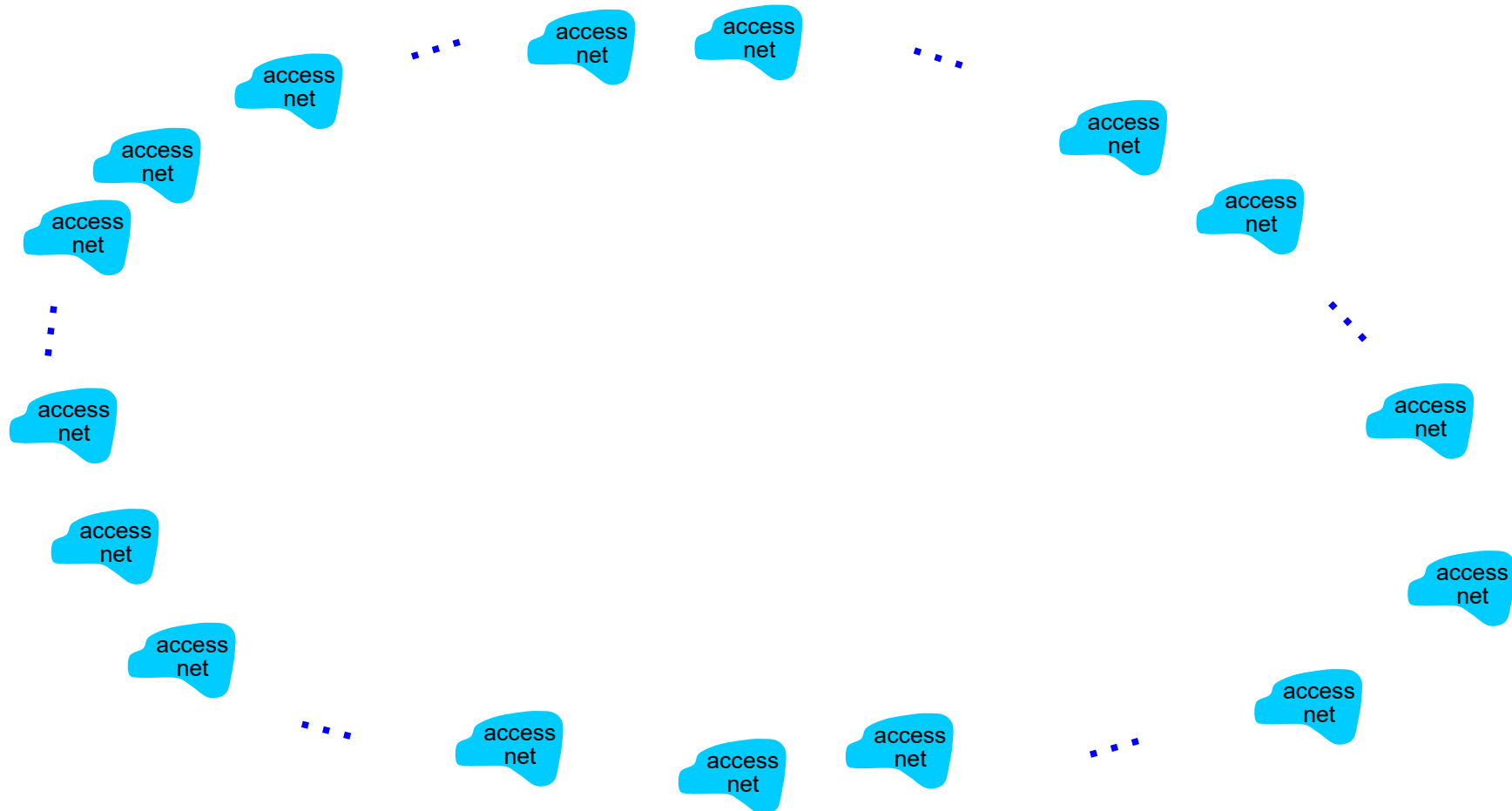


# Internet structure: a “network of networks”

- Hosts connect to Internet via **access** Internet Service Providers (ISPs)
  - residential, enterprise (company, university, commercial) 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
- Let's take a stepwise approach to describe current Internet structure

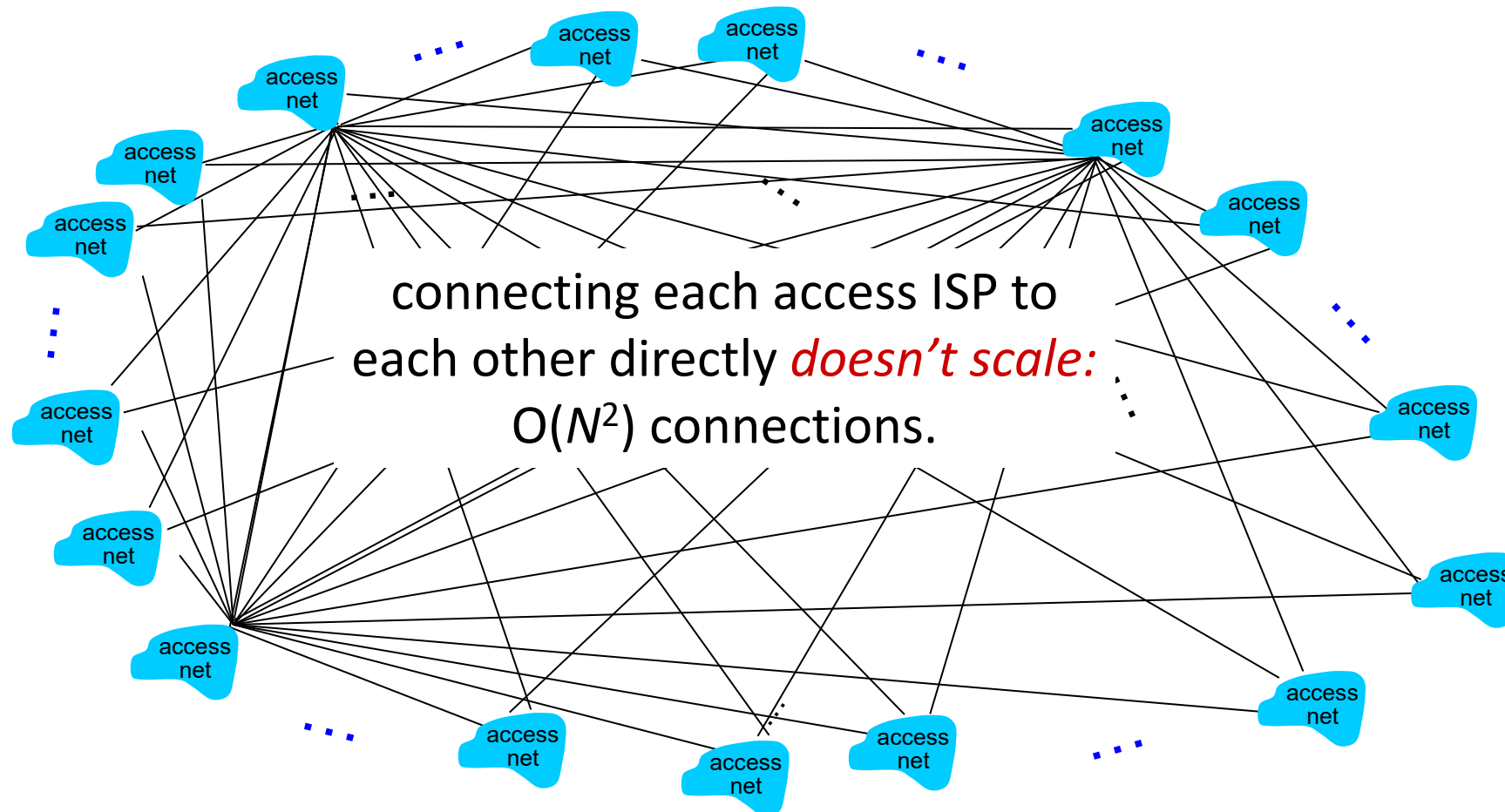
# Internet structure: a “network of networks”

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



# Internet structure: a “network of networks”

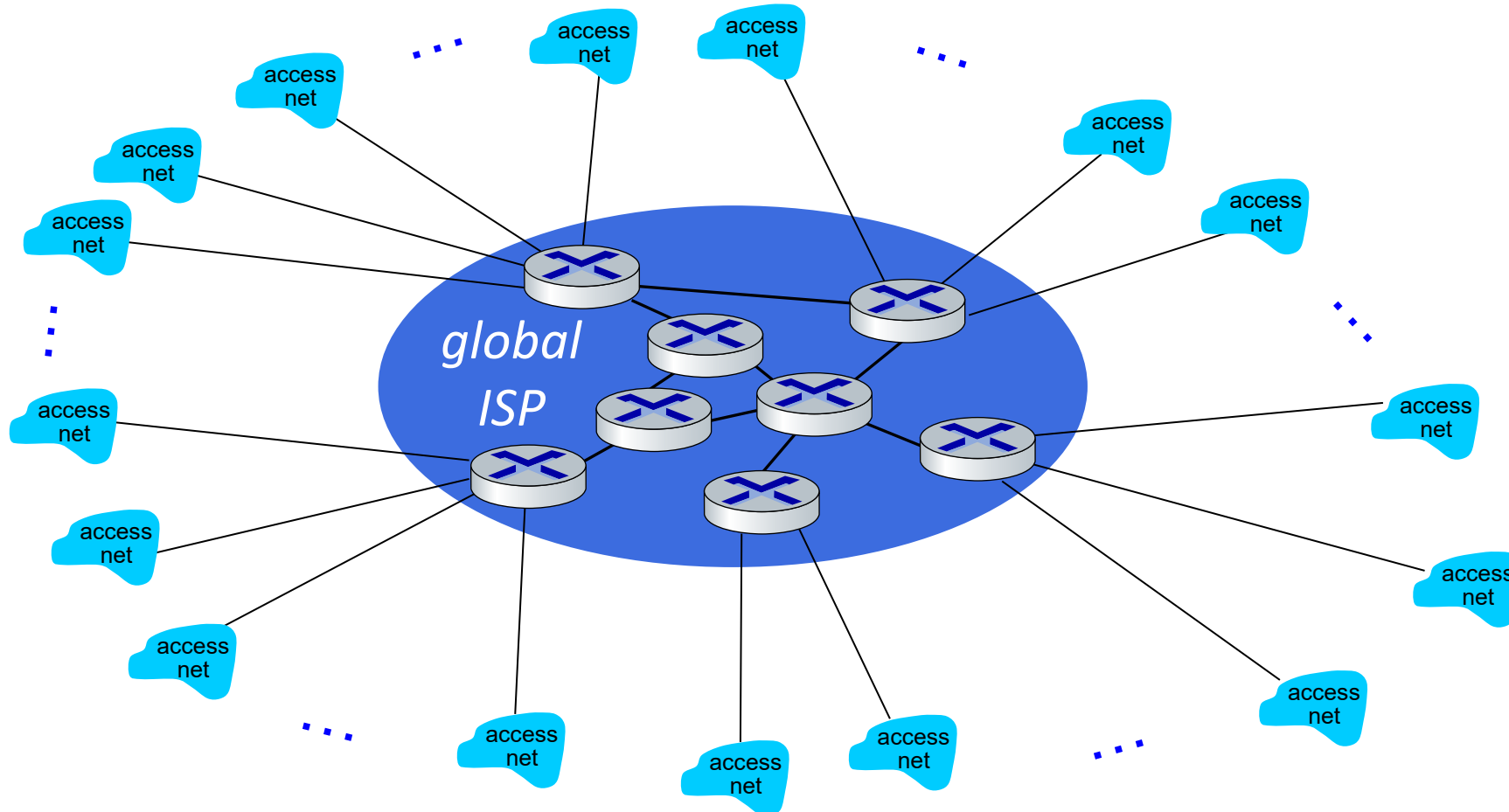
*Question:* given *millions* of access ISPs, how to connect them together?



# Internet structure: a “network of networks”

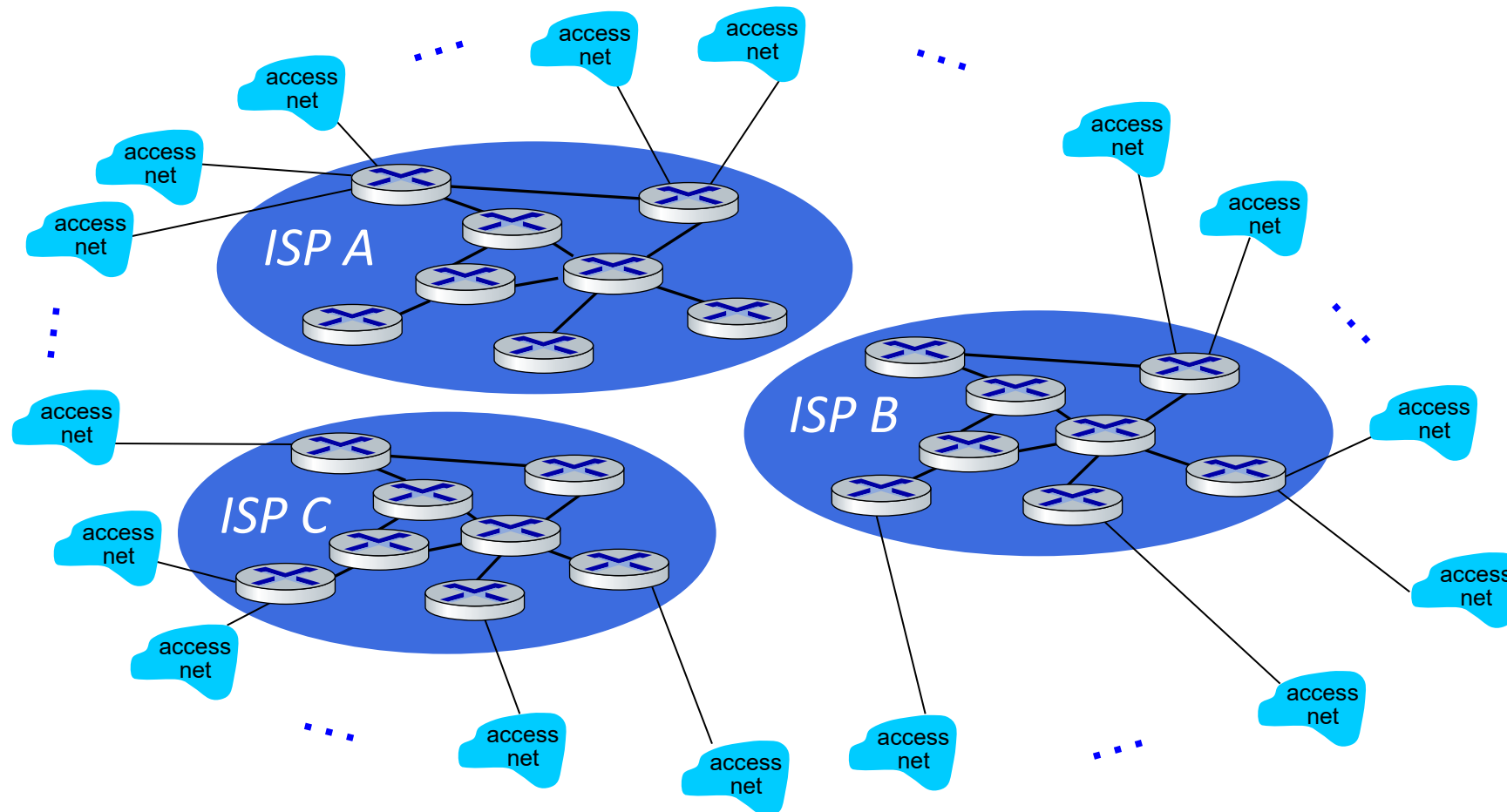
*Option: connect each access ISP to one global transit ISP?*

*Customer and provider ISPs have economic agreement.*



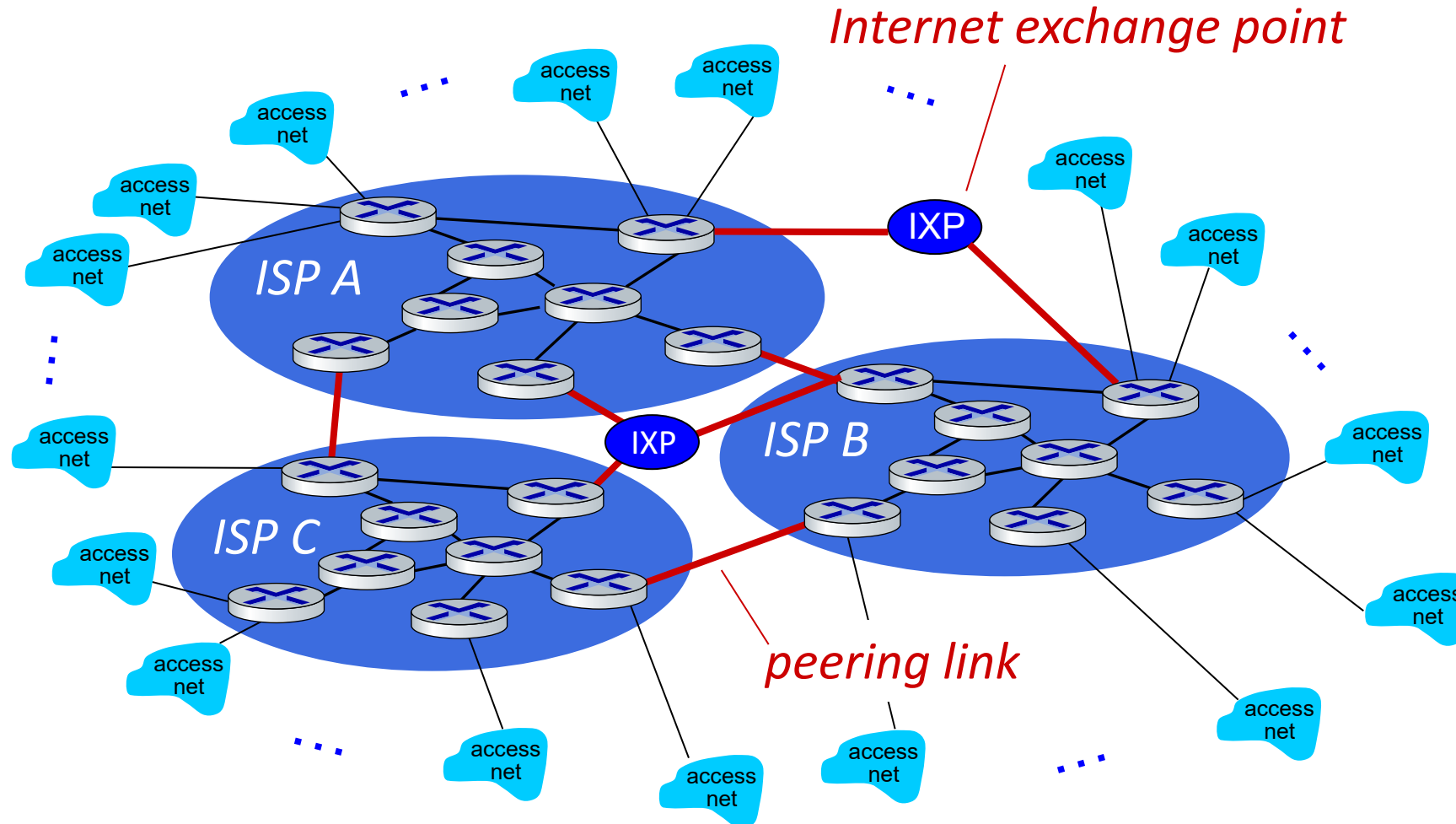
# Internet structure: a “network of networks”

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



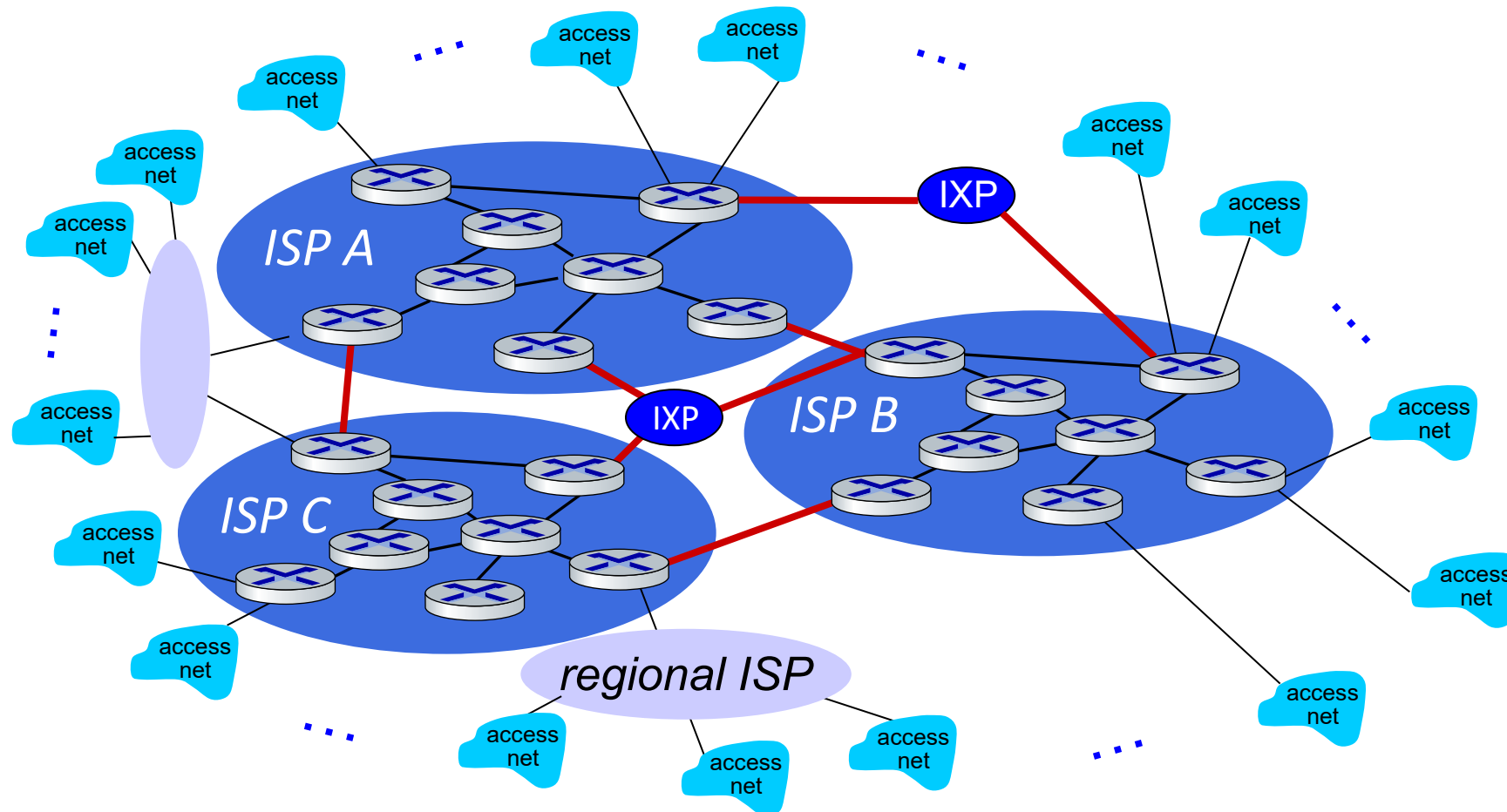
# Internet structure: a “network of networks”

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



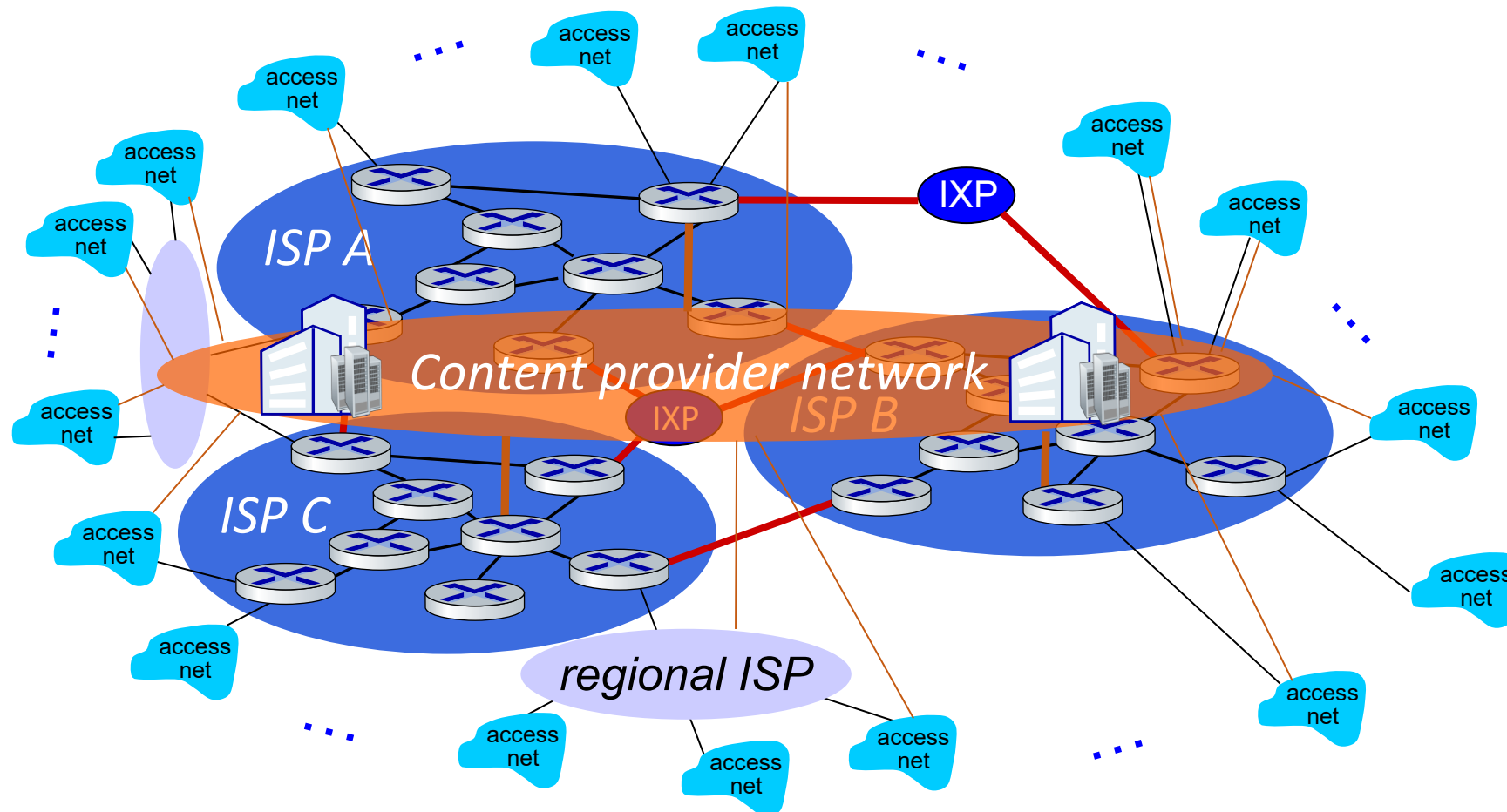
# Internet structure: a “network of networks”

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

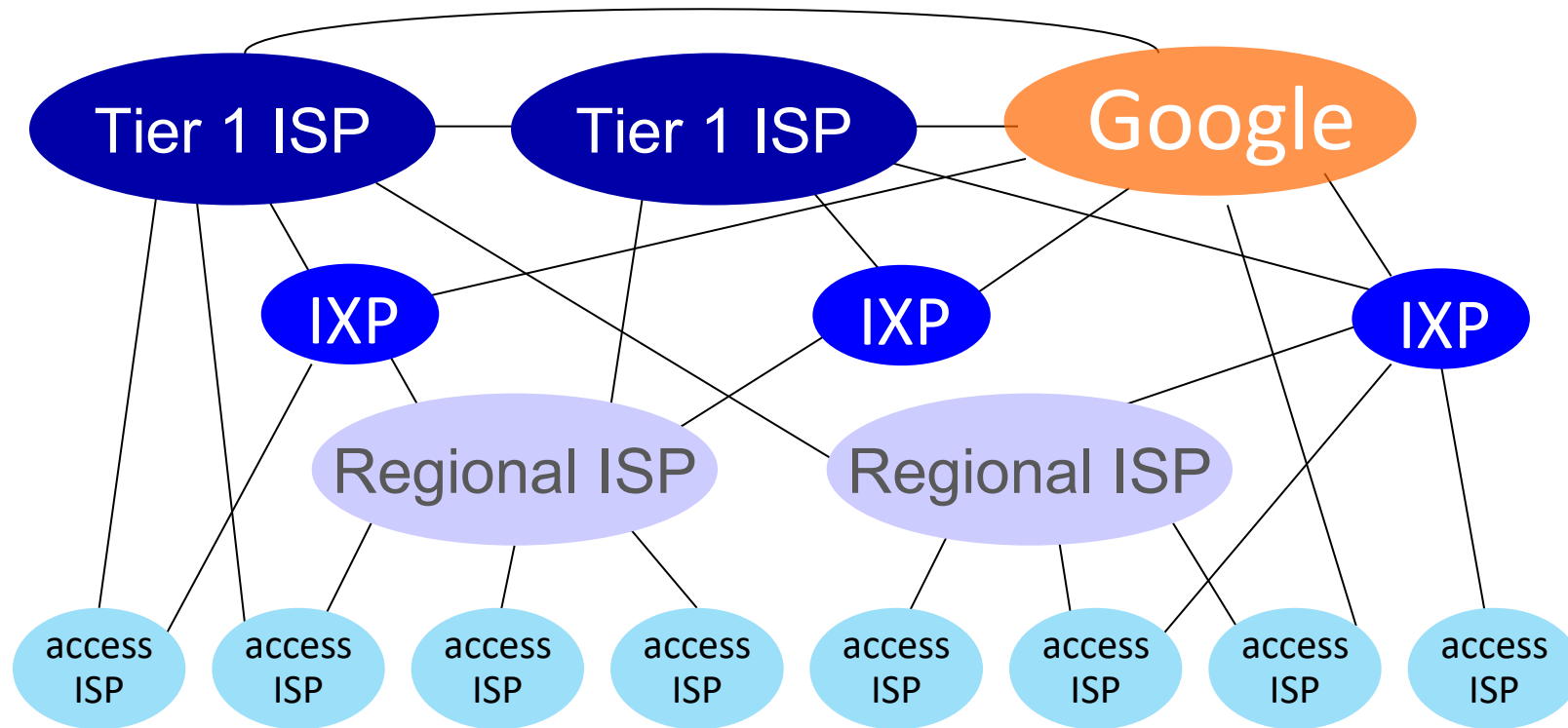


# Internet structure: a “network of networks”

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



# Internet structure: a “network of networks”



At “center”: small # of well-connected large networks

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

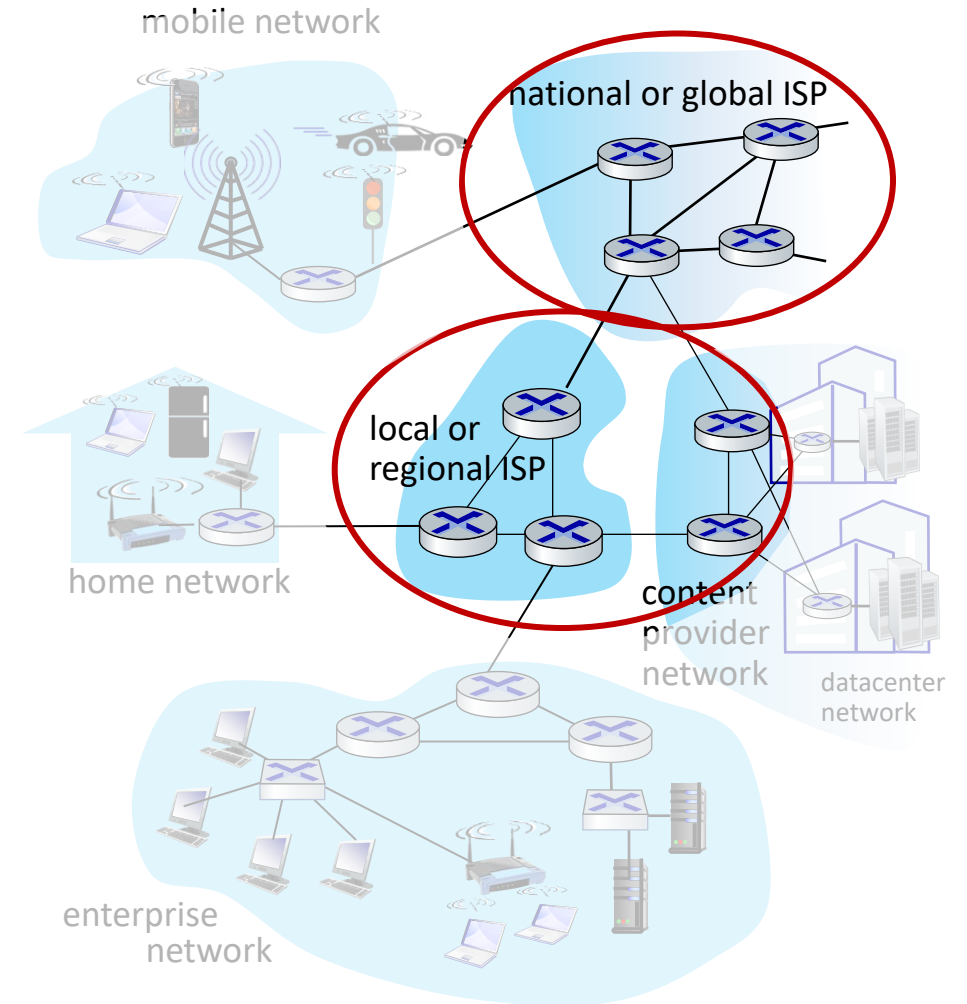
# Chapter 1: roadmap

- What *is* the Internet?
- What *is* a protocol?
- Network edge: hosts, access network, physical media, Internet structure
- **Network core:** packet/circuit switching
- Performance: loss, delay, throughput
- Security
- Protocol layers, service models
- History

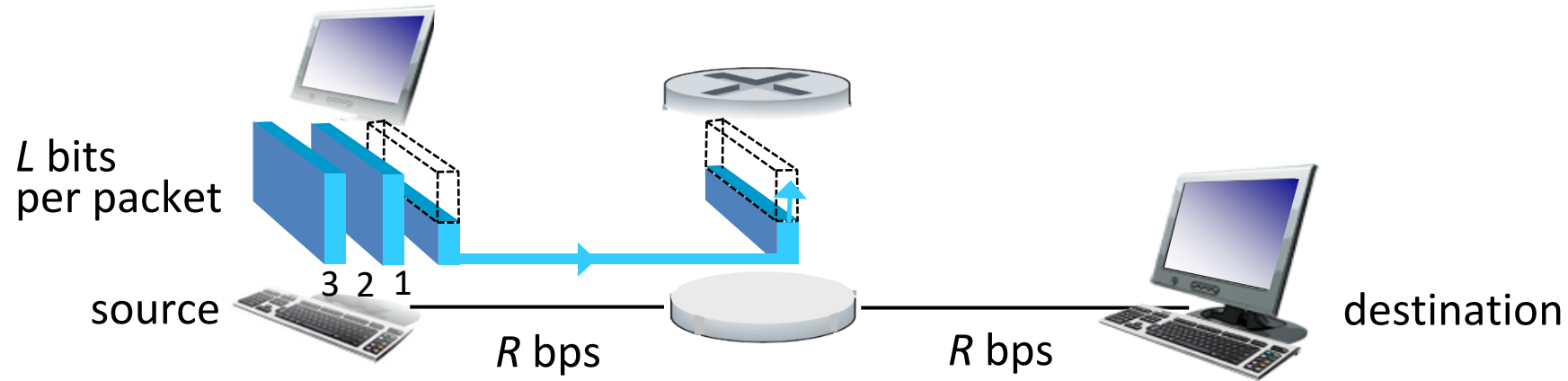


# The network core

- 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



# Packet-switching: store-and-forward

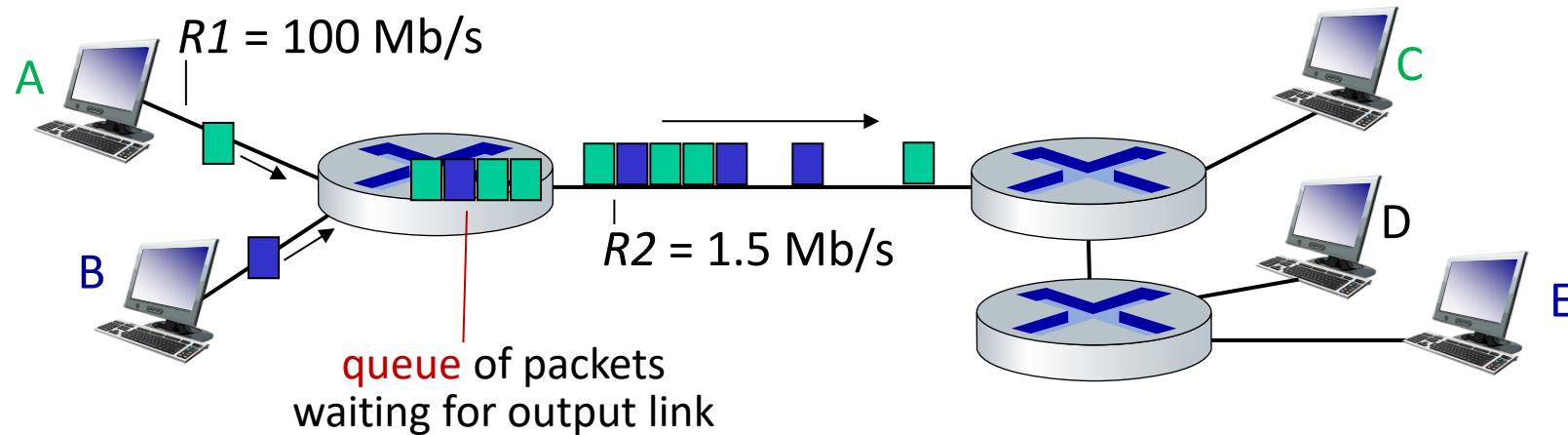


- **Transmission delay:** takes  $L/R$  seconds to transmit (push out)  $L$ -bit packet into link at  $R$  bps
- **Store and forward:** entire packet must arrive at router before it can be transmitted on next link
- **End-end delay:**  $2L/R$  (above), assuming zero propagation delay (more on delay shortly)

## *One-hop numerical example:*

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

# Packet-switching: queueing delay, loss



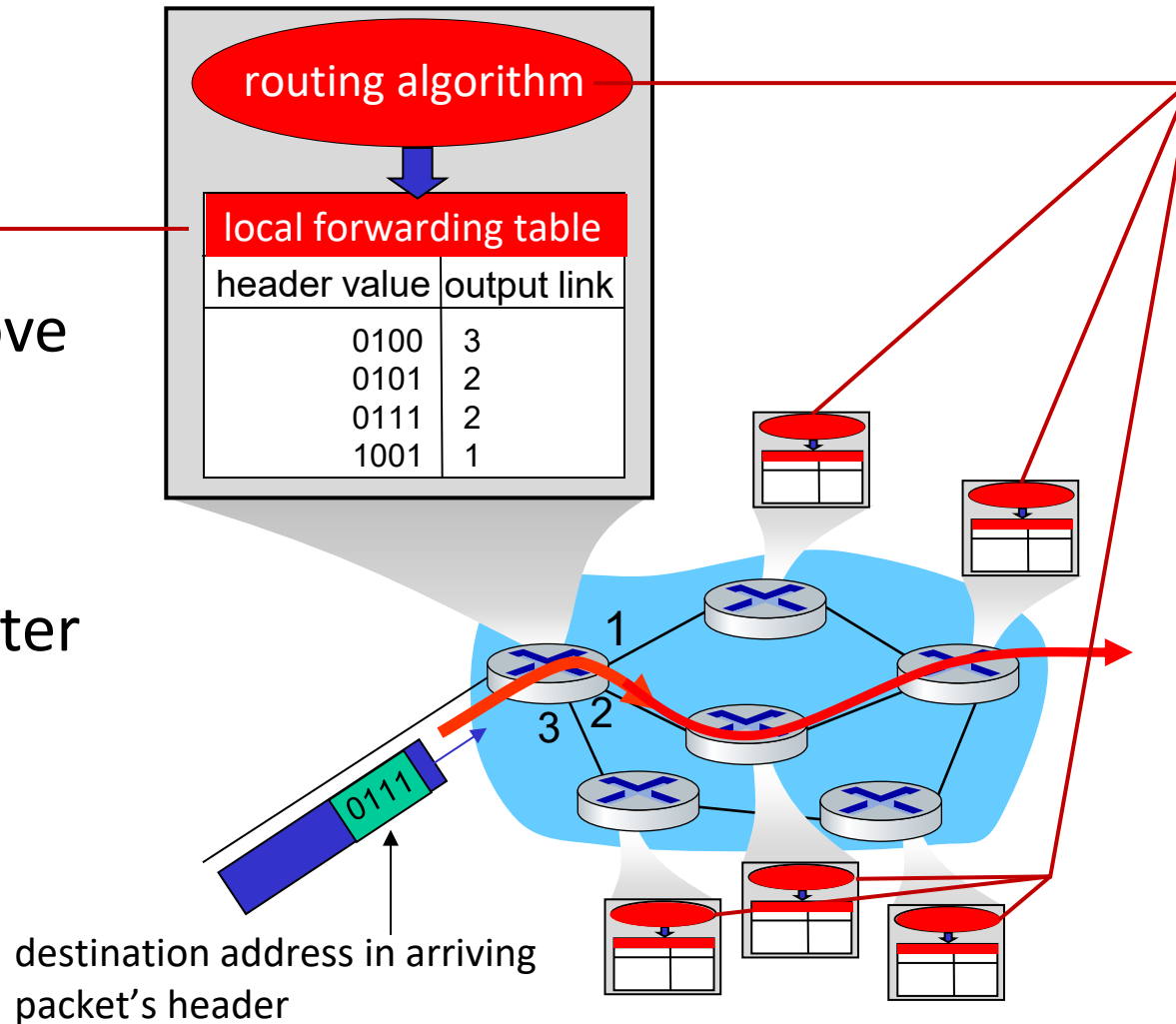
***Packet queuing and loss:*** if arrival rate (in bps) to link exceeds transmission rate (bps) of link for a period of time:

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

# Two key network-core functions

## *Forwarding:*

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



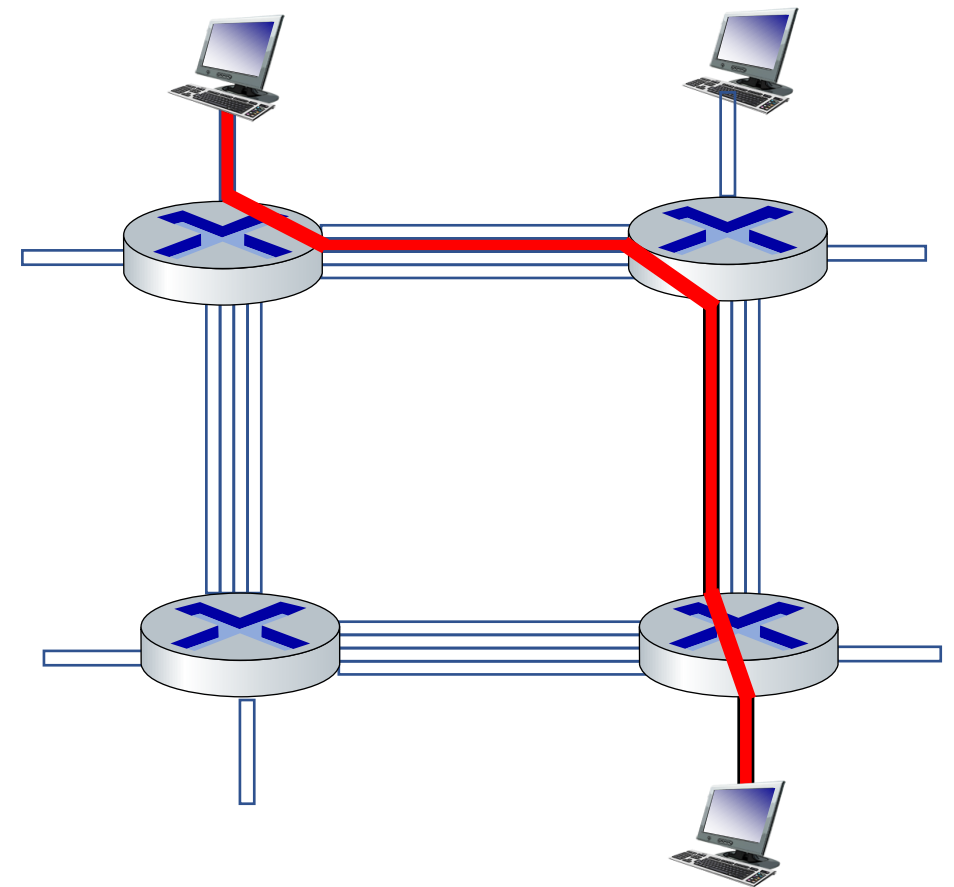
## *Routing:*

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

# Alternative to packet switching: circuit switching

end-end resources allocated to,  
reserved for “call” between source  
and destination

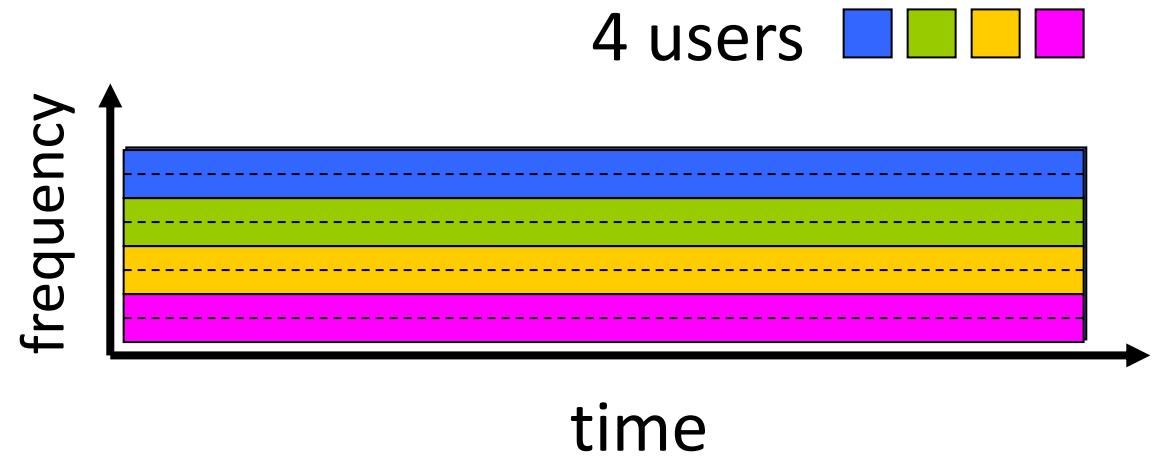
- in diagram, each link has four circuits.
  - call gets 2<sup>nd</sup> circuit in top link and 1<sup>st</sup> circuit in right link.
- dedicated resources: no sharing
  - circuit-like (guaranteed) performance
- circuit segment idle if not used by call (**no sharing**)
- commonly used in traditional telephone networks



# Circuit switching: FDM and TDM

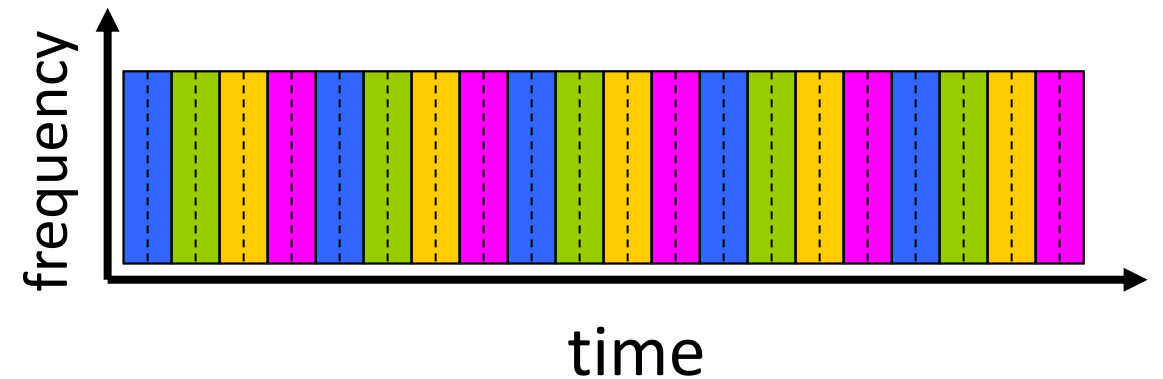
## Frequency Division Multiplexing (FDM)

- optical, electromagnetic frequencies divided into (narrow) frequency bands
- each call allocated its own band, can transmit at max rate of that narrow band



## Time Division Multiplexing (TDM)

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

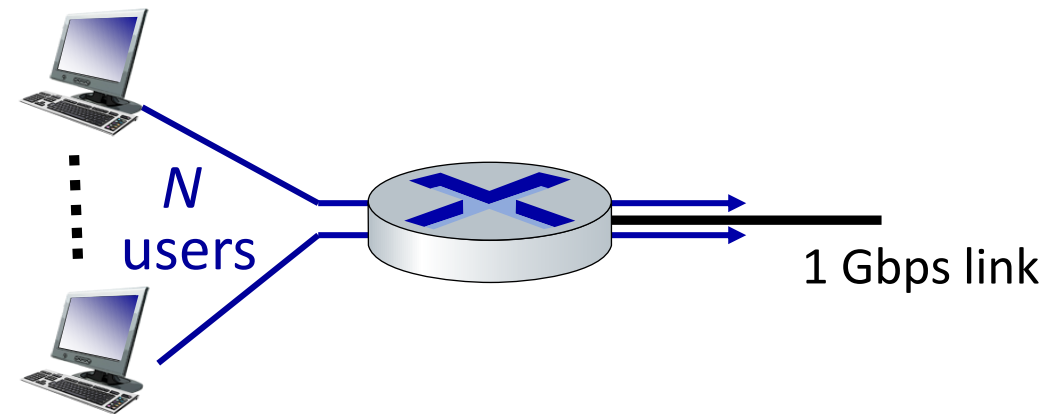


# Packet switching versus circuit switching

*packet switching allows more users to use network!*

Example:

- 1 Gb/s link
- each user:
  - 100 Mb/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 ?

\* Check out the online interactive exercises for more examples: [http://gaia.cs.umass.edu/kurose\\_ross/interactive](http://gaia.cs.umass.edu/kurose_ross/interactive)

- $N = 35$  users
- $P(\text{\# Active Users} > 10) = 1 - P(\text{\# Active Users} = 10) - P(\text{\# Active Users} = 9) - P(\text{\# Active Users} = 8) \dots P(\text{\# Active Users} = 1)$
- $P(\text{\# Active Users} = 10) = C(35, 10) * 0.1^{10} * 0.9^{25}$
- Note: Binomial Distribution

Probability of k out of n ways:

$$P(k \text{ out of } n) = \frac{n!}{k!(n-k)!} p^k (1-p)^{(n-k)}$$

**The General Binomial Probability Formula**

# Packet switching versus circuit switching

## Is packet switching a “winner”?

- great for “bursty” data – sometimes has data to send, but at other times not
  - resource sharing
  - simpler, no call setup
- **excessive congestion possible:** packet delay and loss due to buffer overflow
  - protocols needed for reliable data transfer, congestion control
- **Q: How to provide circuit-like behavior?**
  - bandwidth guarantees traditionally used for audio/video applications

# Chapter 1: roadmap

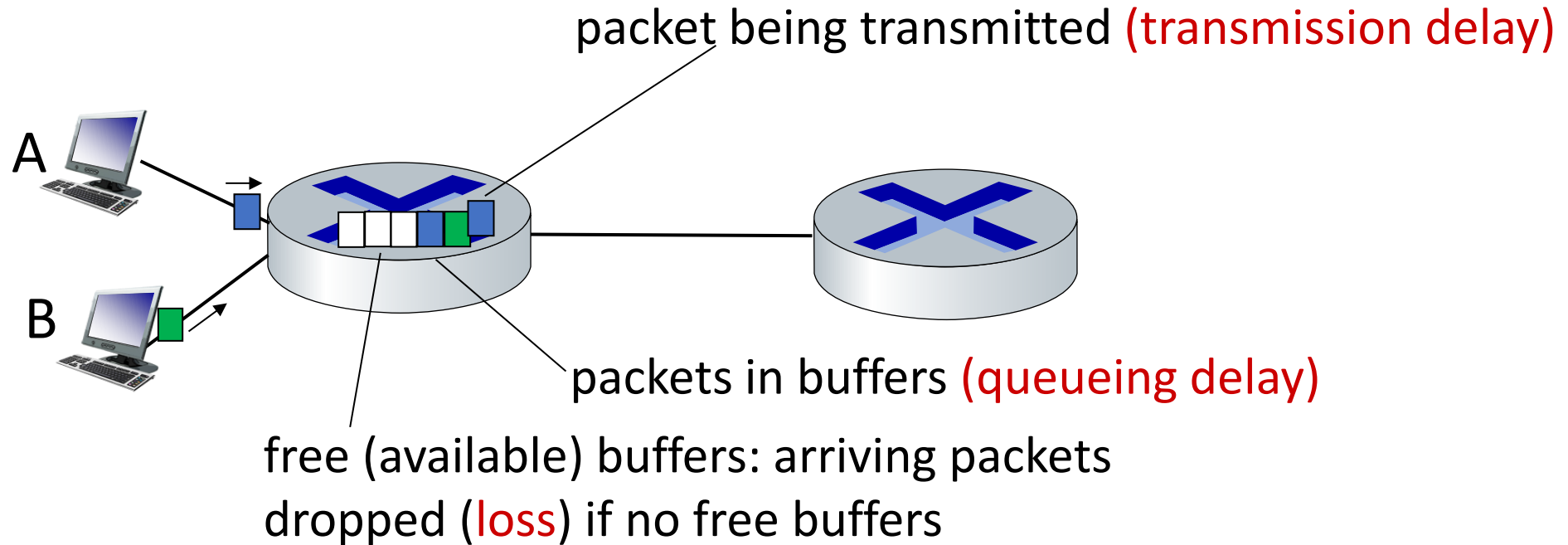
- What *is* the Internet?
- What *is* a protocol?
- Network edge: hosts, access network, physical media
- Network core: packet/circuit switching, internet structure
- **Performance: loss, delay, throughput**
- Security
- Protocol layers, service models
- History



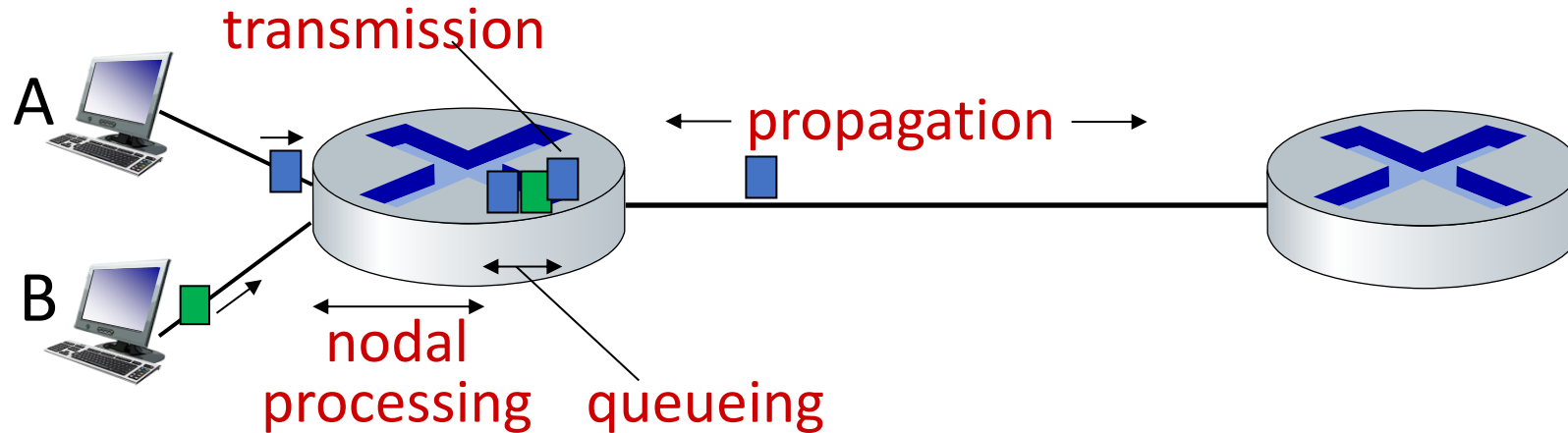
# How do packet loss and delay occur?

packets *queue* in router buffers

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



# Packet delay: four sources



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

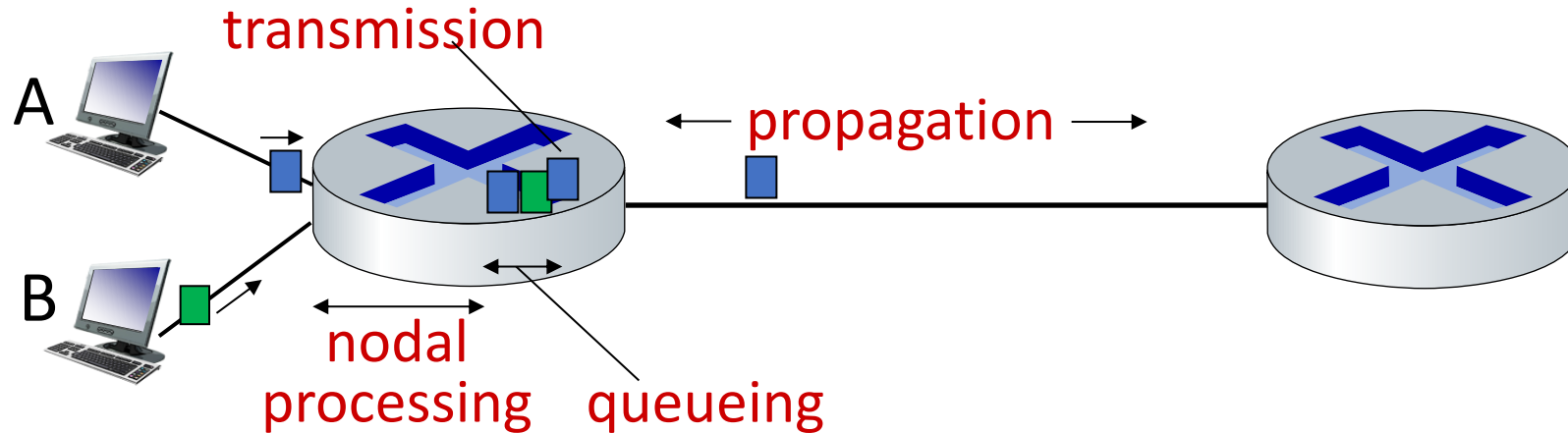
**$d_{\text{proc}}$ : processing delay**

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

**$d_{\text{queue}}$ : queueing delay**

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

# Packet delay: four sources



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

$d_{\text{trans}}$ : transmission delay:

- $L$ : packet length (bits)
- $R$ : link transmission rate (bps)

■  $d_{\text{trans}} = L/R$

$d_{\text{prop}}$ : propagation delay:

- $d$ : length of physical link
- $s$ : propagation speed ( $\sim 2 \times 10^8$  m/sec)

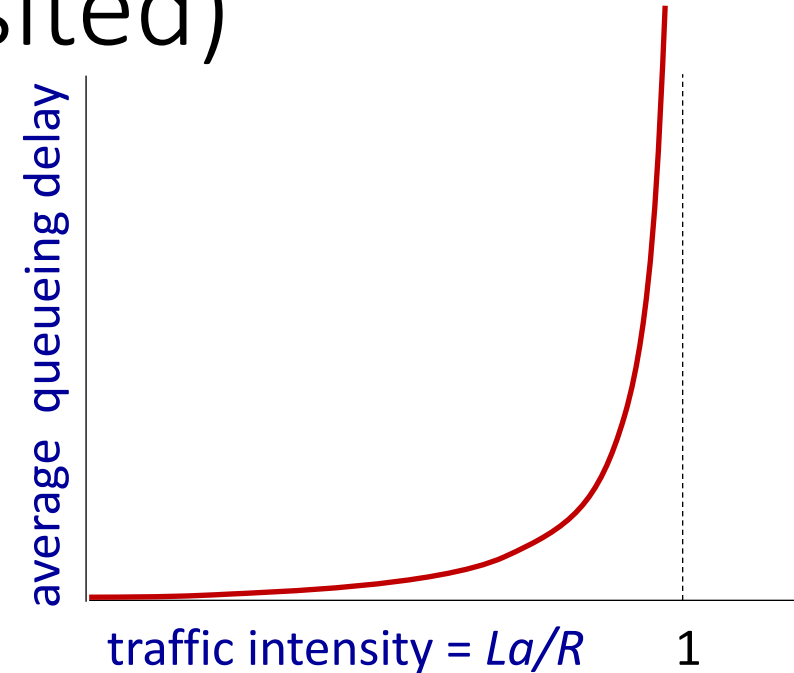
■  $d_{\text{prop}} = d/s$

$d_{\text{trans}}$  and  $d_{\text{prop}}$   
very different

\* Check out the online interactive exercises:  
[http://gaia.cs.umass.edu/kurose\\_ross](http://gaia.cs.umass.edu/kurose_ross)

# Packet queueing delay (revisited)

- $R$ : link bandwidth (bps)
- $L$ : packet length (bits)
- $a$ : average packet arrival rate
- $La/R \sim 0$ : avg. queueing delay small
- $La/R \rightarrow 1$ : avg. queueing delay large
- $La/R > 1$ : more “work” arriving is more than can be serviced - average delay infinite!

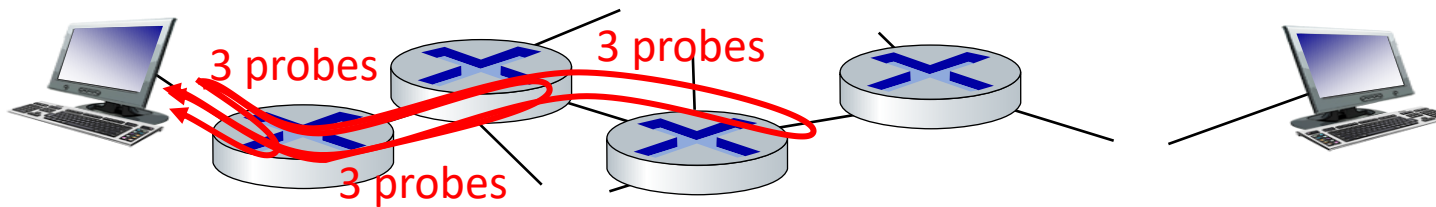


$La/R \sim 0$

$La/R \rightarrow 1$

# “Real” Internet delays and routes

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



# Real Internet delays and routes

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

Tracert in Windows!

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

3 delay measurements  
to border1-rt-fa5-1-0.gw.umass.edu

trans-oceanic link

looks like delays  
decrease! Why?

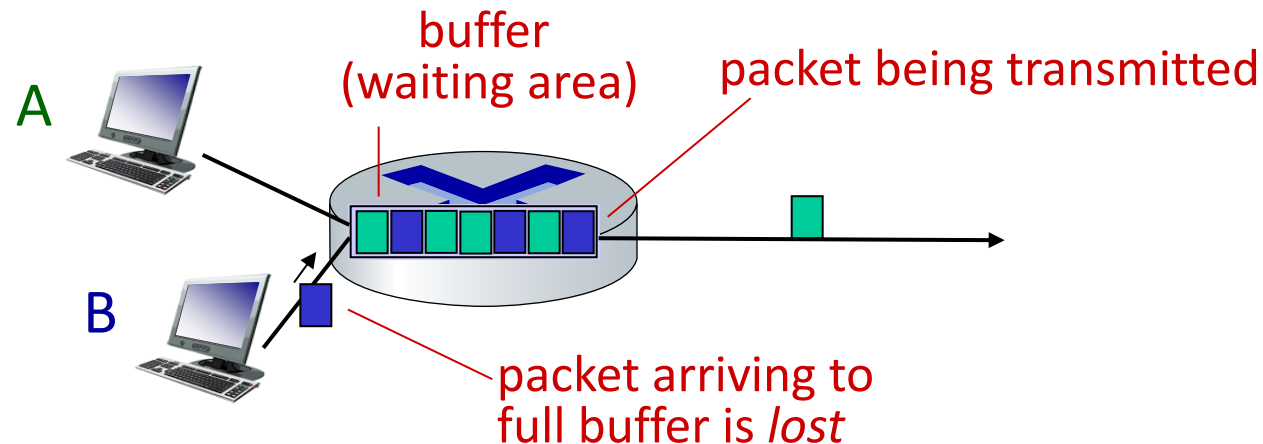
\* means no response (probe lost, router not replying)

```
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
```

\* Do some traceroutes from exotic countries at [www.traceroute.org](http://www.traceroute.org)

# Packet loss

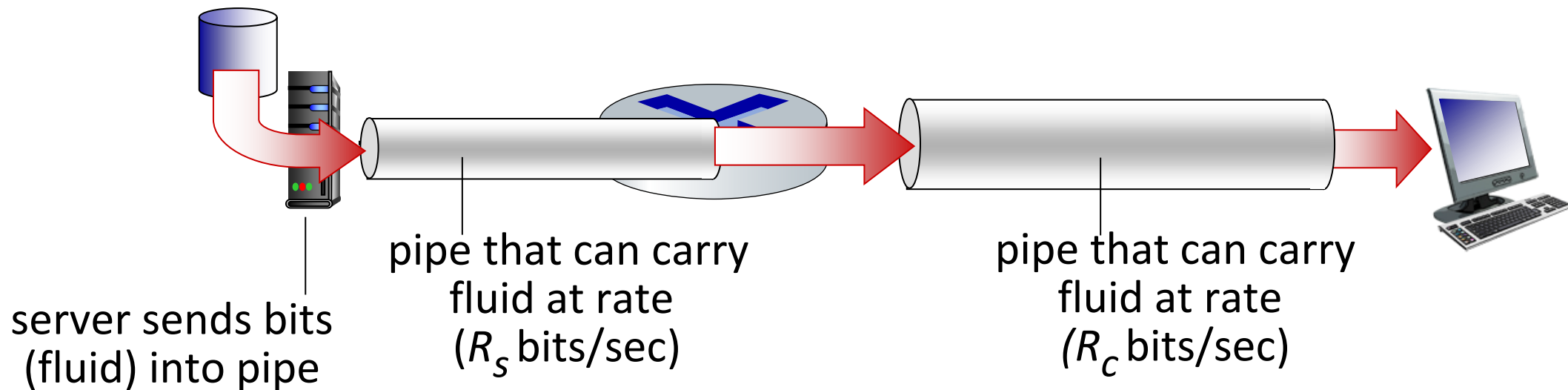
- queue (aka buffer) preceding link in buffer has finite capacity
- packet arriving to full queue dropped (aka lost)
- lost packet may be retransmitted by previous node, by source end system, or not at all



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

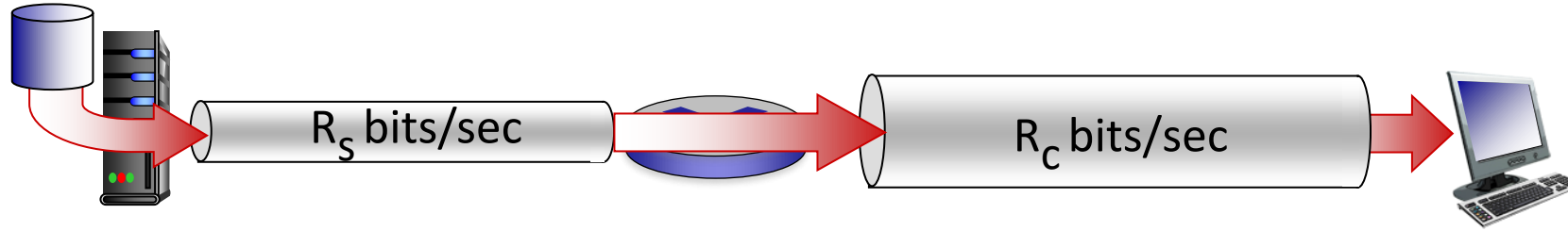
# Throughput

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

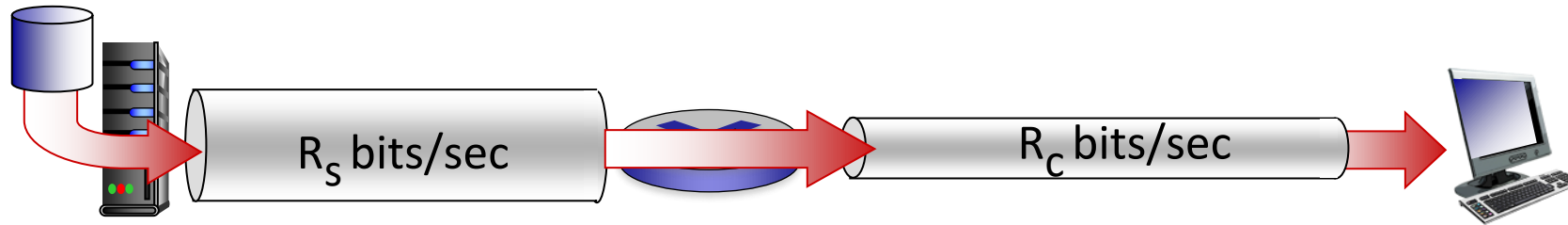


# Throughput

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



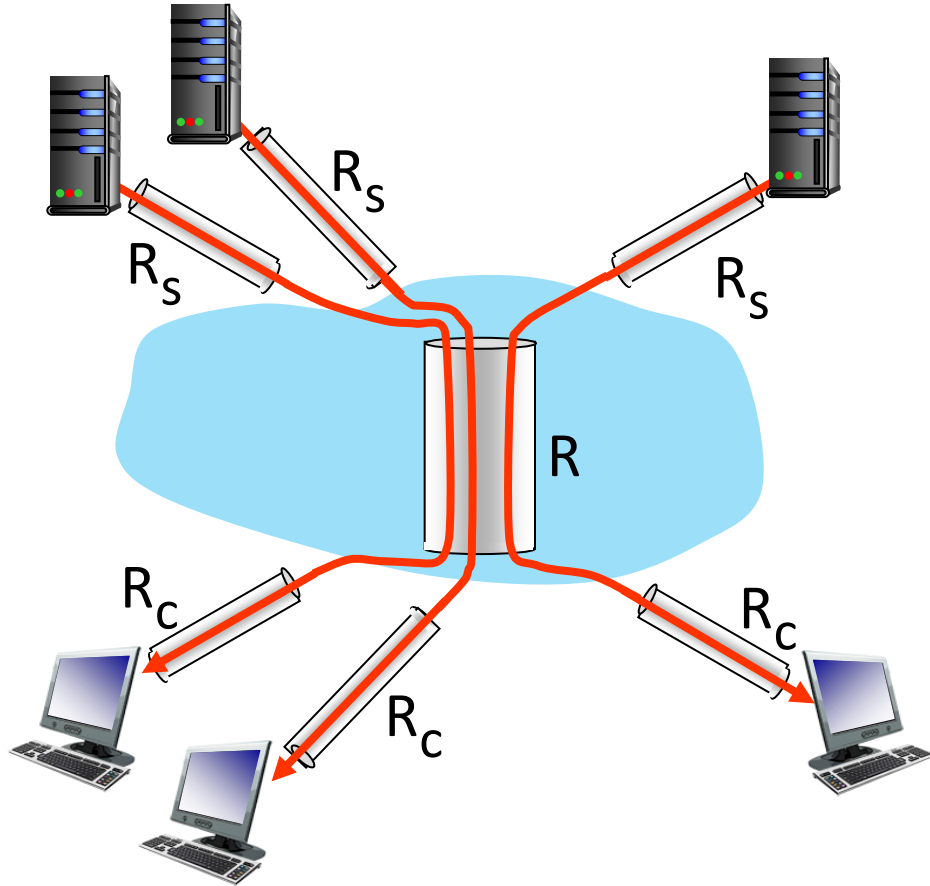
$R_s > R_c$  What is average end-end throughput?



*bottleneck link*

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

# Throughput: network scenario



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

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

\* Check out the online interactive exercises for more examples: [http://gaia.cs.umass.edu/kurose\\_ross/](http://gaia.cs.umass.edu/kurose_ross/)