Lecture 1: introduction

CIS 5617, Fall 2019
Anduo Wang
Based on Slides created by JFK/KWR

Lecture 1: roadmap

1.1 what *is* the Internet?

1.2 network edge
- end systems, access networks, links

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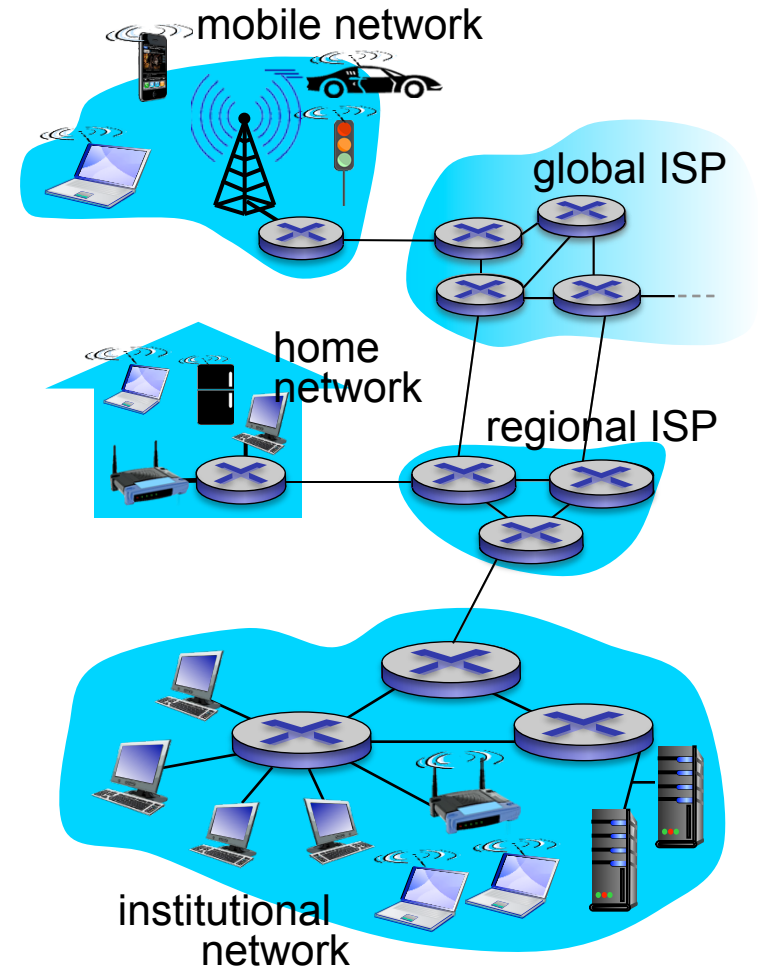- packet switching, circuit switching, network structure

1.4 delay, loss, throughput in networks

1.5 protocol layers, service models

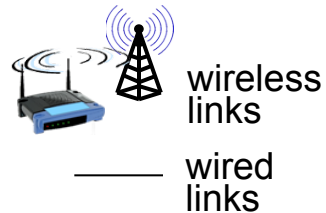*Goal: use Internet as example to get "feel" and terminology*

What's the Internet

- *Internet:* "network of networks"
  - Interconnected ISPs

mobile network

global ISP

home network

regional ISP

institutional network

What's the Internet

- PC
- server
- wireless laptop
- smartphone

- wireless links
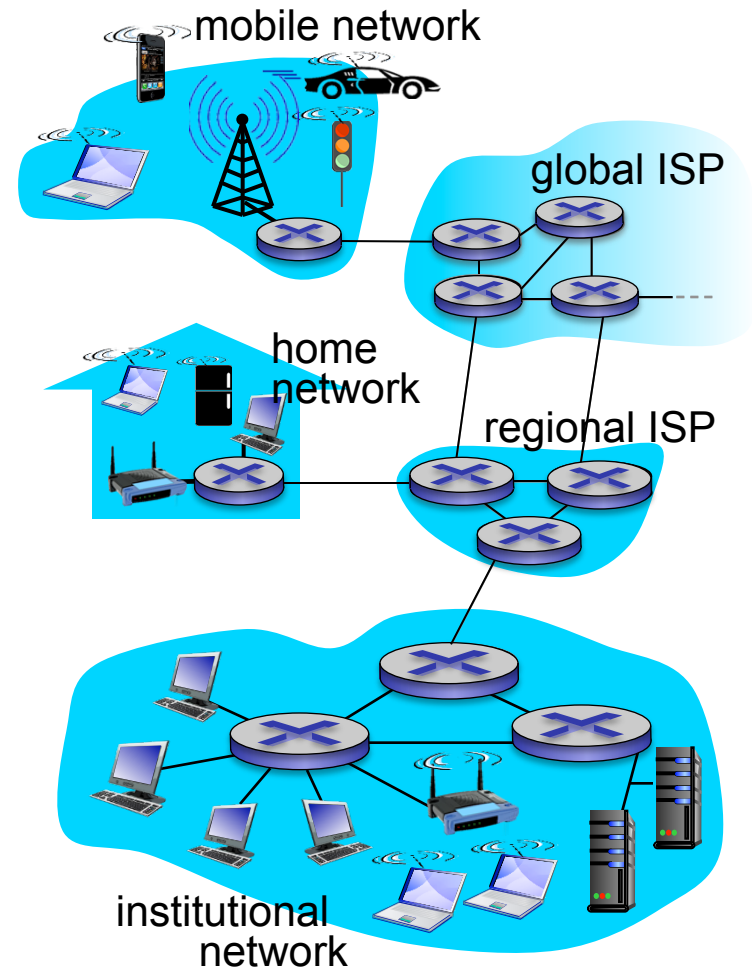- —— wired links

- router

- **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*

mobile network

global ISP

home network

regional ISP

institutional network

What's the Internet

- *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

mobile network

global ISP

home network

regional ISP

institutional network
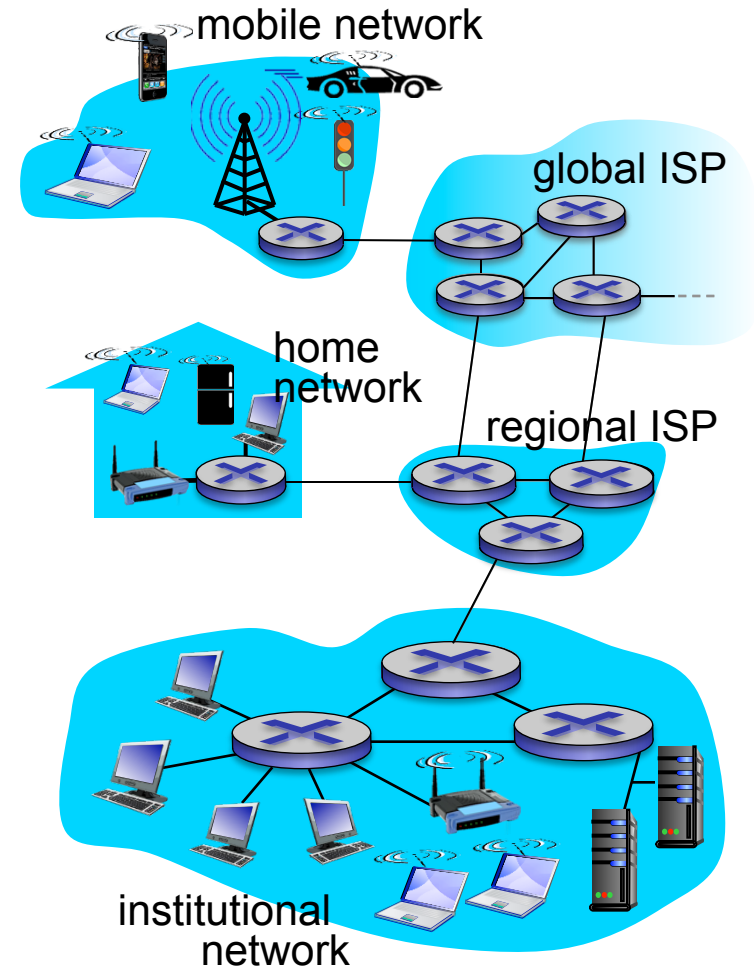
What's a protocol?

human protocols:

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

… 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:
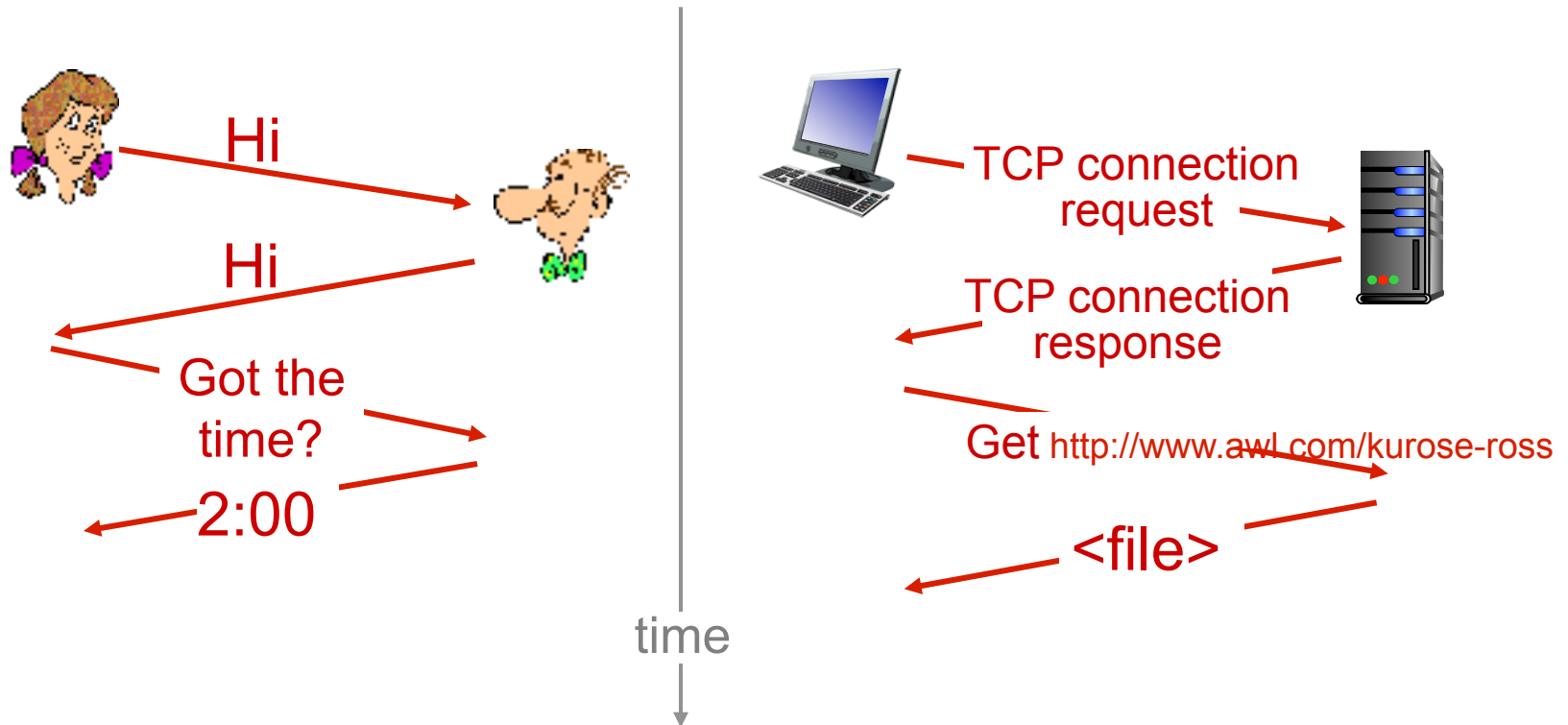
Q: other human protocols?

Lecture 1: roadmap

1.1 what *is* the Internet?

1.2 network edge
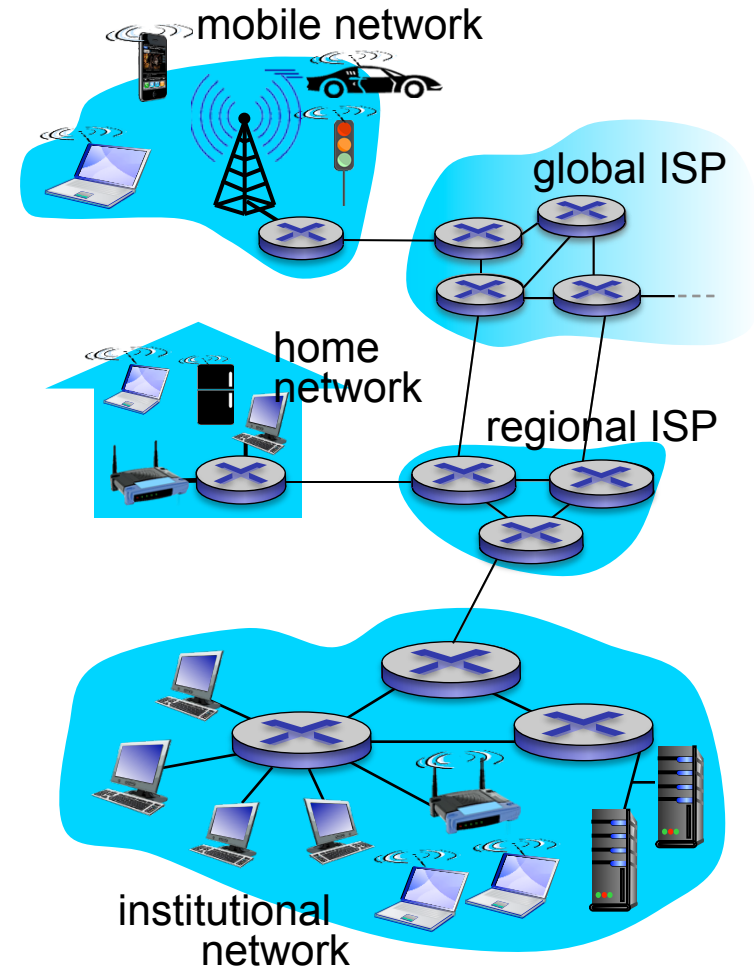 - end systems, access networks, links

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

1.4 delay, loss, throughput in networks

1.5 protocol layers, service models

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

mobile network

global ISP

home network

regional ISP

institutional network

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 network at *transmission rate R*
  - link transmission rate, aka link *capacity, aka link bandwidth*

two packets, L bits each

2   1

host

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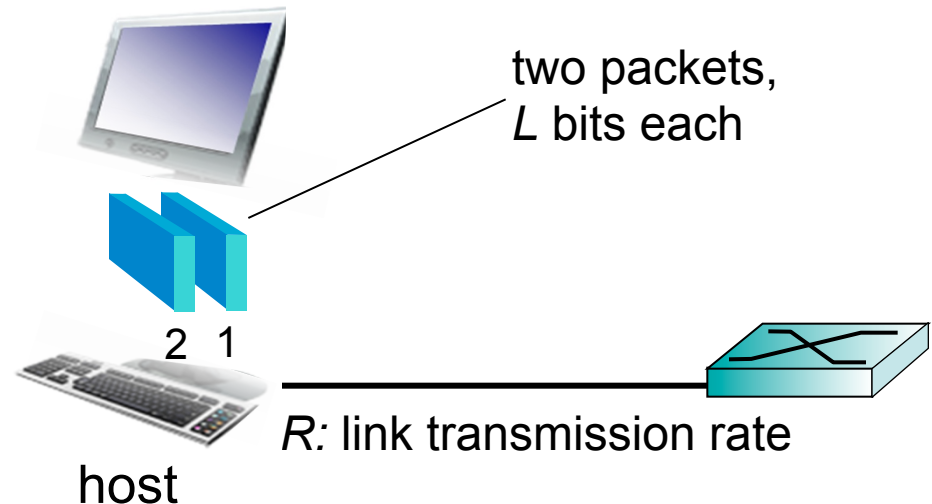
$$\begin{array}{c}\text{packet}\\\text{transmission}\\\text{delay}\end{array} = \begin{array}{c}\text{time needed to}\\\text{transmit } L\text{-bit}\\\text{packet into link}\end{array} = \frac{L \text{ (bits)}}{R \text{ (bits/sec)}}$$

Lecture 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

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
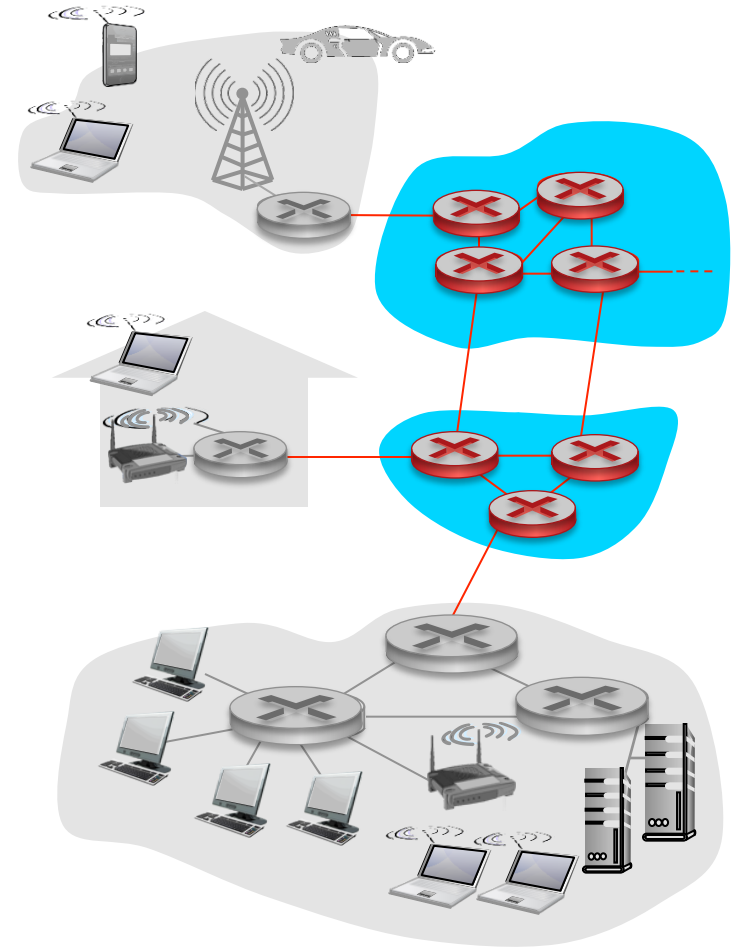
Packet-switching: store-and-forward

L bits
per packet

3 2 1

source

R bps

R bps

destination
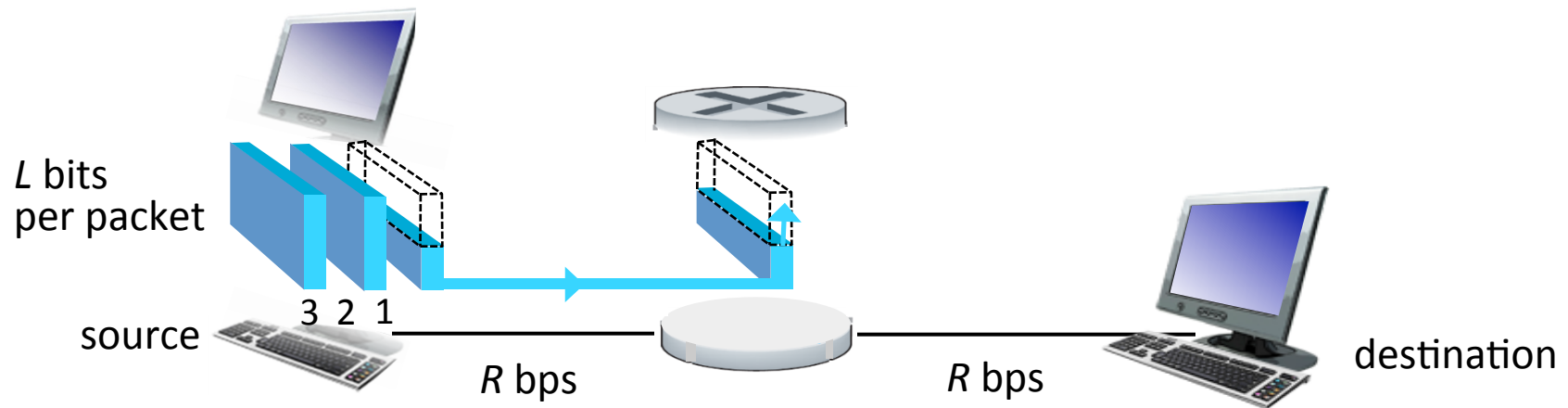
- takes *L/R* seconds to transmit (push out) *L*-bit packet into link at *R* bps

- *store and forward:* entire packet must arrive at router before it can be transmitted on next link

- end-end delay = 2*L/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 …

Packet Switching: queueing delay, loss

A $R = 100$ Mb/s

B

queue of packets
waiting for output link

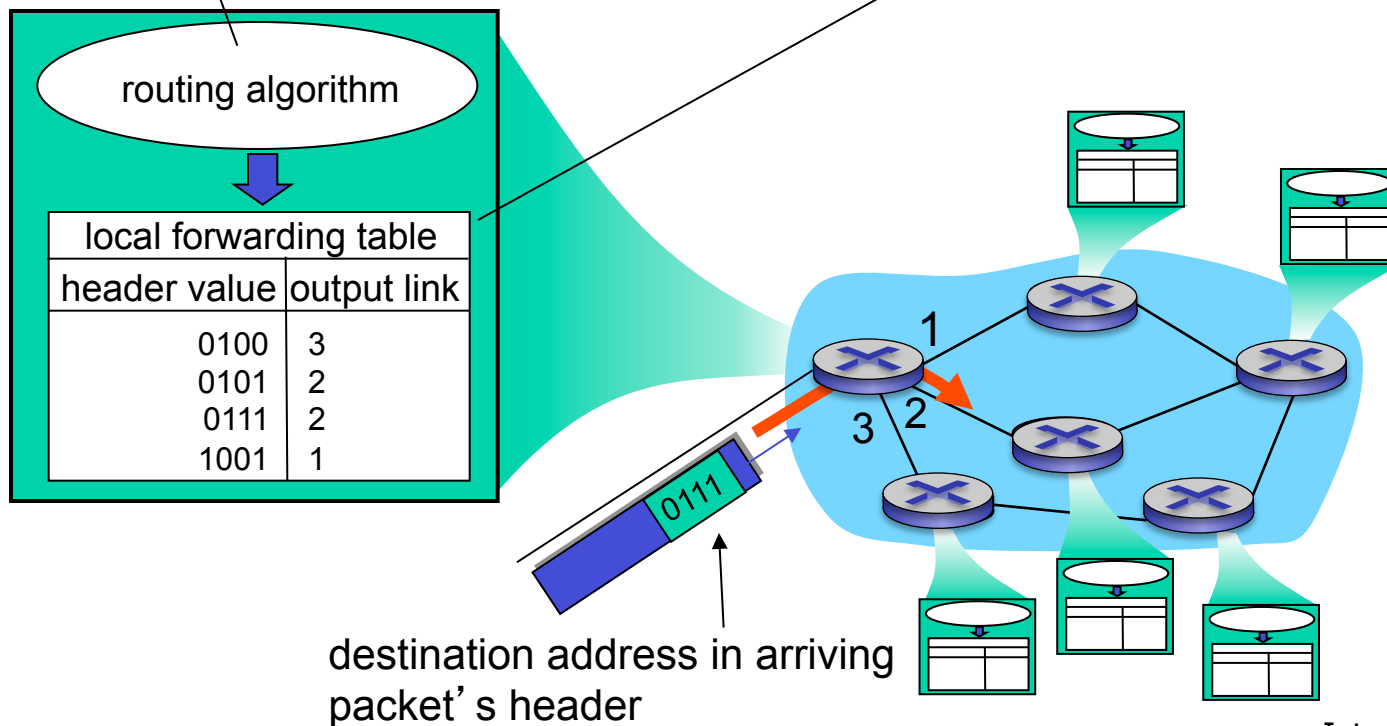$R = 1.5$ Mb/s

C

D

E
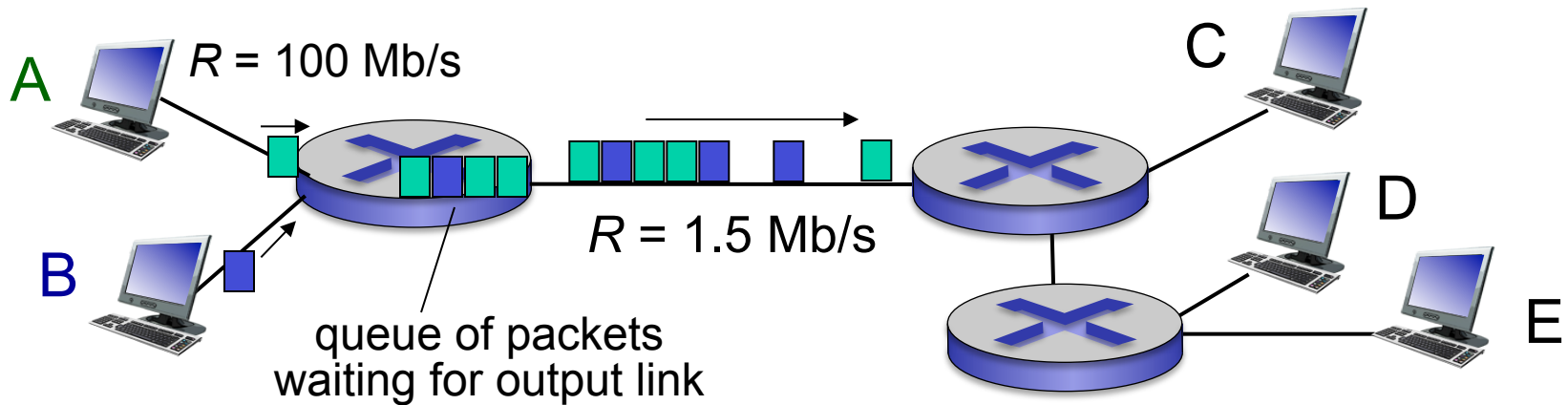
queuing and loss:

- 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

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

routing algorithm

local forwarding table

header value	output link
0100	3
0101	2
0111	2
1001	1

0111

destination address in arriving packet's header

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)*
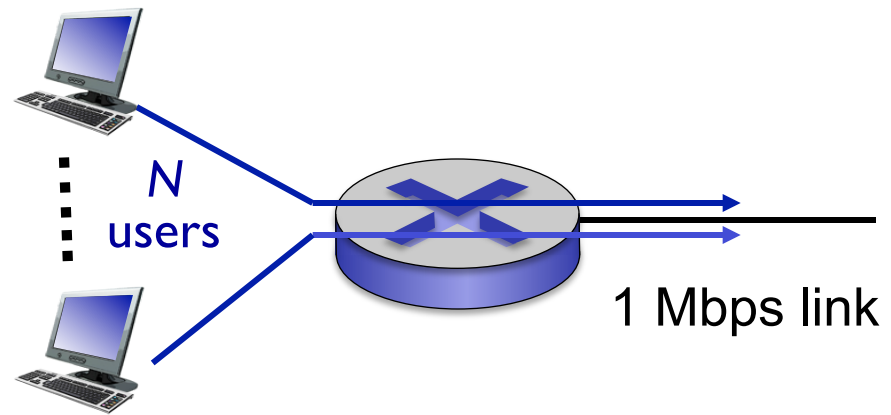- commonly used in traditional telephone networks
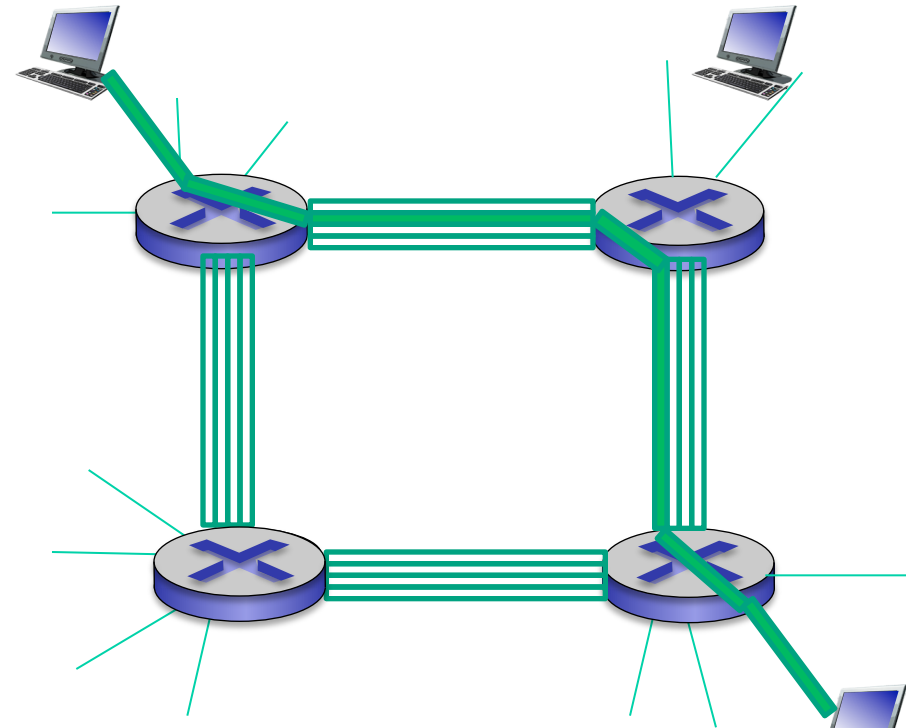
Packet switching versus circuit switching

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 *

N users

1 Mbps link

Q: how did we get value 0.0004?

Q: what happens if > 35 users ?

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

Packet switching versus circuit switching

is packet switching a "slam dunk winner?"

- great for bursty data
  - 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

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

break

# Lecture 1: roadmap

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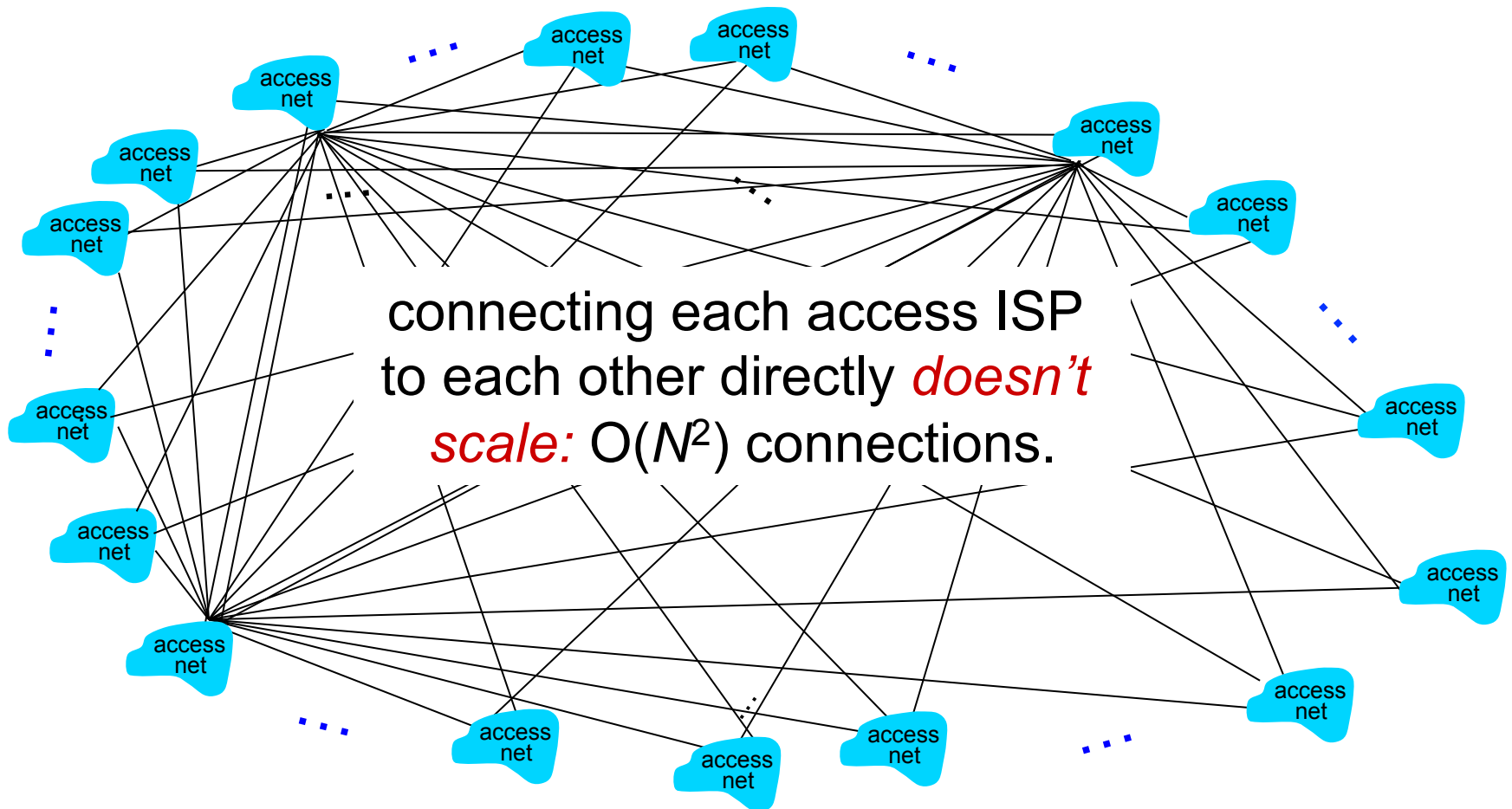- *Let's take a stepwise approach to describe current Internet structure*

Internet structure: network of networks

Question: given *millions* of access ISPs, how to connect them together?
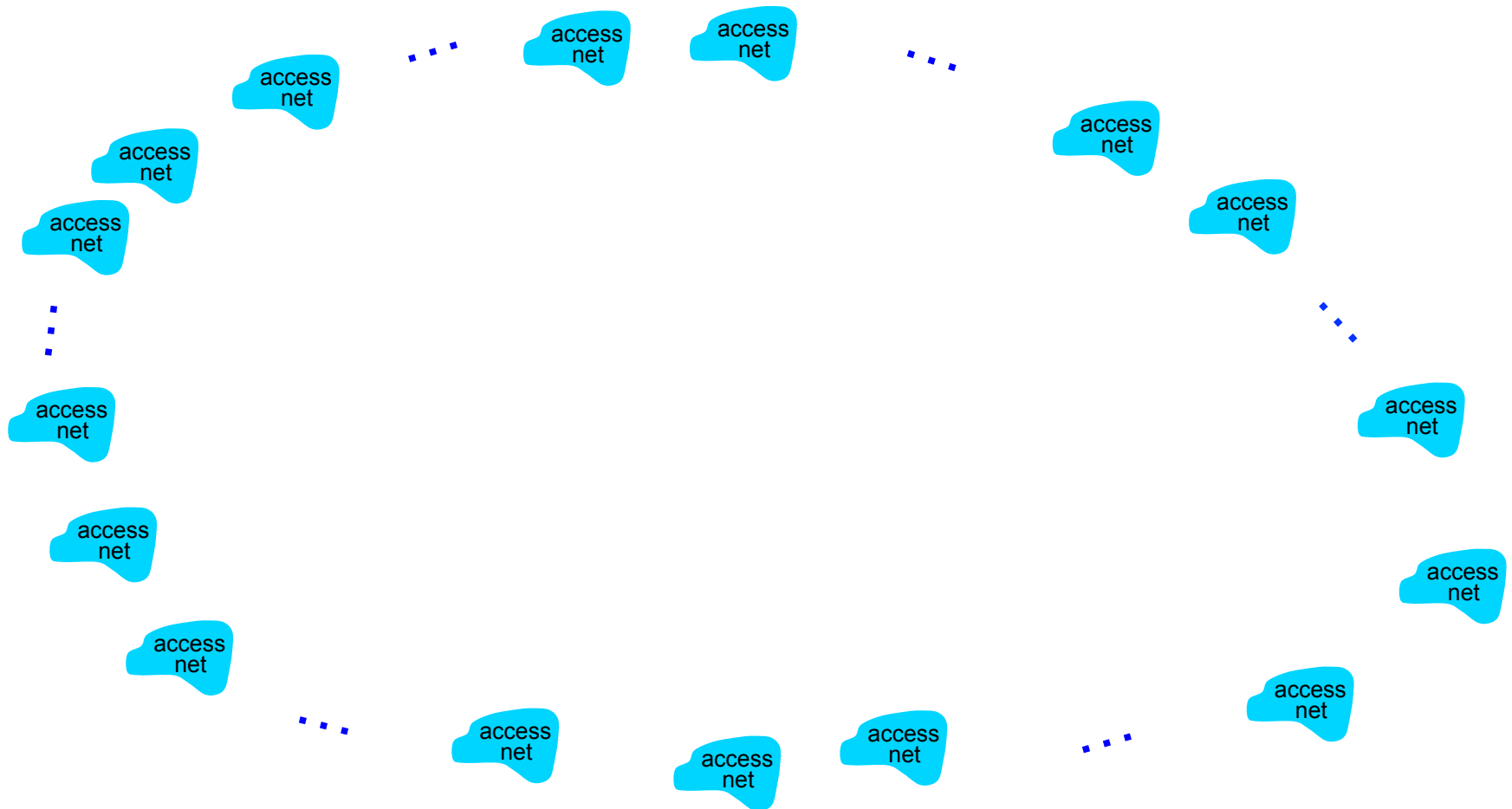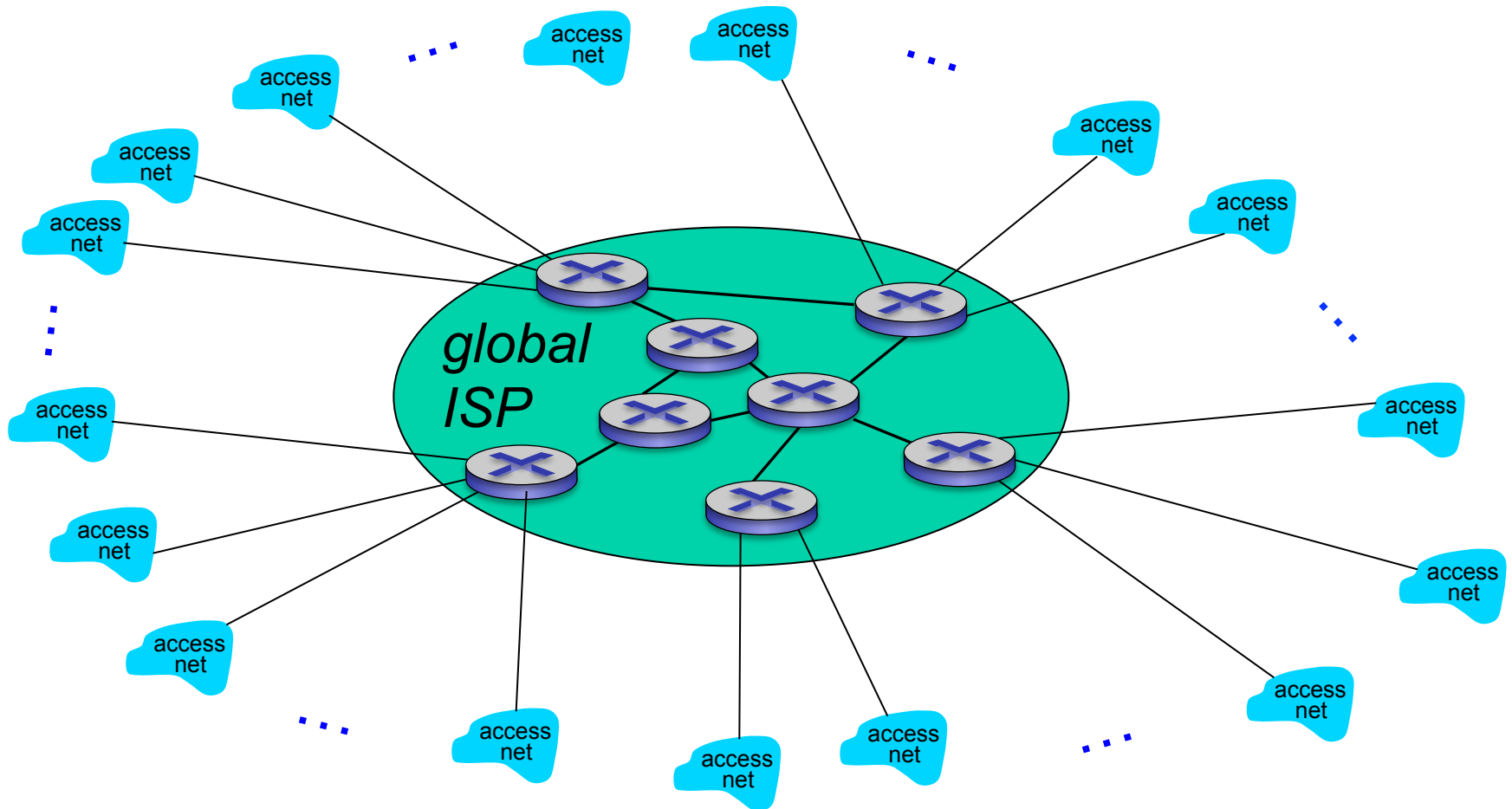
Internet structure: network of networks

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.

access net (repeated for each node in the diagram)

Internet structure: network of networks

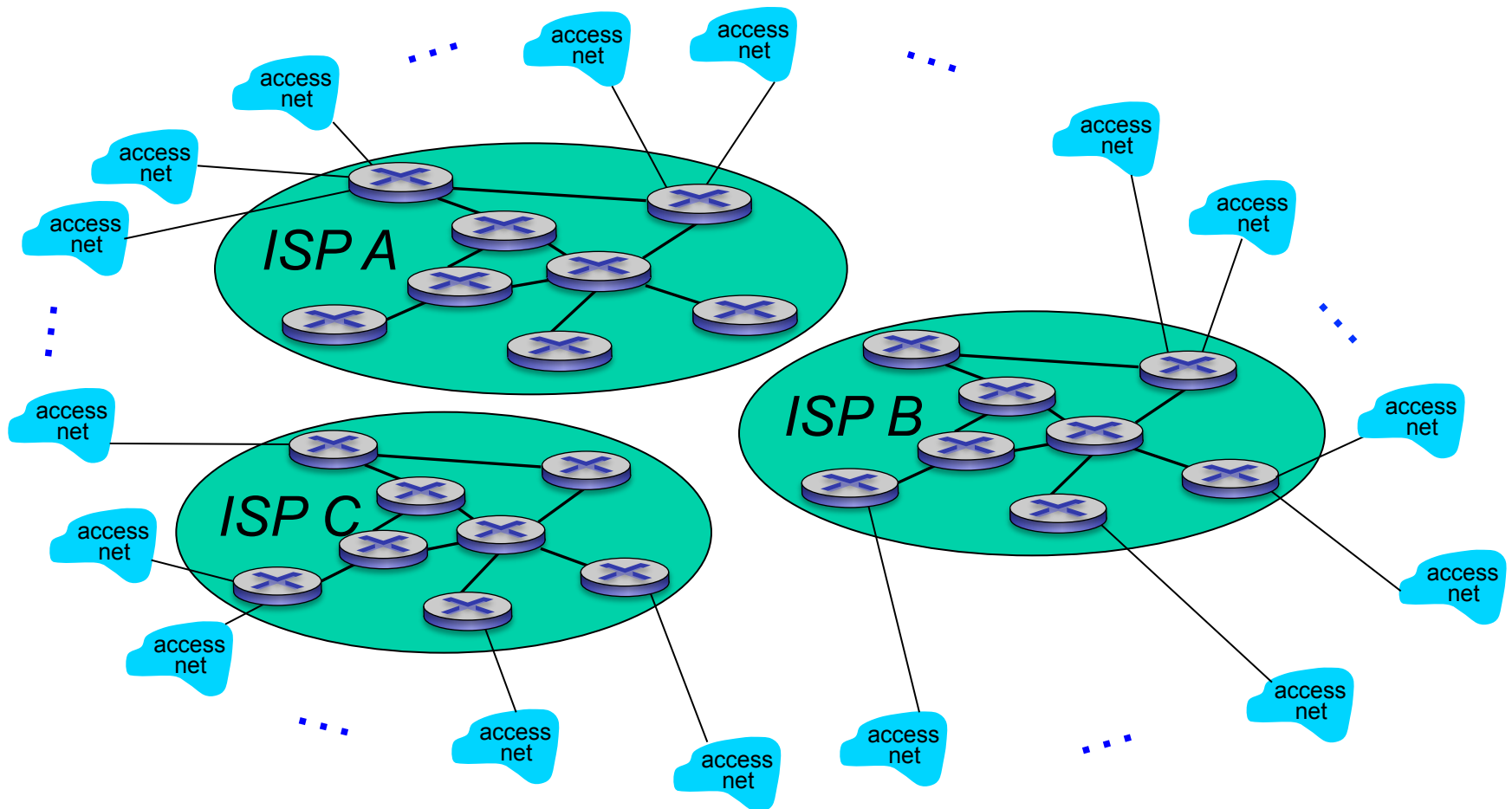*Option:* connect each access ISP to one global transit ISP?
Customer and *provider* ISPs have economic agreement.

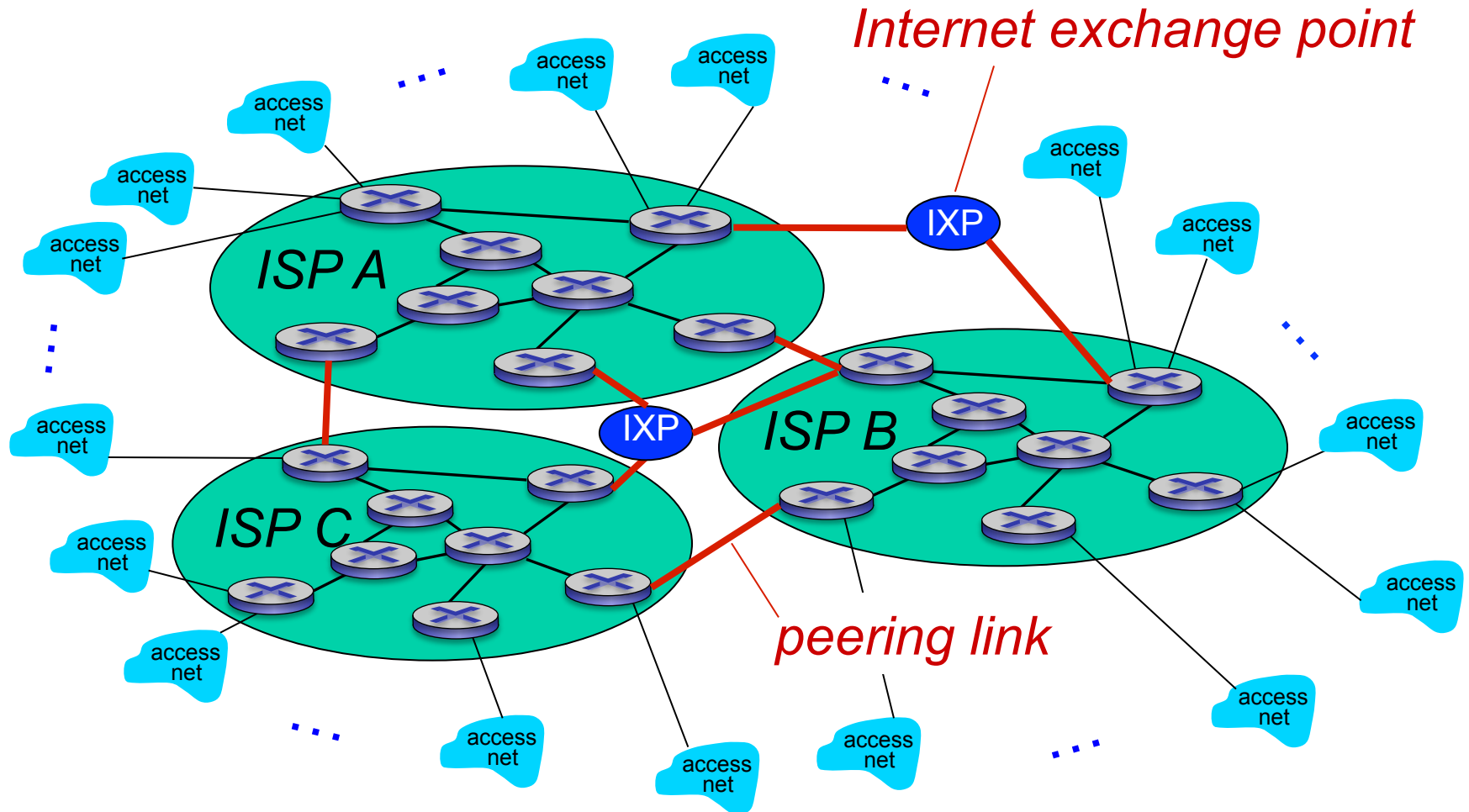# Internet structure: network of networks

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 exchange point

peering link

Internet structure: network of networks

… and regional networks may arise to connect access nets to ISPs
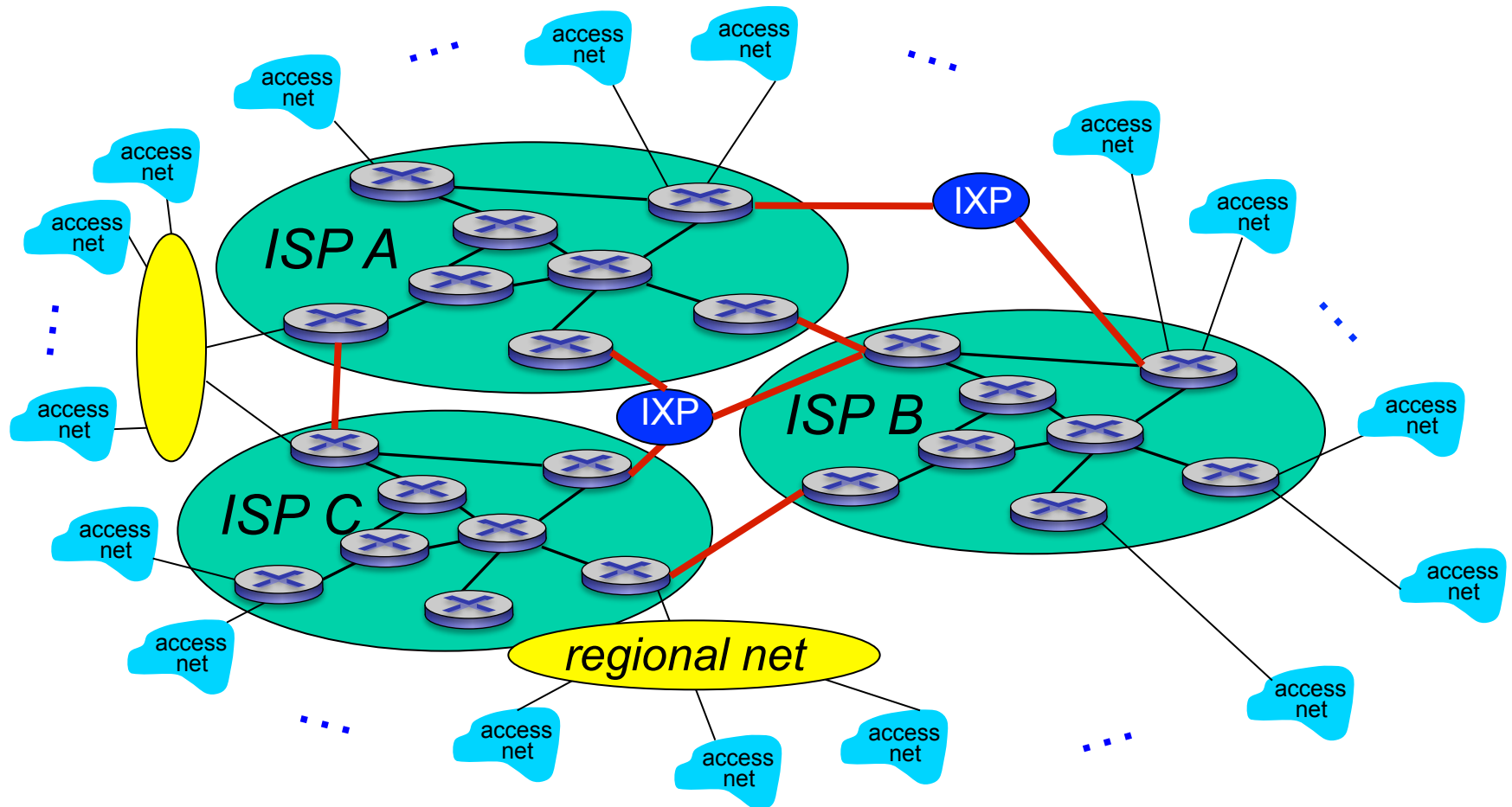
Internet structure: network of networks

… and content provider networks (e.g., Google, Microsoft, Akamai) may run their own network, to bring services, content close to end users
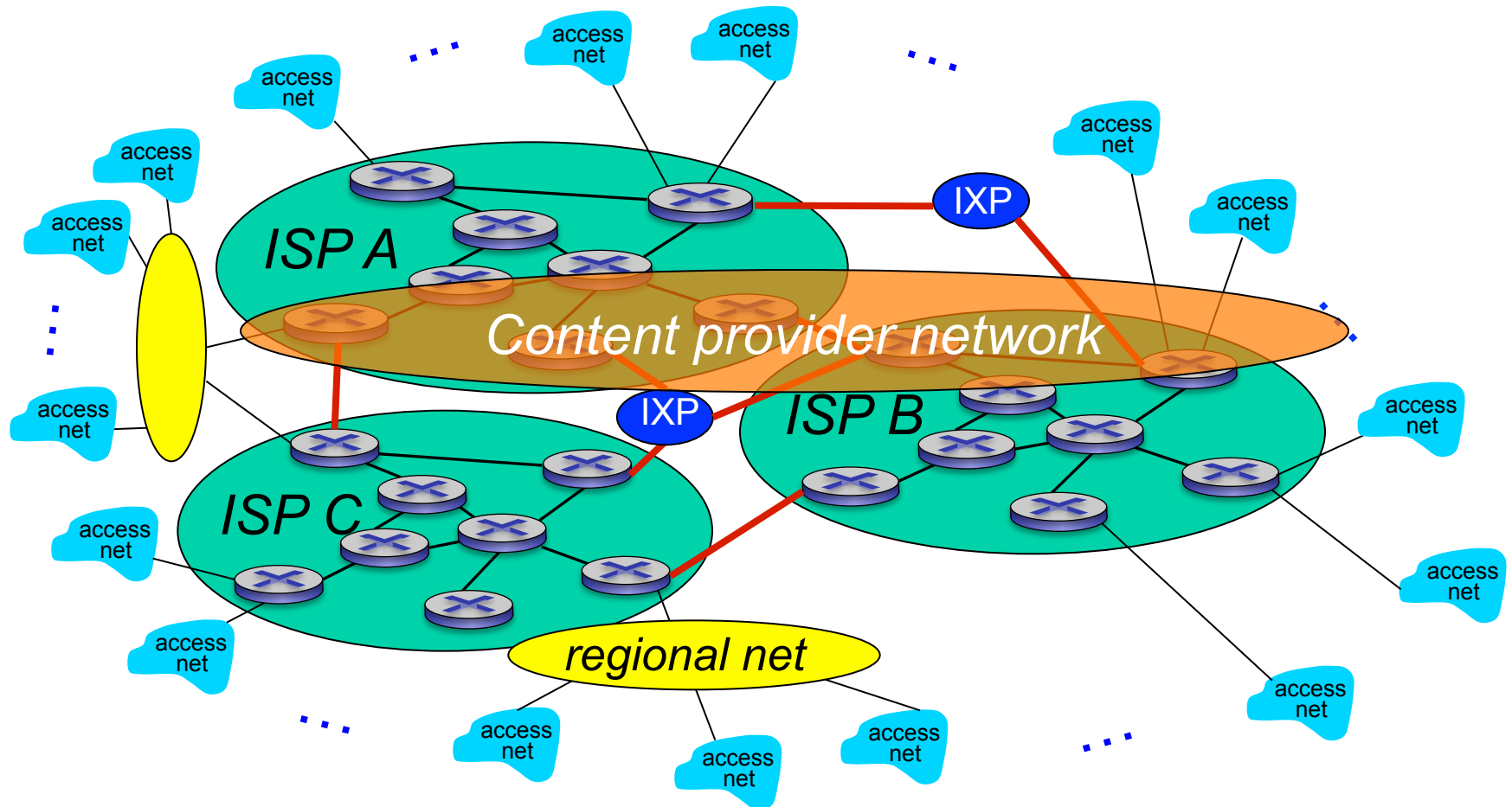
Internet structure: network of networks
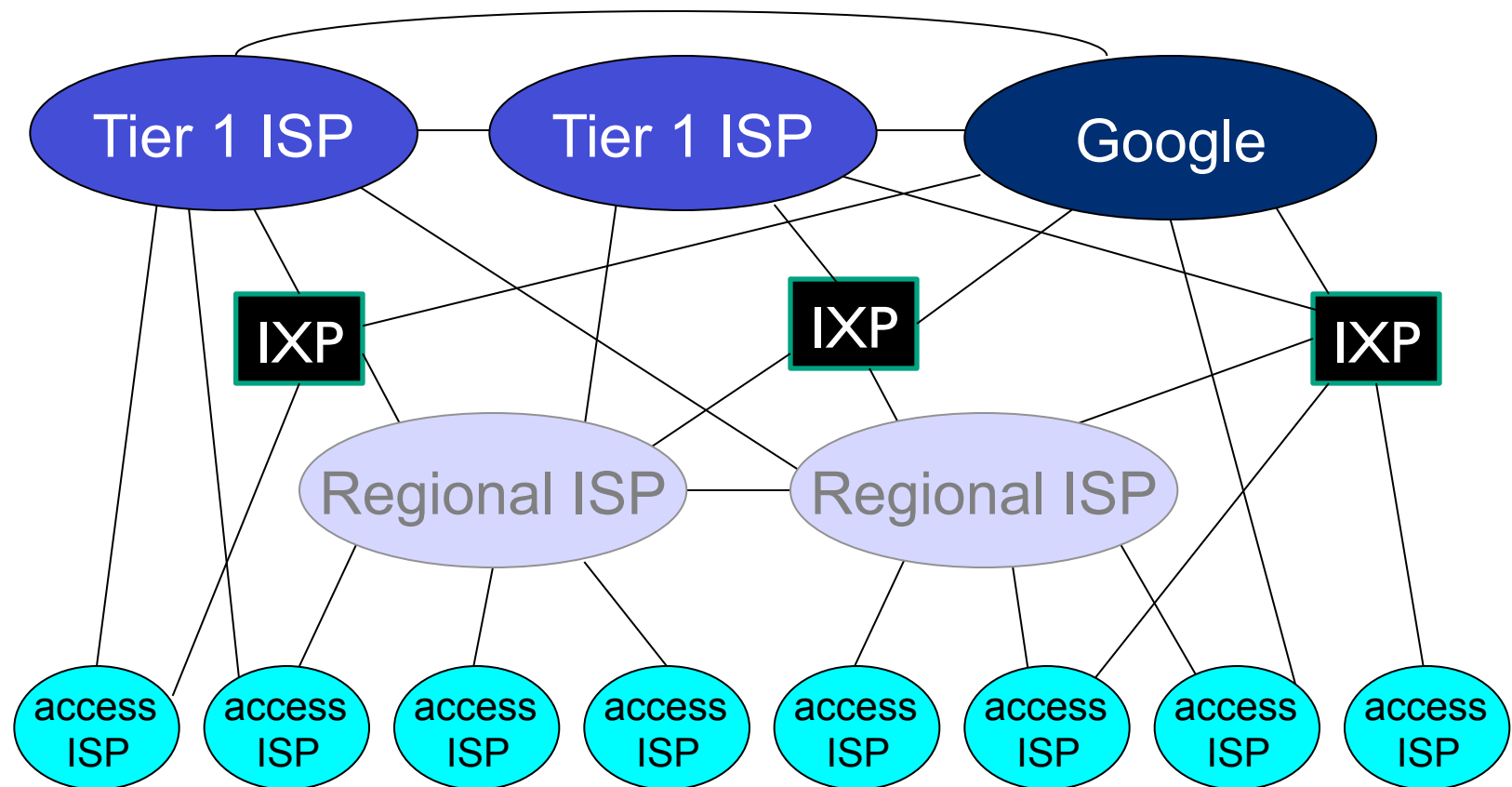
- at center: small # of well-connected large networks
  - "tier-1" commercial ISPs (e.g., Level 3, Sprint, AT&T, NTT), national & international coverage
  - content provider network (e.g., Google): private network that connects it data centers to Internet, often bypassing tier-1, regional ISPs

Lecture 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

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

packet being transmitted (delay)

A

B

packets queueing (delay)

free (available) buffers: arriving packets dropped (loss) if no free buffers
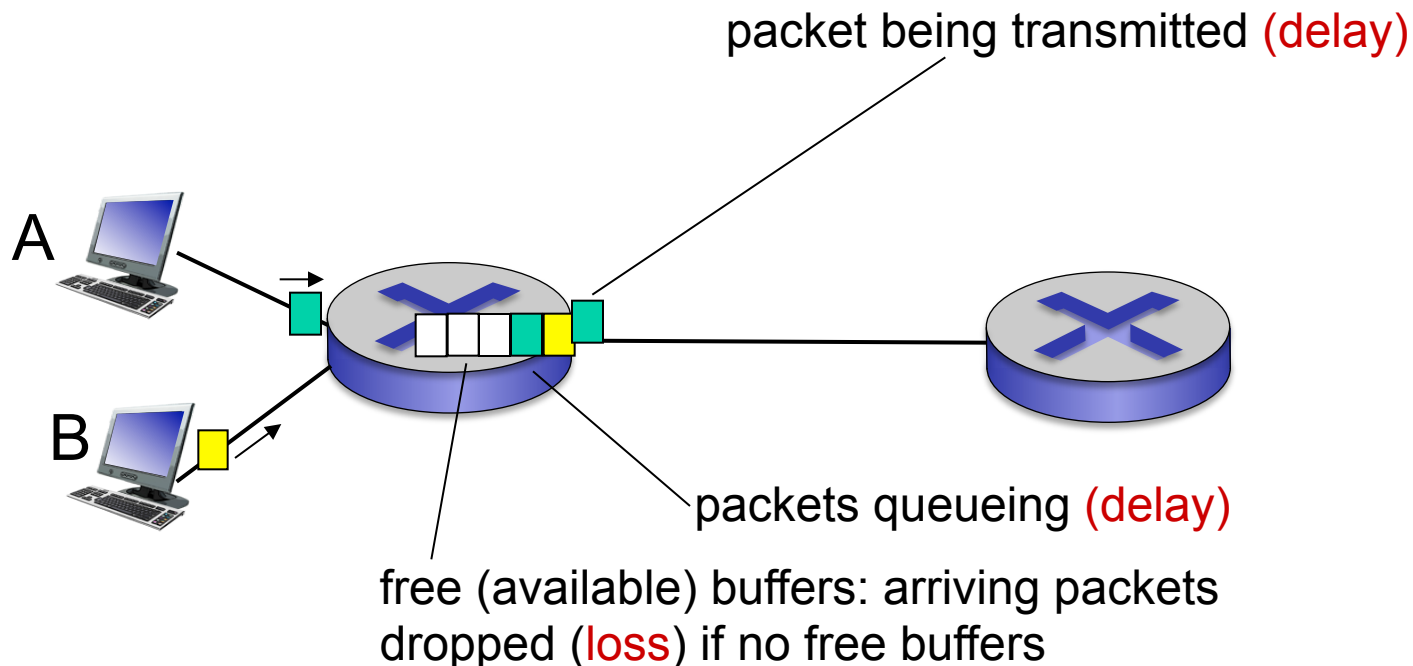
Four sources of packet delay
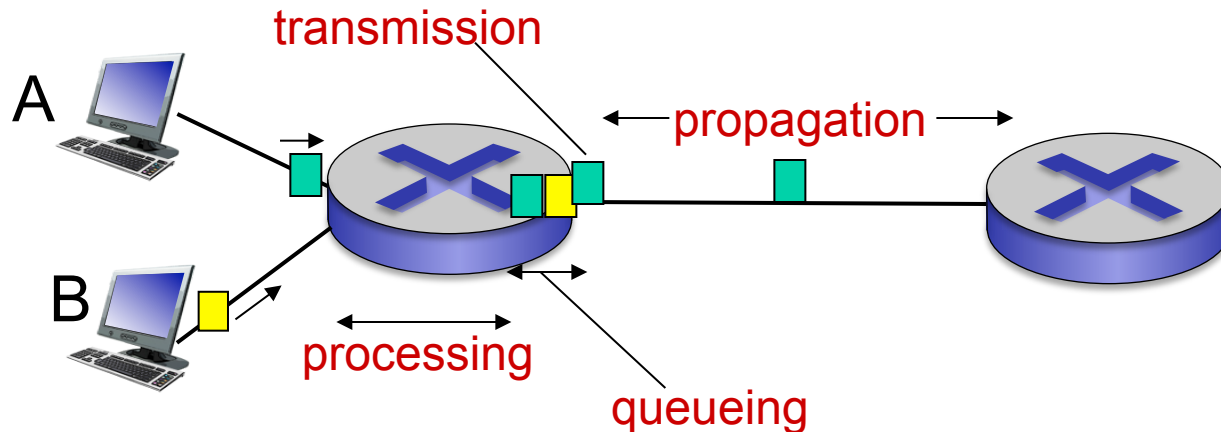
$$d_{nodal} = d_{proc} + d_{queue} + d_{trans} + d_{prop}$$

d_{proc}: processing delay
- 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

transmission

A

propagation

B

processing
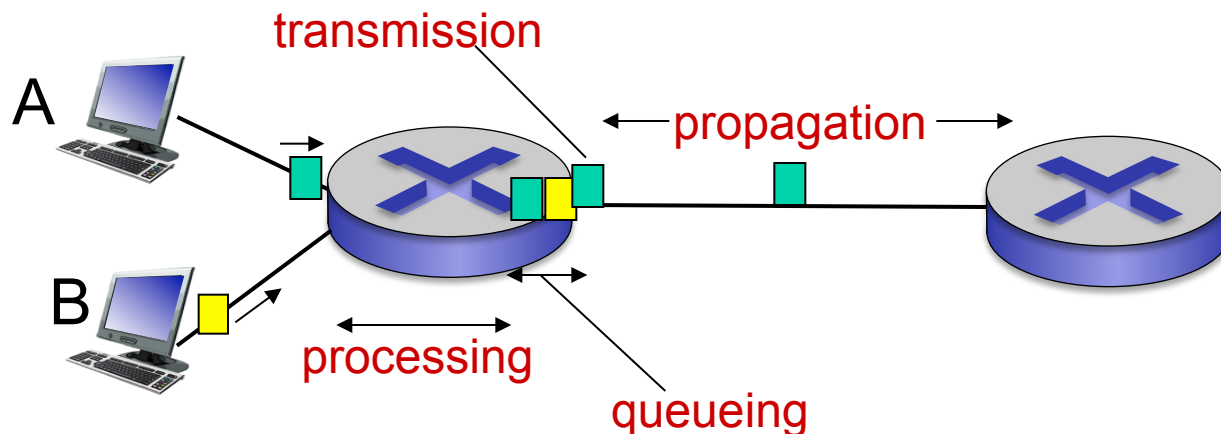
queueing

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

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 (~2x10^8 m/sec)
- $d_{prop} = d/s$

Queueing delay (revisited)

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

average queueing delay

traffic intensity = *La/R*

La/R

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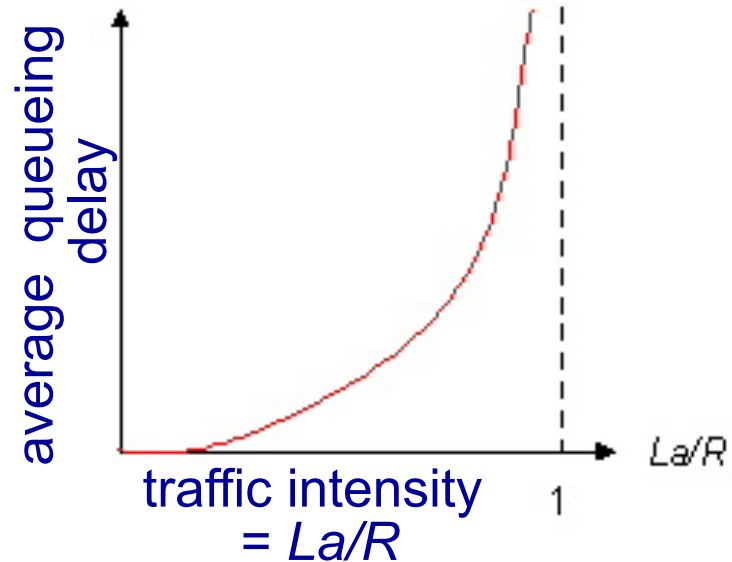
- *La/R* ~ 0: avg. queueing delay small
- *La/R* -> 1: avg. queueing delay large
- *La/R* > 1: more "work" arriving than can be serviced, average delay infinite!

La/R ~ 0

La/R -> 1

"Real" Internet delays and routes

- what do "real" Internet delay & loss look like?
- **traceroute** program: provides delay measurement from source to router along end-end Internet path towards destination. For all *i*:
  - sends three packets that will reach router *i* on path towards destination
  - router *i* will return packets to sender
  - sender times interval between transmission and reply.

3 probes
3 probes
3 probes

"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

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 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

A

B

buffer
(waiting area)

packet being transmitted

packet arriving to
full buffer is *lost*
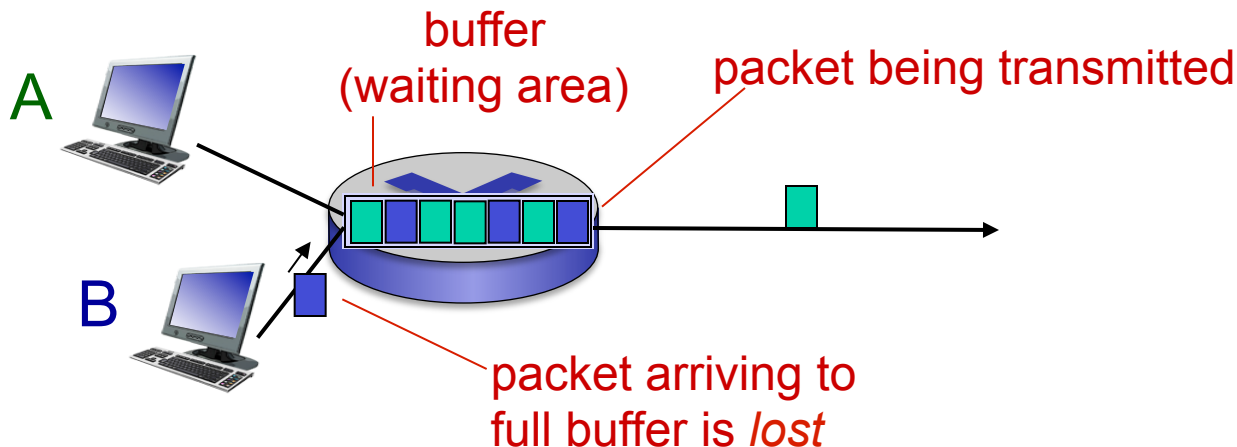
Throughput

- *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

server sends bits
(fluid) into pipe

pipe that can carry
fluid at rate
R_S bits/sec)

pipe that can carry
fluid at rate
R_C bits/sec)
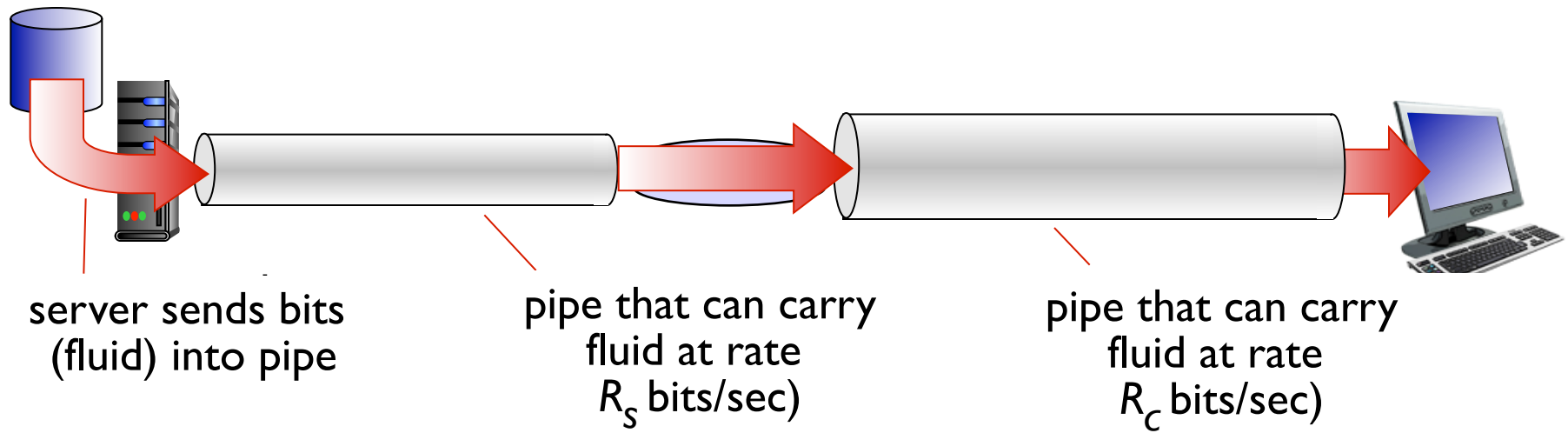
Throughput (more)

- $R_s < R_c$ What is average end-end throughput?
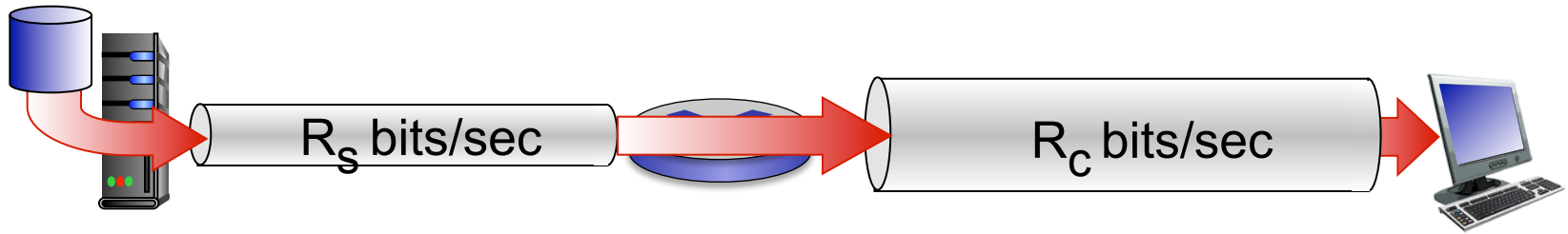
- $R_s > R_c$ What is average end-end throughput?
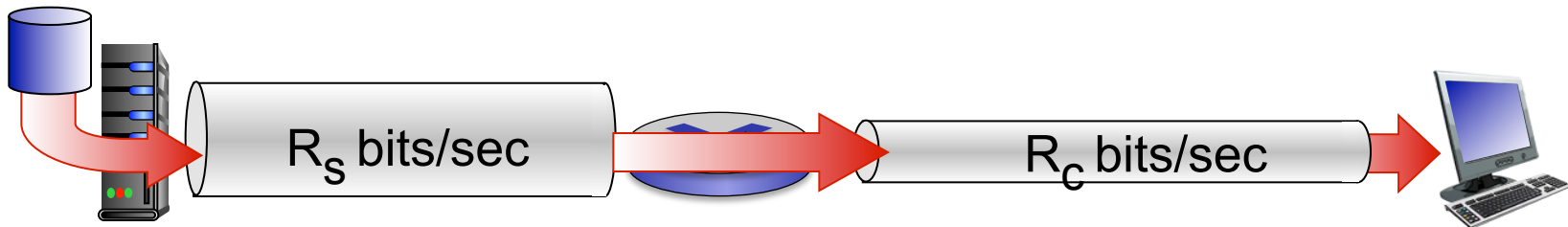
bottleneck link

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

Throughput: Internet scenario

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

R_s

R_s

R_s

R_s

R

R_c

R_c

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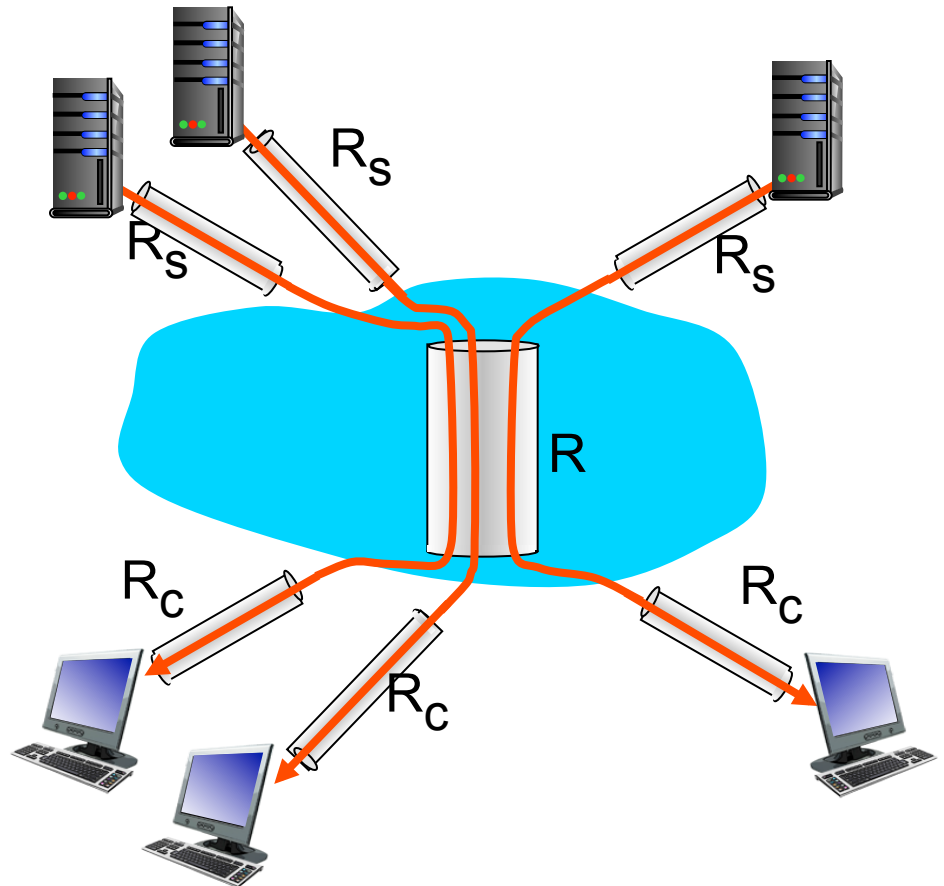
10 connections (fairly) share backbone bottleneck link R bits/sec

Lecture 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

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

ticket (purchase)	ticket (complain)
baggage (check)	baggage (claim)
gates (load)	gates (unload)
runway takeoff	runway landing
airplane routing	airplane routing

airplane routing
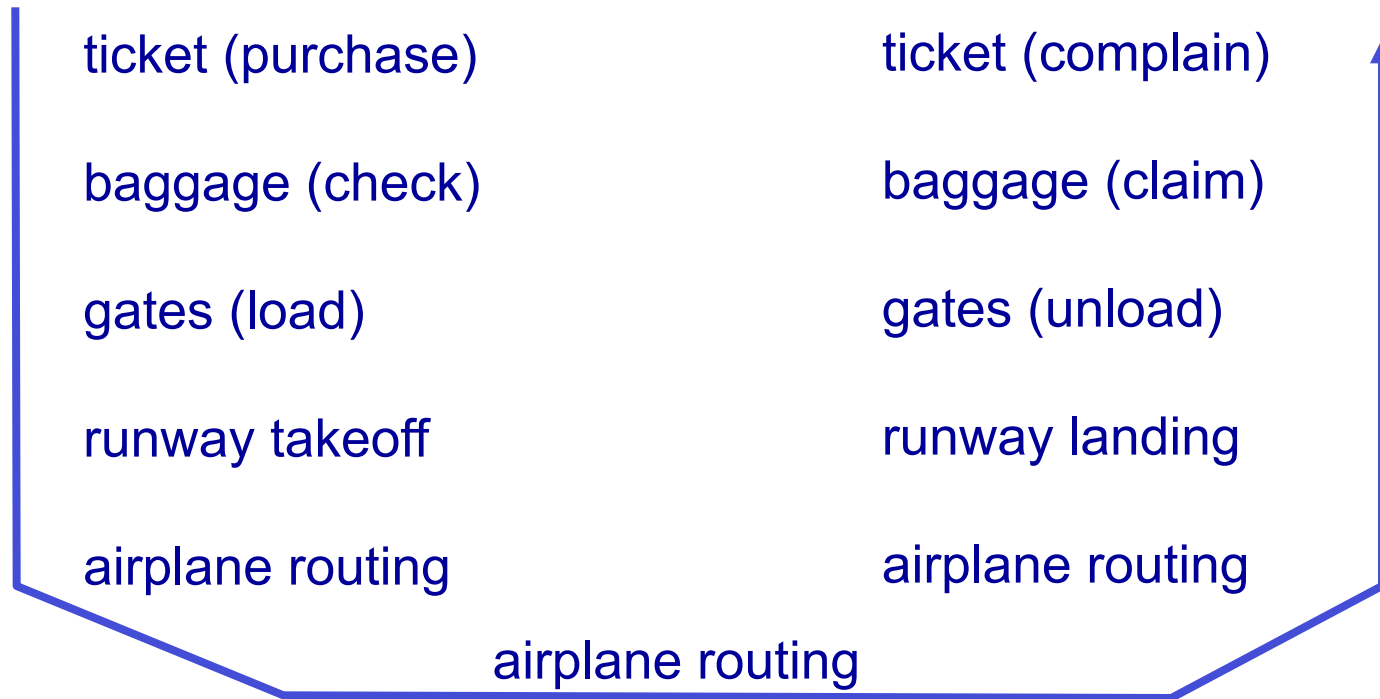
- a series of steps

Layering of airline functionality

departure airport	intermediate air-traffic control centers	arrival airport	
ticket (purchase)		ticket (complain)	ticket
baggage (check)		baggage (claim	baggage
gates (load)		gates (unload)	gate
runway (takeoff)		runway (land)	takeoff/landing
airplane routing	airplane routing airplane routing	airplane routing	airplane routing
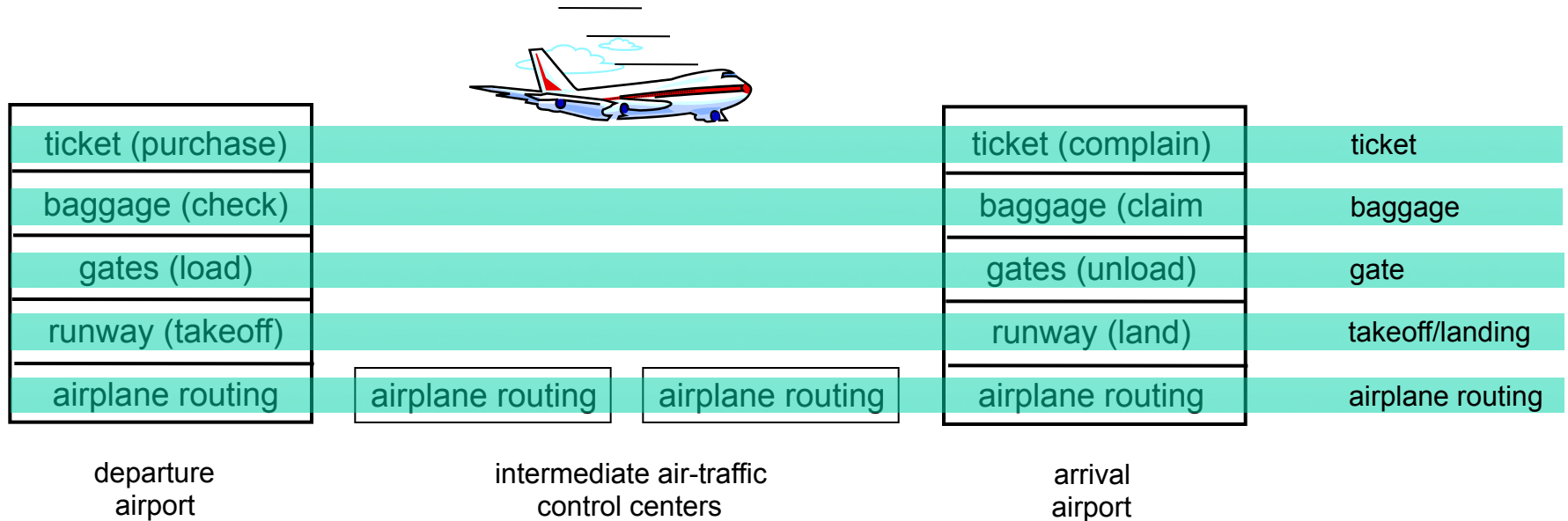
layers: each layer implements a service
- via its own internal-layer actions
- relying on services provided by layer below

Why layering?

dealing with complex systems:

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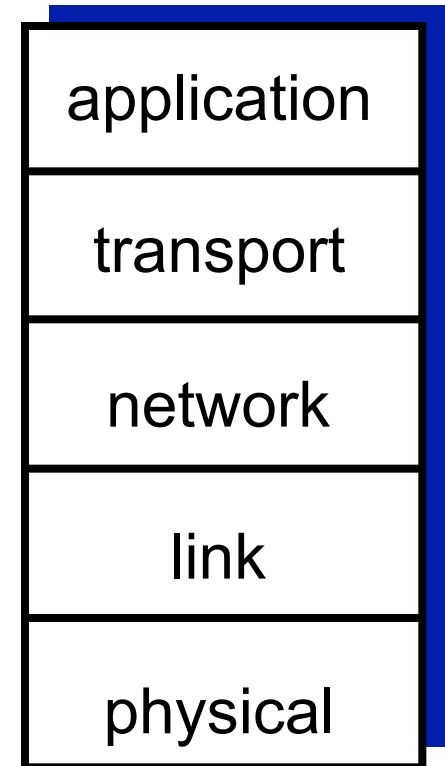- modularization eases maintenance, updating of system
  - change of implementation of layer's service transparent to rest of system
  - e.g., change in gate procedure doesn't affect rest of system
- layering considered harmful?

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"

application
transport
network
link
physical

Encapsulation

message   M

segment H_t M

datagram H_n H_t M

frame H_l H_n H_t M

source

| application |
| transport |
| network |
| link |
| physical |

| link |
| physical |

switch

destination

M

H_t M

H_n H_t M

H_l H_n H_t M

| application |
| transport |
| network |
| link |
| physical |

H_n H_t M

H_l H_n H_t M

| network |
| link |
| physical |

H_n H_t M

router