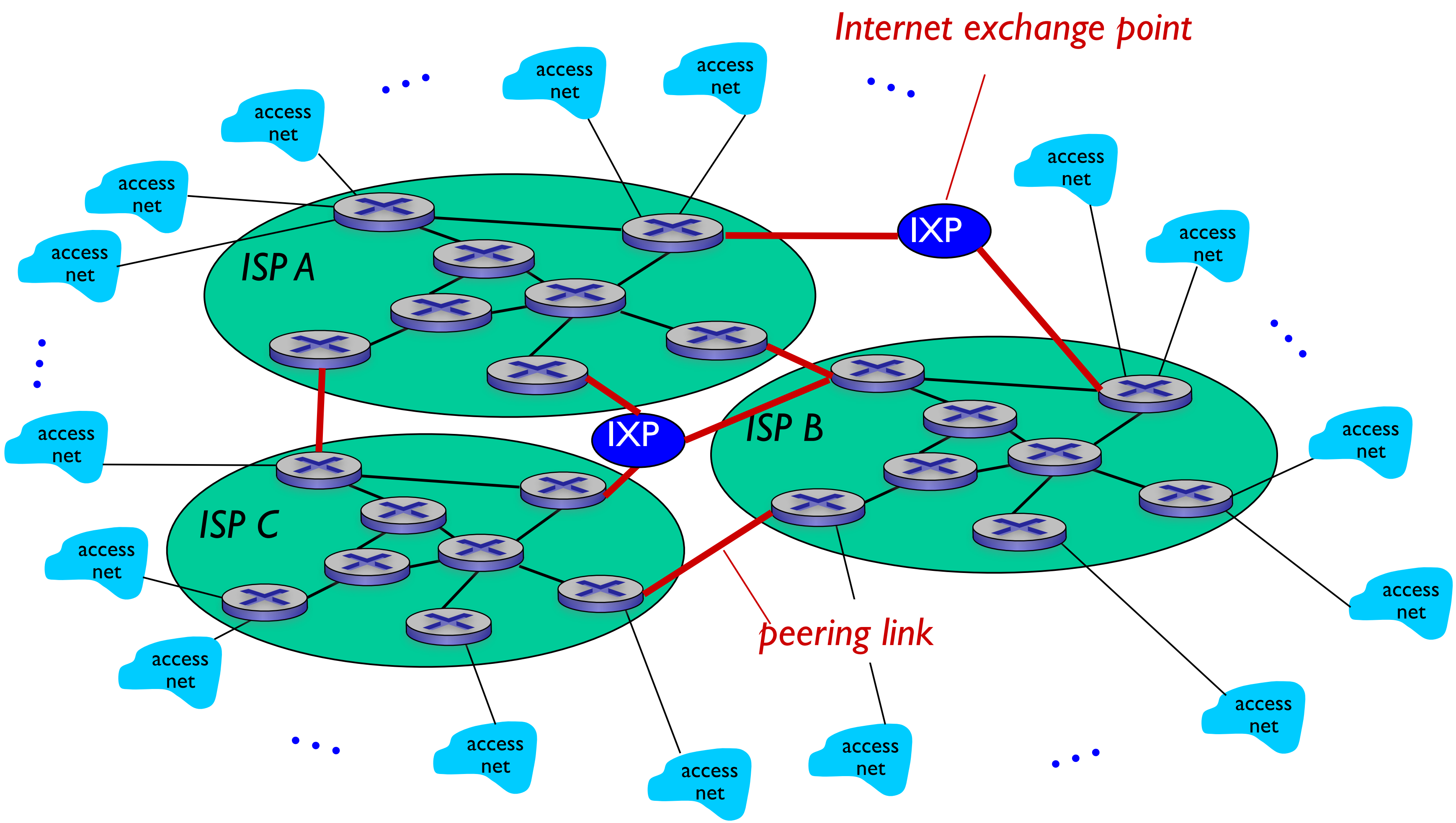
Internet structure: network of networks

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



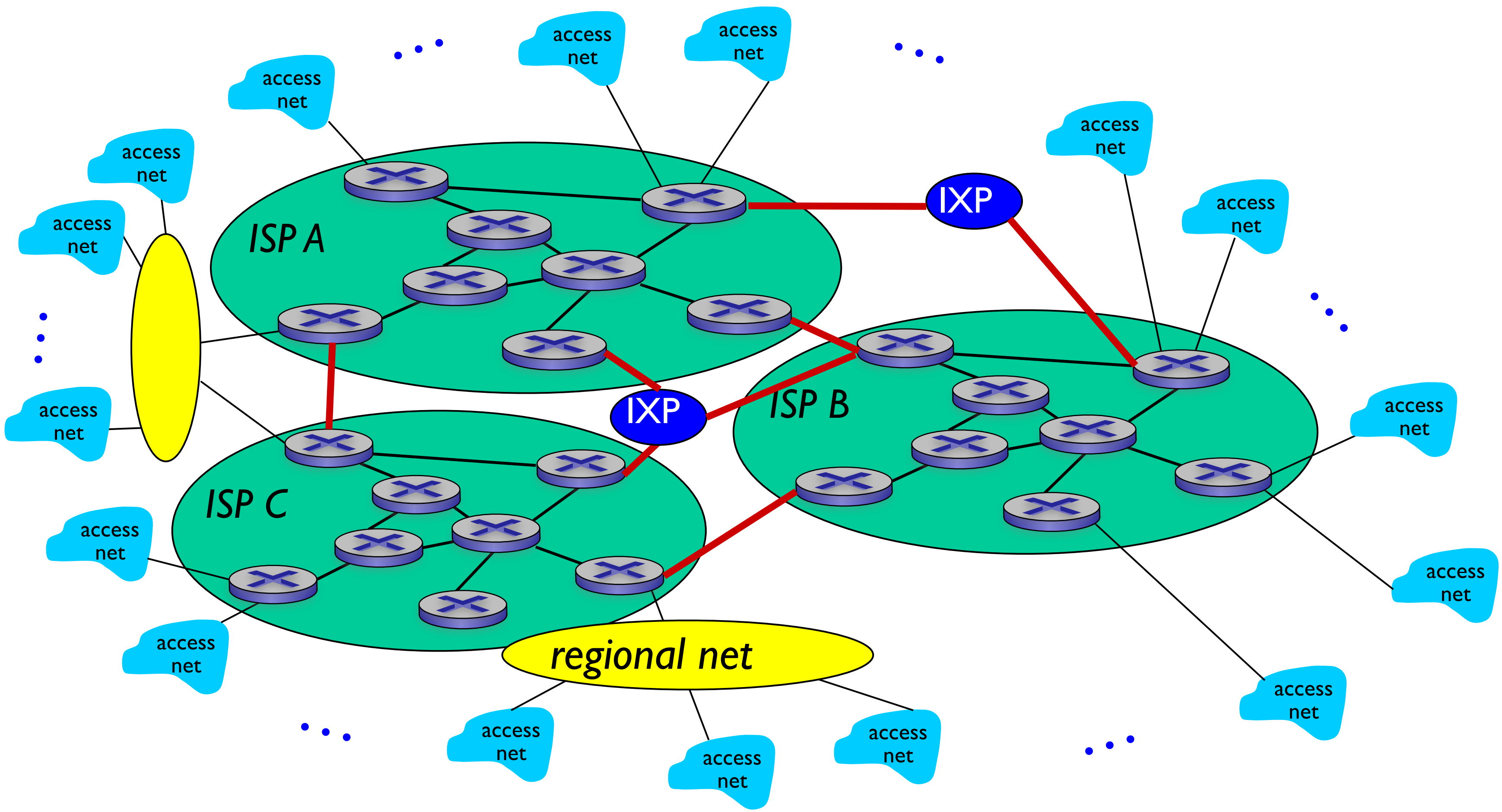
Internet structure: network of networks

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



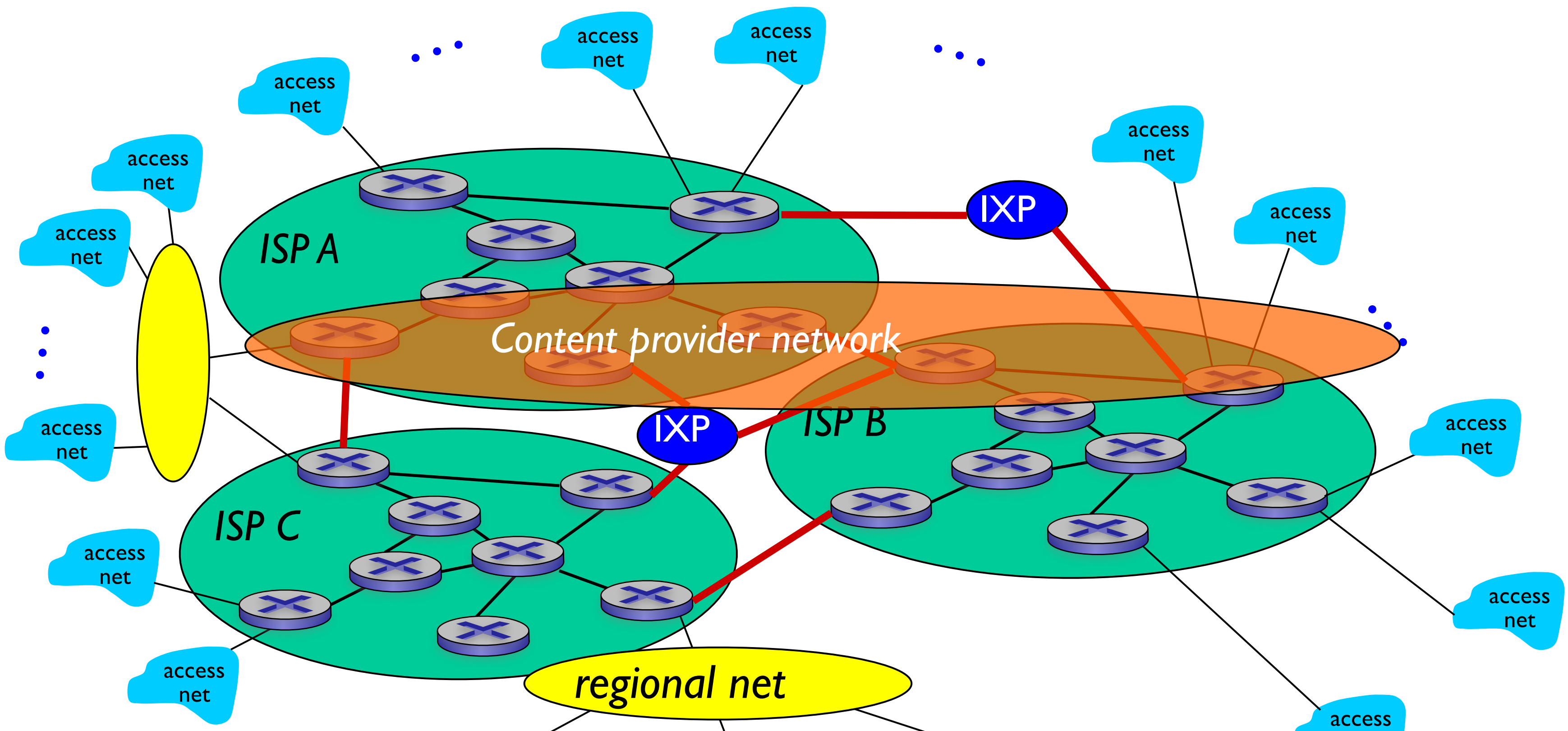
Internet structure: network of networks

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



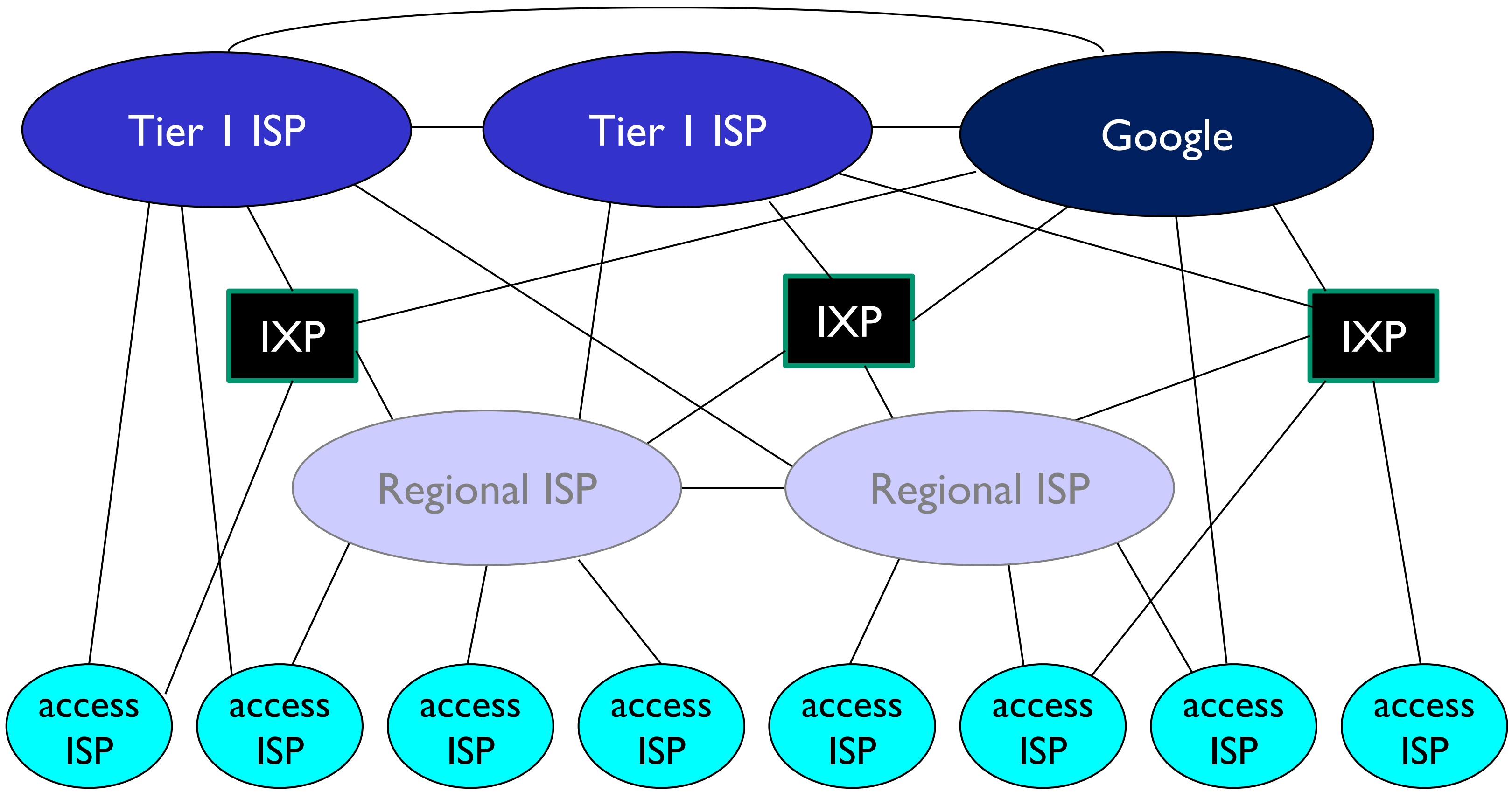
Internet structure: network of networks

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



Key networking challenge (we'll come back to this):
How to discover and access remote hosts and services?

Internet structure: network of networks



- at center: small # of well-connected large networks
  - "tier-1" commercial ISPs (e.g., Level 3, Sprint, AT&T, NTT), with national & international coverage
  - content provider network (e.g., Google): private network that connects its servers to Internet, often bypassing tier-1, regional ISPs

Chapter 1: roadmap

1.1 what *is* the Internet?

1.2 network edge

- end systems, access networks, links

Next

1.3 network core

- network structure, packet switching, circuit switching

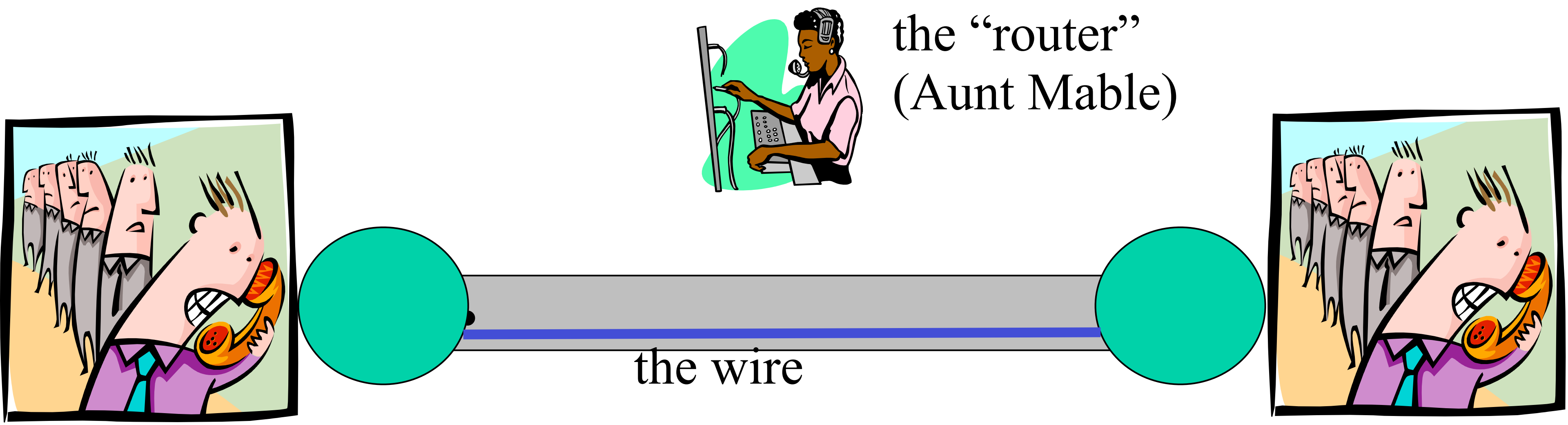
1.4 delay, loss, throughput in networks

1.5 protocol layers, service models

1.6 networks under attack: security

1.7 history

Back in the Old Days...

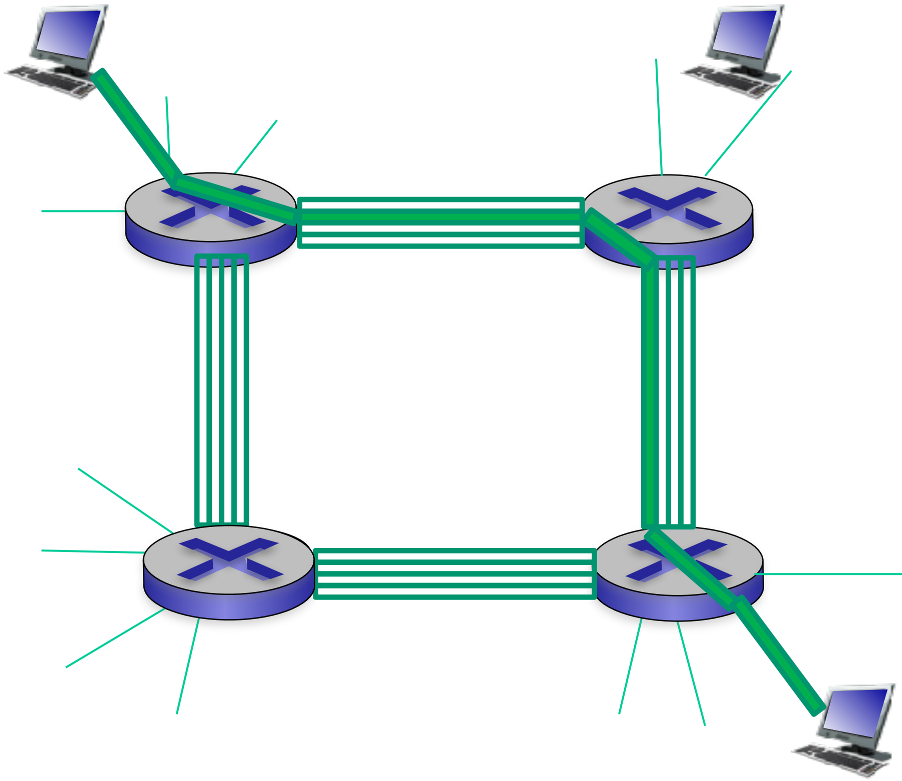


1920s telephony: *circuits*---a physical wire from one end to the other

Alternative (compared to Internet, which we'll discuss next) 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 while reserved but not used by call *(no sharing)*
- commonly used in traditional telephone networks



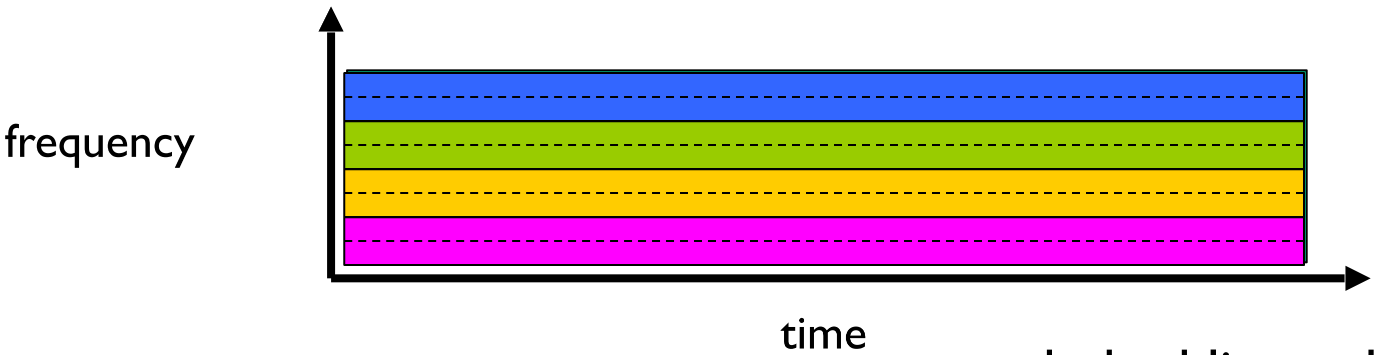
How can we use circuit switching but still support more simultaneous users than we have physical circuits?

Circuit switching: FDM versus TDM

Example:

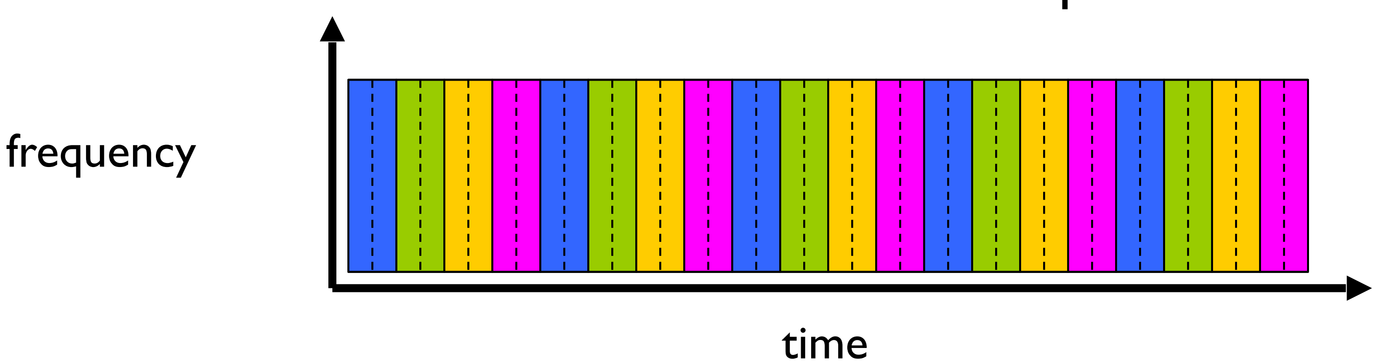
4 users

FDM (Frequency Division Multiplexing)

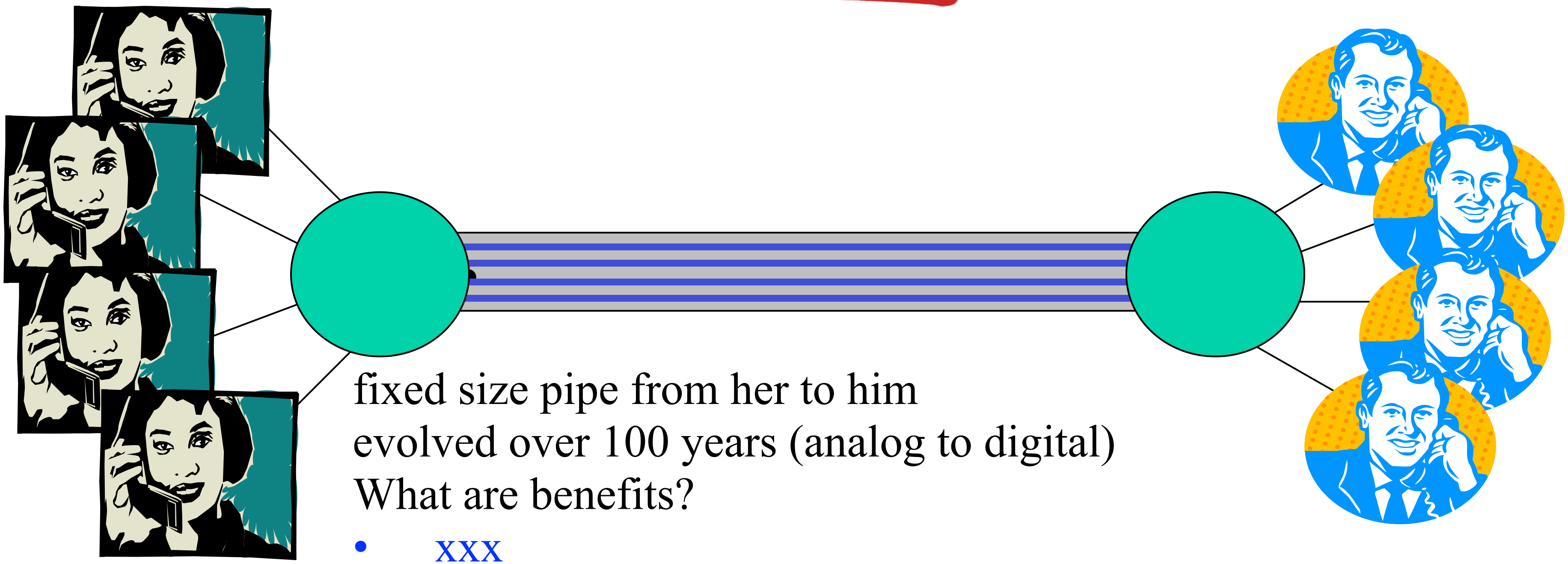


dashed line: subdivide for upstream vs downstream

TDM (Time Division Multiplexing)



Logical Network View

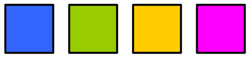


fixed size pipe from her to him
evolved over 100 years (analog to digital)
What are benefits?

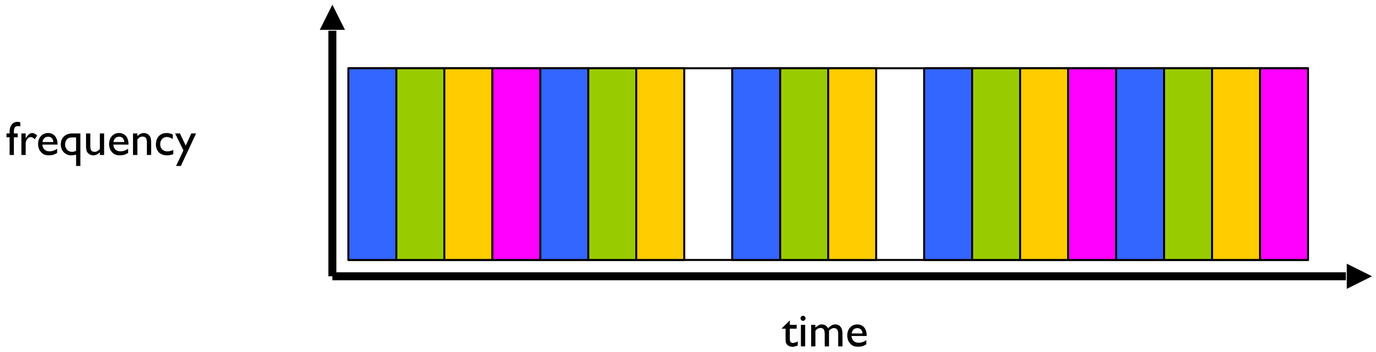
- xxx

Circuit switching: Silent periods -> wasteful

Example:

4 users 

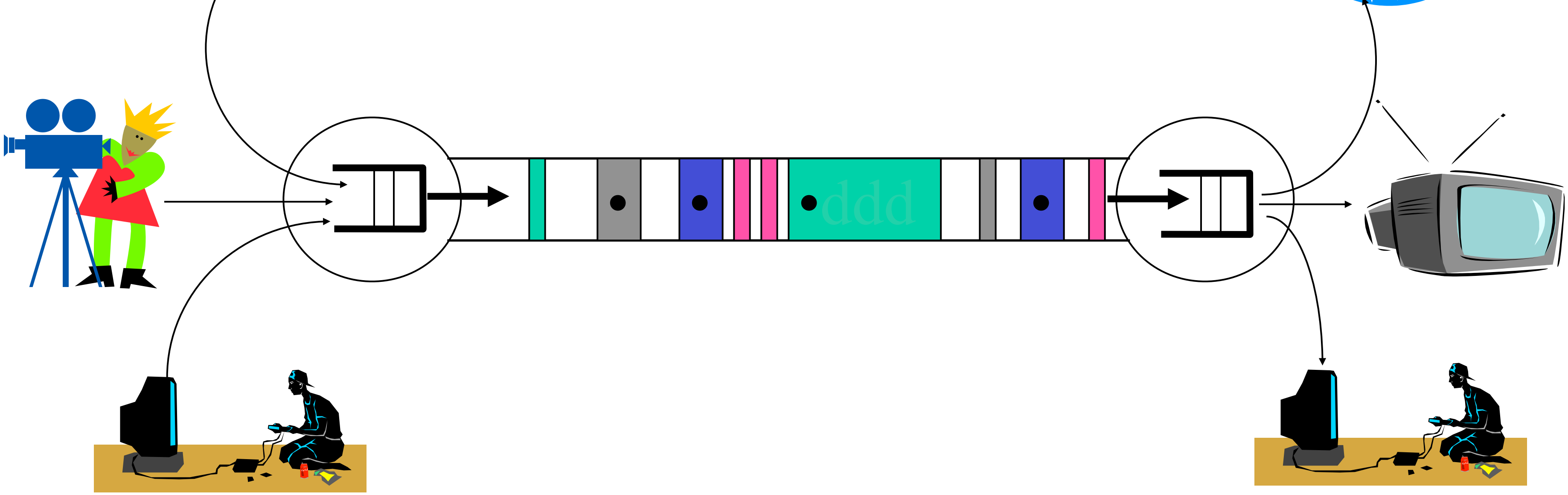
TDM (Time Division Multiplexing)



Packet Switching (Internet)

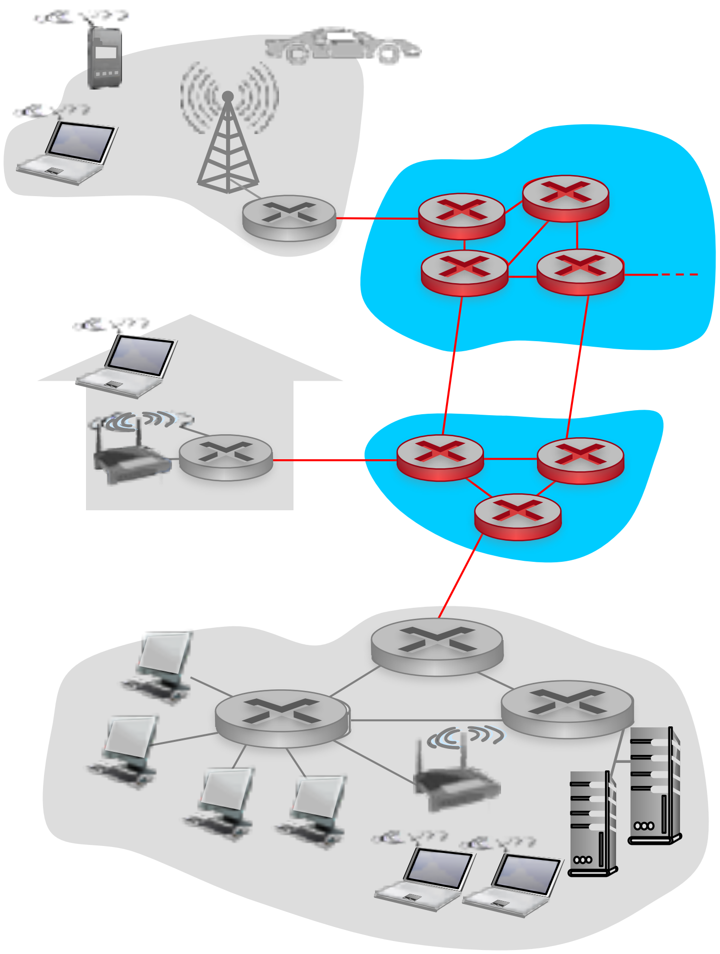
differences:

- *packets* as low-level component
- *multiple* kinds of traffic



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
    - after transmission, link is available for other packets (including from other sources)

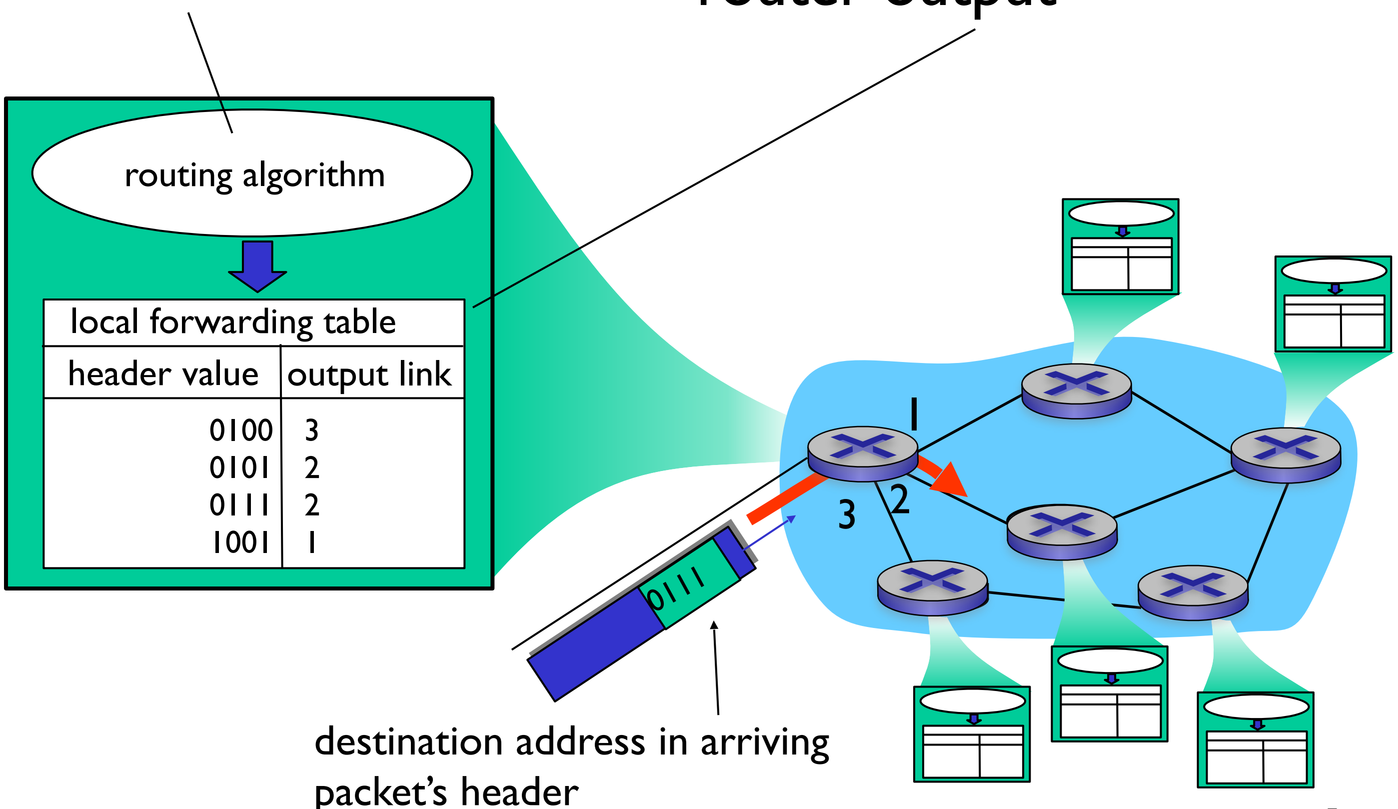


Two key network-core functions

Key networking challenge (class will come back to this):
How to learn consistent, quality routes in dynamic, distributed setting?

routing: determines source-destination route taken by packets
- *routing algorithms*

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

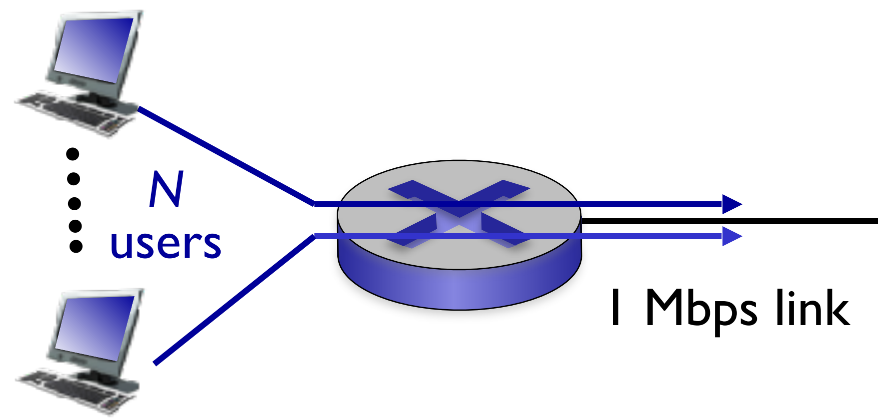


Packet switching versus circuit switching

packet switching allows more users to use network!

example:

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



- *How many users can circuit-switching support?*
  -
- *How many users can packet switching support?*
  -

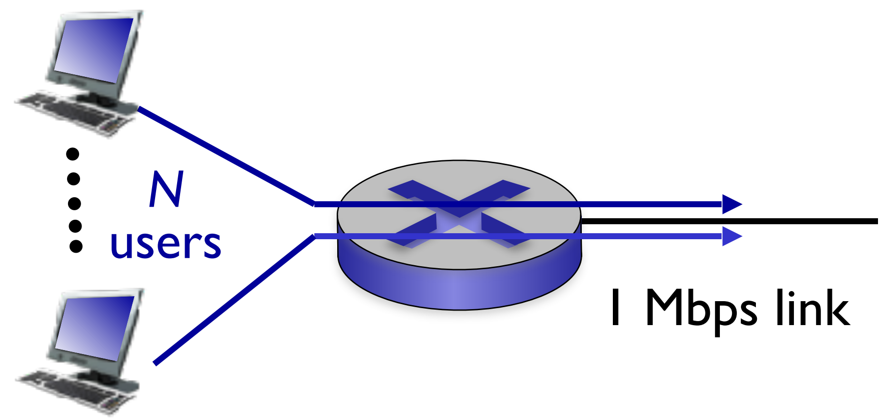
* Check out the online interactive exercises for more examples: http://gaia.cs.umass.edu/kurose_ross/interactive/

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
- *How many users can circuit-switching support?*
  - 10 users, 100% reliable
- *How many users can packet switching support?*
  - with 100% reliability, 10
  - with 99.9+% reliability, 38
  - with 35 users, probability > 10 active at same time is less than .0004



Q: how did we get value 0.0004?*

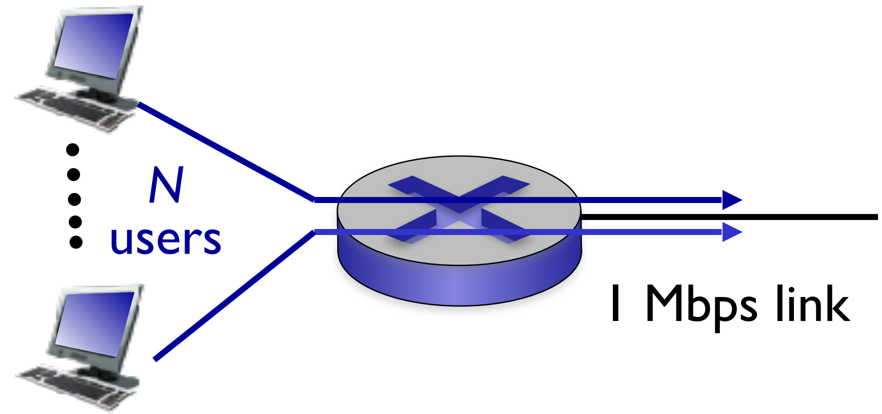
* Check out the online interactive exercises for more examples: http://gaia.cs.umass.edu/kurose_ross/interactive/

Packet switching versus circuit switching

- 1 Mb/s link
- each user:
  - 100 kb/s when "active"
  - active 10% of time
- *How many users can packet switching support?*
  - with 100% reliability, 10
  - with 35 users, probability > 10 active at same time is less than .0004:
  - 0.9^{35} [probability 0 users sending]

    $+ 0.1(0.9)^{34} \times 35$
    [prob of only 1 user x # of users]

    $+ 0.1^2(0.9)^{33} \times$ (# combos of 2 users)
    + …



Q: how did we get value 0.0004?*

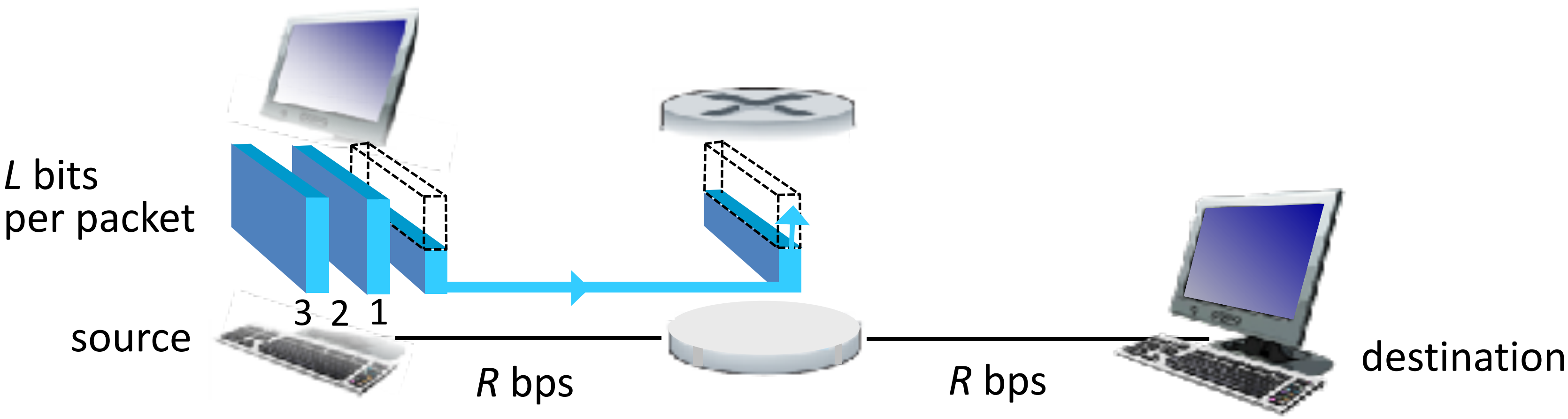
$$p = .1, N = 35, T = 10$$

$$Pr[\text{n>T}] = 1 - \sum_{i=0}^{i=T} \binom{N}{i} p^i (1-p)^{(N-i)}$$

Q: what happens if > 35 users ?
- chance of collision increases

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

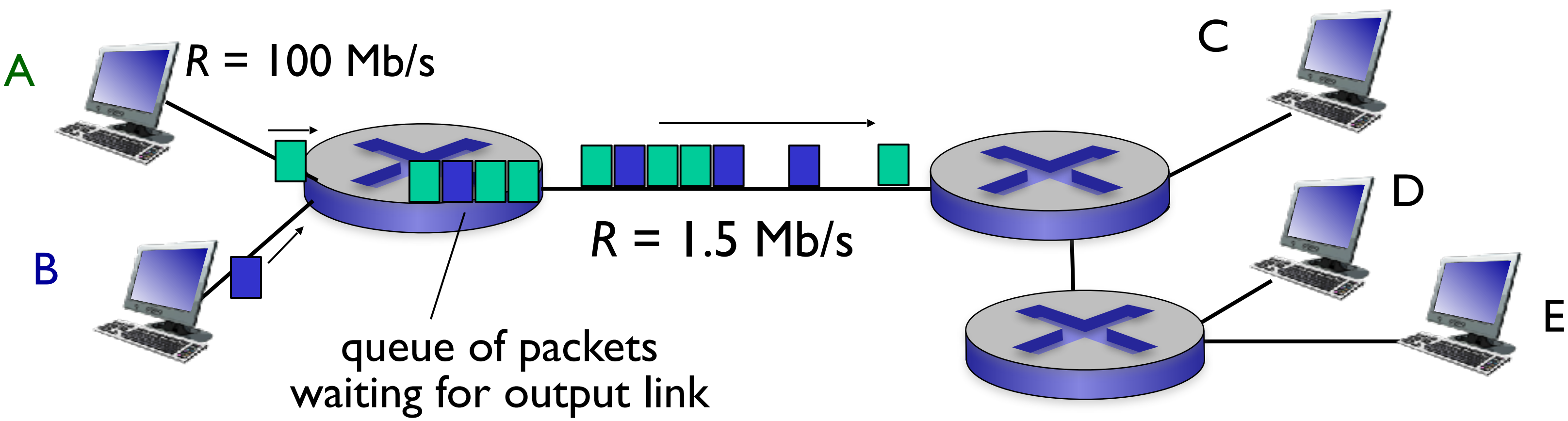
Packet-switching: store-and-forward



- 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 router can transmit it onward
  - *buffers* the packet until ready to send

- *(assuming 0 propagation delay, we'll discuss that later)*
  end-to-end delay of packet?
  - **2L/R**
- ***Delay for destination to receive all 3 packets?***
  - **4L/R**

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
    - "*queueing delay*"
    - like going to check out at grocery store and having to wait for customer(s) in front of you
  - packets can be dropped (lost) if memory (buffer) fills up

Packet switching versus circuit switching

is packet switching a "slam dunk winner?"

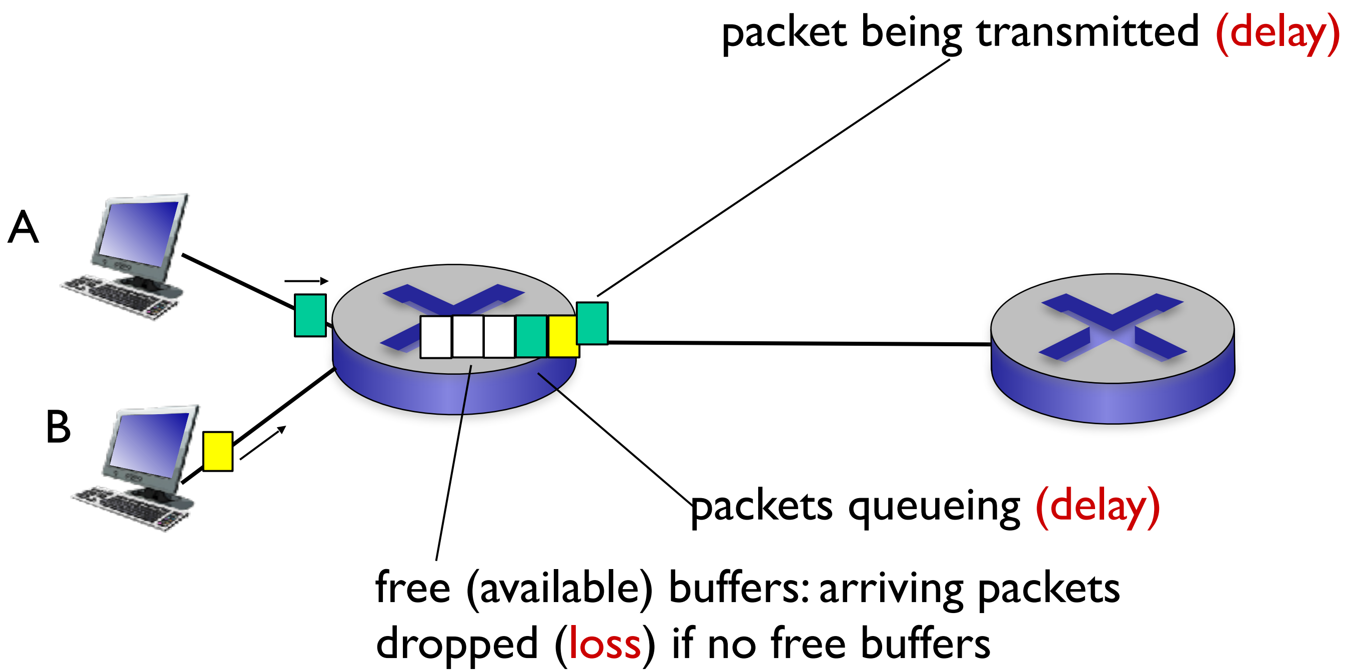
- great for bursty data
  - resource sharing
  - simpler, no call setup
- easy to support diverse and dynamic applications
- excessive congestion possible: packet delay and loss
  - protocols needed for reliable data transfer, congestion control
- *Q:* How to provide circuit-like behavior?
  - bandwidth 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)?

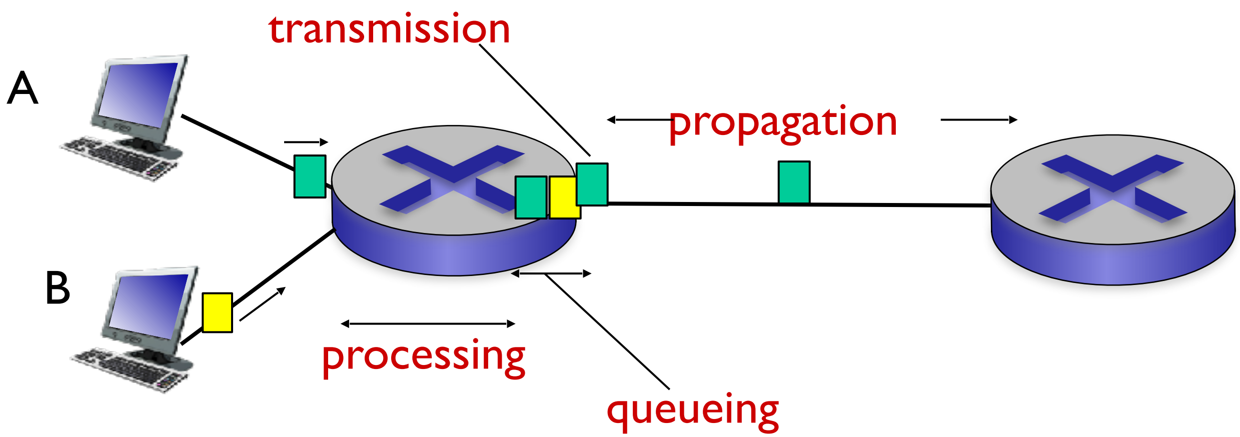
How do loss and delay occur?

packets *queue* in router buffers

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



4 sources of nodal (per node) delay



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

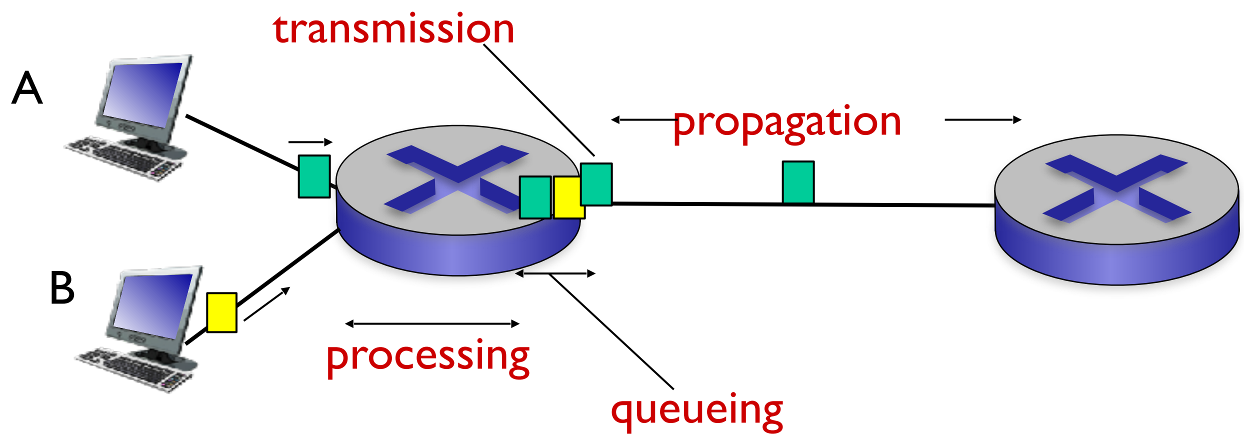
d_{proc}: processing

- 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

4 sources of nodal (per node) 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

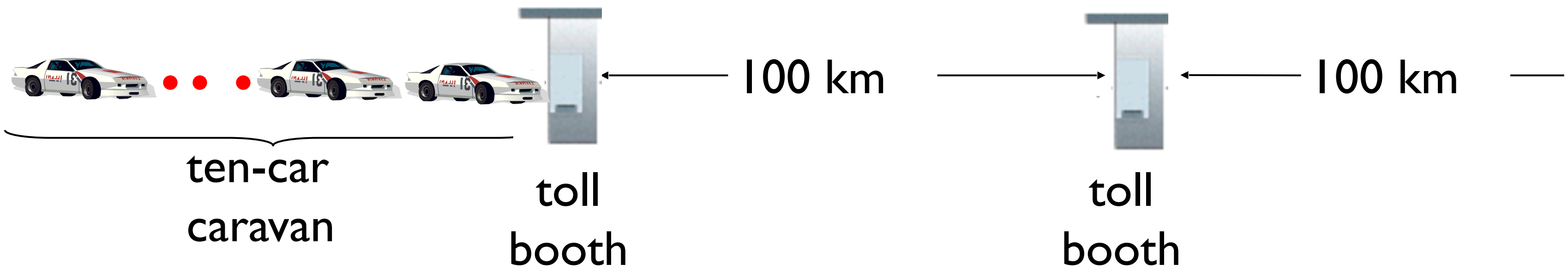
d_{prop}: propagation delay:

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

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

* Check out the Java applet for an interactive animation on trans vs. prop delay

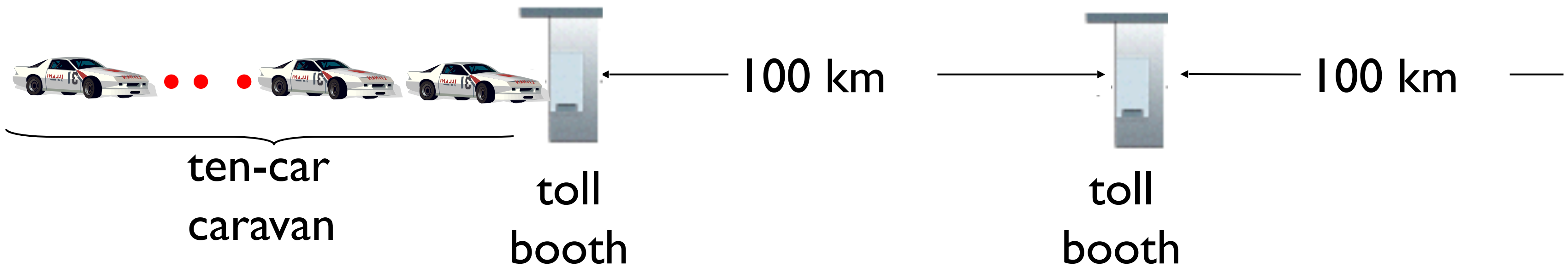
Caravan analogy



- 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

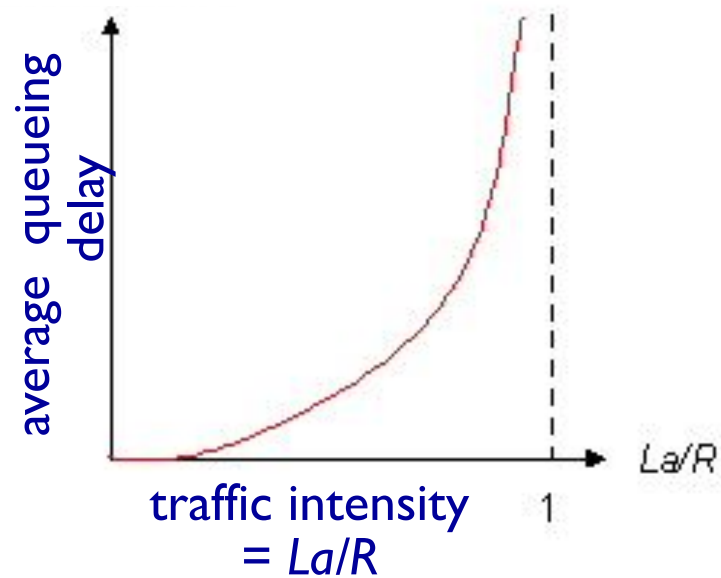
Caravan analogy (more)



- suppose cars now "propagate" at 1000 km/hr
- and suppose toll booth now takes one 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 (packets per second)



- *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!

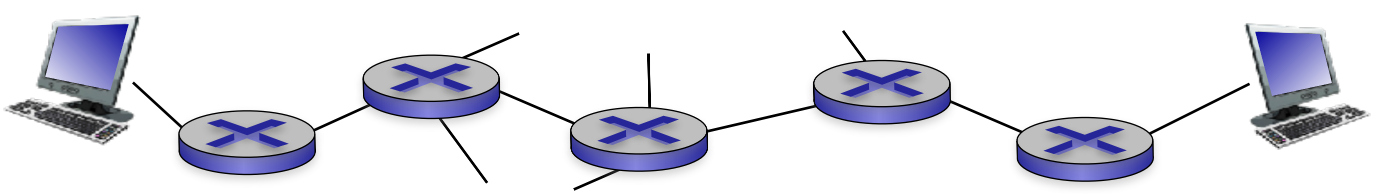
La/R ~ 0

La/R -> *1*

* Check online interactive animation on queuing and loss

"Real" Internet delays and routes

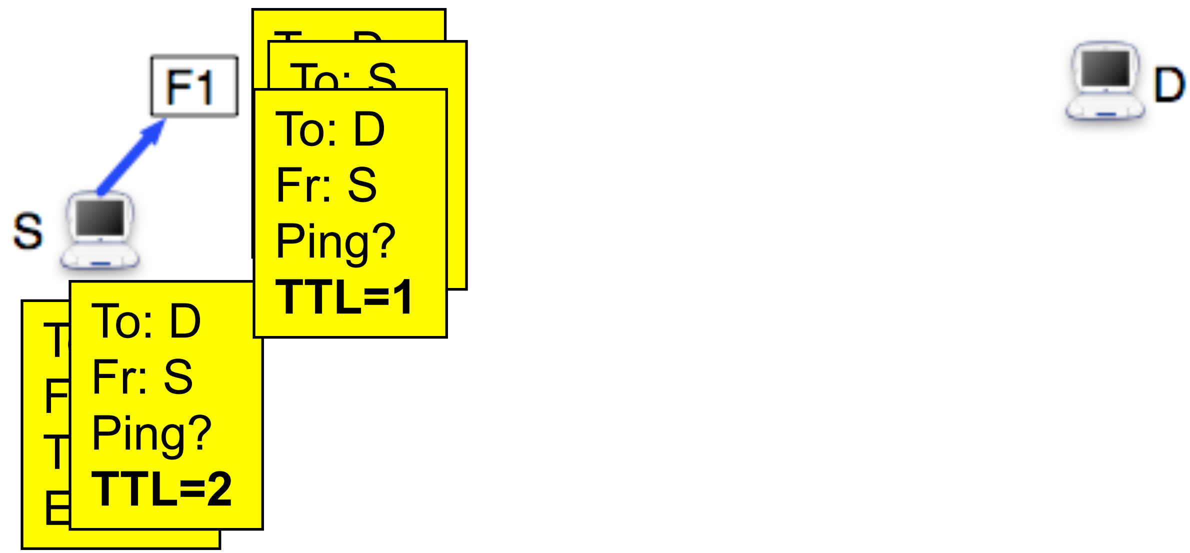
- what do "real" Internet delay & loss look like?
  - Protocols do not provide much visibility
  - Networks do not have incentive to divulge
  - Therefore, need tools that measure routes
- `traceroute` program: provides delay measurement from source to router along end-end Internet path towards destination.



Traceroute Uses TTL to Measure Path

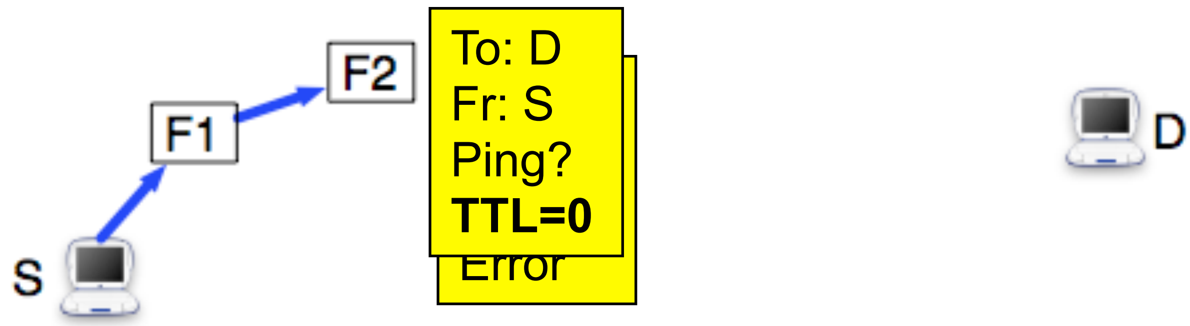
- All Internet packets have a time-to-live (TTL) field
- Each router on the path decrements the field
- If TTL=0, router discards packet, sends error to source
- Traceroute uses errors at each TTL to build route

Traceroute Uses TTL to Measure Path



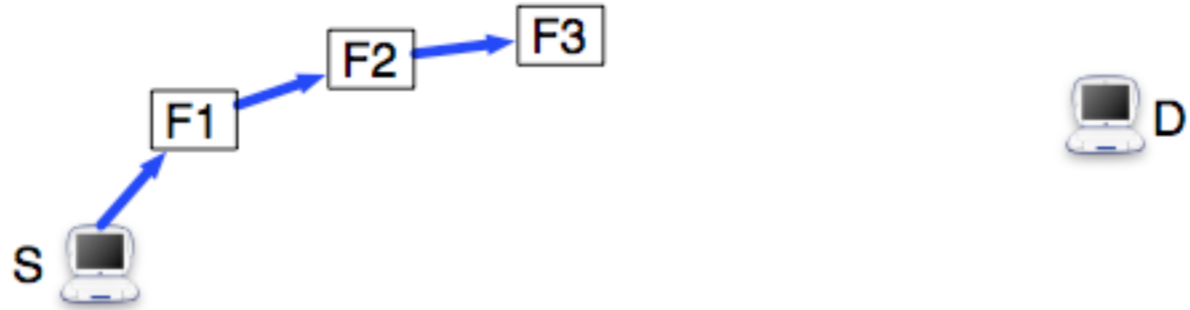
- All Internet packets have a time-to-live (TTL) field
- Each router on the path decrements the field
- If TTL=0, router discards packet, sends error to source
- Traceroute uses errors at each TTL to build route

Traceroute Uses TTL to Measure Path



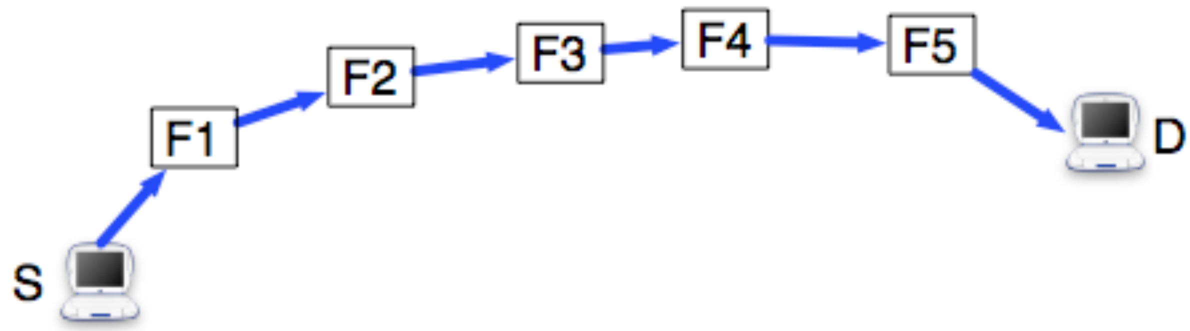
- All Internet packets have a time-to-live (TTL) field
- Each router on the path decrements the field
- If TTL=0, router discards packet, sends error to source
- Traceroute uses errors at each TTL to build route

Traceroute Uses TTL to Measure Path



- All Internet packets have a time-to-live (TTL) field
- Each router on the path decrements the field
- If TTL=0, router discards packet, sends error to source
- Traceroute uses errors at each TTL to build route

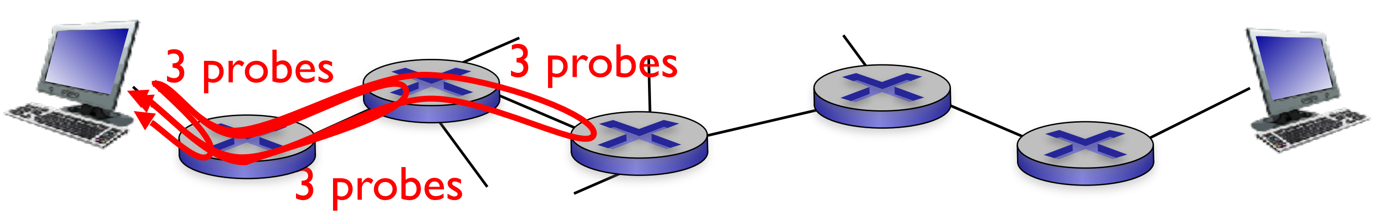
Traceroute Uses TTL to Measure Path



- All Internet packets have a time-to-live (TTL) field
- Each router on the path decrements the field
- If TTL=0, router discards packet, sends error to source
- Traceroute uses errors at each TTL to build route

“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.



“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

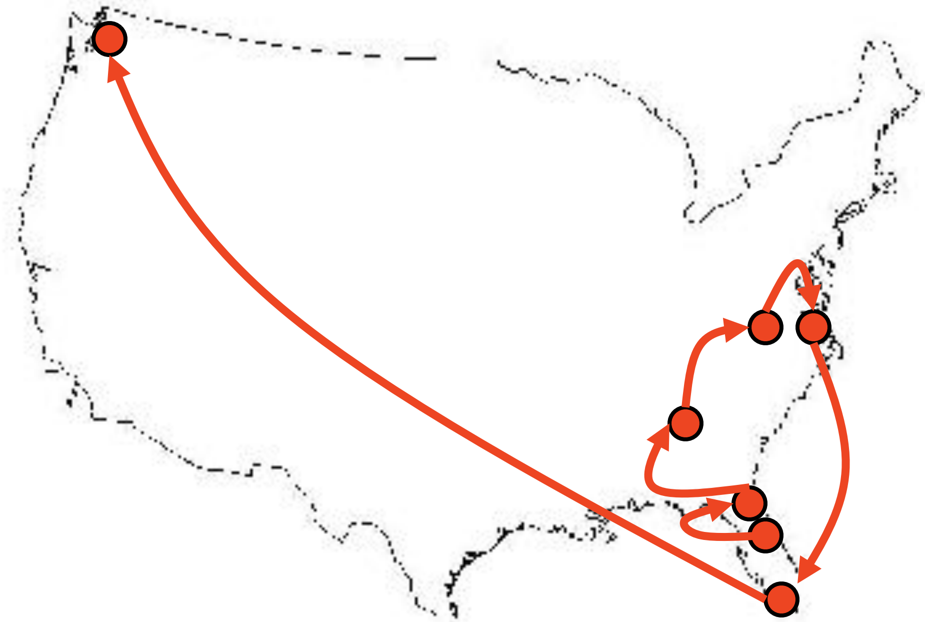
* means no response (probe lost, router not replying)

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

"Real" Internet delays, routes

- 150ms round-trip time Orlando to Seattle, 2-3x expected
  - E.g., Content provider detects poor client performance
- Issue traceroute, check if indirect

Hop no.	DNS name / IP address
1	132.170.3.1
2	198.32.155.89
3	**JAX-FL**...net.flrnet.org
4	**ATLANTA**ix.cox.com
5	**ASH**...as.cox.net
6	core2...**WDC**.pnap.net
7	cr1.**WDC**...internap.net
8	cr2-cr1.**WDC**...internap.net
9	cr1.**MIA**...internap.net
10	cr1.**SEA**...internap.net



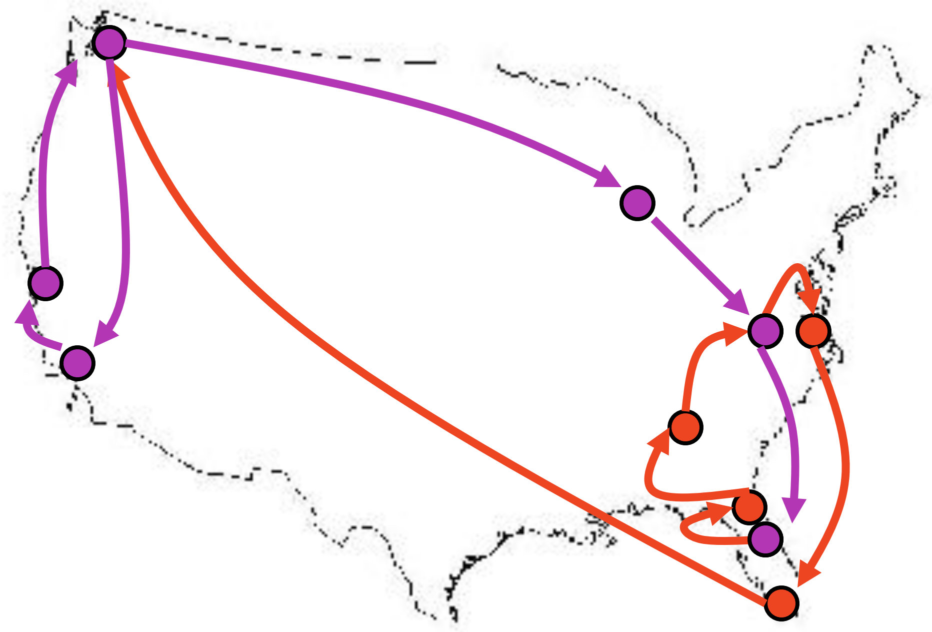
- Indirectness: FL->DC->FL
  But only explains half of latency inflation

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

"Real" Internet delays, routes

- What about Seattle back to Orlando?
  - Internet paths often asymmetric (routing section will explore why)
- Issue traceroute in opposite direction

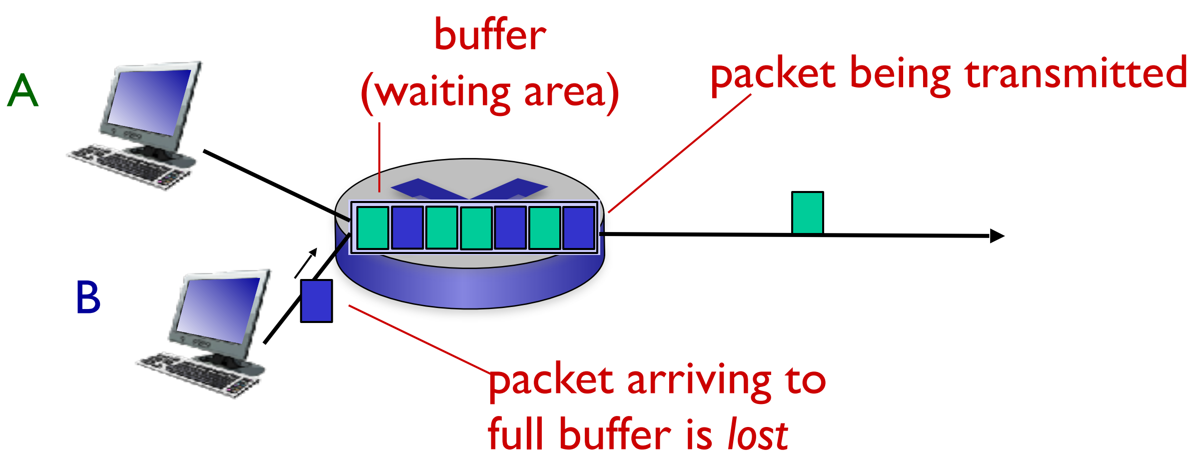
Hop no.	DNS name / IP address
1	cr1.**SEA**...internap.net
2	cr1.**SEA**...internap.net
3	internap...**LSANCA01**.transitrail.net
4	te4...**LSANCA01**.transitrail.net
5	te4...**PLALCA01**.transitrail.net
6	te4...**STTLWA01**.transitrail.net
7	te4...**CHCGIL01**.transitrail.net
8	te2...**ASBNVA01**.transitrail.net
9	132.170.3.1
10	planetlab2.eecs.**UCF.EDU**



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

Packet loss

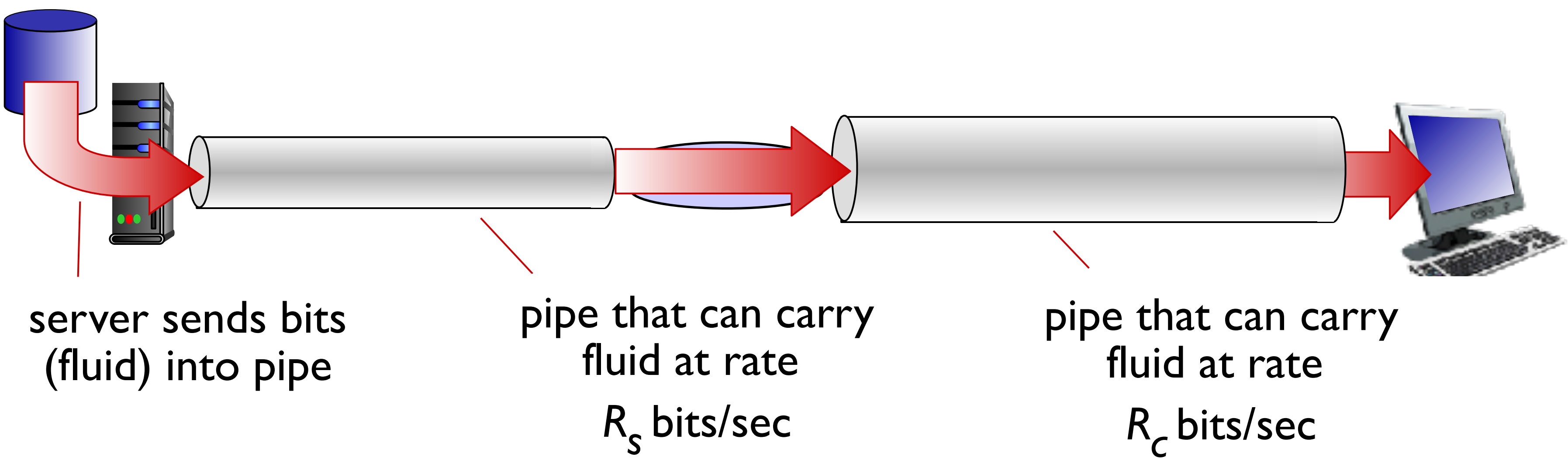
- queue (aka buffer) preceding link in buffer has finite capacity
- packet arriving to full queue dropped (aka lost)
- lost packet may be retransmitted by previous node, by source end system, or not at all



* Check out the Java applet for an interactive animation on queuing and loss

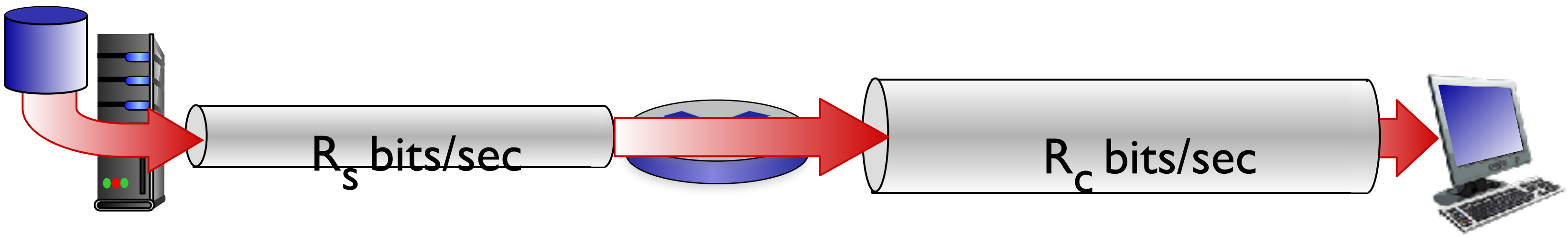
Throughput

- *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

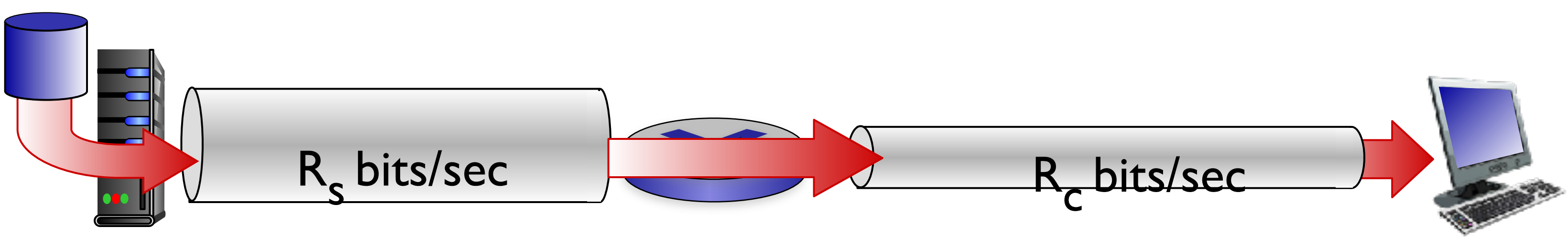


Throughput (more)

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



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

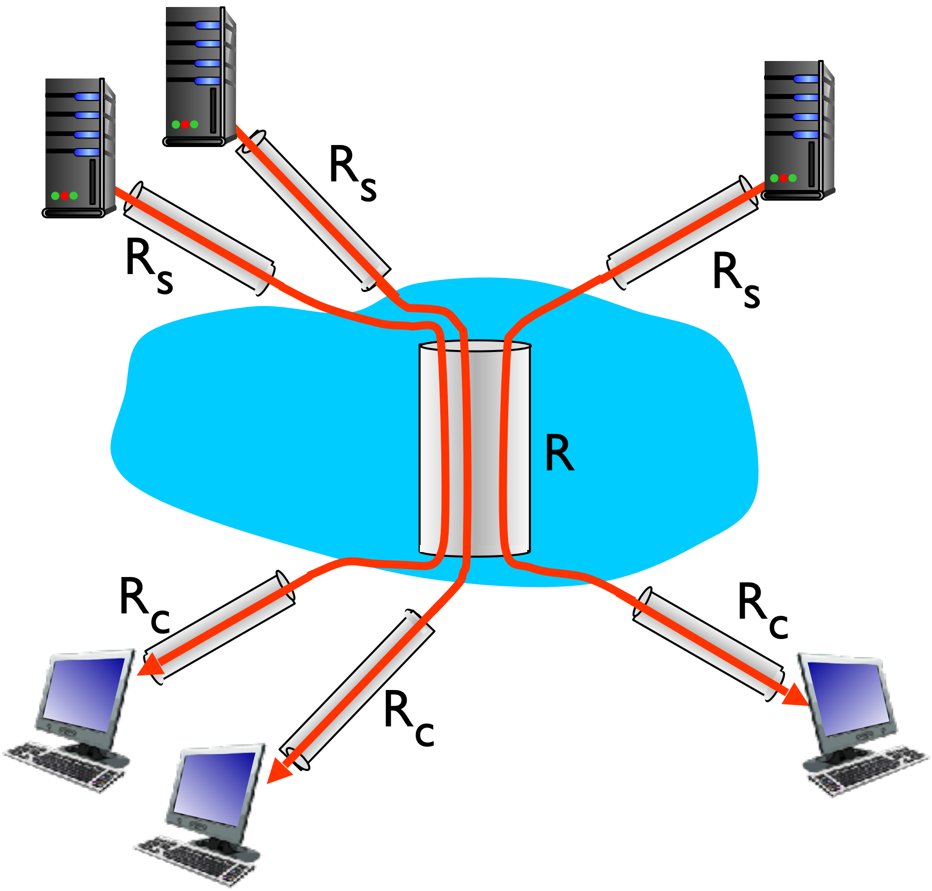


bottleneck link

link on end-end path that constrains throughput

Throughput: Internet scenario

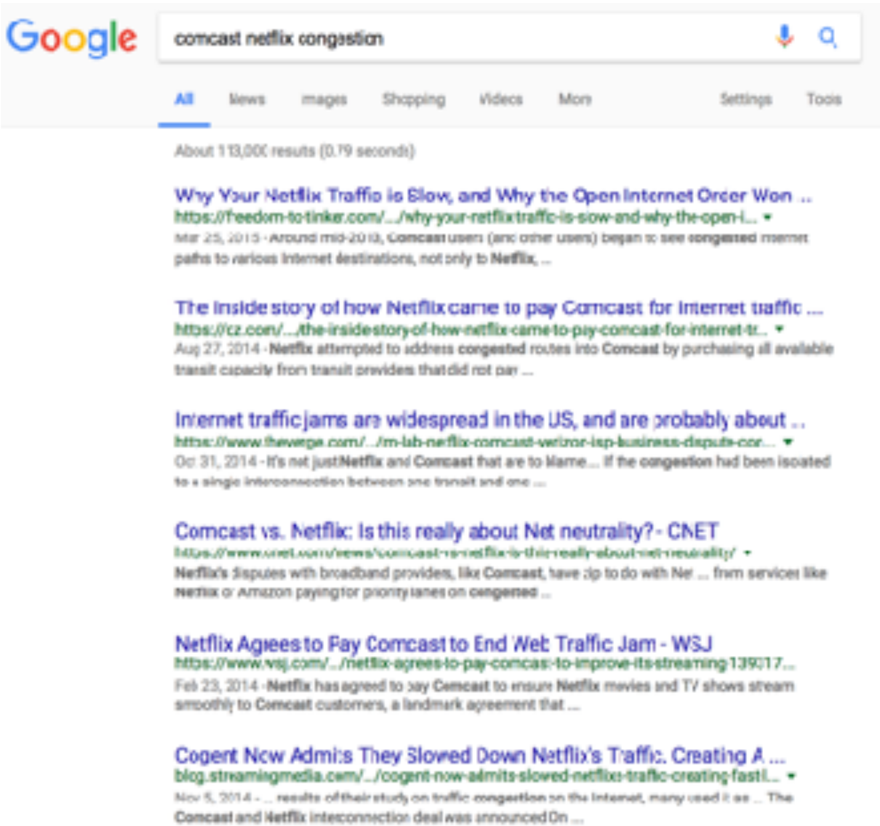
- per-connection end-end throughput: $min(R_c,R_s,R/10)$
- in practice: R_c or R_s is often bottleneck
  - Why?
  - When might it not be?
  - How will this change over time?

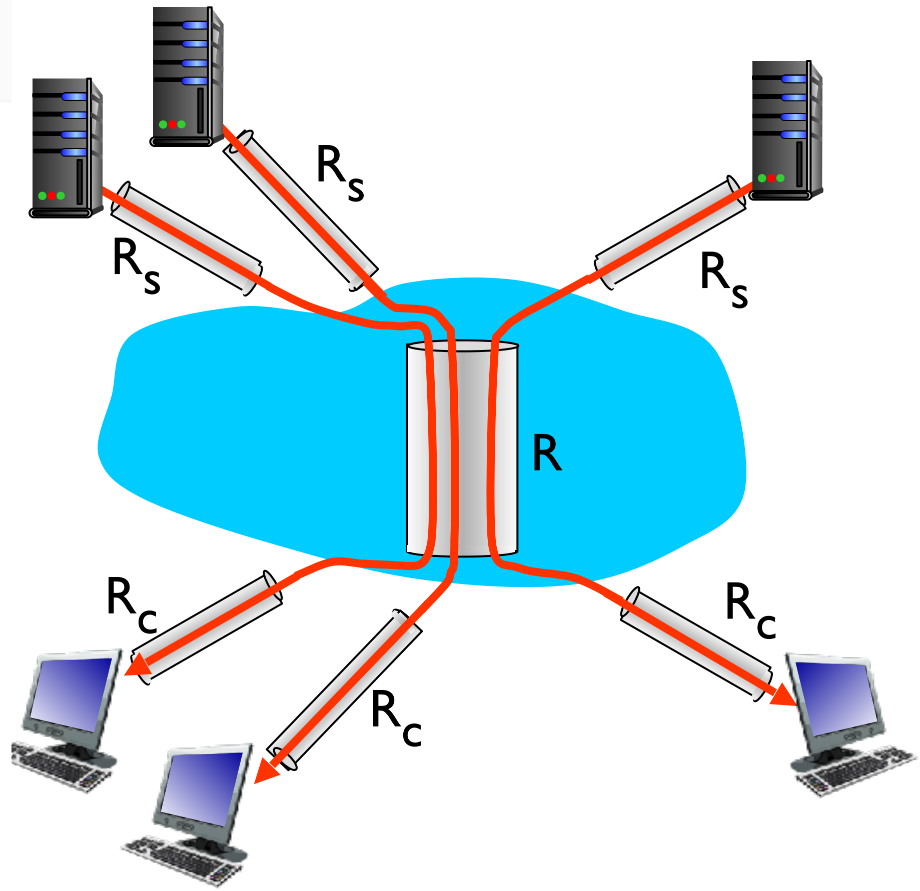


10 connections (fairly) share backbone bottleneck link R bits/sec

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

Throughput: Internet scenario





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

Chapter 1: roadmap

1.1 what *is* the Internet?

1.2 network edge

- end systems, access networks, links

1.3 network core

- network structure, packet switching, circuit switching

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?

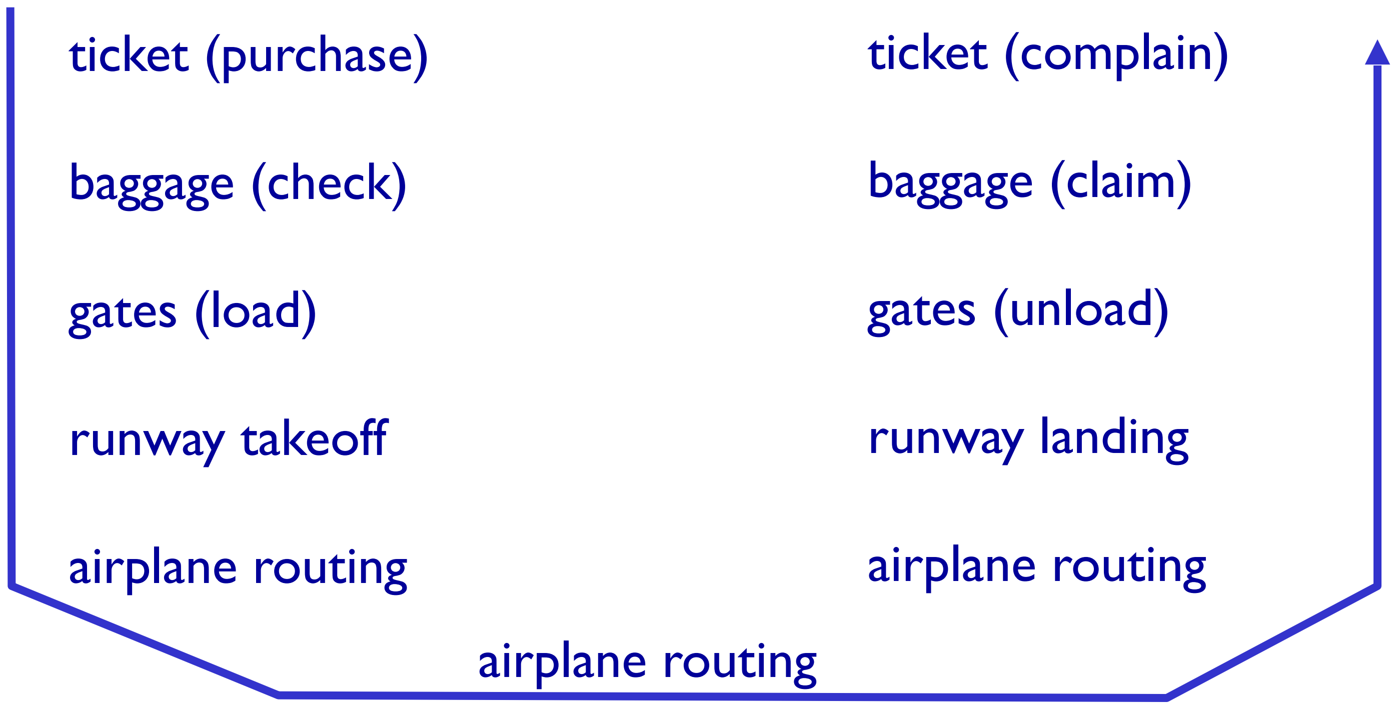
…. or at least our discussion of networks?

Organization of air travel

1. purchase ticket
2. check baggage
3. load plane gates
4. runway takeoff
5. airplane routing
6. runway landing
7. unload plane
8. claim baggage

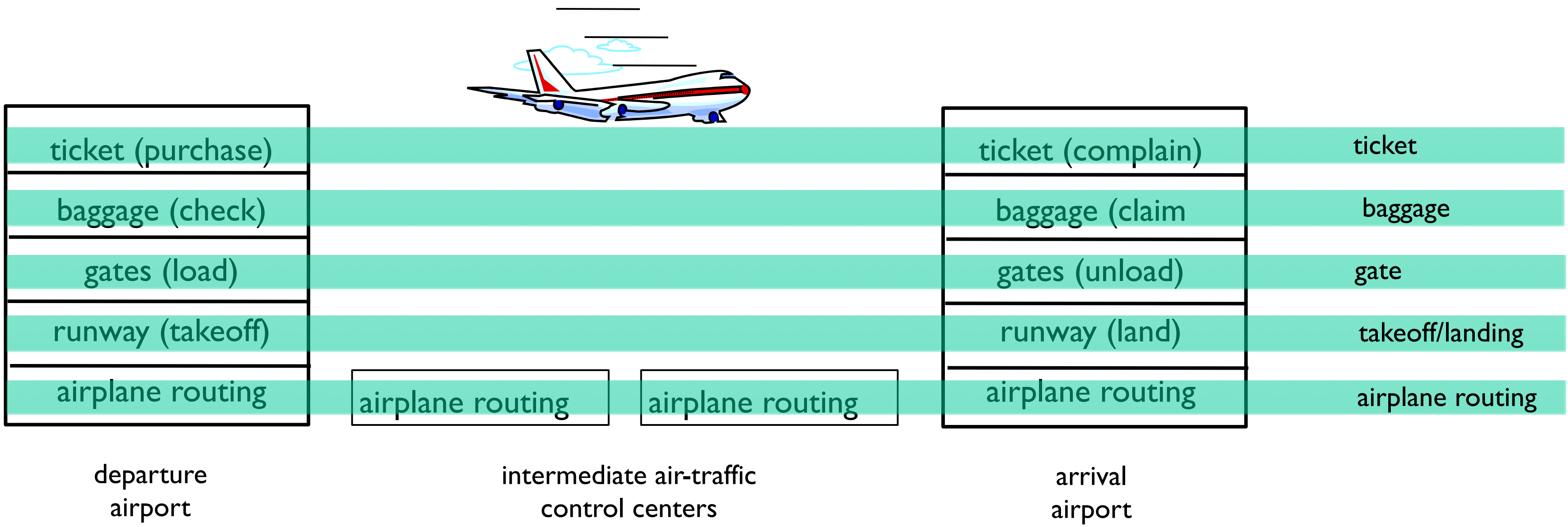
- a series of steps

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?

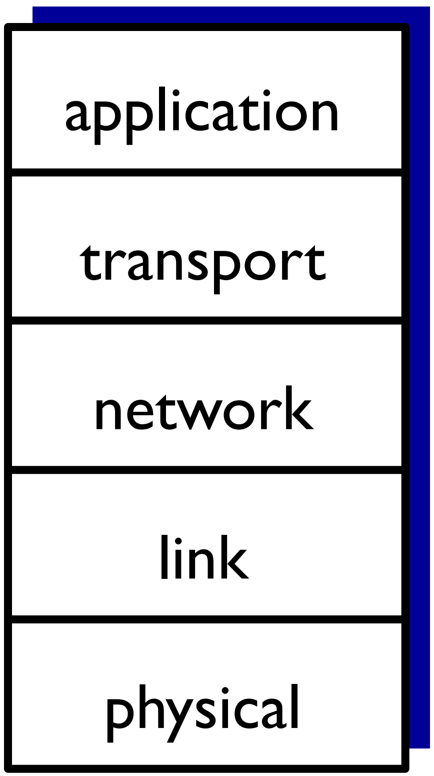
Why layering?

dealing with complex systems:

- explicit structure allows identification, relationship of complex system's pieces
  - layered *reference model* for discussion
- modularization eases maintenance, updating, reuse 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

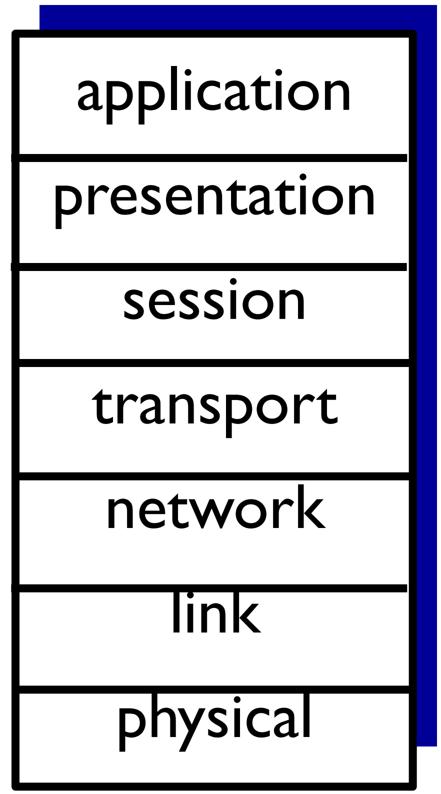
Internet protocol stack (5 layer)

- *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"



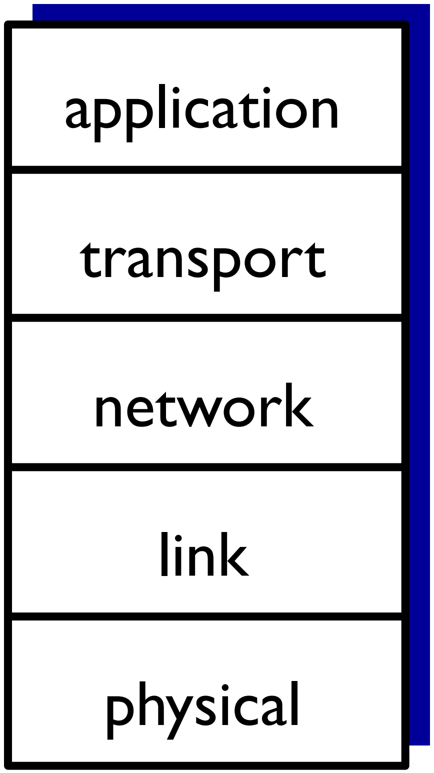
ISO/OSI reference model (7 layer)

- *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?

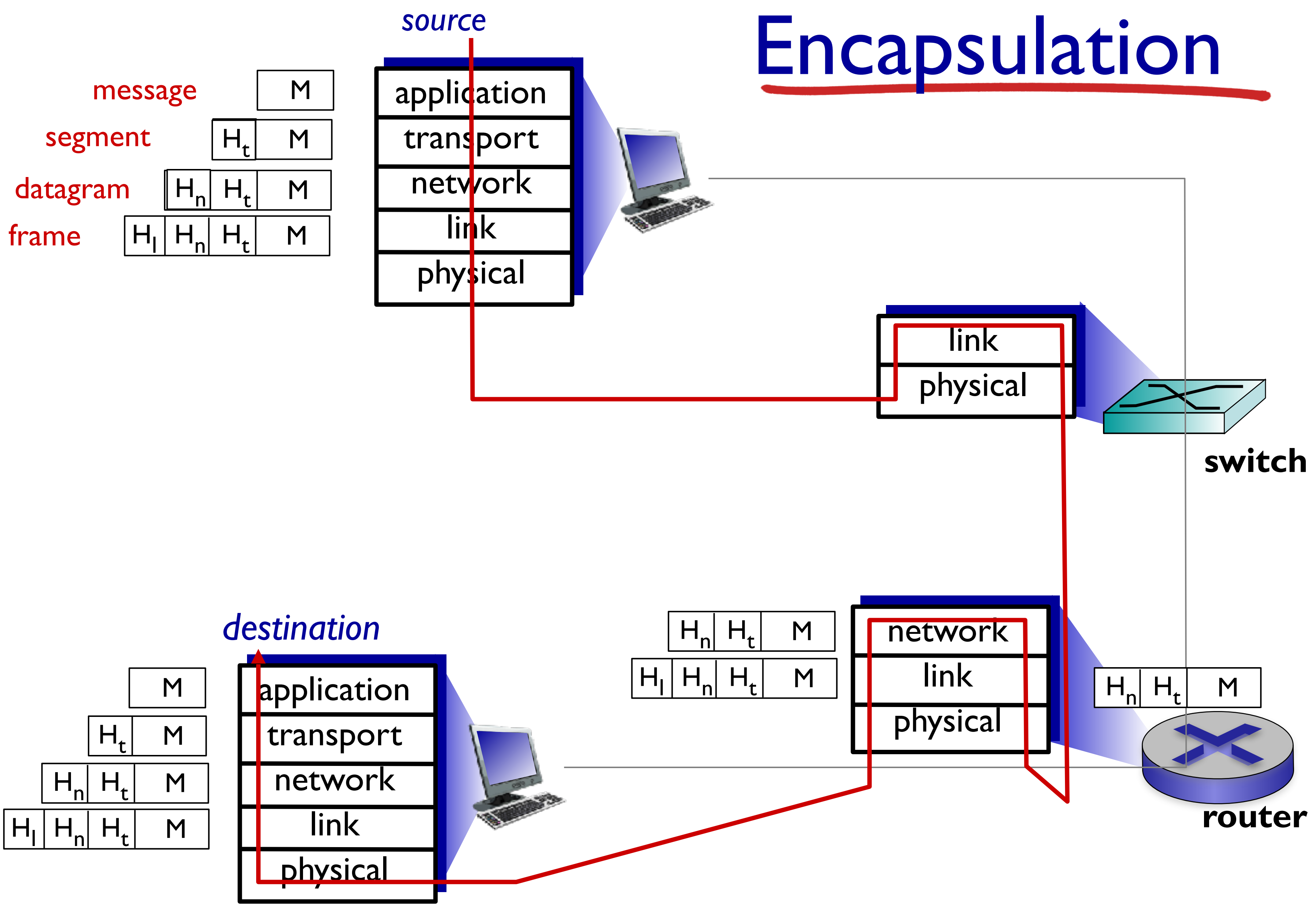


Internet protocol stack (5 layer)

- *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"



Encapsulation



Why ***NOT*** layering?

dealing with complex systems:

- explicit structure allows identification, relationship of complex system's pieces
  - layered *reference model* for discussion
- modularization eases maintenance, updating, reuse 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?***
  - upper layer may duplicate functionality of lower
  - upper may not need functionality of lower
  - one layer might want info only visible at another layer

Chapter 1: roadmap

1.1 what *is* the Internet?
1.2 network edge
- end systems, access networks, links

1.3 network core
- network structure, packet switching, circuit switching

1.4 delay, loss, throughput in networks
1.5 protocol layers, service models
1.6 networks under attack: security
1.7 history (not covered in lecture)

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

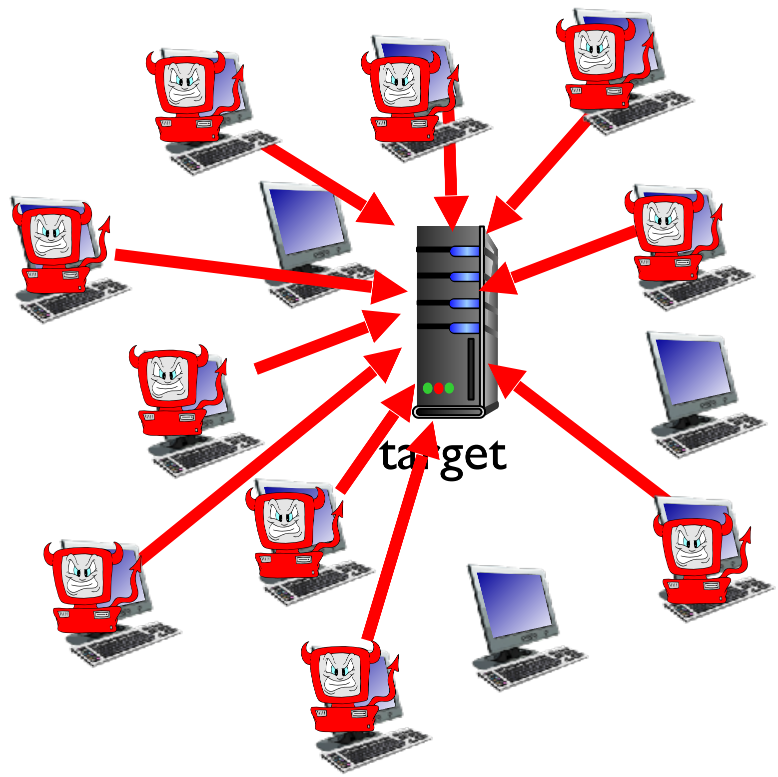
Bad guys: attack server, network infrastructure

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)

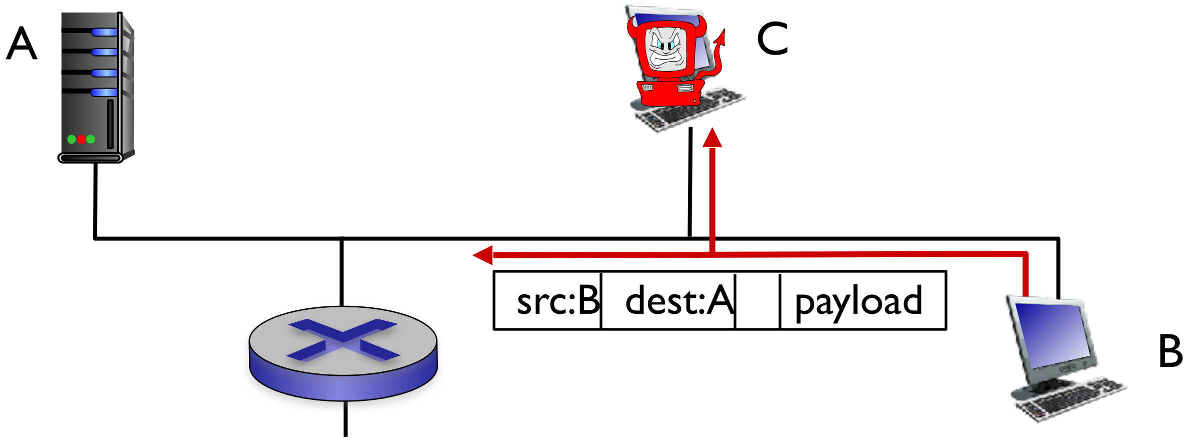
3. send packets to target from compromised hosts



Bad guys can sniff packets

packet "sniffing":

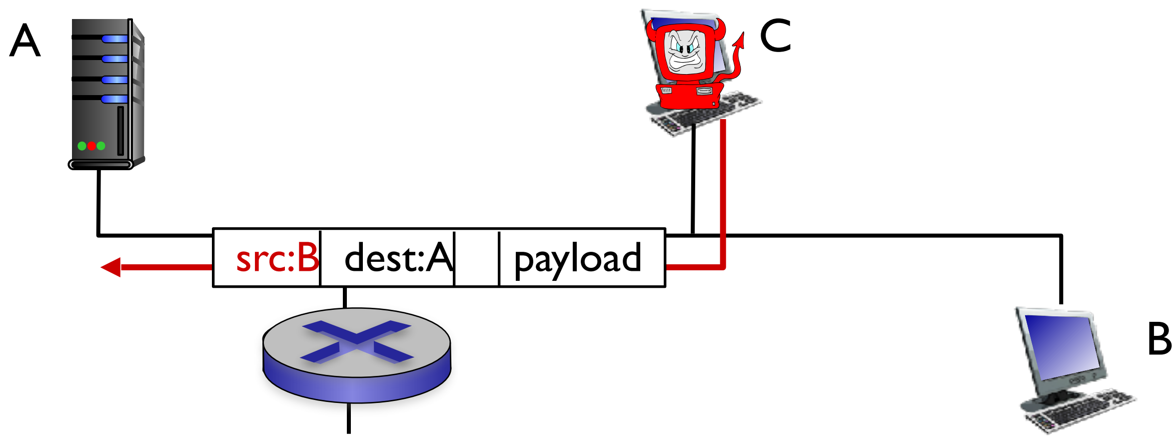
- broadcast media (shared Ethernet, wireless)
- promiscuous network interface reads/records all packets (e.g., including passwords!) passing by



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

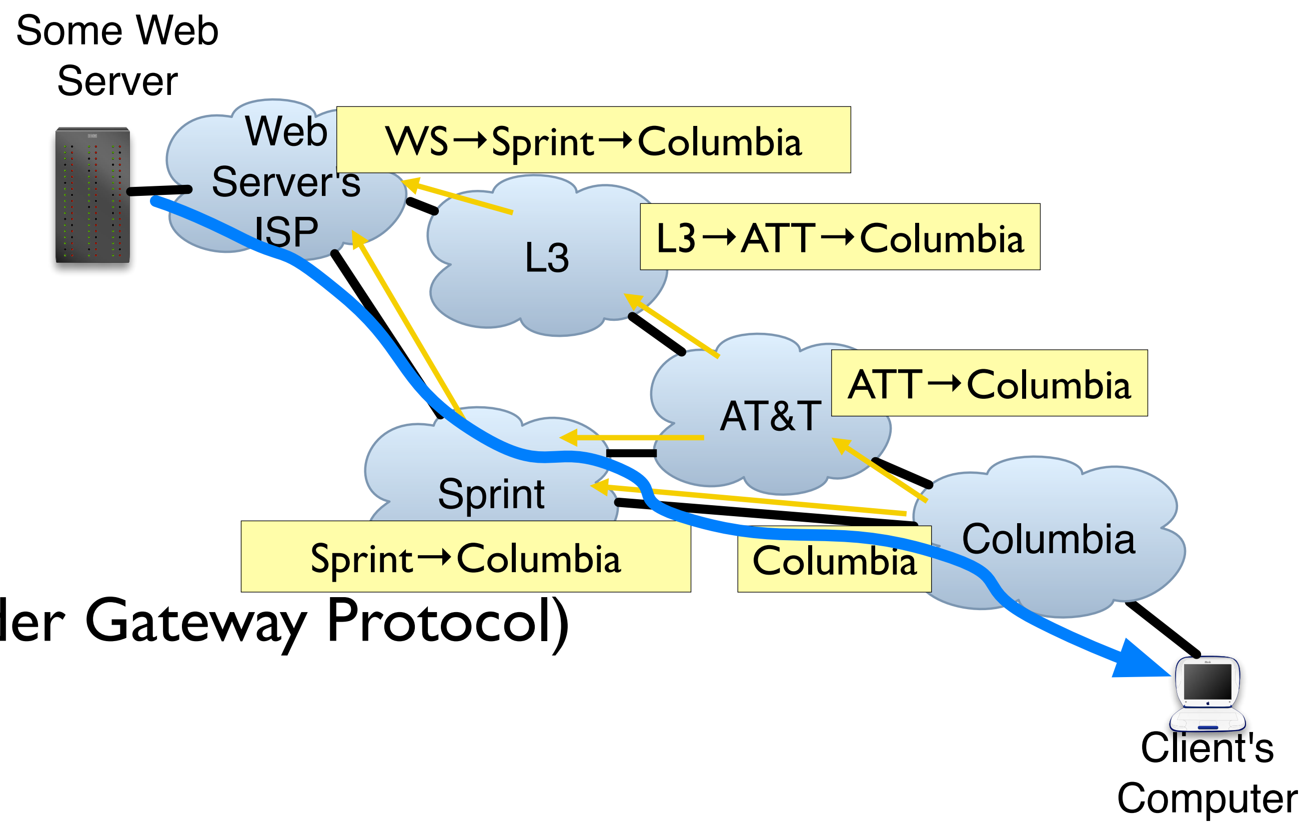
Bad guys can use fake addresses

IP spoofing: send packet with false source address



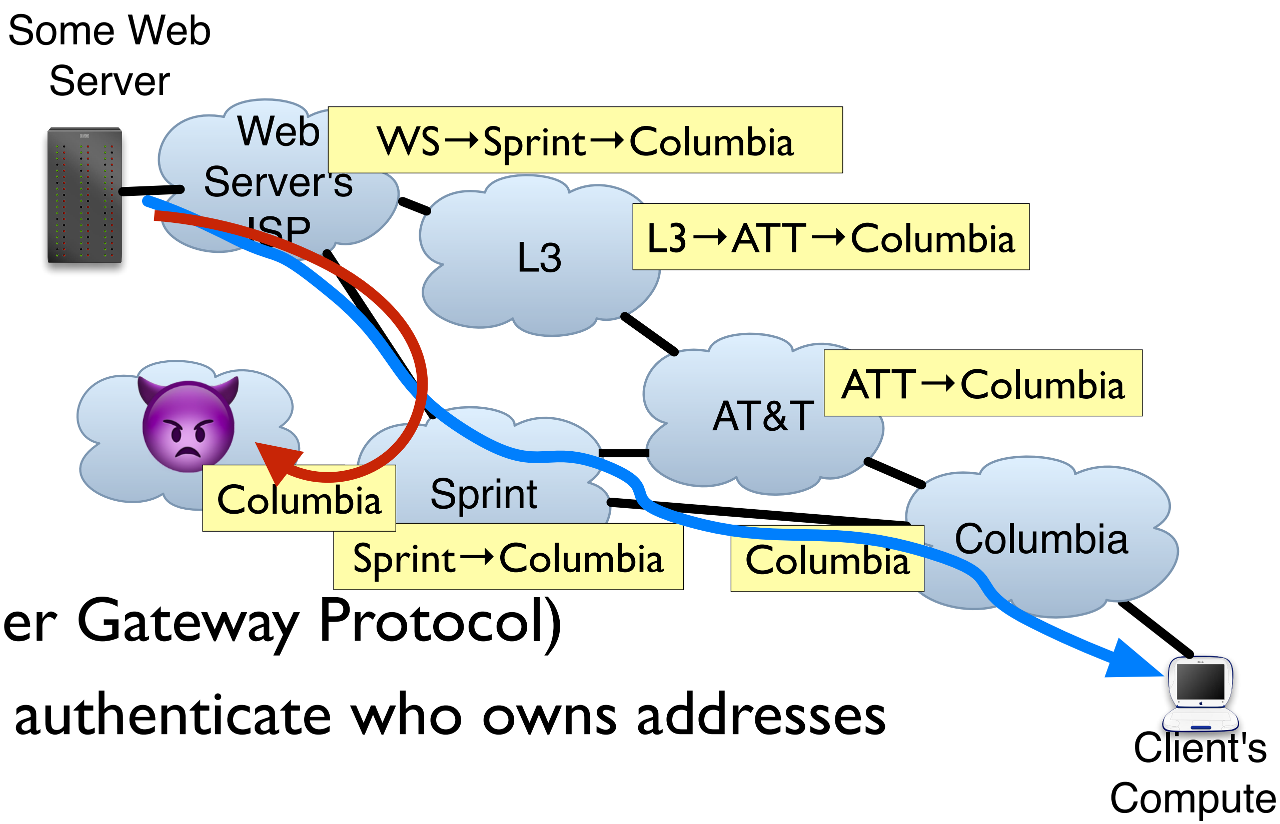
… *lots more on security (throughout, Chapter 8)*

Bad guys can claim they control addresses



BGP (Border Gateway Protocol)

Bad guys can claim they control addresses



BGP (Border Gateway Protocol)

- does not authenticate who owns addresses

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 (see book)

you now have:

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

Chapter 1: roadmap

1.1 what *is* the Internet?
1.2 network edge
- end systems, access networks, links

1.3 network core
- network structure, packet switching, circuit switching

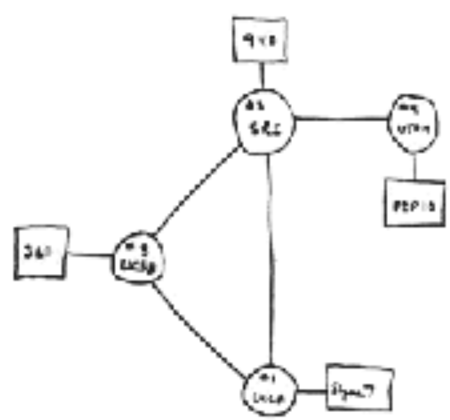
1.4 delay, loss, throughput in networks
1.5 protocol layers, service models
1.6 networks under attack: security
1.7 history

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!

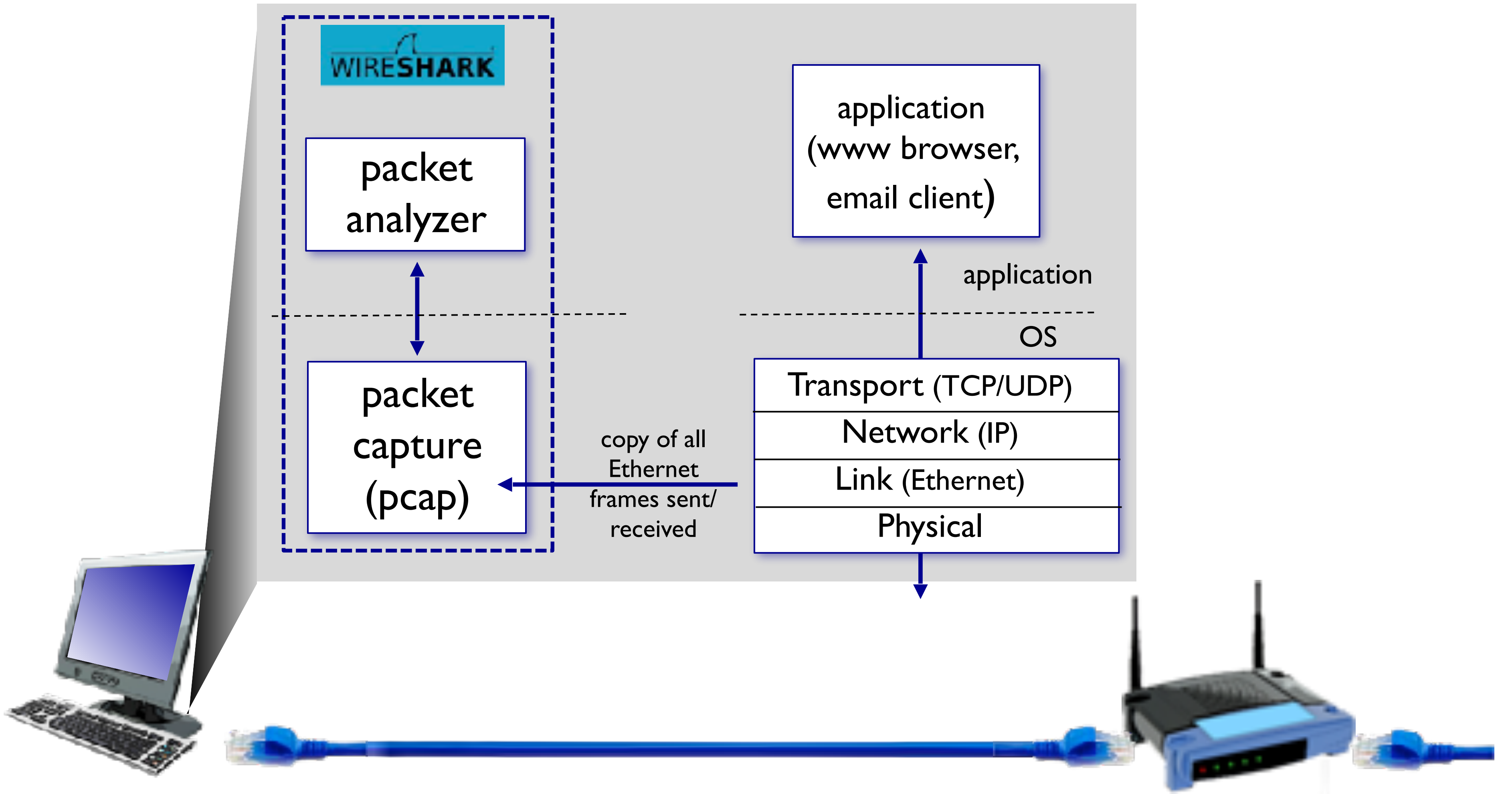
- Internet overview
- what's a protocol?
- network edge, core, access network
  - packet-switching versus circuit-switching
  - Internet structure
- performance: loss, delay, throughput
- layering, service models
- security
- history

you now have:

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

Uploading course materials to sites such as CourseHero, Chegg or Github is academic misconduct at Columbia (see pg 10 of Columbia guide).

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