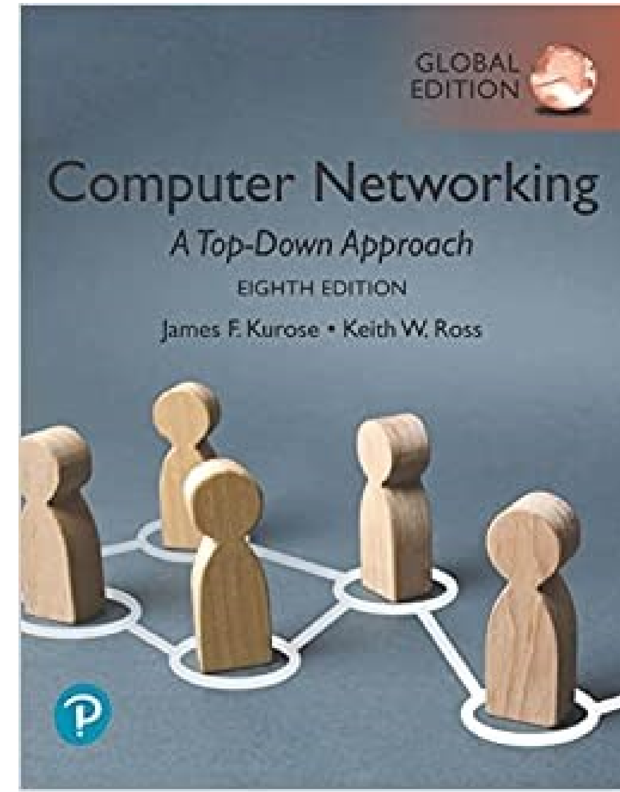
Chapter 1
Introduction – part 2

School of Computing
Gachon Univ.
Joon Yoo



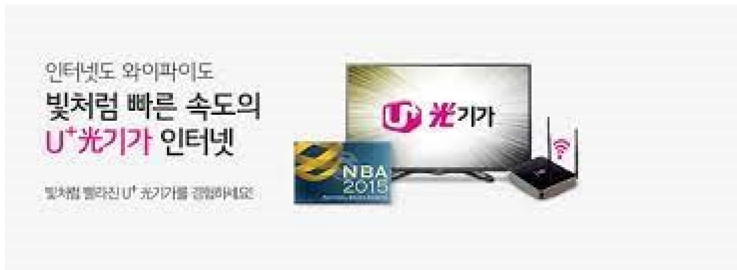
Computer Networking: A Top-Down Approach
8th edition (Global edition)
Jim Kurose, Keith Ross
Pearson, 2021

Many slides from J.F Kurose and K.W. Ross

가천대학교
Gachon University

Question

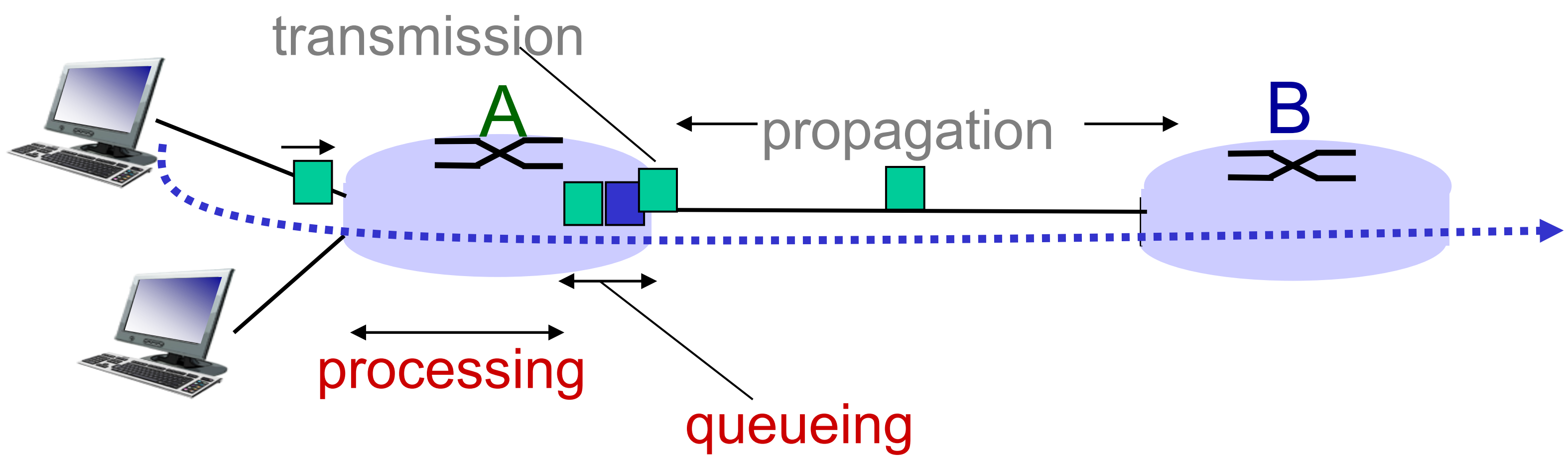
How do we measure the performance of a network?



Chapter 1: roadmap

가천대학교 Gachon University

Four sources of packet delay



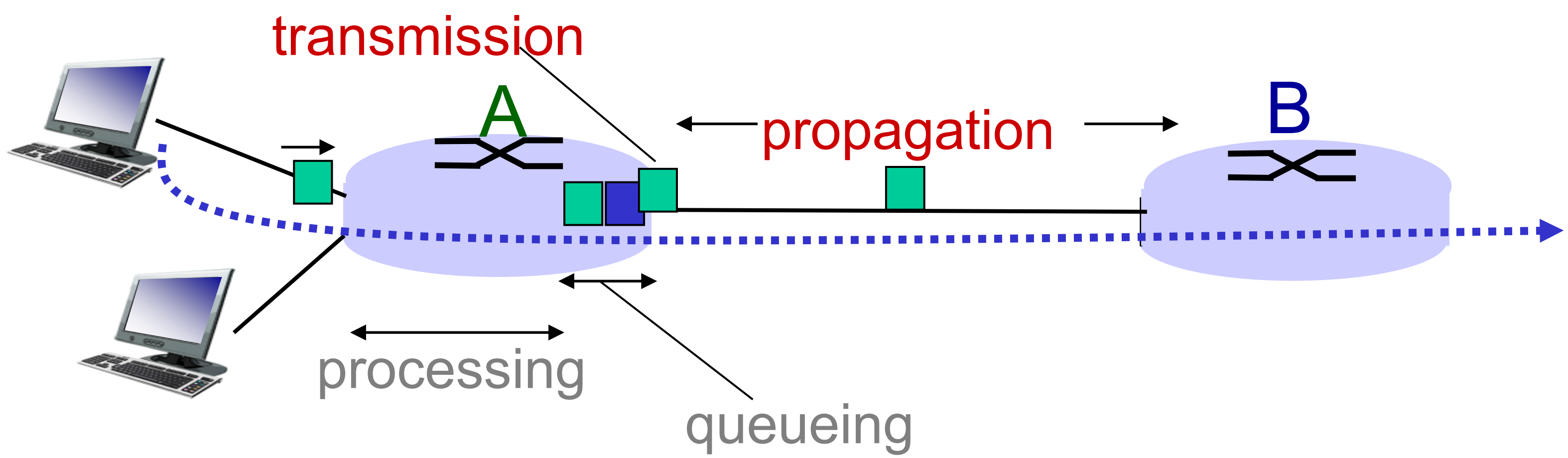
$$\boxed{d_{nodal} = d_{proc} + d_{queue} + d_{trans} + d_{prop}}$$

1. d_{proc}: processing delay
- examines packet header
- determine where to direct packet (=output link), e.g., link to router B.
- typically usec

2. d_{queue}: queueing delay
- time waiting at output link for transmission
- depends on earlier arriving packets that are queued and waiting
- typically usec ~ msec

Gachon University 가천대학교

Four sources of packet delay



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

3. d_{trans}: transmission delay:

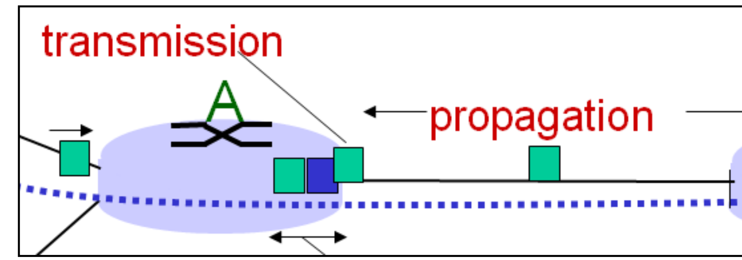
- time to transmit all the packet's bits into the link (e.g., link AB)
- $d_{trans} = L/R$
  - L: packet length (bits)
  - R: link *bandwidth (bps)*

4. d_{prop}: propagation delay:

- time to propagate from beginning of the link to next router (e.g., router B)
- $d_{prop} = d/s$
  - d: length of physical link
  - s: propagation speed in medium (~2×10^8 m/sec) – similar to speed of light

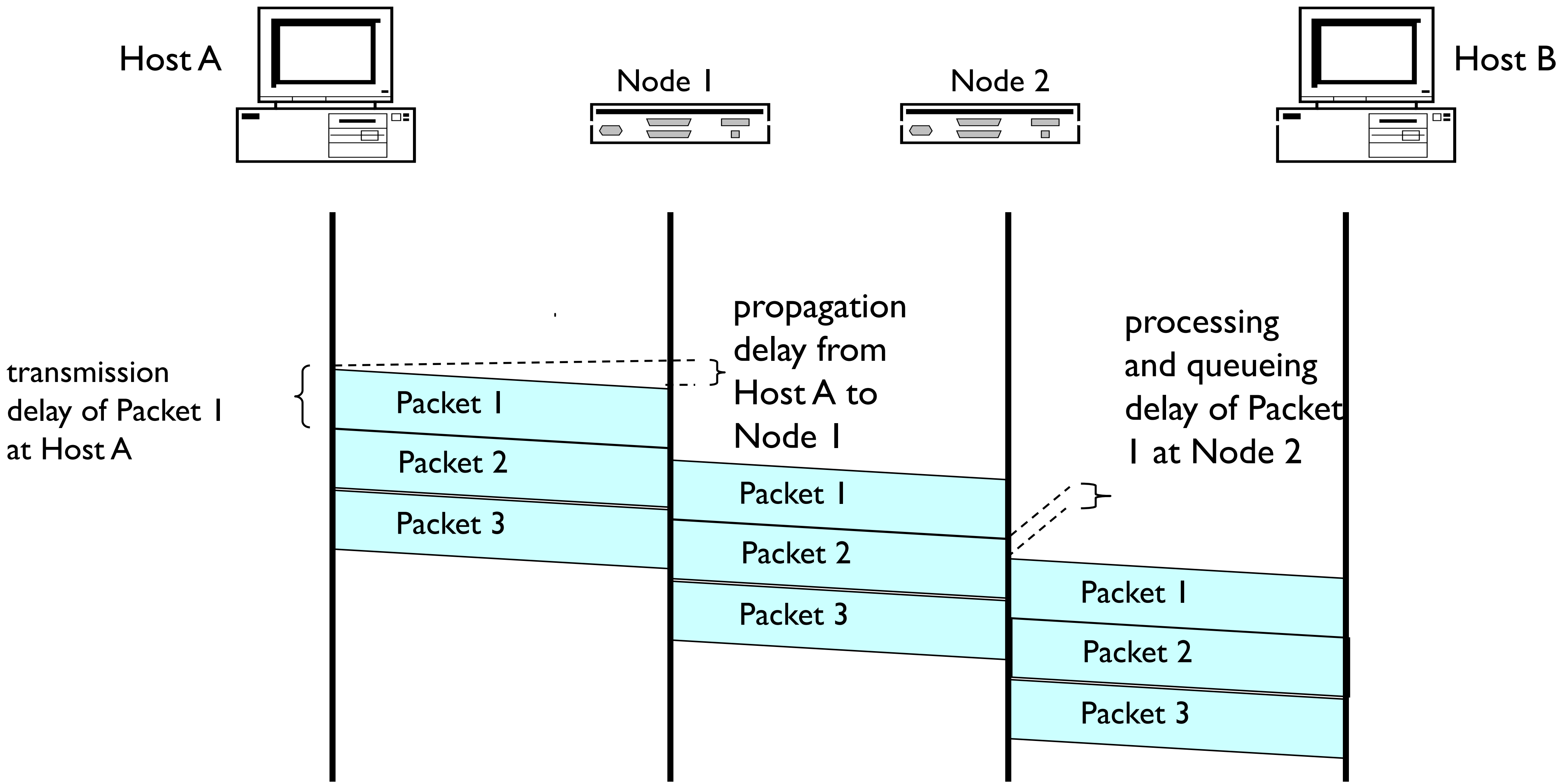
Transmission vs. Propagation Delay

- Transmission (전송) delay
  - Time for router to push out the packet
  - Function of packet's length (L bits) and transmission rate (R bps) of link
    - $d_{trans} = L/R$
  - Nothing to do with distance between two routers



- Propagation (전파) delay
  - Time a bit to propagate from one router to the next
  - Function of distance between the two routers
    - $d_{prop} = d/s$
  - Nothing to do with packet's length or the transmission rate of link

Timing Diagram of Packet Switching



Nodal delay

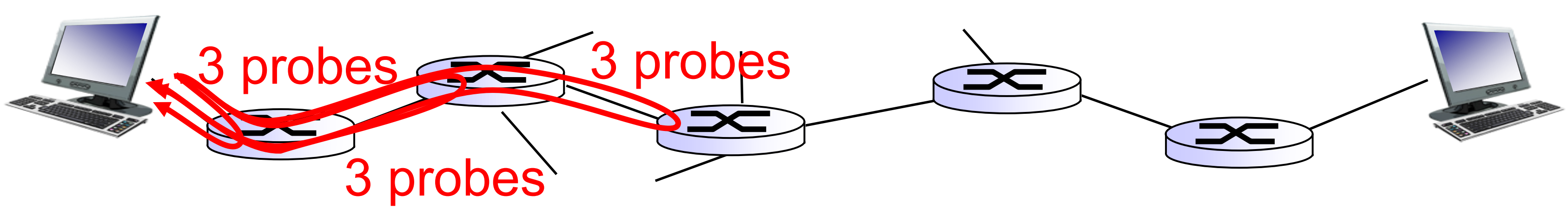
$$\boxed{d_{\text{nodal}} = d_{\text{proc}} + d_{\text{queue}} + d_{\text{trans}} + d_{\text{prop}}}$$

- d_{proc} = processing delay
  - typically a few microsecs or less
- d_{queue} = **queuing delay**
  - depends on congestion of network

- d_{trans} = transmission delay = L/R
  - negligible for 10Mbps and higher
  - but significant for low-speed links (e.g., dial-up modem, 2G)
- d_{prop} = propagation delay
  - a few microsecs (e.g., same campus) to hundreds of msecs (e.g., satellite link)

"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

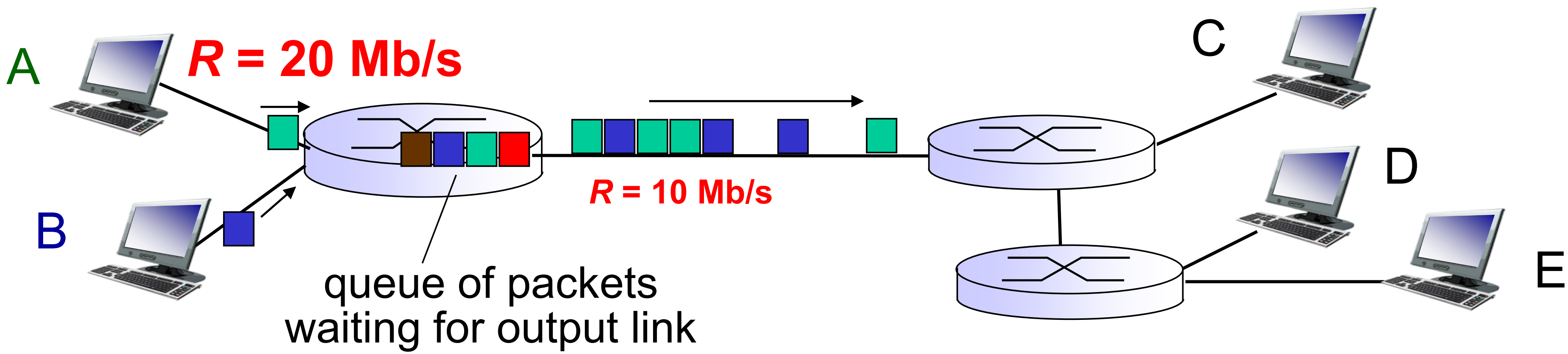
1 cs-gw (128.119.240.254) 1 ms 1 ms 2 ms
2 border1-rt-fa5-1-0.gw.umass.edu (128.119.3.145) 1 ms 1 ms 2 ms
3 cht-vbns.gw.umass.edu (128.119.3.130) 6 ms 5 ms 5 ms
4 jn1-at1-0-0-19.wor.vbns.net (204.147.132.129) 16 ms 11 ms 13 ms
5 jn1-so7-0-0-0.wae.vbns.net (204.147.136.136) 21 ms 18 ms 18 ms
6 abilene-vbns.abilene.ucaid.edu (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

Packet Switching: queueing delay, loss



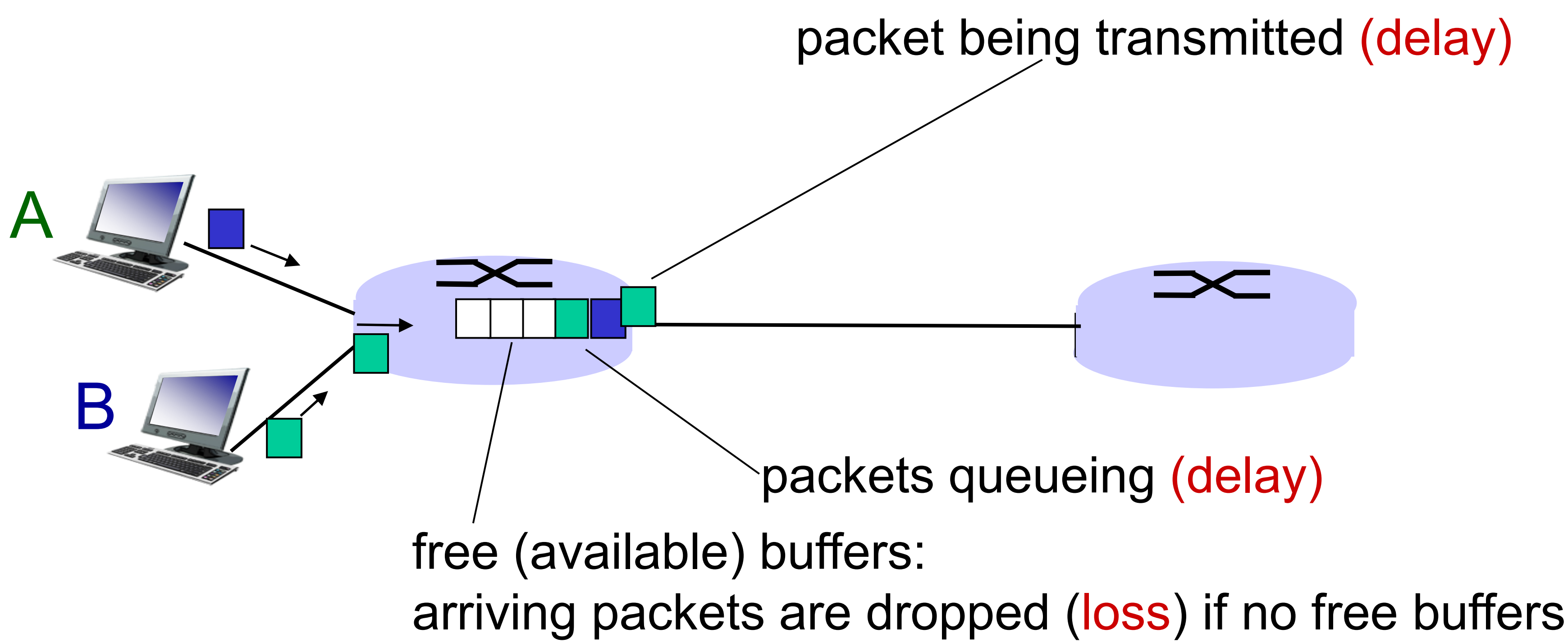
queuing and loss:

- 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**
  - packets can be dropped if memory (buffer) fills up – **loss**

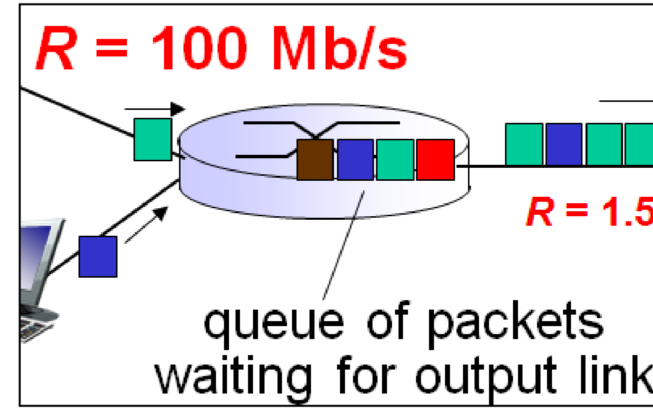
How do loss and delay occur?

packets *queue* in router buffers

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

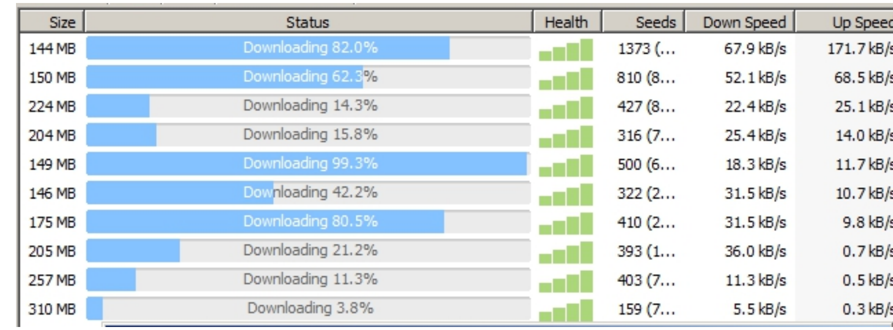


loss and delay



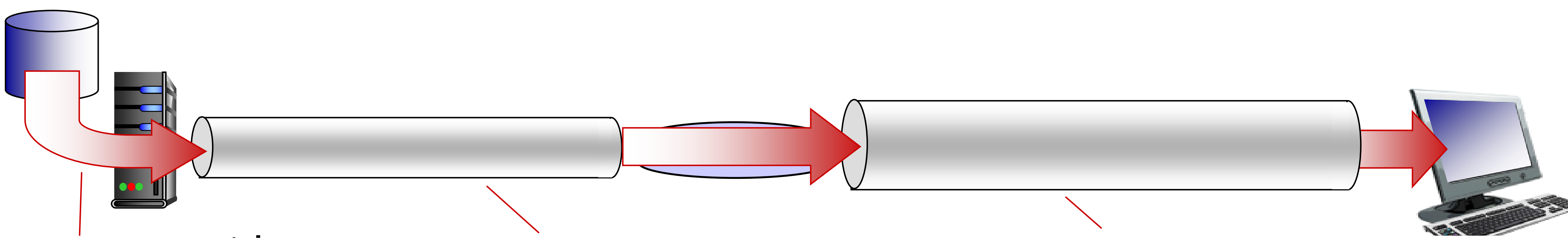
- Queueing delay
  - 10 packets arrive at router almost at the same time
    - Currently the router buffer is empty
  - First packet has no queueing delay
  - 10th packet has to wait for 9 other packets to be transmitted – **queueing delay** occurs
- Loss
  - If the router buffer (i.e., memory) is full
  - Next arriving packet will be **dropped** – **packet loss** occurs
- Queueing delay and loss will increase as traffic intensity (i.e., congestion) increases

Throughput



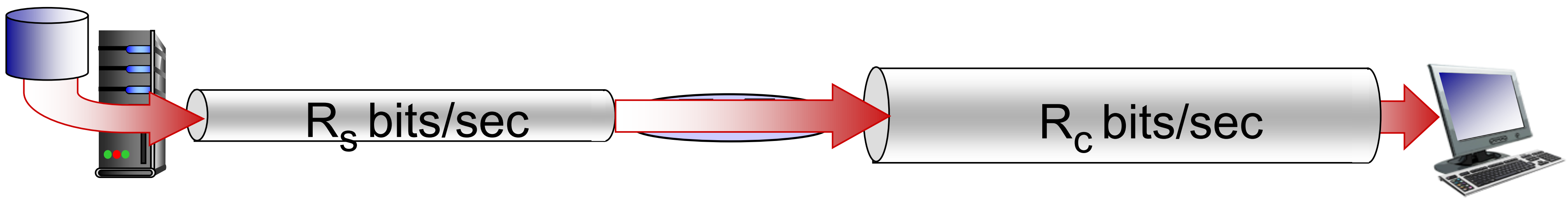
Size	Status	Health	Seeds	Down Speed	Up Speed
144 MB	Downloading 82.0%		1373 (...	67.9 kB/s	171.7 kB/s
150 MB	Downloading 62.3%		810 (8...	52.1 kB/s	68.5 kB/s
224 MB	Downloading 14.3%		427 (8...	22.4 kB/s	25.1 kB/s
204 MB	Downloading 15.8%		316 (7...	25.4 kB/s	14.0 kB/s
149 MB	Downloading 99.3%		500 (6...	18.3 kB/s	11.7 kB/s
146 MB	Downloading 42.2%		322 (2...	31.5 kB/s	10.7 kB/s
175 MB	Downloading 80.5%		410 (2...	31.5 kB/s	9.8 kB/s
205 MB	Downloading 21.2%		393 (1...	36.0 kB/s	0.7 kB/s
257 MB	Downloading 11.3%		403 (7...	11.3 kB/s	0.5 kB/s
310 MB	Downloading 3.8%		159 (7...	5.5 kB/s	0.3 kB/s

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

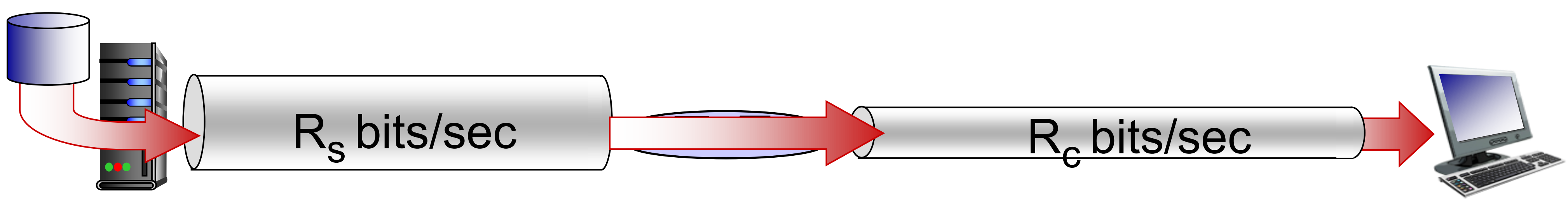


Throughput (more)

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



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



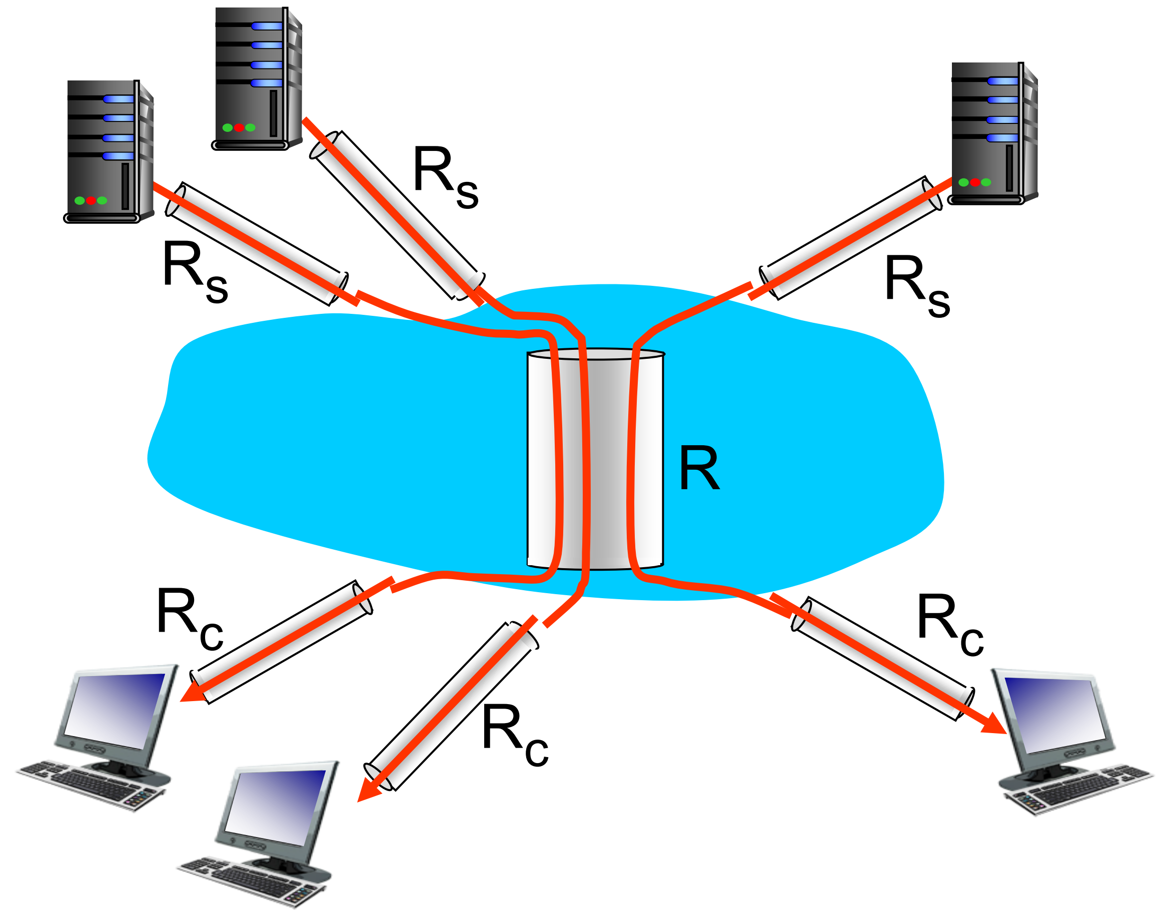
bottleneck link

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

Throughput: Internet scenario

- per-connection end-end throughput: $\min(R_c, R_s, R/10)$
- in practice: R_c or R_s is often bottleneck
  - R is often much larger
  - Even R/10, R/100, … is larger

- Throughput depends on the transmission rates of links on the data path



10 connections

Chapter 1: roadmap

1.1 what *is* the Internet?

1.2 network edge

- end systems, access ispworks, 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":

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

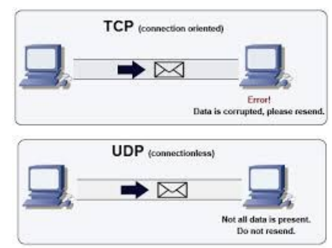
Question:

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

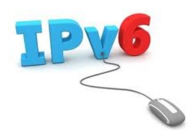
…. or at least our discussion of networks?

Internet protocol stack

- *application:* supporting various network applications
  - HTTP, SMTP, FTP, DNS
- *transport:* process-process data transfer
  - TCP, UDP

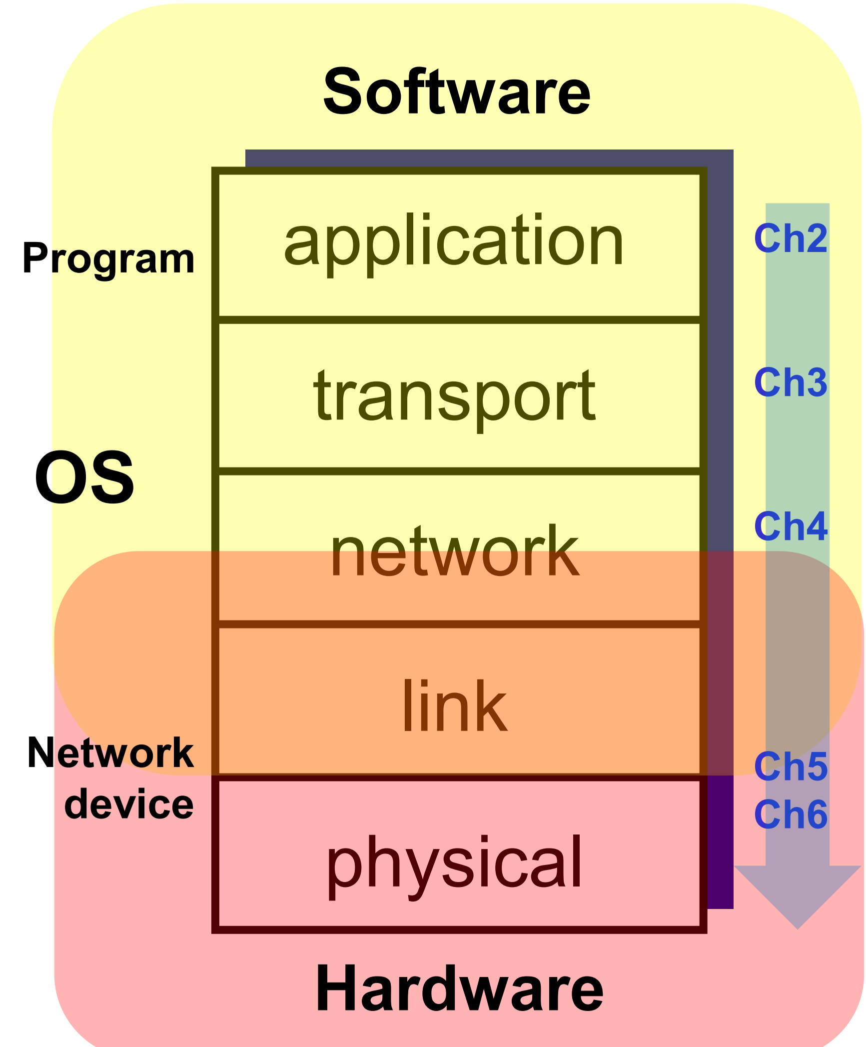


- *network:* routing of datagrams from source to destination
  - IP, routing protocols



- *link:* data transfer between neighboring network elements
  - Ethernet, 802.11 (WiFi)

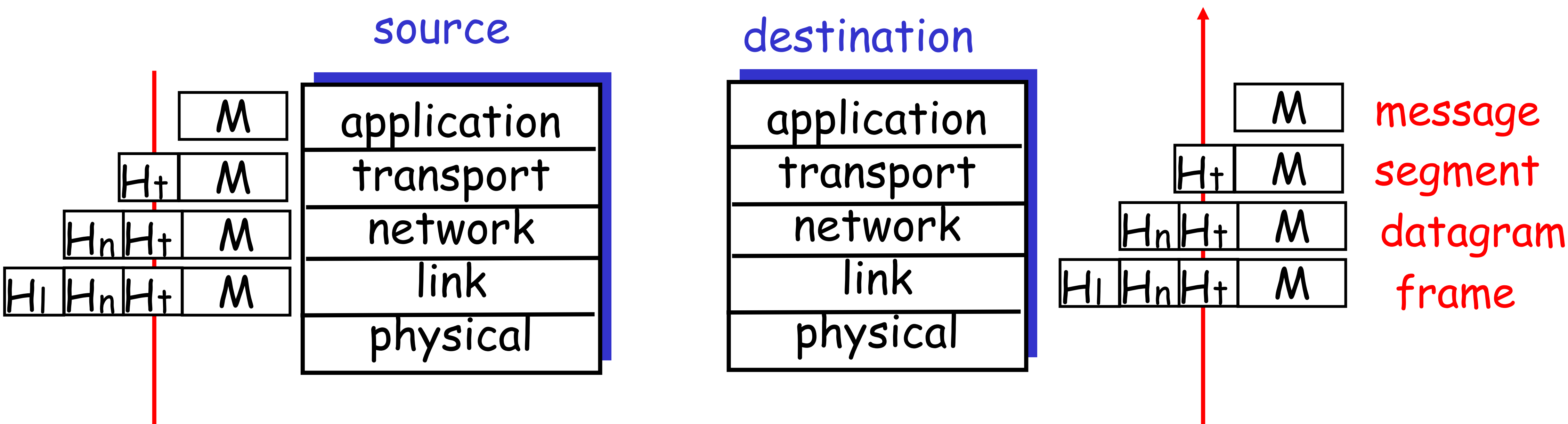
- *physical:* bits "on the wire"



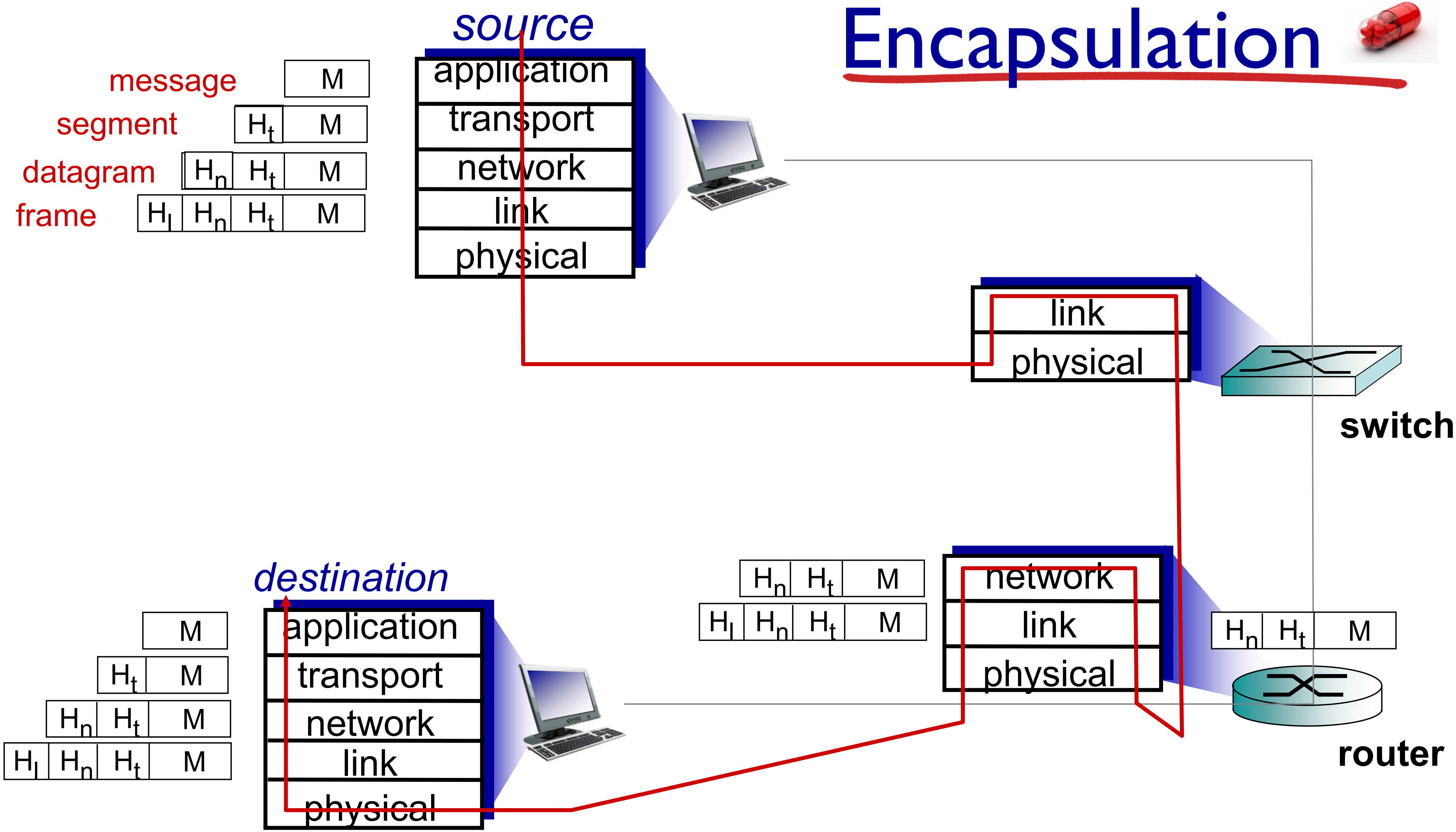
Protocol layering and data

Each layer takes data from above

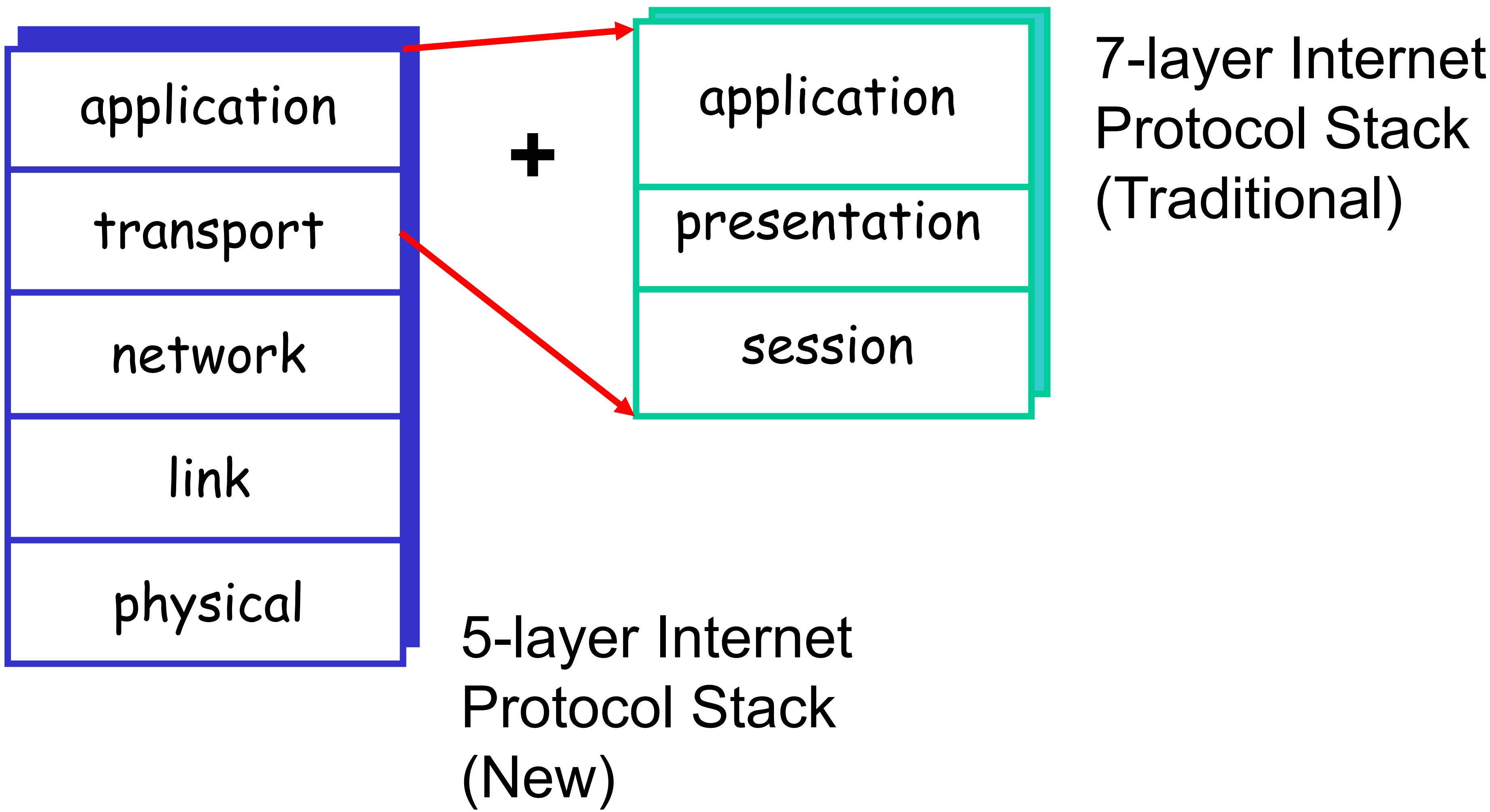
- adds header information to create new data unit
- passes new data unit to layer below



Encapsulation



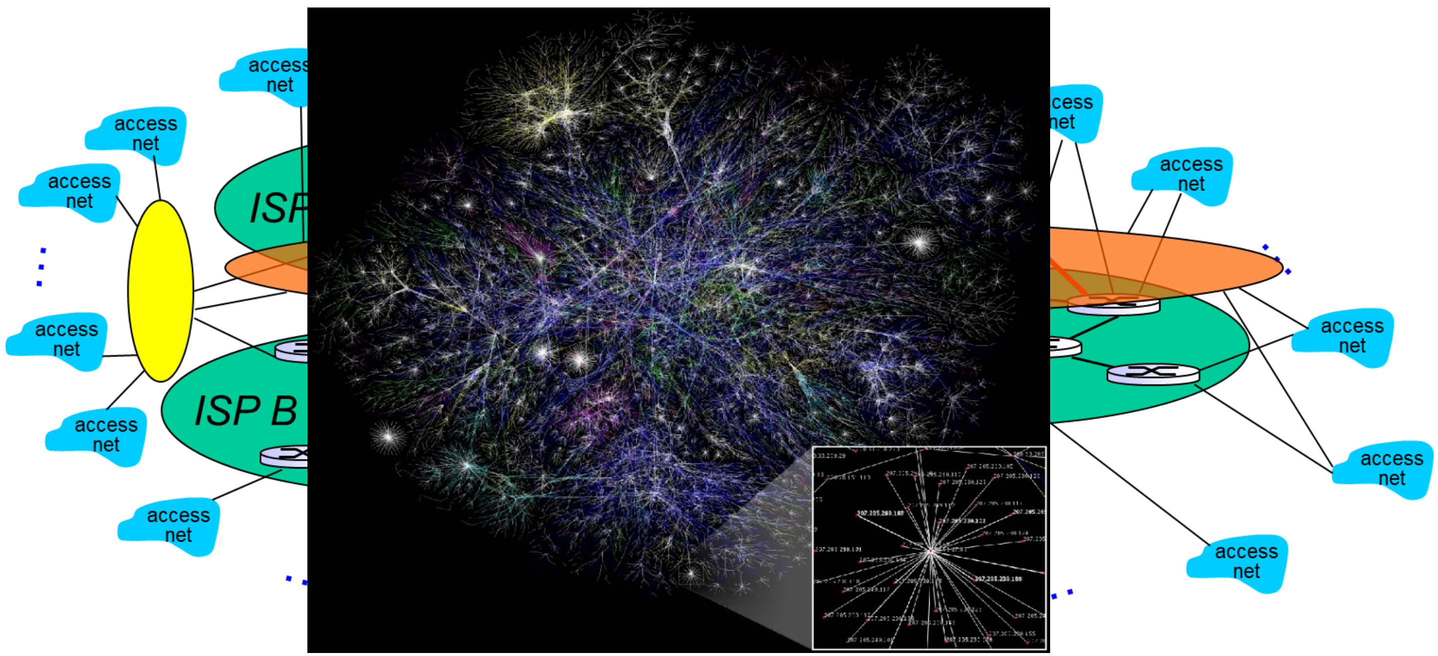
ISO 7-layer reference model



가천대학교
Gachon University

Question

Who takes care of the complex Internet?



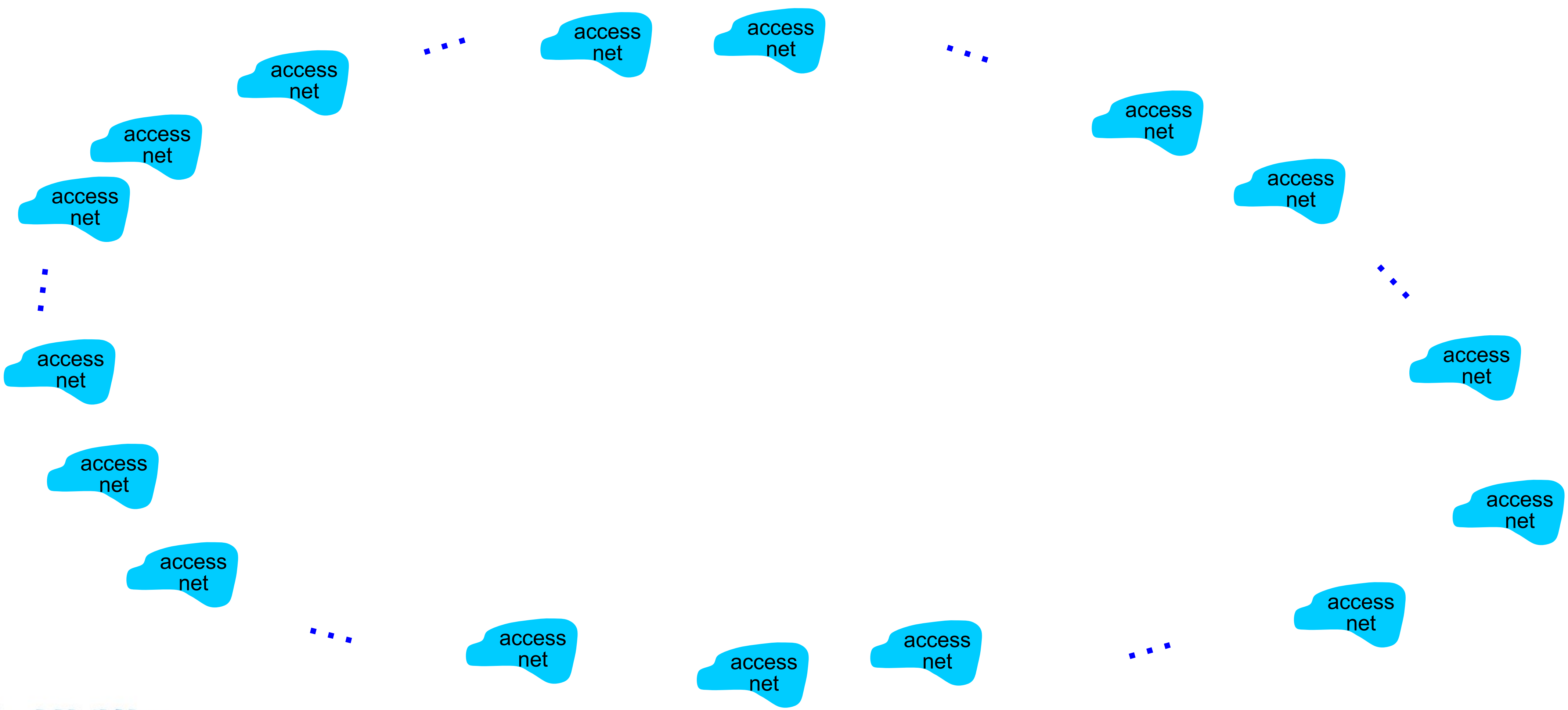
Internet structure: network of networks (Ch. 1.3.3)

- End systems connect to Internet via access **ISPs** (**Internet Service Providers**)
  - DSL, cable, FTTH, Wi-Fi, cellular, …
  - Telco (e.g., AT&T, Sprint, KT, SKT), Residential (e.g., KT, SK, LG), company and university ISPs
- Access ISPs in turn must be interconnected.
  - Network of Networks
  - 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

ISP is an "administrative view" of the network

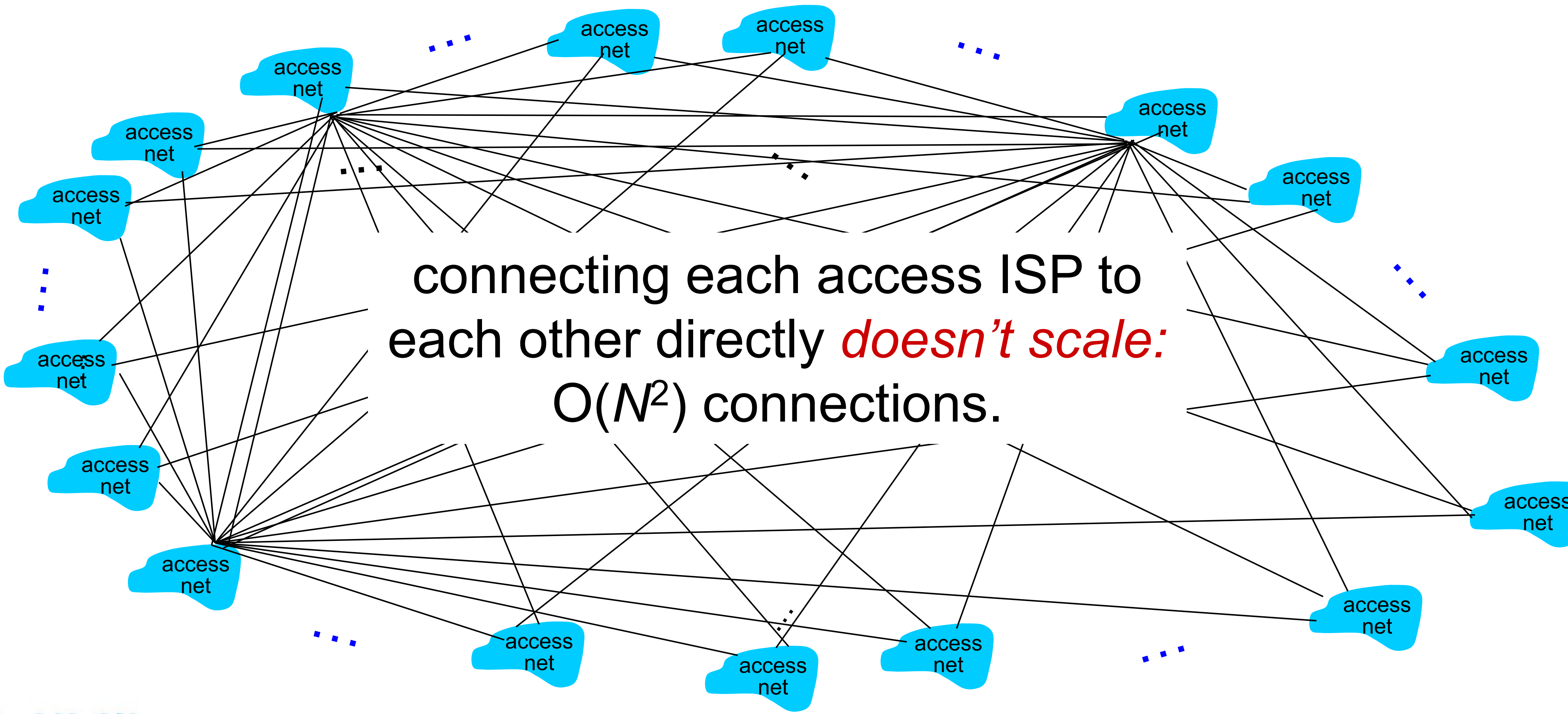
Internet structure: network of networks

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



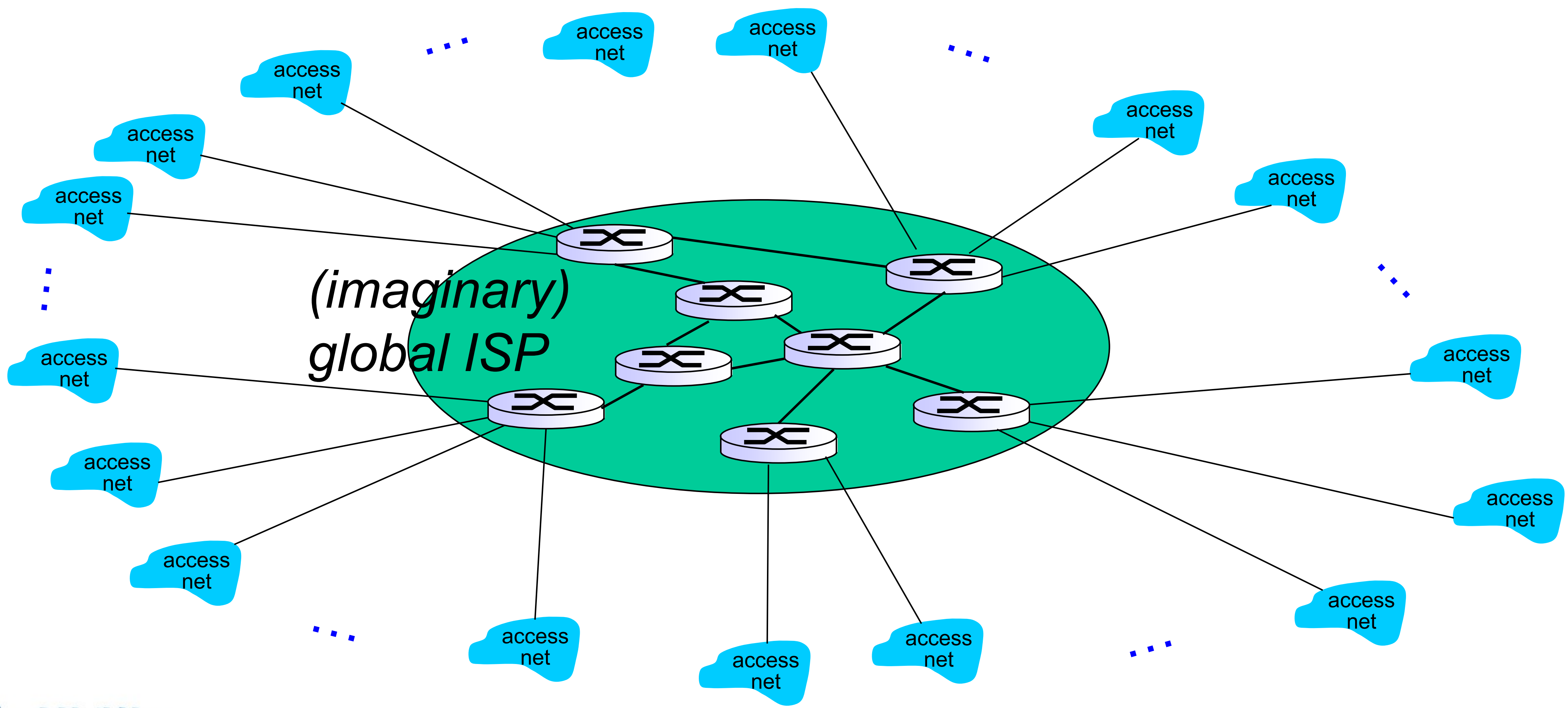
Internet structure: network of networks

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



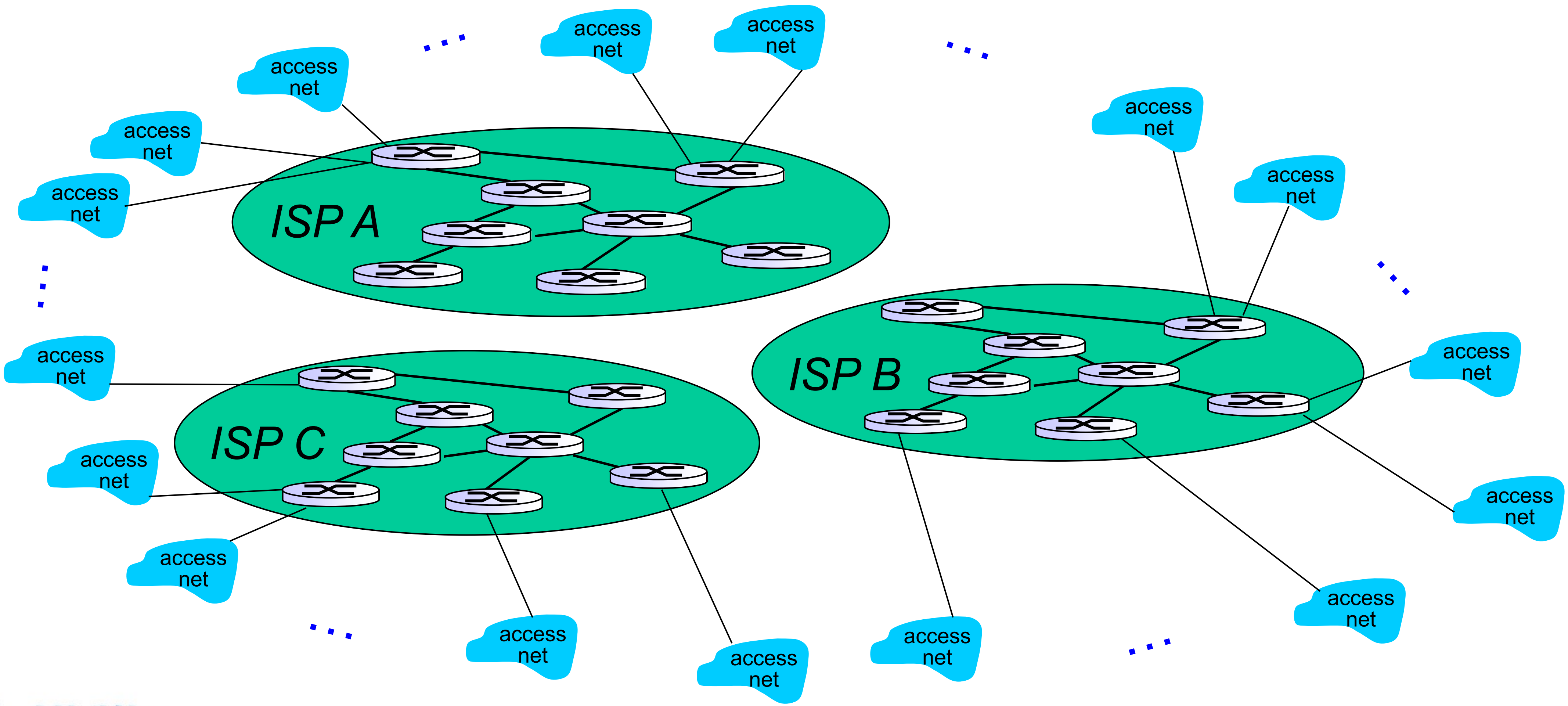
Internet structure: network of networks

Option: connect each access ISP to a global transit ISP? Customer (who?) and provider ISP (who?) have economic agreement.



Internet structure: network of networks

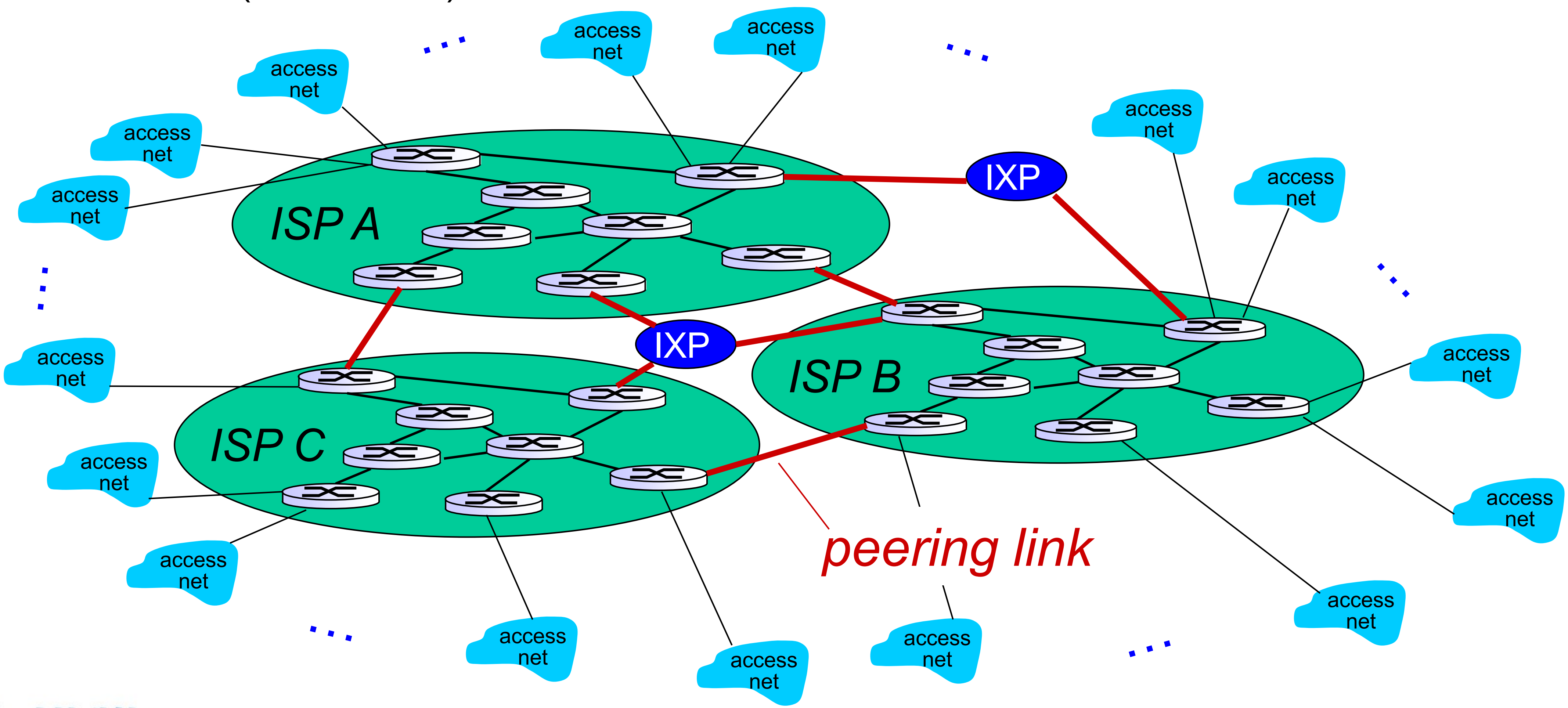
But if one global ISP is viable business, there will be competitors called **tier-1 ISPs**. – around a dozen tier-1 ISPs world-wide. (e.g., Level 3 communications, Sprint, AT&T, NTT), national & international coverage



가천대학교
Gachon University

Internet structure: network of networks

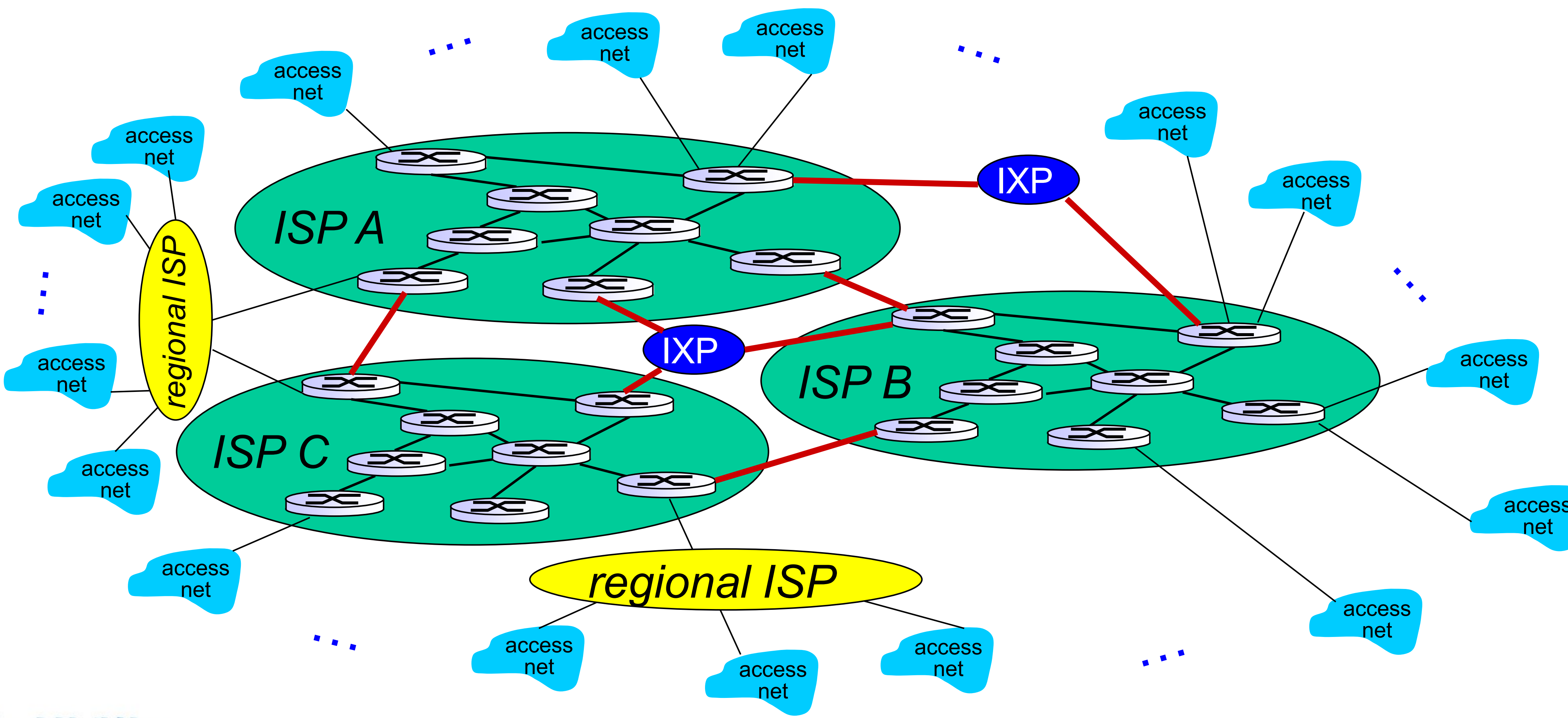
Tier-1 ISPs must be interconnected by **Internet exchange point** (**IXP**): 3rd party stand-alone building with its own switches, over 600 IXPs in the Internet (as of 2020)



가천대학교
Gachon University

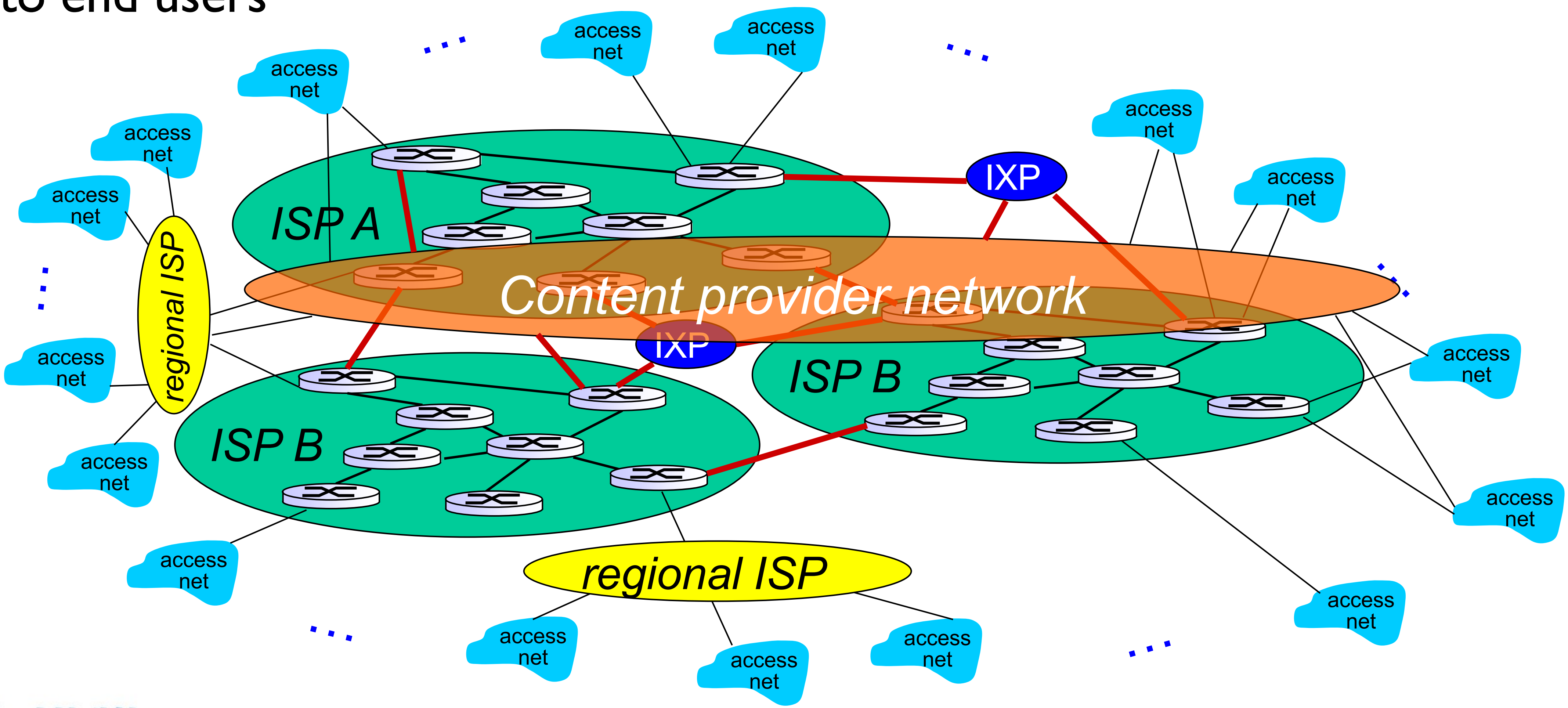
Internet structure: network of networks

A few tier-1 ISPs cannot be close to all access IPSs.
regional ISPs may arise to connect access ISPs in the region.

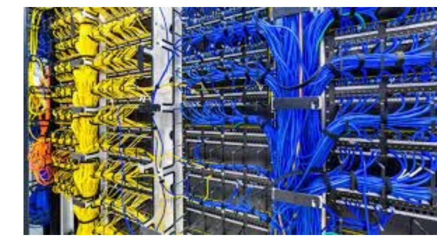


Internet structure: network of networks

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

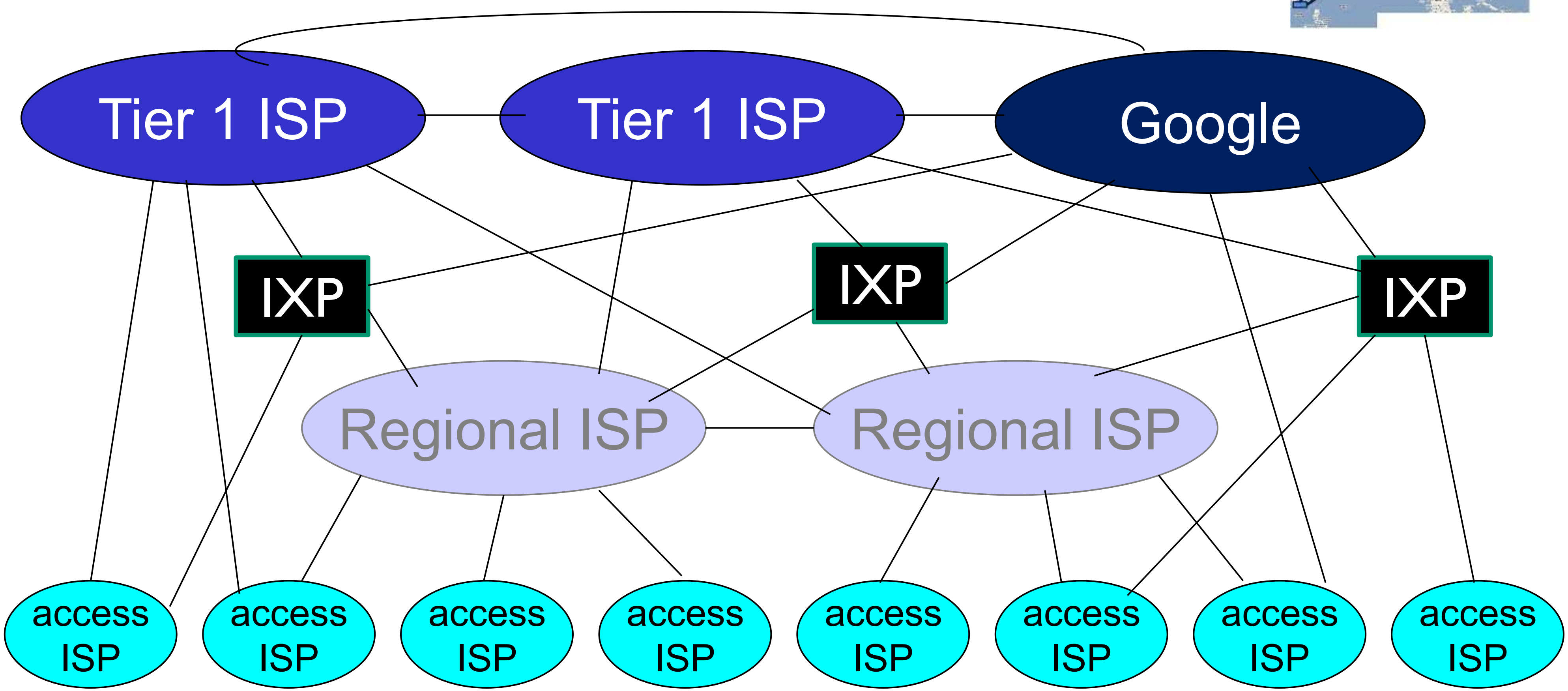


Data Centers and Cloud Computing (Ch. 1.2)



- Internet companies such as Google, Microsoft, Amazon, and Alibaba have built data centers, each hosting hundreds and thousand of hosts
- Data centers are not only connected to the Internet, but also internally include complex data center networks that interconnect data center hosts
- Data center service example: Amazon
  - **Amazon e-commerce** page for users
  - **Parallel computing** infrastructures for Amazon-specific data processing tasks
  - **Cloud computing** to other companies
    - (Initial) Netflix and Airbnb ran their entire web services in Amazon Web Service (AWS)

Internet structure: network of networks



- at center: small # of well-connected large networks
  - content provider network (e.g, Google): private network that connects it data centers to Internet, often bypassing tier-1, regional ISPs

Chapter 1: roadmap

1.1 what *is* the Internet?

1.2 network edge

- end systems, access ispworks, 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

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!

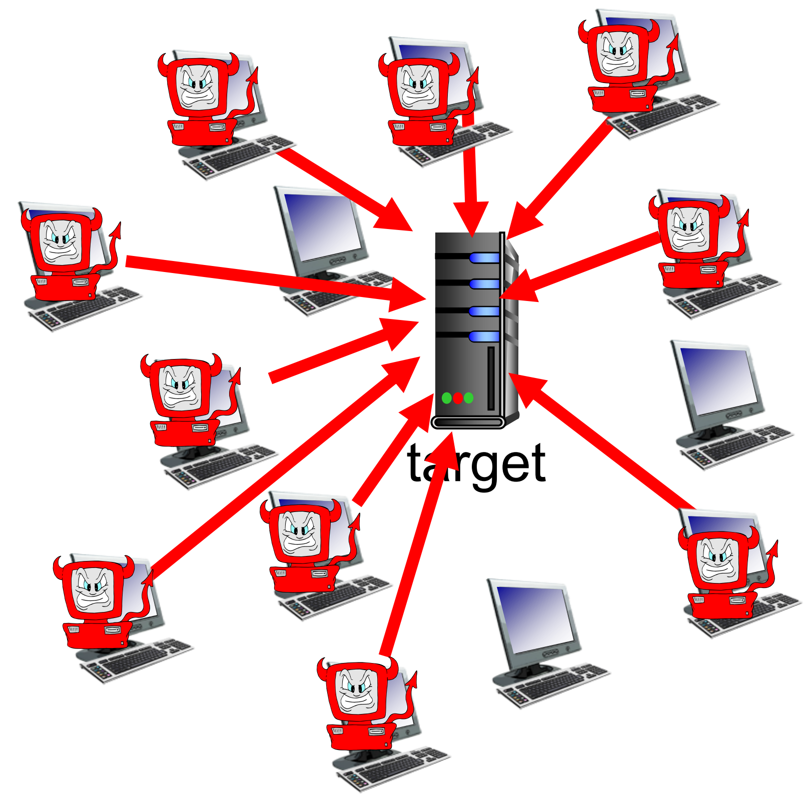
Bad guys: put malware into hosts via Internet

- Malware
  - *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: can record keystrokes, web sites visited, upload info to collection site
- infected host can be enrolled in botnets;
  - A network of thousands of similarly compromised devices
  - The bad guys control the leverage for spam e-mail distribution or Distributed DoS (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

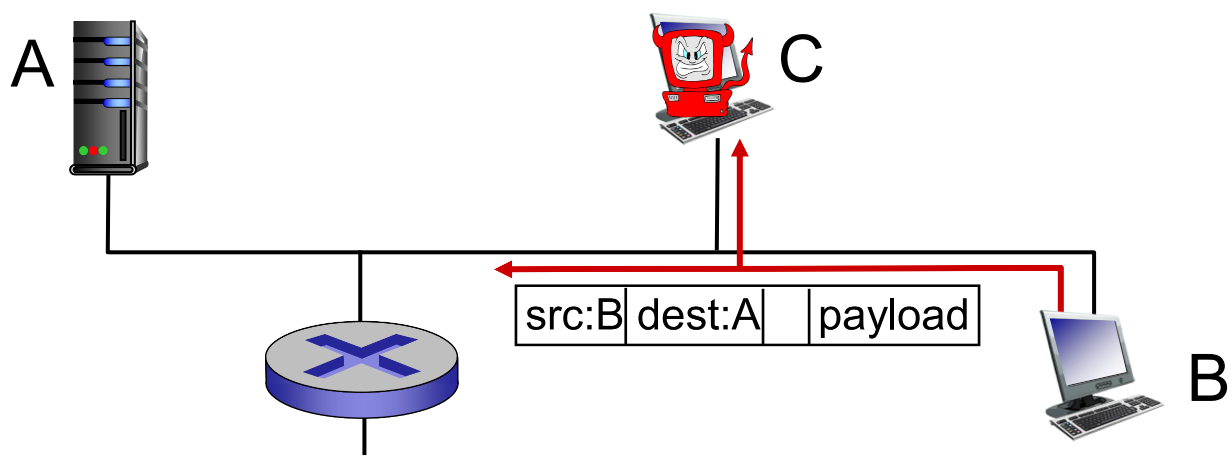
1. select target
2. break into hosts around the network (comprising a botnet)
3. send packets to target from compromised hosts
4. results in vulnerability attack, bandwidth flooding, and/or connection flooding



Bad guys can sniff packets

packet "sniffing":

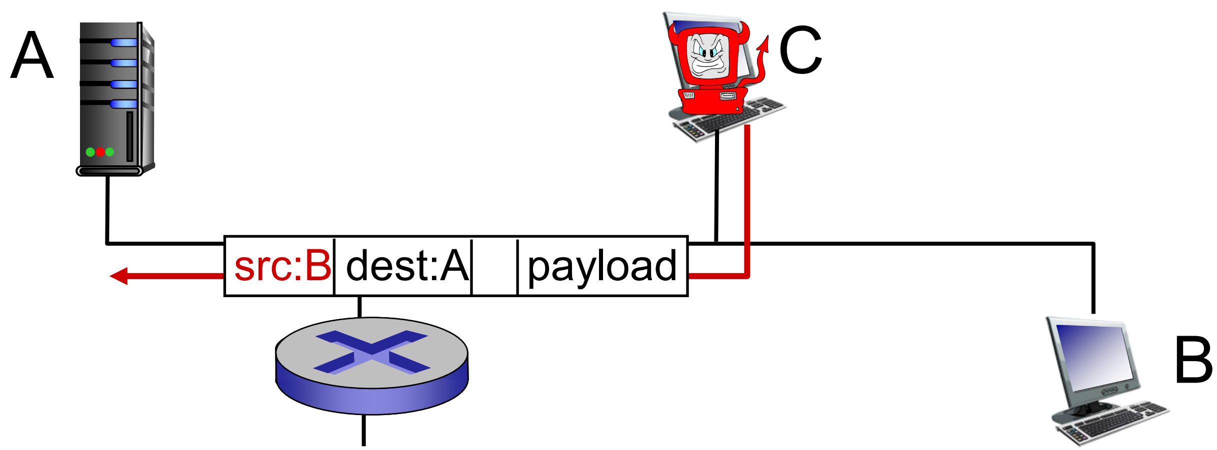
- broadcast media (shared Ethernet, wireless)
- promiscuous network interface reads/records all packets (e.g., including passwords!) passing by



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

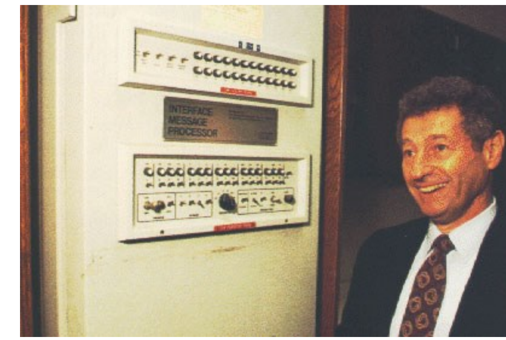
Bad guys can use fake addresses

IP spoofing: send packet with false source address



Brief Internet history

1961-1990: Early packet-switching, protocols



- 1961: Kleinrock - queueing theory shows effectiveness of packet-switching
- 1969: first packet-switched ARPAnet node operational (15 nodes in 1972)
  - login UCLA to SRI (Stanford Research Institute) system – crashing the system
- 1974: Cerf and Kahn - architecture for interconnecting networks
  - defined today's TCP/IP Internet architecture
- 1983: deployment of TCP/IP
- 1982: smtp e-mail protocol defined
- 1983: DNS defined for name-to-IP-address translation
- 1988:TCP congestion control
- 100,000 hosts connected to confederation of networks

Internet history

1990, 2000's - present: the Web, new apps

- early 1990's: ARPAnet decommissioned
- early 1990s: World Wide 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

2005-present

- 20.35B devices attached to Internet (2017)
- emergence of online social networks:
  - Facebook: 1.47B active users (2018)
- online-video streaming
  - YouTube: 1.57B active users (2018)
  - Netflix: 37% of US Internet traffic (2015)

South Korea

- 1982년 5월, 전길남 교수가 서울대학교 컴퓨터공학과와 구미 전자기술연구소(KIET)의 두 개의 중형 컴퓨터에 IP 주소를 할당 받아 전용선으로 연결하고 이를 패킷 교환 방식으로 연결하는데 성공
- 데이콤(현 LG유플러스)이 1986년 PC 통신 '천리안' 서비스를 시작
- 연구 목적으로 일부만 사용할 수 있었던 인터넷은 1994년 한국통신(현 KT)이 코넷(KORNET)이라는 이름으로 인터넷 상용 서비스를 개시

https://ko.wikipedia.org/wiki/대한민국의_인터넷

CH1: summary

covered a "ton" of material!

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

you now have:

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

가천대학교
Gachon University