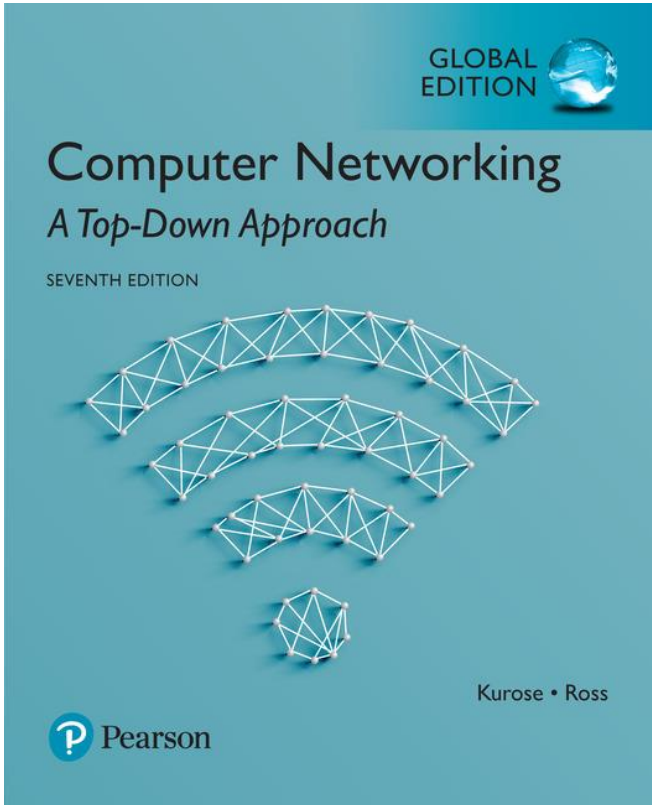
Chapter 1
Introduction



Computer Networking: A Top Down Approach

7th Edition, Global Edition
Jim Kurose, Keith Ross
Pearson
April 2016

A note on the use of these Powerpoint slides:

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

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

Thanks and enjoy! JFK/KWR

Chapter 1: introduction

our goal:

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

overview:

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

Chapter 1: roadmap

1.1 what *is* the Internet and protocol?

1.2 network edge

- end systems, access networks, links

1.3 network core

- packet switching, circuit switching, network structure

1.4 delay, loss, throughput in networks

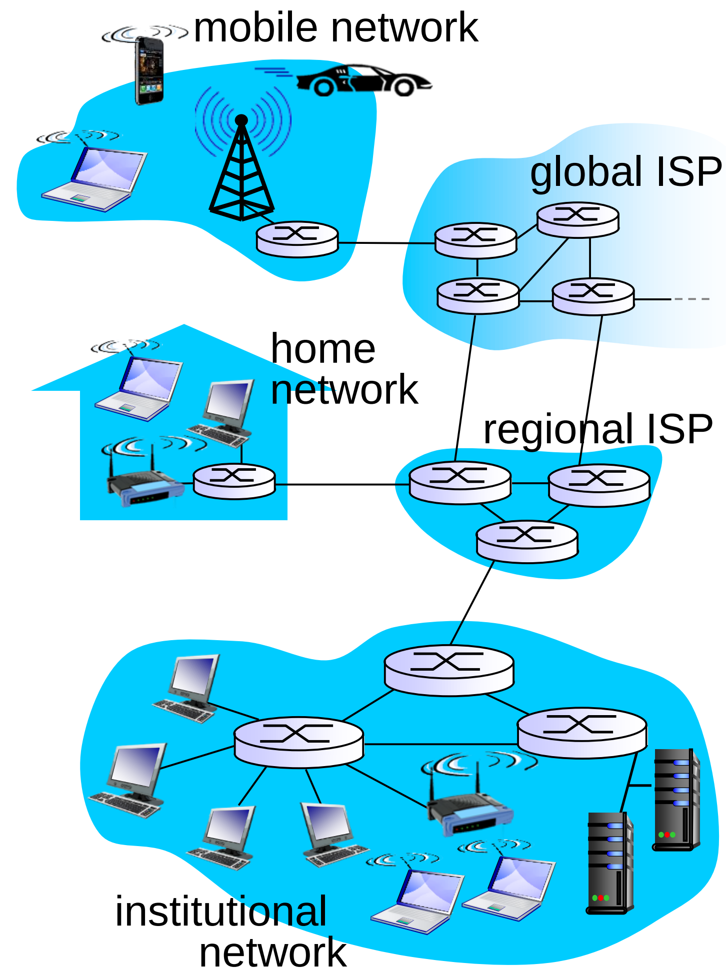
1.5 protocol layers, service models

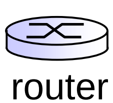
1.6 networks under attack: security

1.7 history

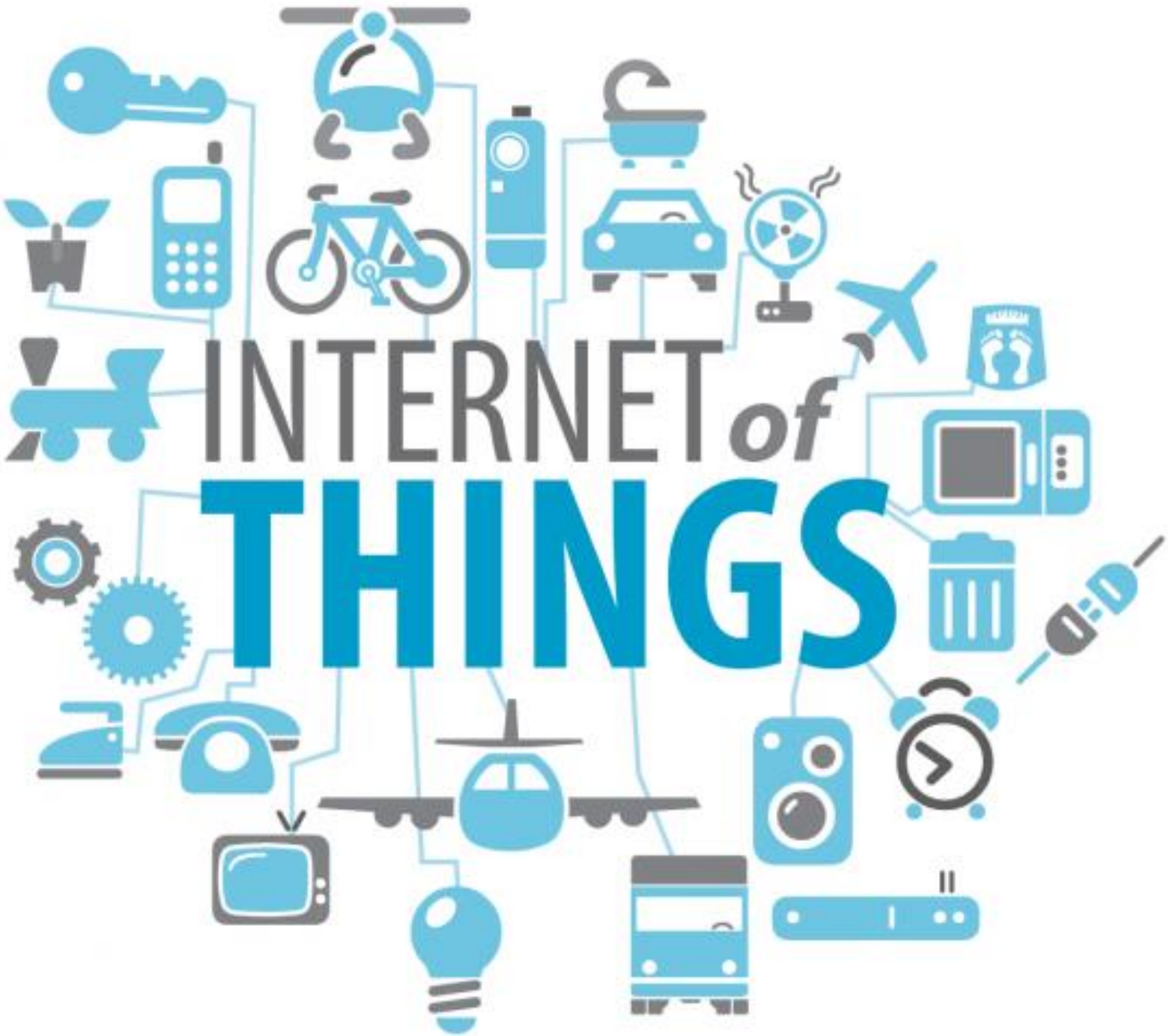
What's the Internet: "nuts and bolts" view

- millions of connected computing devices:
  - *hosts* = *end systems (end devices)*
  - running *network apps*
- *communication links*
  - fiber, copper, radio, satellite
  - transmission rate: *bandwidth*



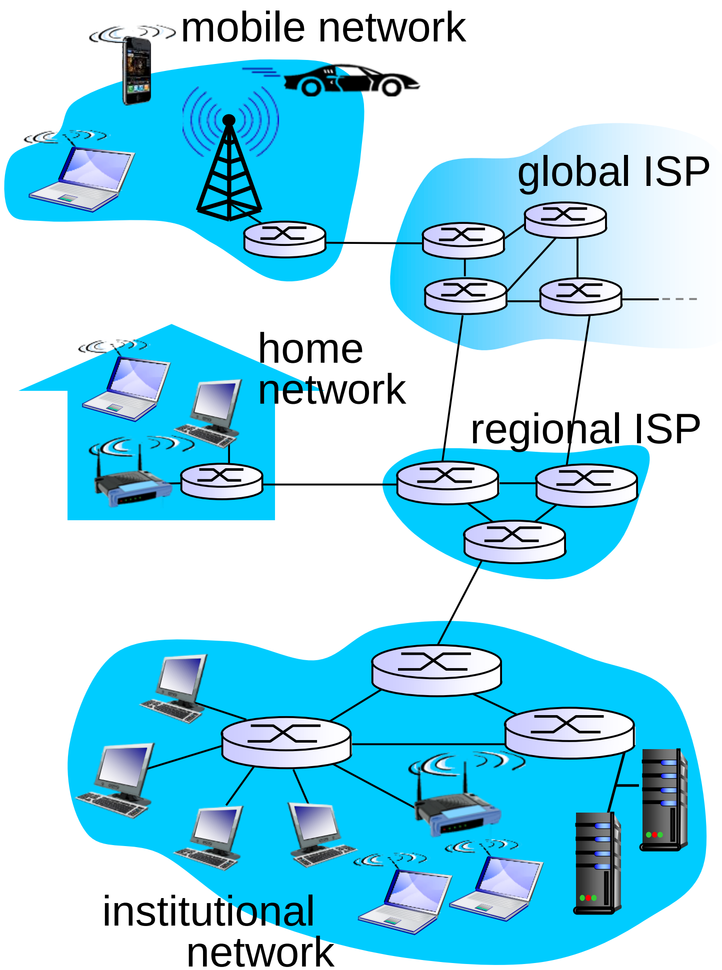


- *Packet switches:* forward packets (chunks of data)
  - *routers* and *switches*


INTERNET of
THINGS

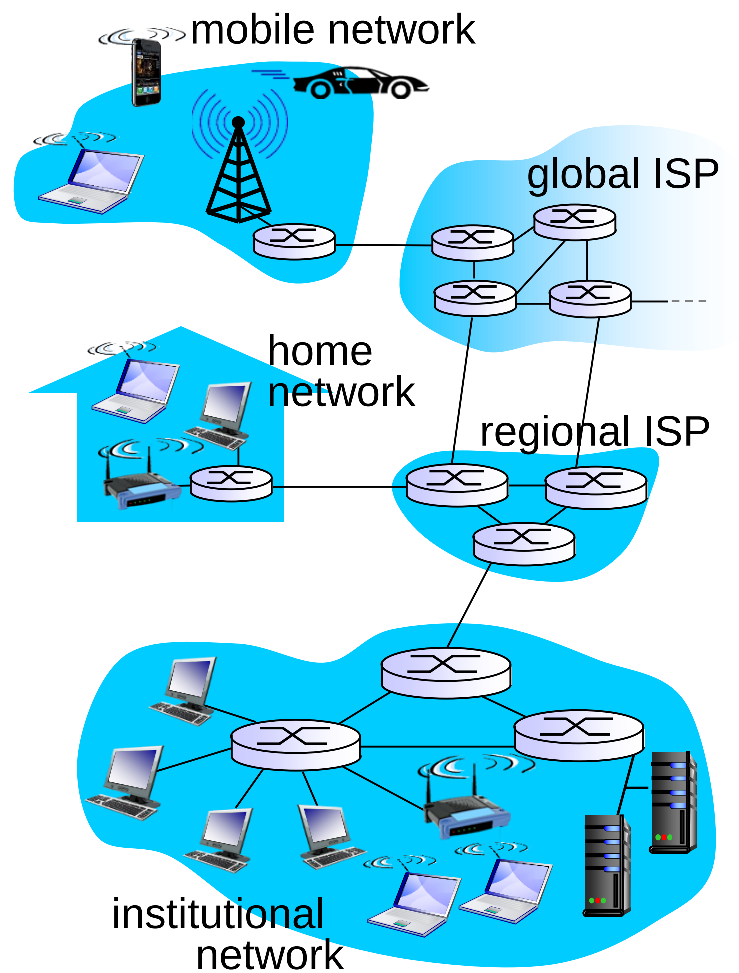
What's the Internet: "nuts and bolts" view

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



What's the Internet: a service view

- *<u>Infrastructure</u> that provides services to applications:*
  - Web, VoIP, email, games, e-commerce, social nets, …
- *provides <u>programming interface</u> to apps (APIs)*
  - **hooks** that allow sending and receiving app programs to "connect" to Internet
  - provides service options, analogous to postal service



What's a protocol?

human protocols:
- "what's the time?"
- "I have a question"
- Introductions

Network Protocol is a set of rules that governs the exchange of messages between communicating parties.

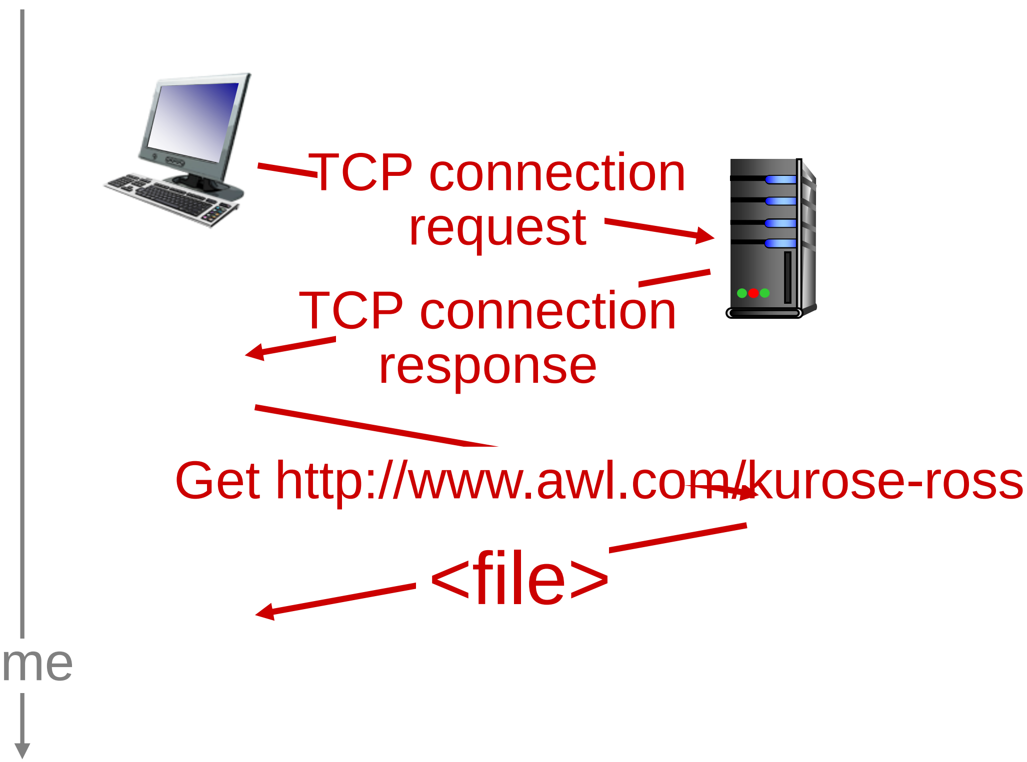
- *Specifically, protocols define* format, order of msgs sent and received among network entities, and actions taken *on msg transmission, receipt*

… specific msgs sent

… specific actions taken when msgs received, or other events

What's a protocol?

a computer network protocol:



Chapter 1: roadmap

1.1 what *is* the Internet and protocol?

1.2 network edge

- end systems, access networks, links

1.3 network core

- packet switching, circuit switching, network structure

1.4 delay, loss, throughput in networks

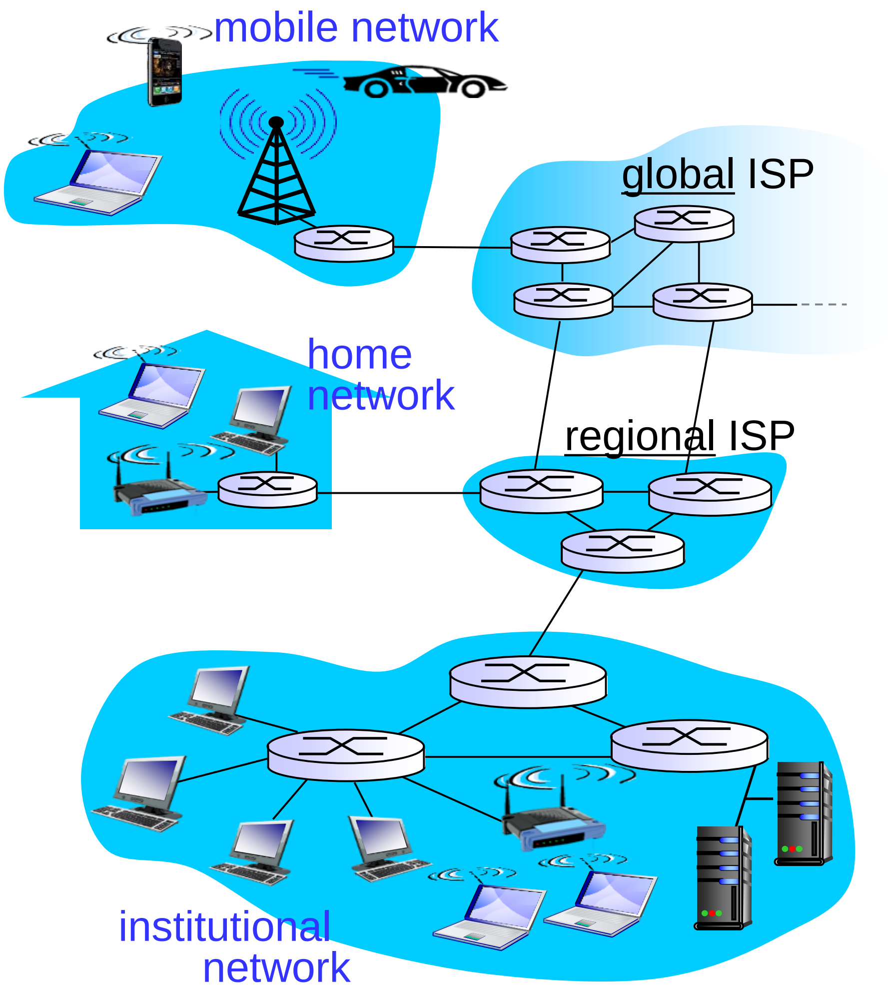
1.5 protocol layers, service models

1.6 networks under attack: security

1.7 history

Network 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



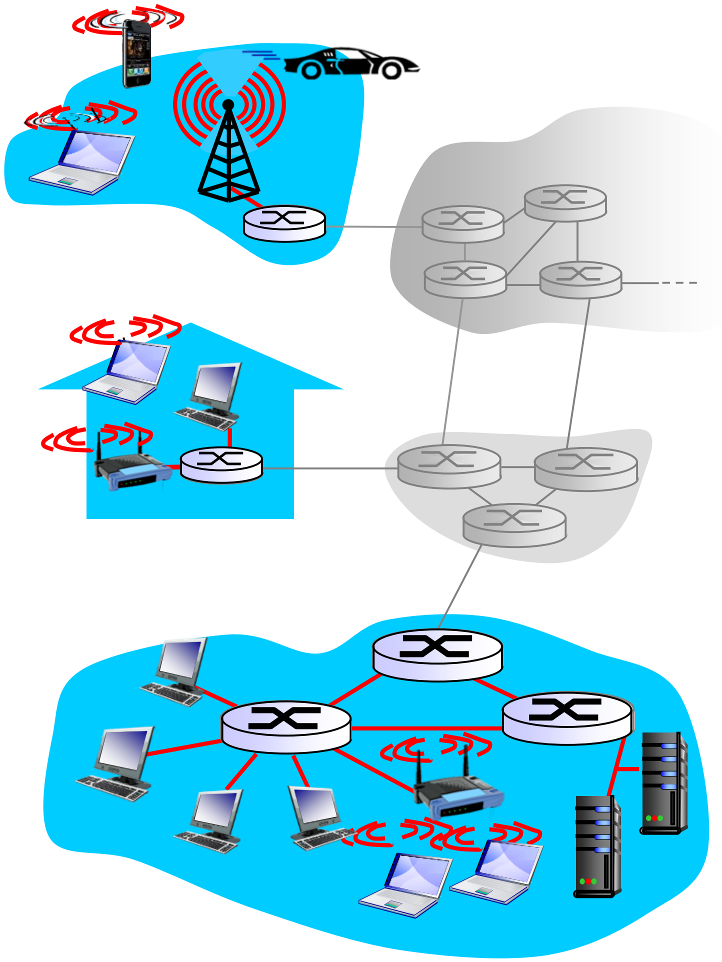
Access networks and physical media

Q: How to connect end systems to edge router?

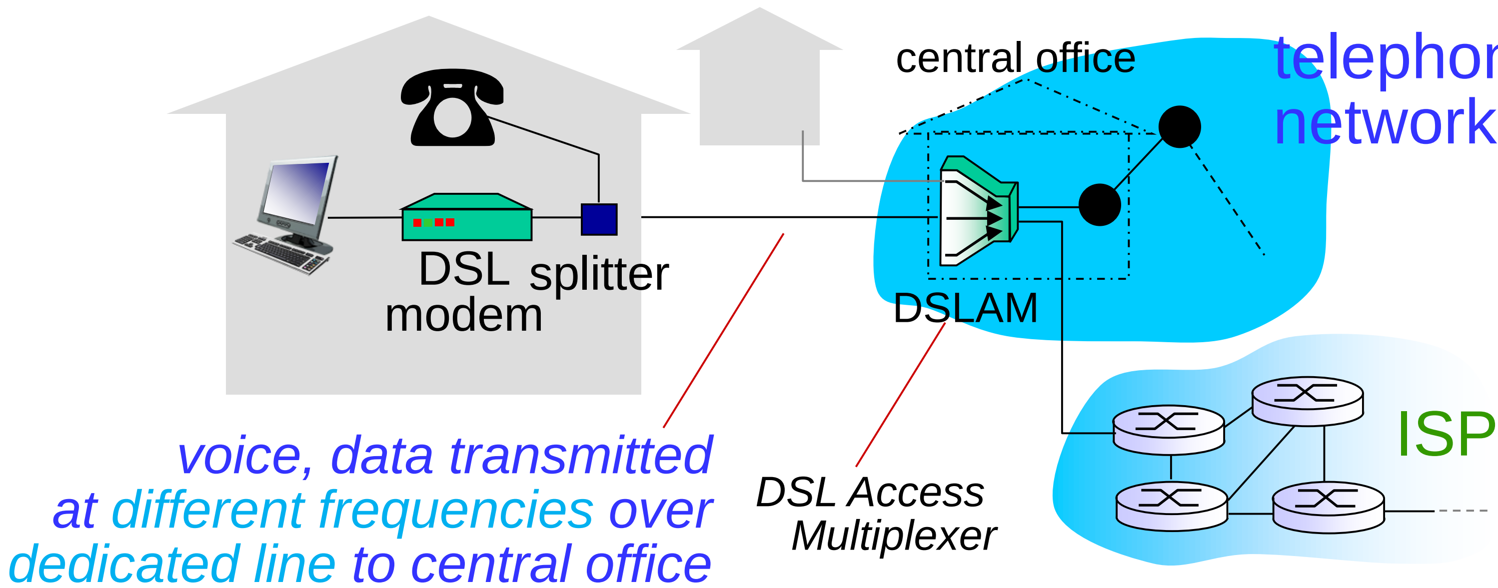
- residential access nets
- institutional access networks (school, company)
- mobile access networks

keep in mind:

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

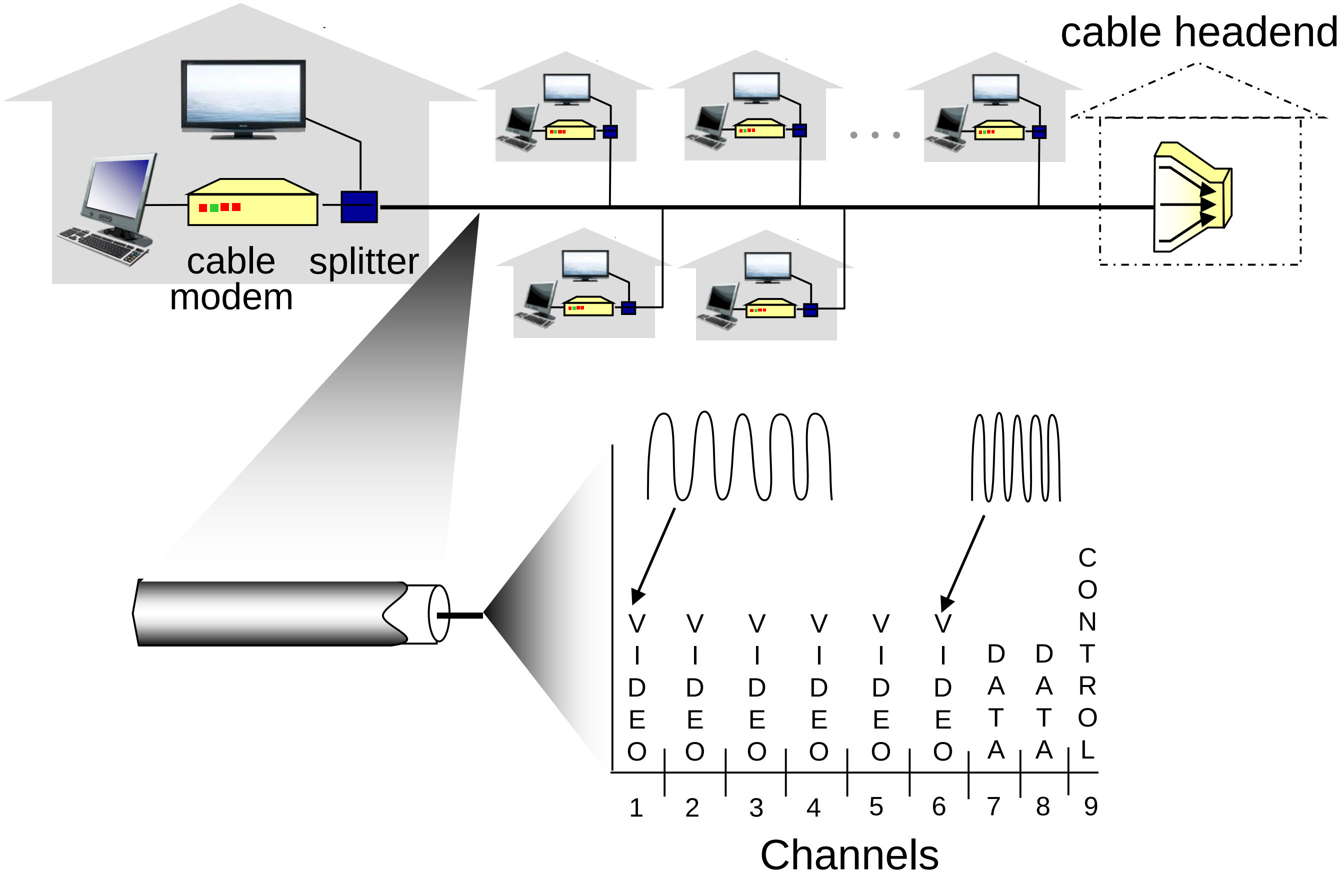


Access net: digital subscriber line (DSL)



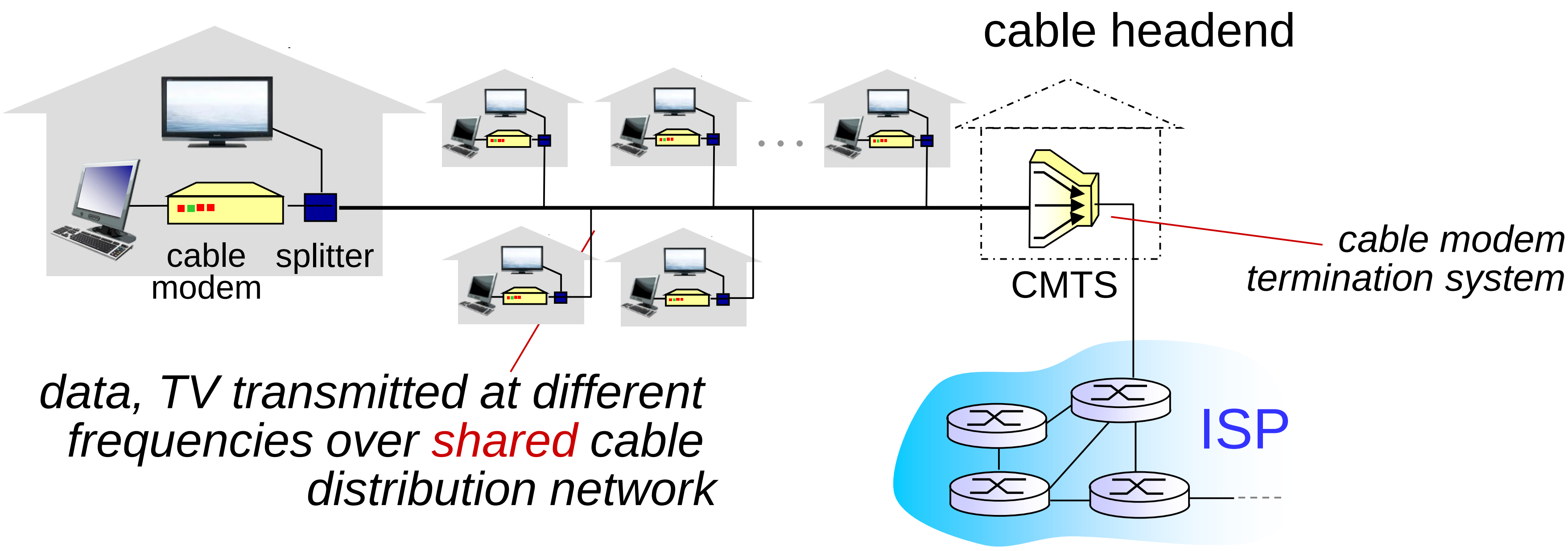
- use *existing* telephone line to central office DSLAM (access multiplexer)
  - data over DSL phone line goes to Internet
  - voice over DSL phone line goes to telephone net
- Up to 400Mbps upstream and downstream transmission rates
- Asymmetric upstream and downstream!

Access net: cable network



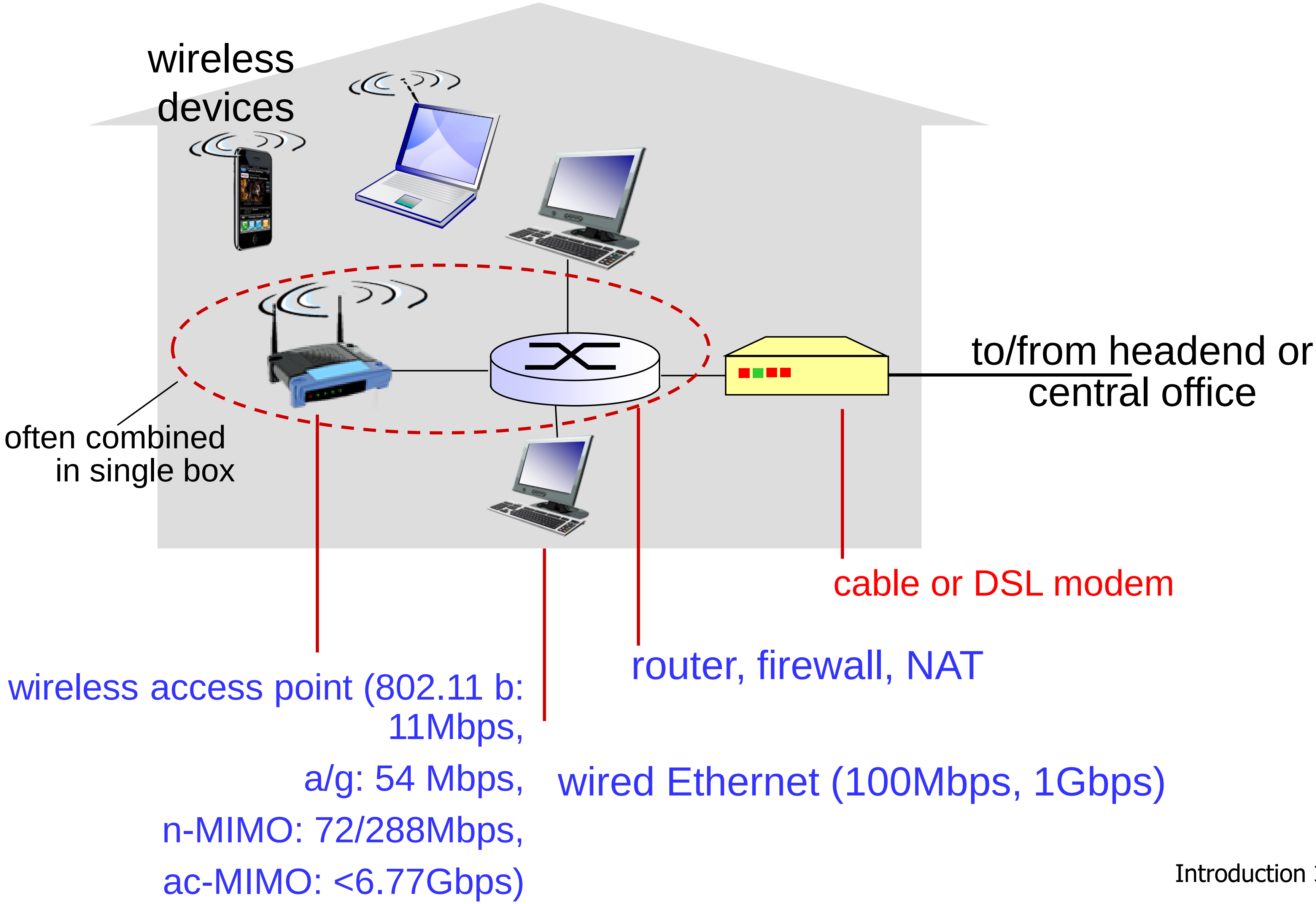
frequency division multiplexing: different **channels** transmitted in different **frequency bands**

Access net: cable network

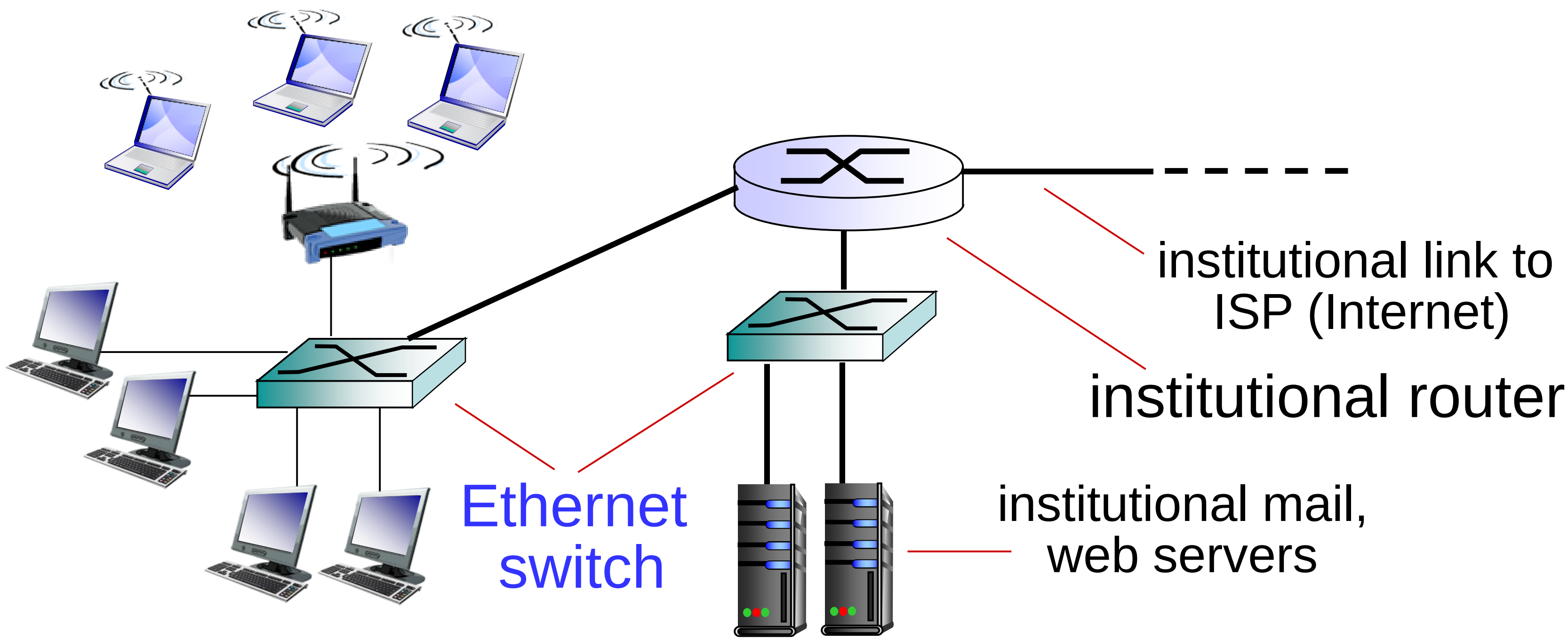


- HFC: hybrid fiber coax
  - **asymmetric**: up to >30Mbps downstream transmission rate, >2 Mbps upstream transmission rate
- network of cable, fiber attaches homes to ISP router
  - homes *share access network* to cable headend
  - unlike DSL, which has dedicated access to central office

Access net: home network



Enterprise access networks (Ethernet)

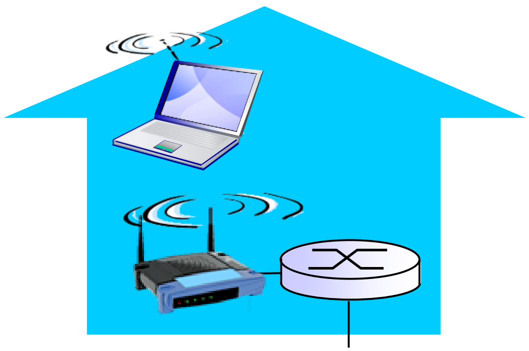


- typically used in companies, universities, etc
- 10Mbps, 100Mbps, 1Gbps, 10Gbps, 100Gbps transmission rates
- end systems typically connect into Ethernet switch

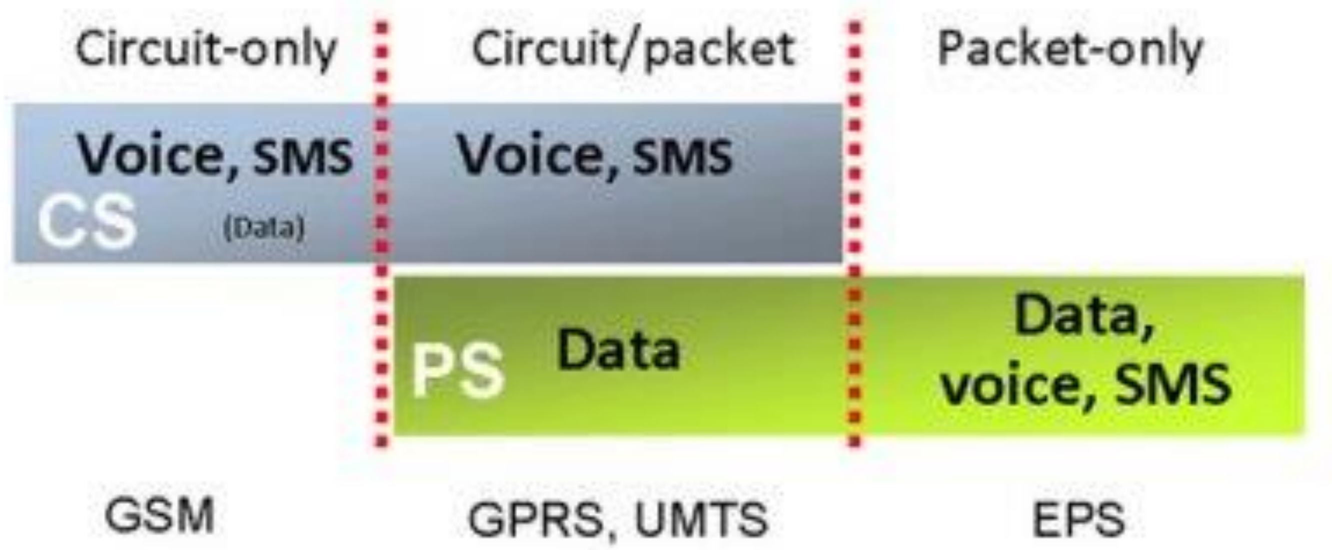
Wireless access networks

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

wireless LANs:
- within building (100 ft)
- 802.11b/g/a/n/ac/ax (WiFi): 11, 54 Mbps, up to 6-7Gbps transmission rate



wide-area wireless access
- provided by telco (cellular) operator, 10's km
- between 1 and 10 Mbps
- 3G, 4G (LTE), 5G

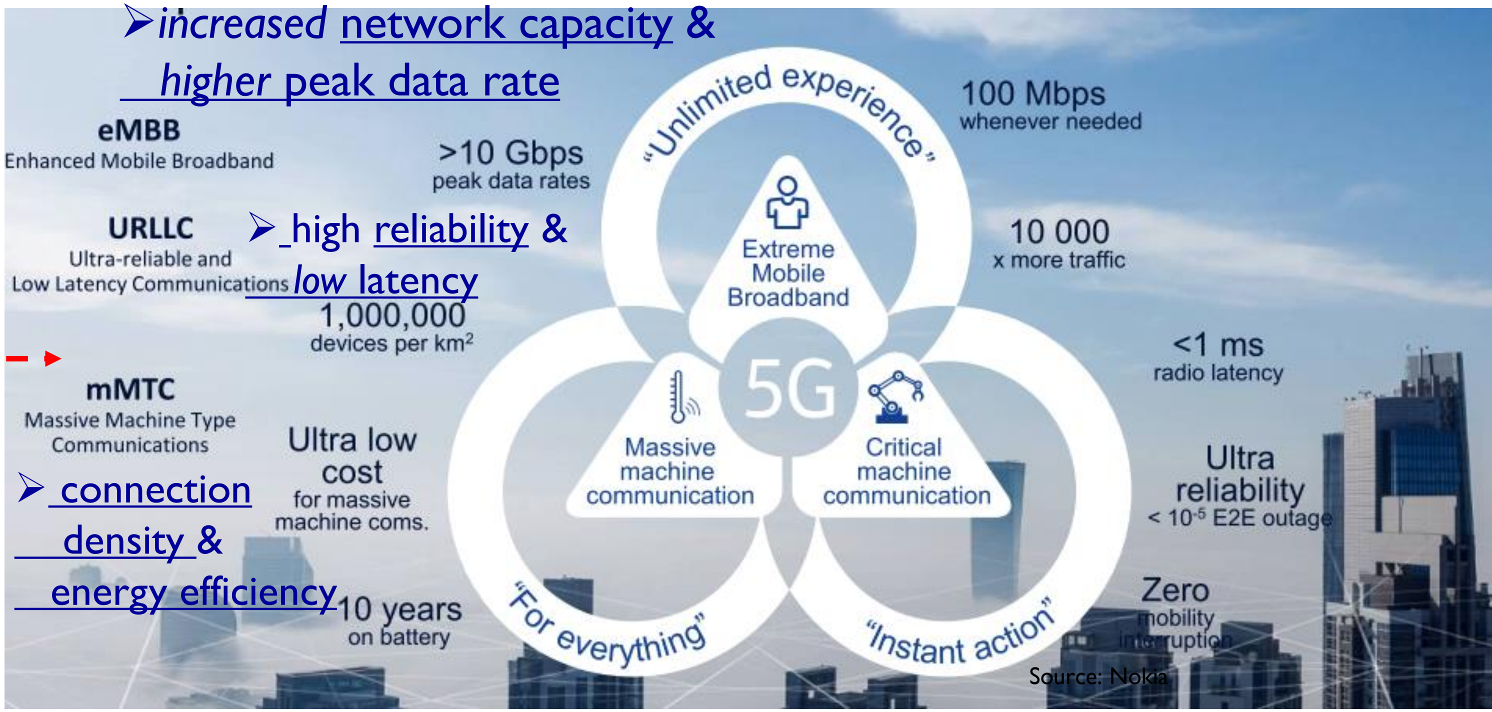

Circuit-only
Circuit/packet
Packet-only
Voice, SMS
CS
(Data)
Voice, SMS
PS
Data
Data,
voice, SMS
GSM
GPRS, UMTS
EPS

2G 3G 4G

5G: Creation of A Variety of New Services without Limits

→ **High bandwidth, Low latency, High deployment density, Service-oriented.**

- World Economic Forum (WEF) termed the "Fourth Industrial Revolution.
- A world of enterprise **Internet of Things (IoT)** deployments

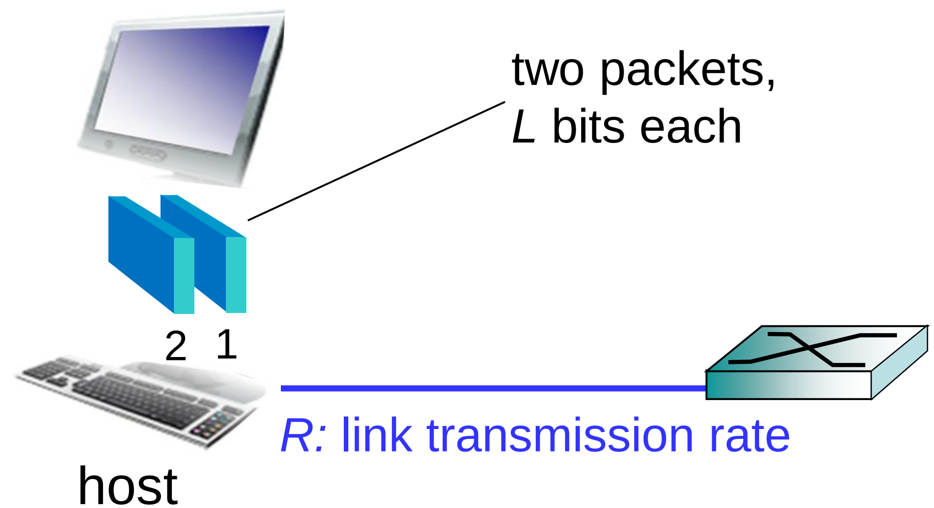


- *increased* network capacity & *higher* peak data rate
- high reliability & *low* latency
- connection density & energy efficiency

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*
  - link transmission rate, aka link *capacity, aka link bandwidth*



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

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

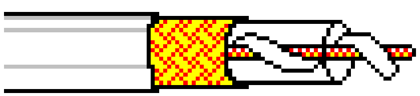
twisted pair (TP)

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

Physical media: coax, fiber

coaxial cable:

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



fiber optic cable:

- glass fiber carrying light pulses, each pulse a bit
- high-speed operation:
  - high-speed point-to-point transmission (e.g., 10's-100's Gpbs transmission rate)
- low error rate:
  - repeaters spaced far apart
  - immune to electromagnetic noise

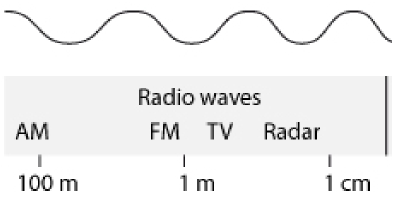
Physical media: radio

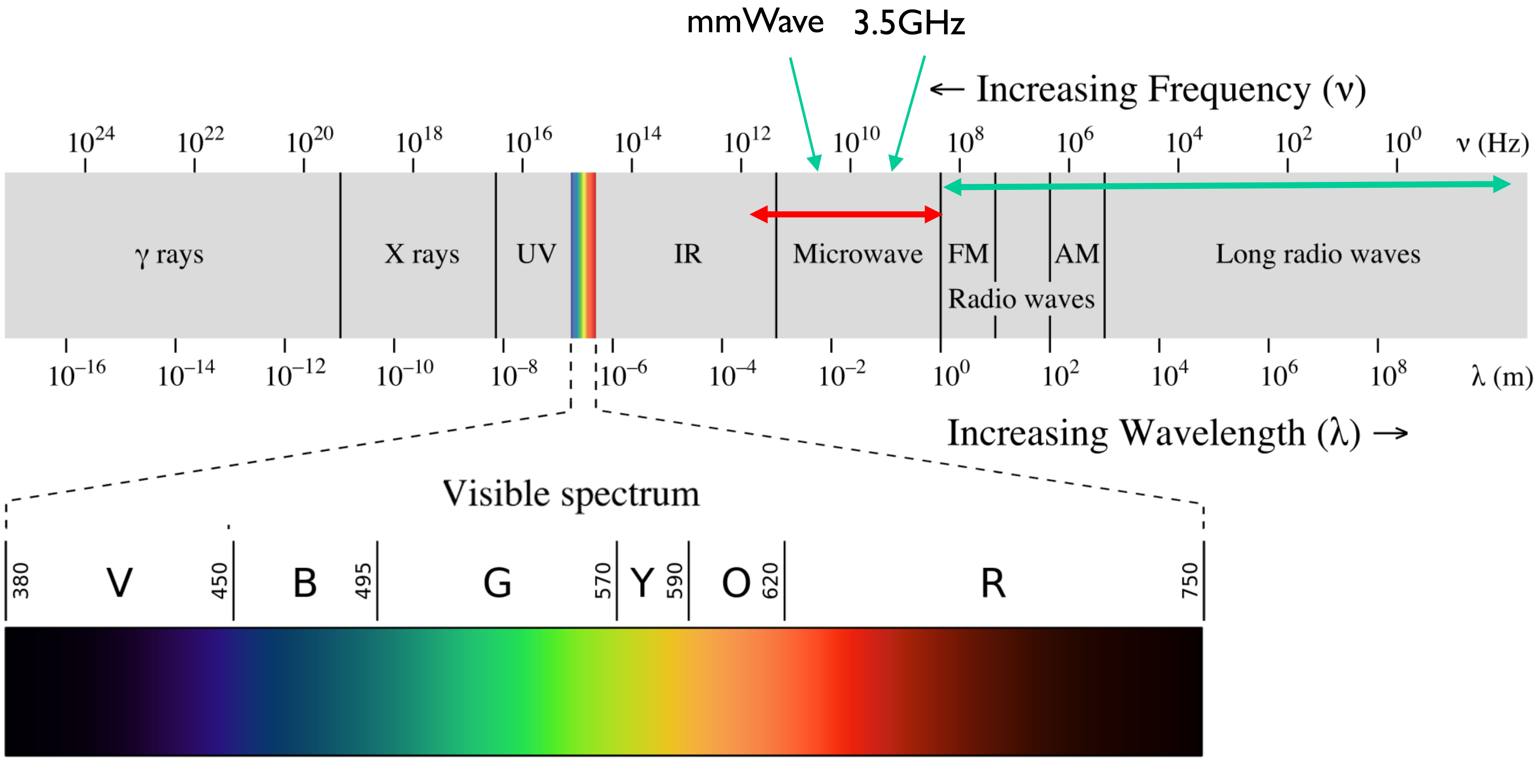
- signal carried in electromagnetic spectrum
- no physical "wire"
- bidirectional
- propagation environment effects:
  - reflection
  - obstruction by objects
  - interference

radio link types:

- terrestrial microwave
  - e.g. up to 45 Mbps channels
- LAN (e.g., WiFi)
  - 11Mbps, 54 Mbps, 600Mbps, 1Gbps, 6.9Gbps, 9.6Gbps
- wide-area (e.g., cellular)
  - 3G cellular: ~ few Mbps
- satellite
  - Kbps to 45Mbps channel (or multiple smaller channels)
  - 270 msec end-end delay
  - geosynchronous versus low altitude

ElectroMagnetic Spectrum





Chapter 1: roadmap

1.1 what *is* the Internet?

1.2 network edge

- end systems, access networks, links

1.3 network core

- packet switching, circuit switching, network structure

1.4 delay, loss, throughput in networks

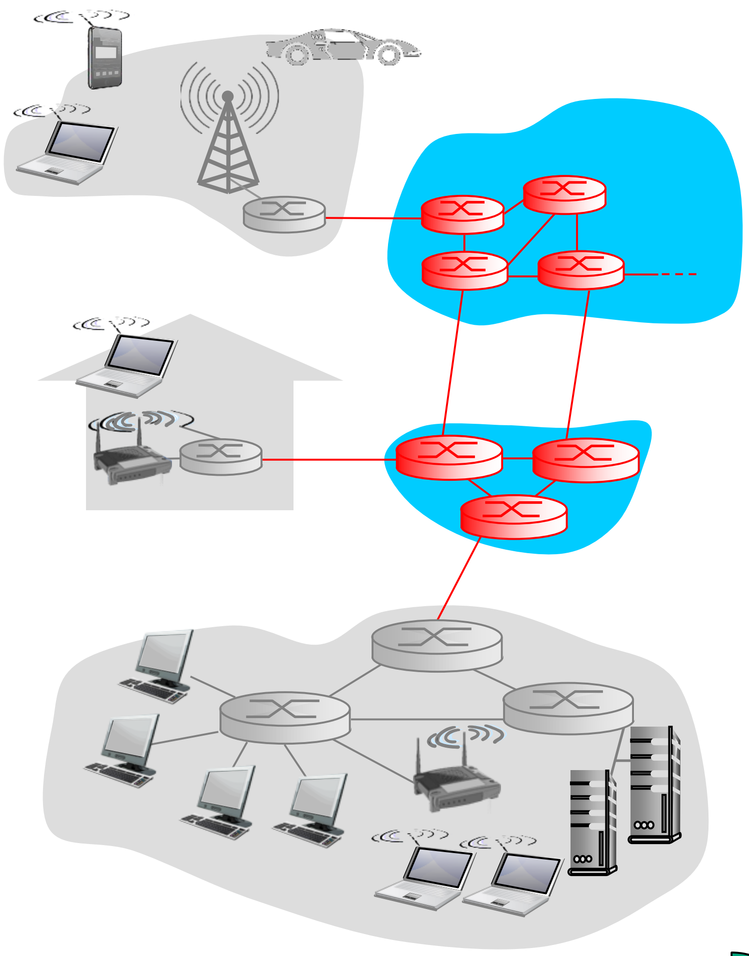
1.5 protocol layers, service models

1.6 networks under attack: security

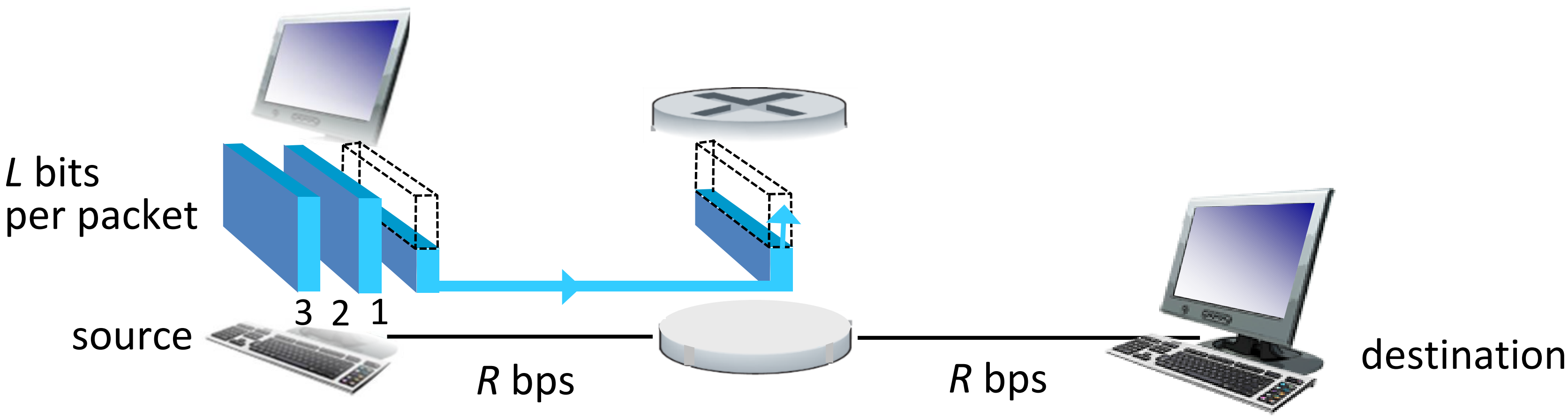
1.7 history

The network core

- **mesh of interconnected <u>routers</u>**
- 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



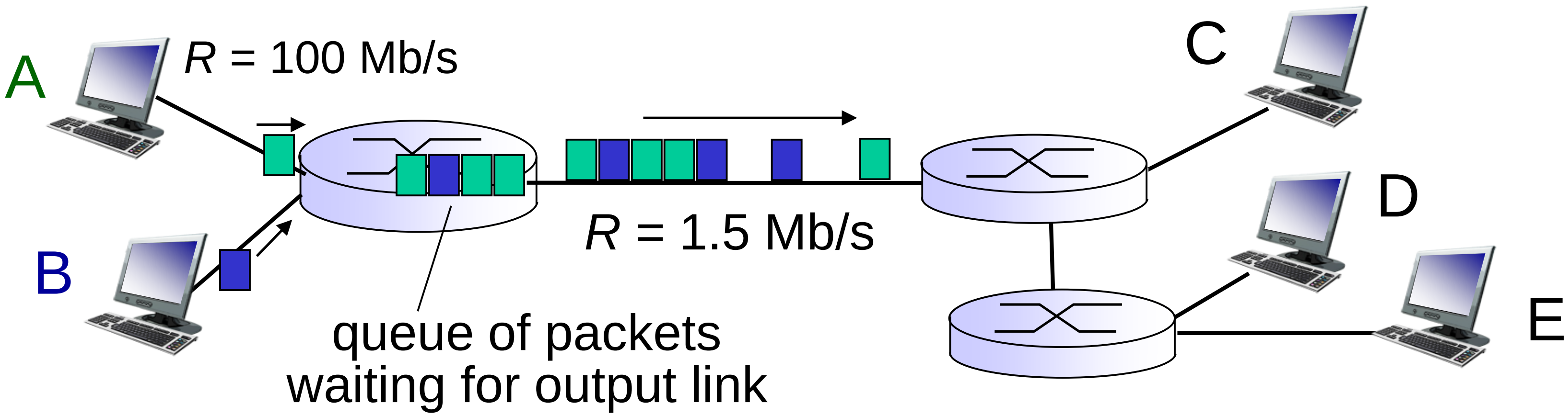
- takes L/R seconds to transmit (push out) L-bit packet into link at R bps
- *store and forward:* entire packet must arrive at router before it can be transmitted on next link
- end-end delay = $2L/R$ (assuming zero propagation delay)

more on delay shortly …

one-hop numerical example:

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

Packet Switching: queueing delay, loss



queueing and loss:

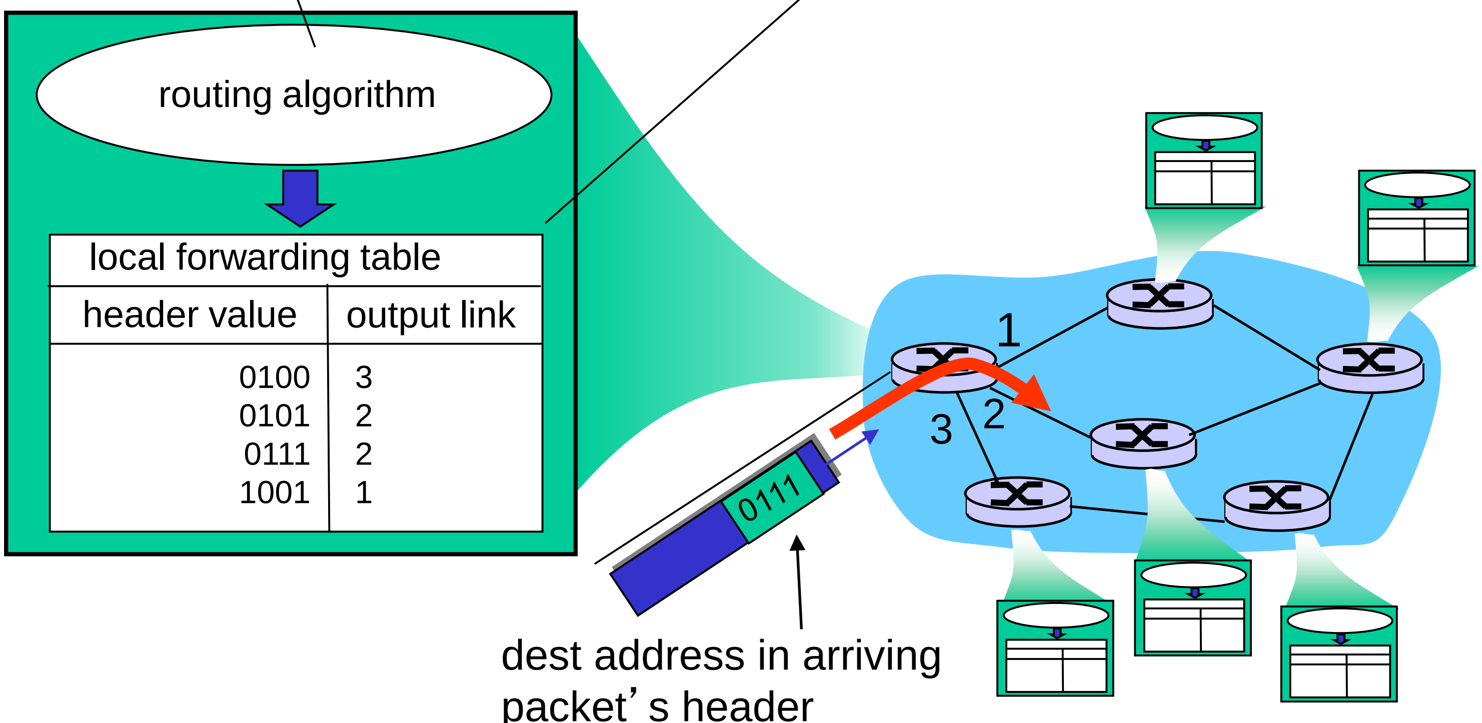
- If **arrival rate** (in bits) to link exceeds **transmission rate** of link ***for a period of time***:
  - packets will **queue**, wait to be transmitted on link
  - packets can be **dropped** (lost) if memory (buffer) fills up

Two key network-core functions

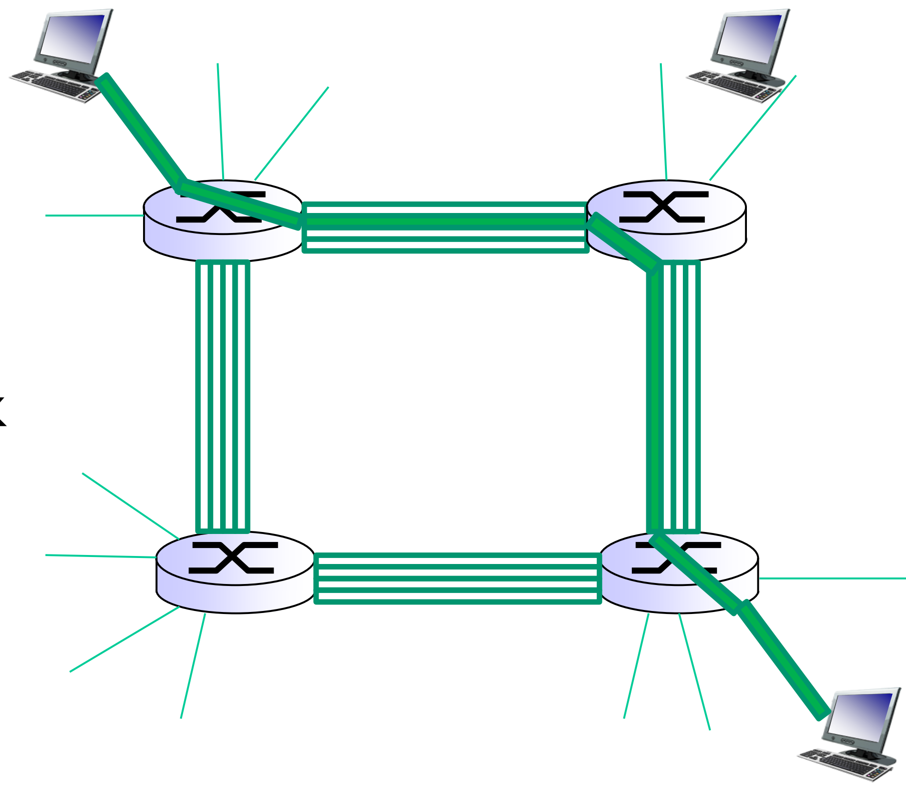
routing: determines source-destination route taken by packets

- *routing algorithms*

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



Alternative core: circuit switching



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

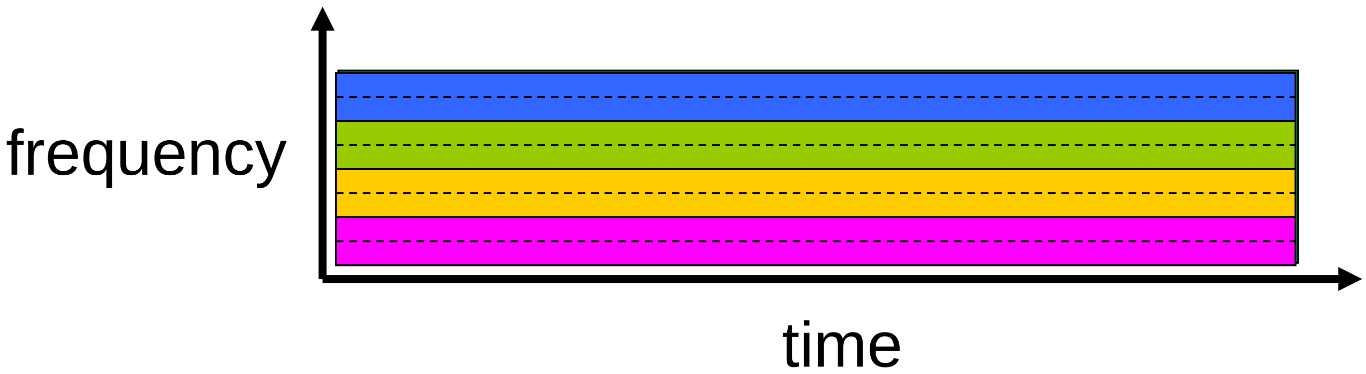
- In diagram, each link has four circuits.
  - call gets 2nd circuit in top link and 1st circuit in right link.
- **dedicated** resources: no sharing
  - **circuit-like (guaranteed) performance**
- circuit segment idle if not used by call *(no sharing)*
- Commonly used in **traditional telephone networks**

Circuit switching: FDM versus TDM

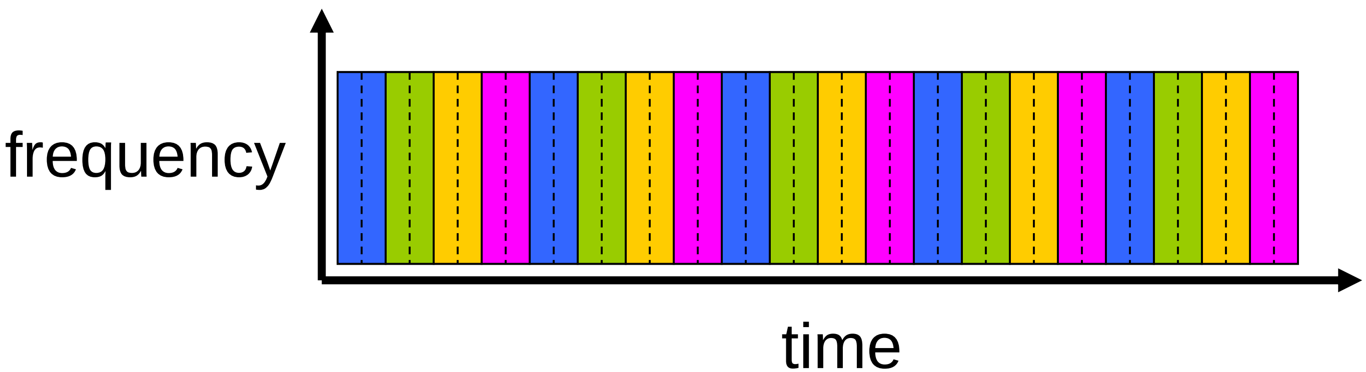
Example:

4 users

FDM



TDM

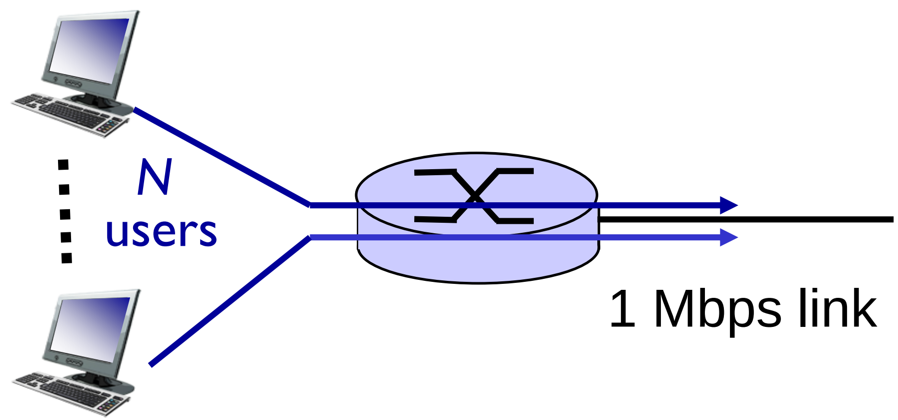


Packet switching versus circuit switching

packet switching ***allows more users to use network!***

example:

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



- *circuit-switching:*
  - 10 users
- *packet switching:*
  - with 35 users, probability > 10 active at same time is less than .0004 *

Q: how did we get value 0.0004?

Q: what happens if > 35 users ?

Packet switching versus circuit switching

is packet switching a "slam dunk winner?"

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

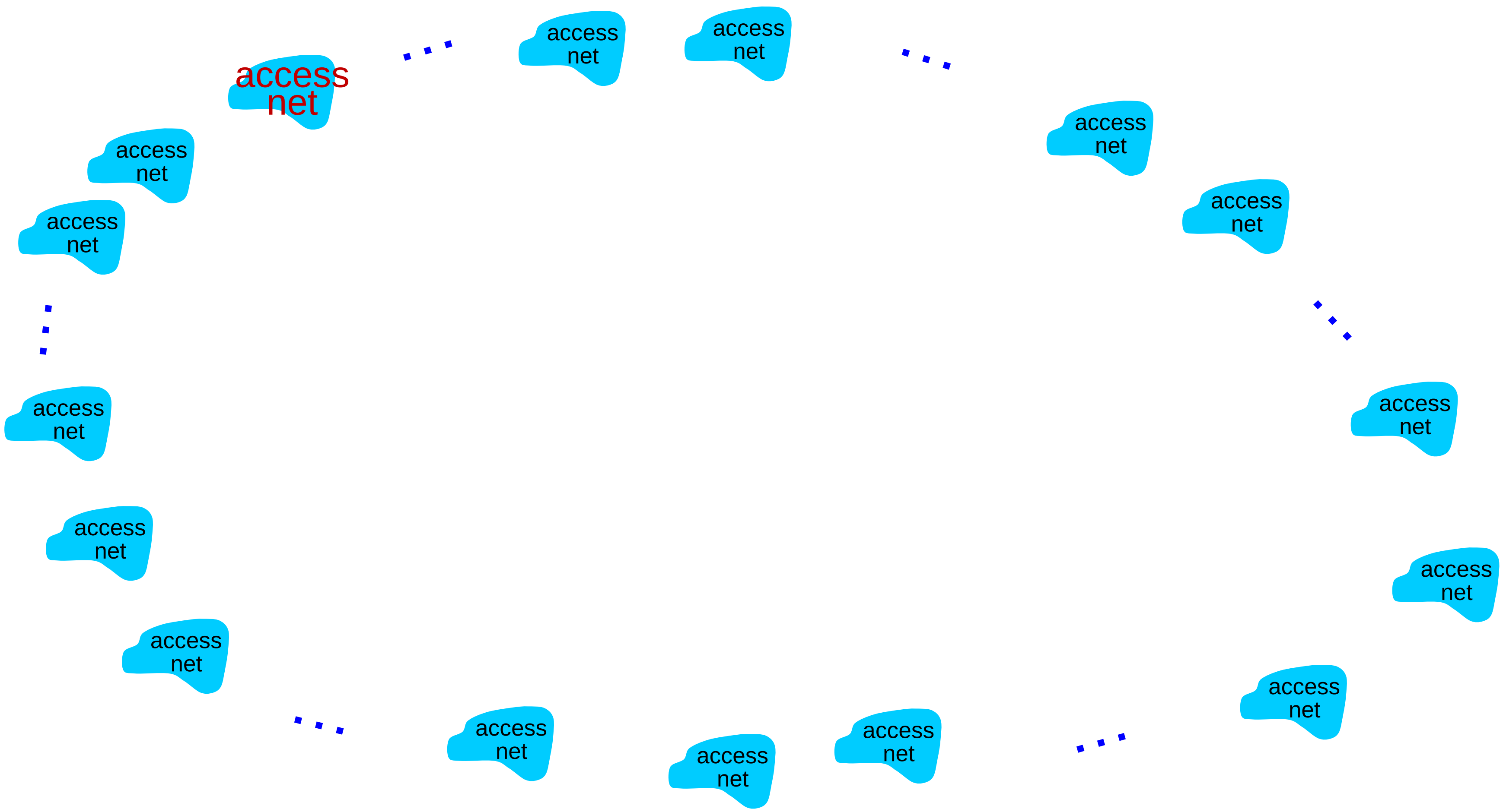
Q: human analogies of reserved resources (circuit switching) versus on-demand allocation (packet-switching)?

Internet structure: network of networks

- End systems connect to Internet via access ISPs (Internet Service Providers)
  - Residential, company and university ISPs, wireless networks
- 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

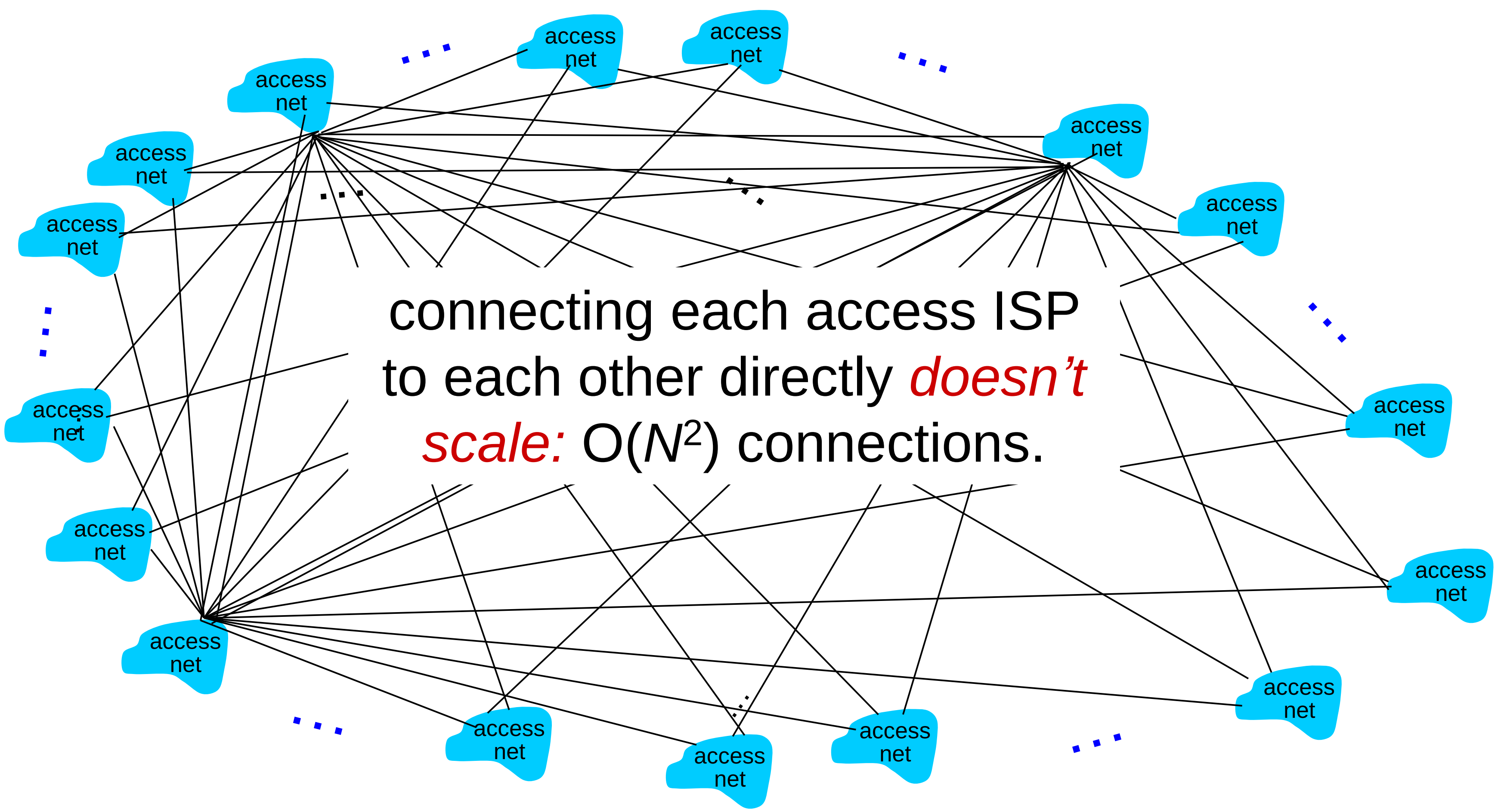
Internet structure: network of networks

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



Internet structure: network of networks

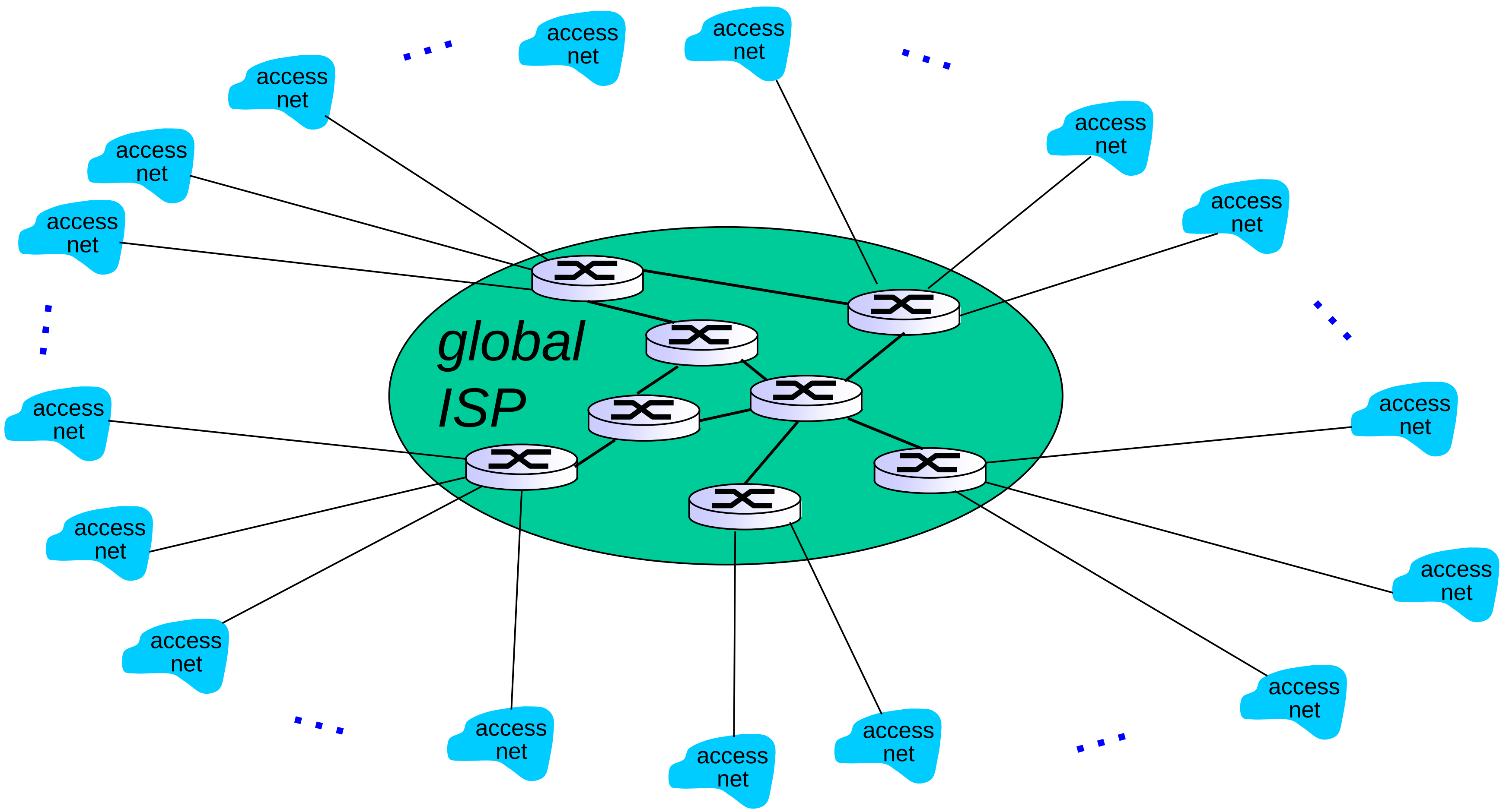
Option: connect each access ISP to every other access ISP?



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

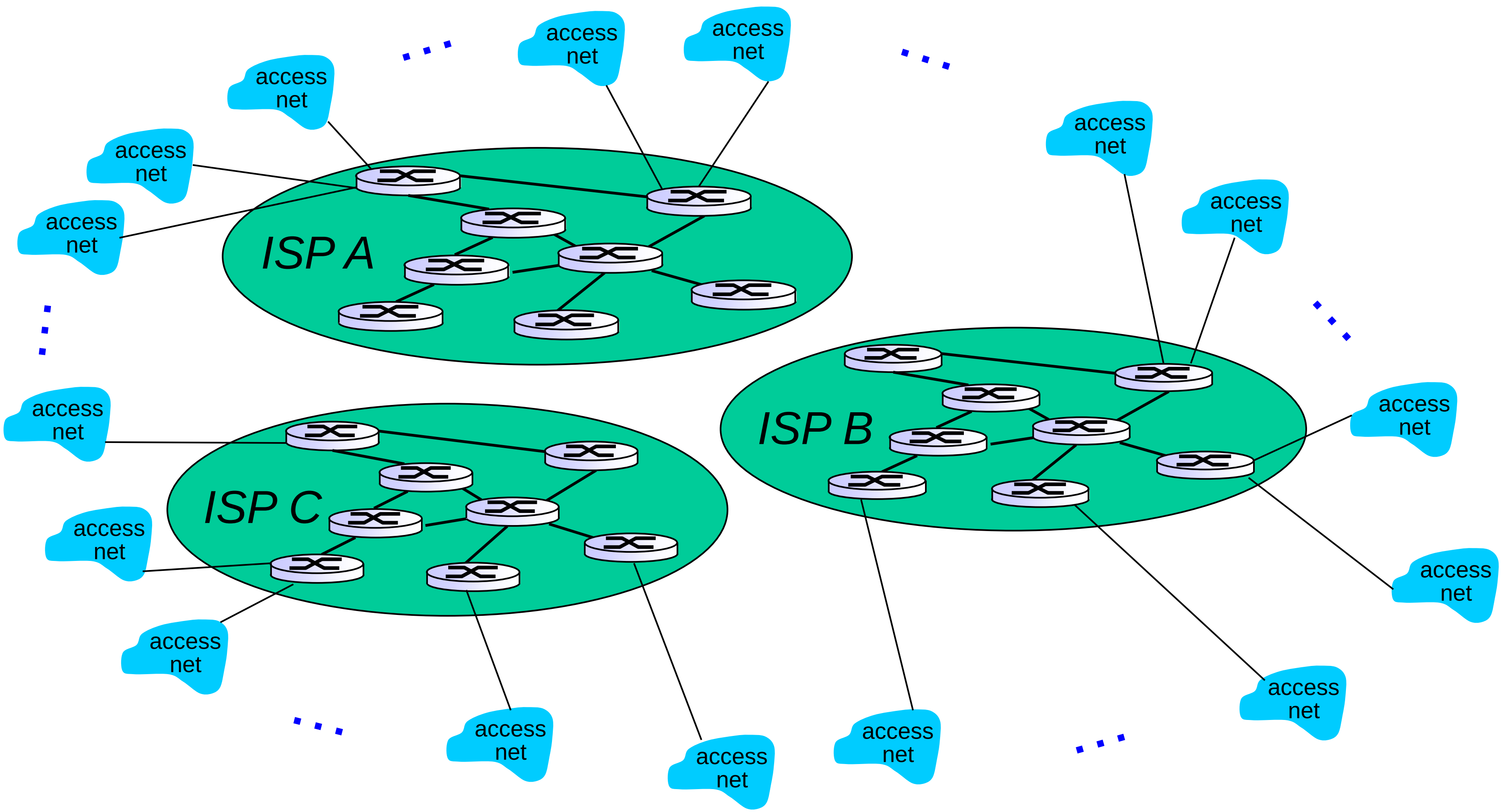
Internet structure: network of networks

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

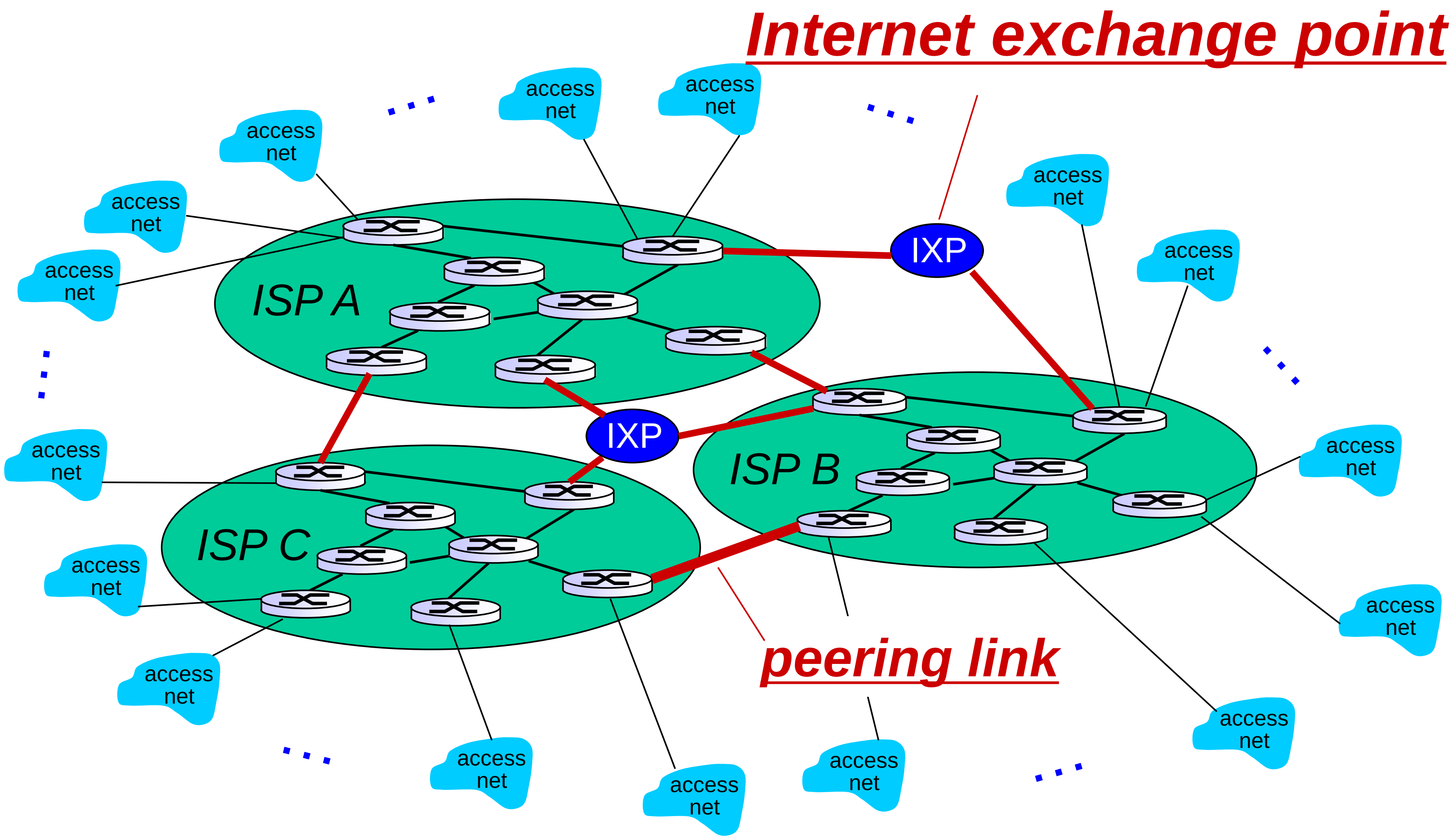


Internet structure: network of networks

One vs. multiple global ISPs ….

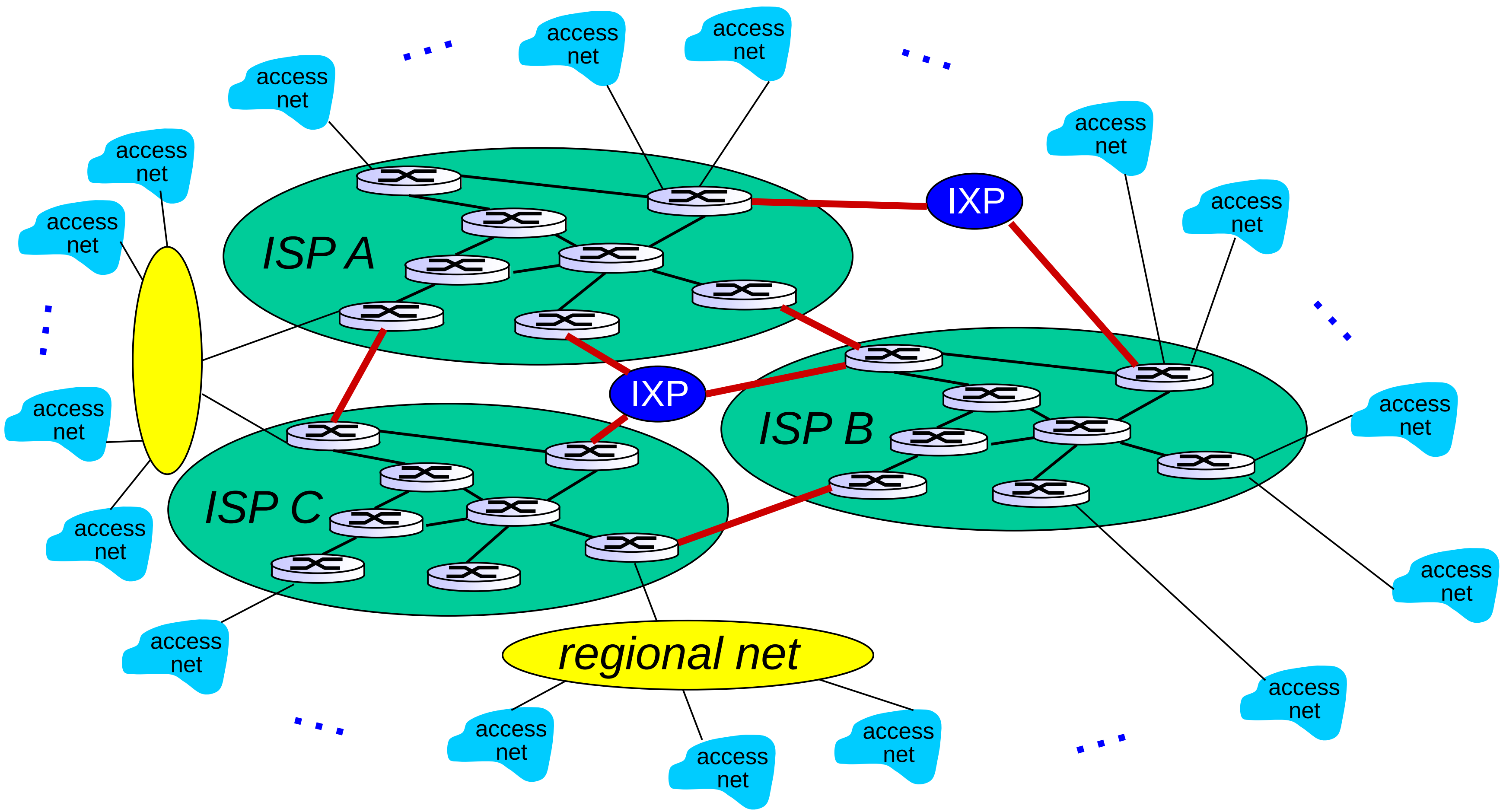


Internet structure: network of networks



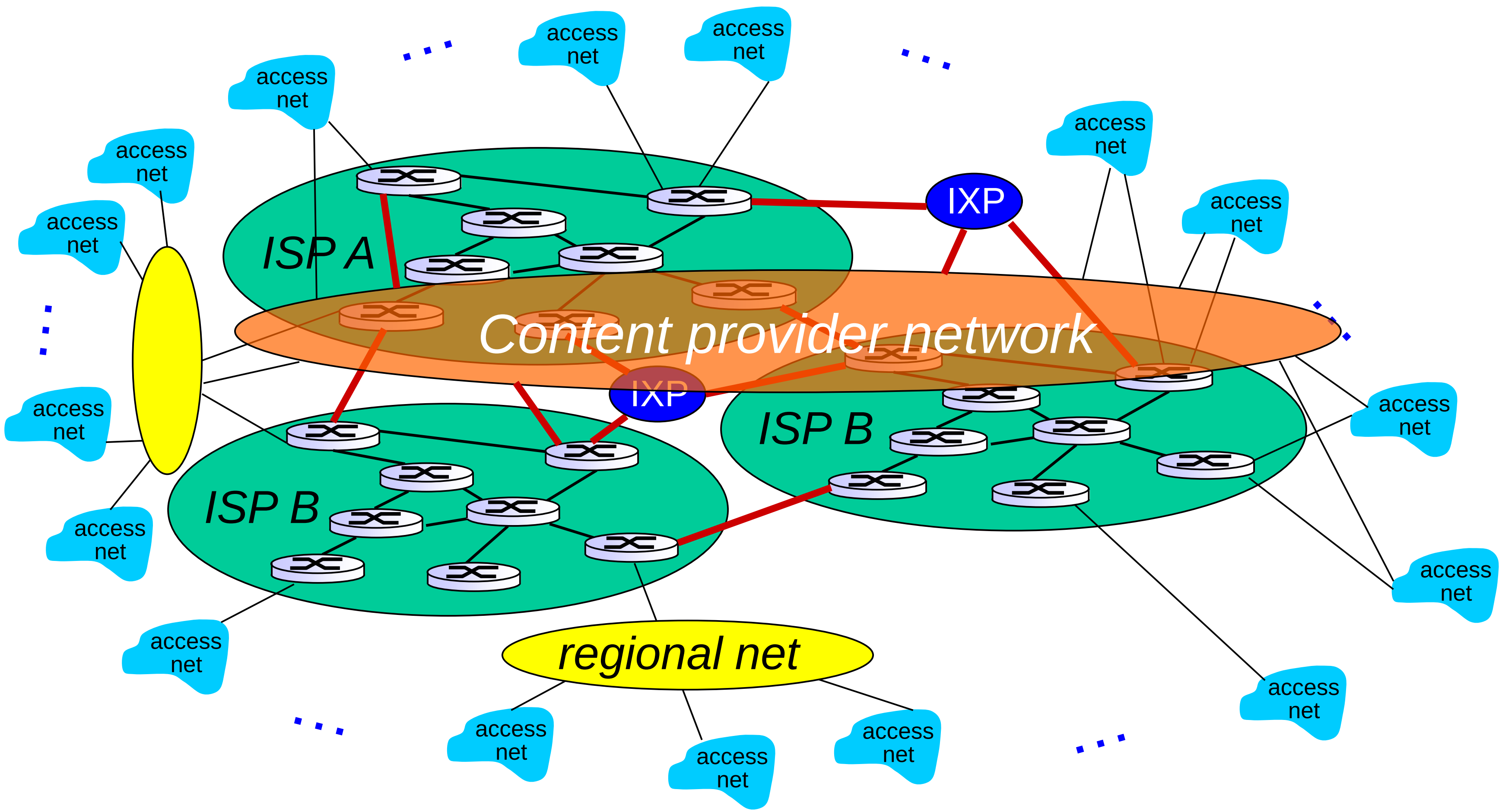
Internet structure: network of networks

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

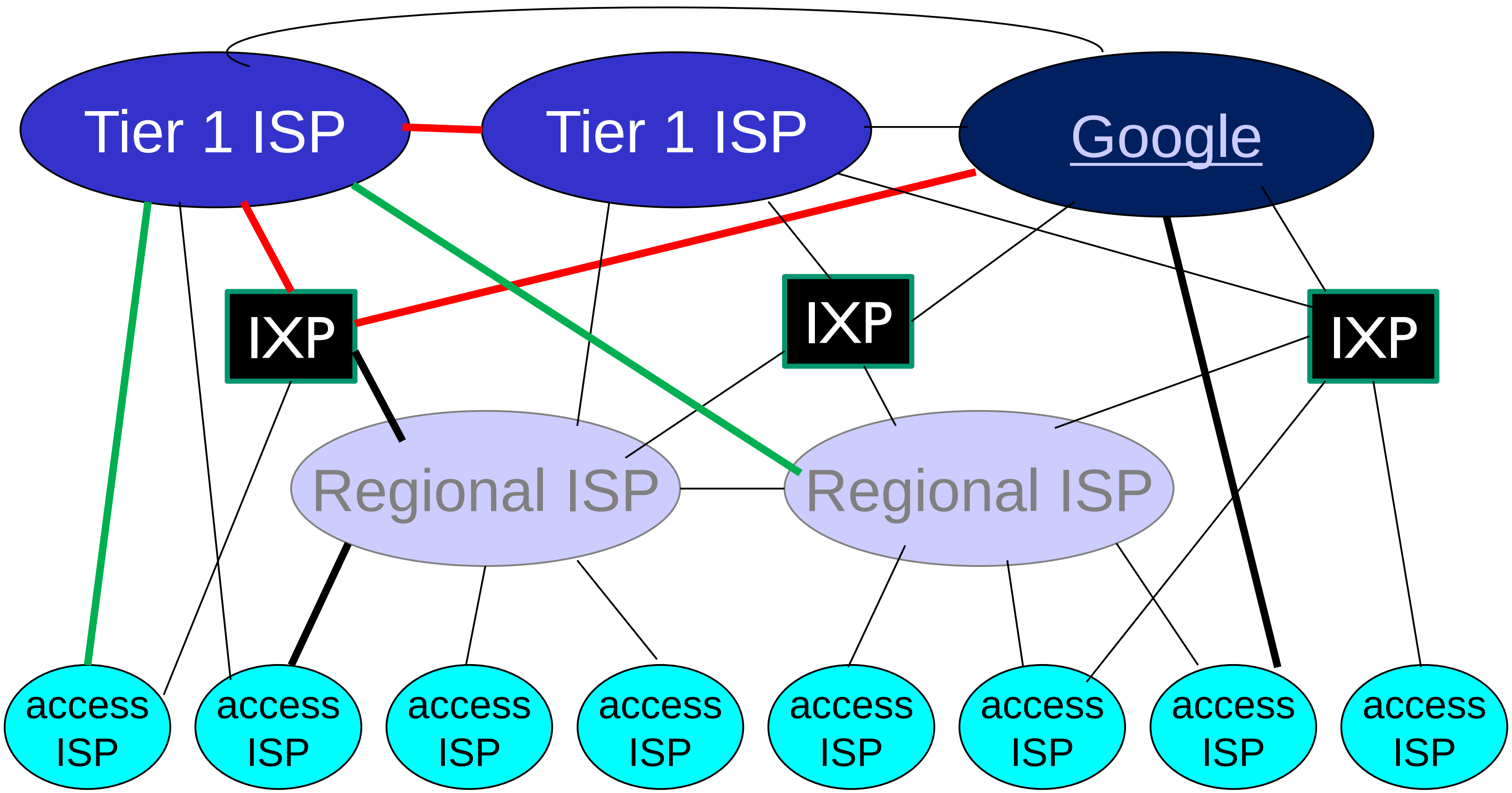


Internet structure: network of networks

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

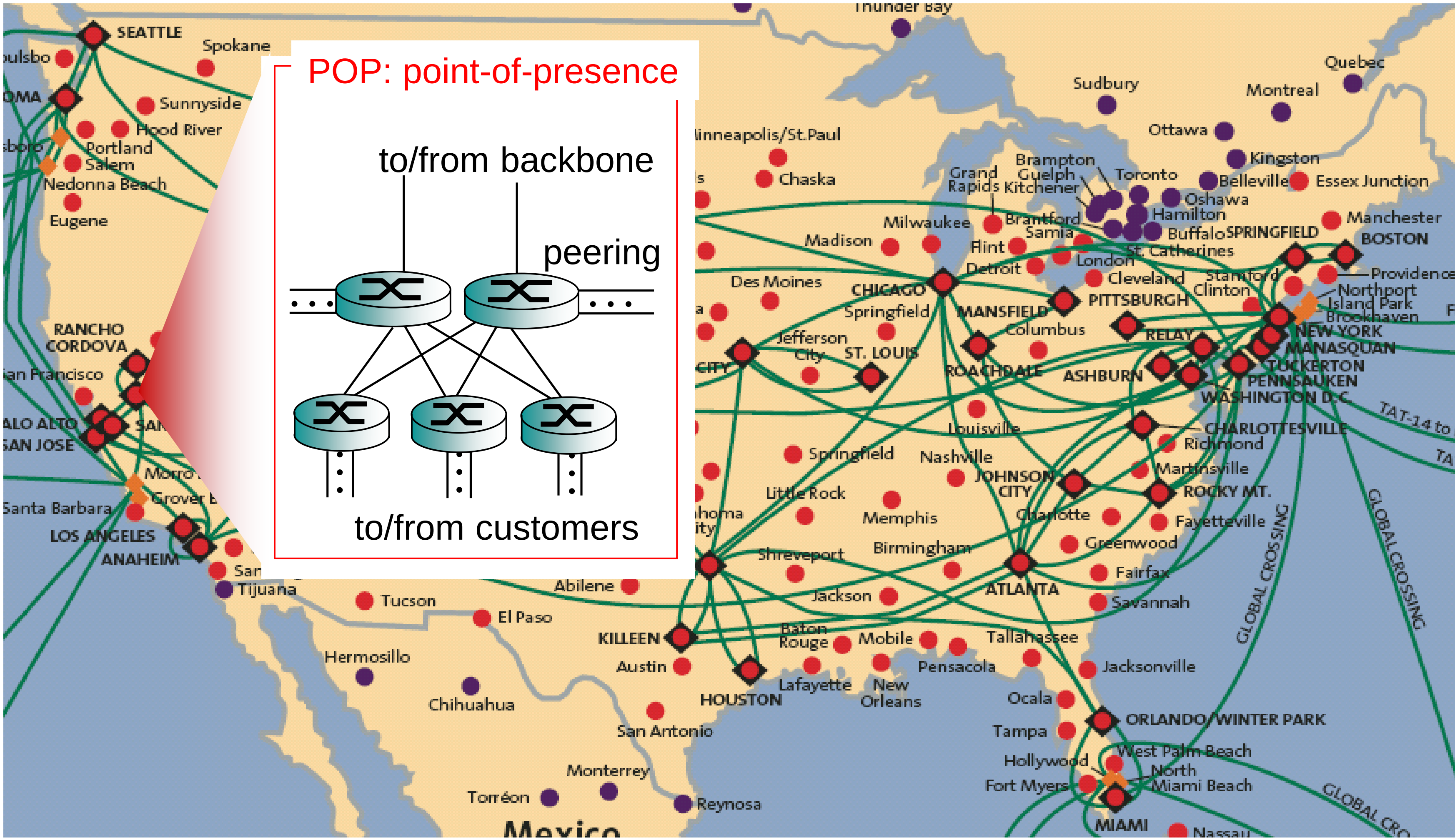


Internet structure: network of networks



- 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 its data centers to Internet, often bypassing tier-1, regional ISPs

Tier-1 ISP: e.g., Sprint



Chapter 1: roadmap

1.1 what *is* the Internet and protocol?

1.2 network edge

- end systems, access networks, links

1.3 network core

- packet switching, circuit switching, network structure

1.4 delay, loss, throughput in networks

1.5 protocol layers, service models

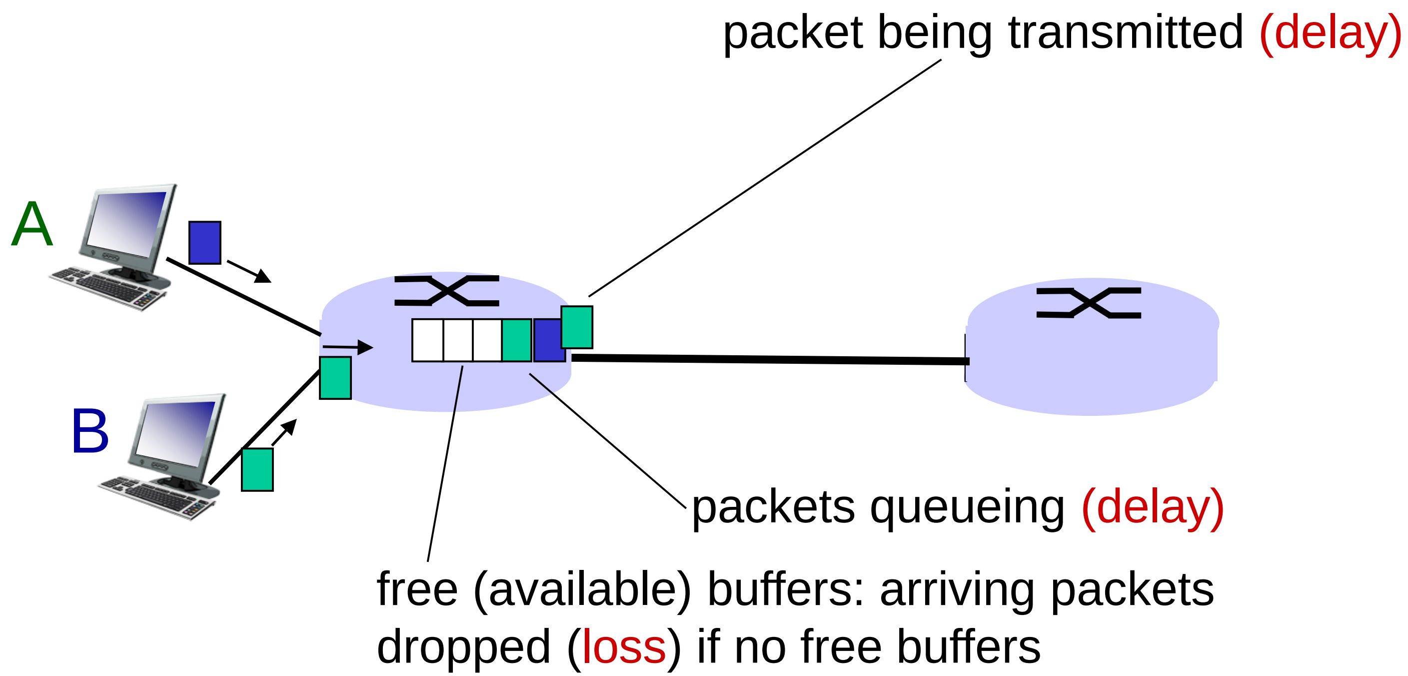
1.6 networks under attack: security

1.7 history

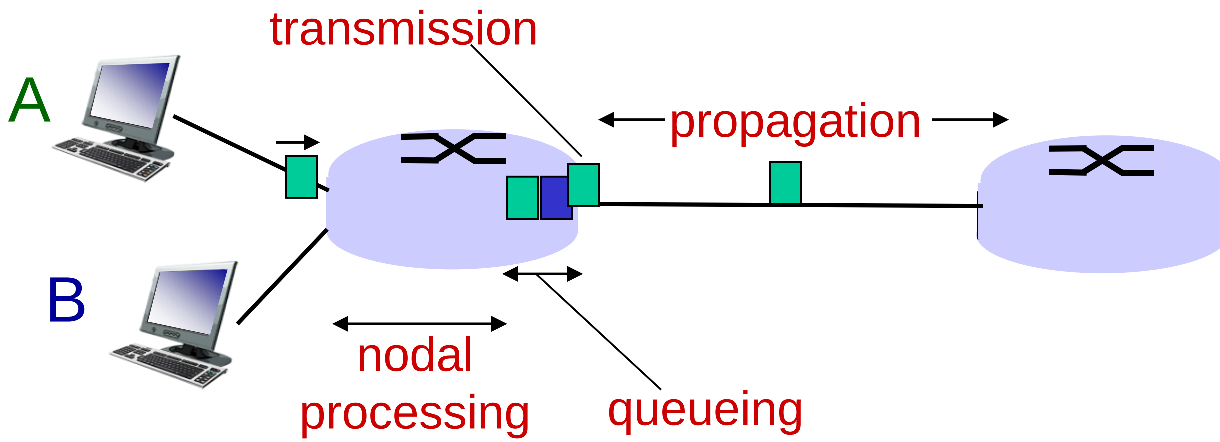
How do **loss** and **delay** occur?

packets *queue* in router buffers

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



Four sources of packet delay



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

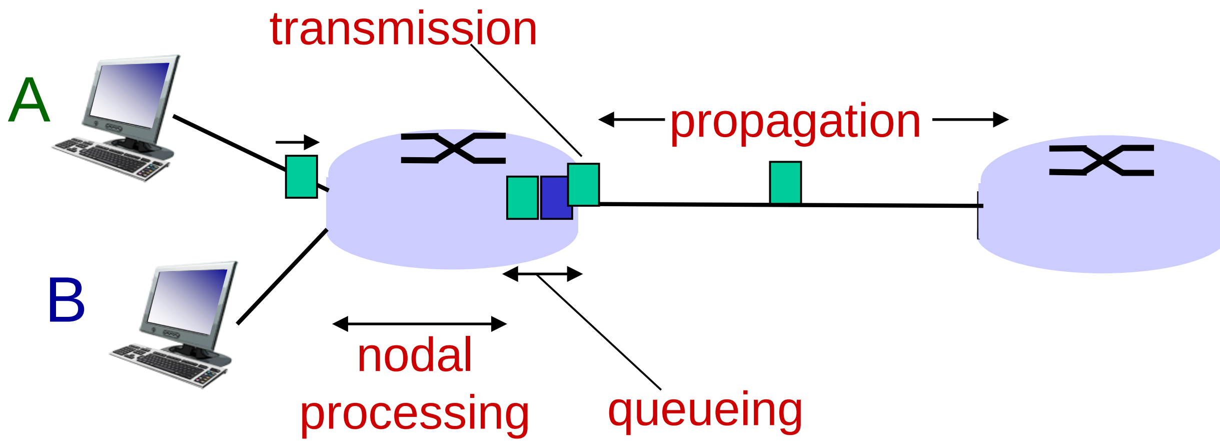
d_{proc}: nodal processing

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

d_{queue}: queueing delay

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

Four sources of packet delay



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

d_{trans}: transmission delay:

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

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

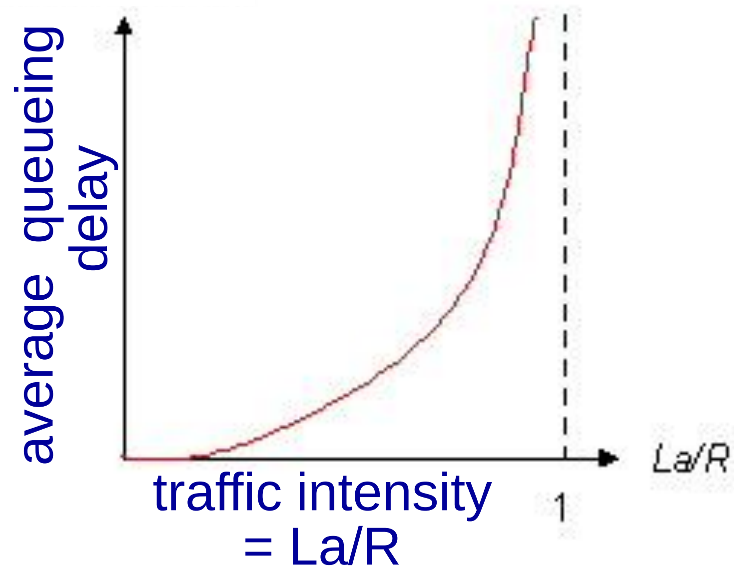
d_{prop}: propagation delay:

- *d*: length of physical link
- *s*: propagation speed in medium (~2x10^8 m/sec)
- $d_{prop} = d/s$

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

Queueing delay (revisited)

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



- *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!

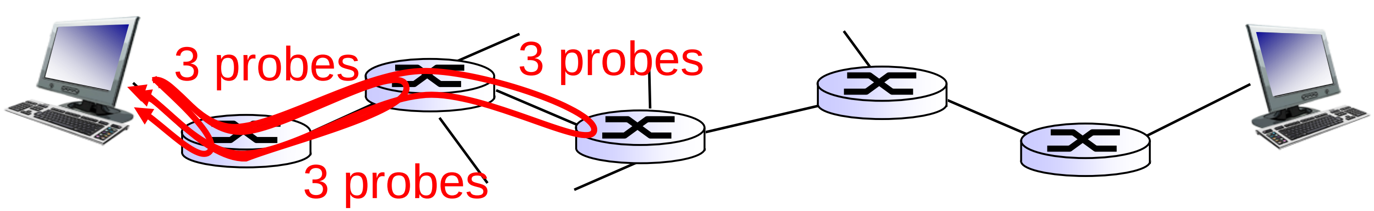

La/R ~ 0


La/R -> 1

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

"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

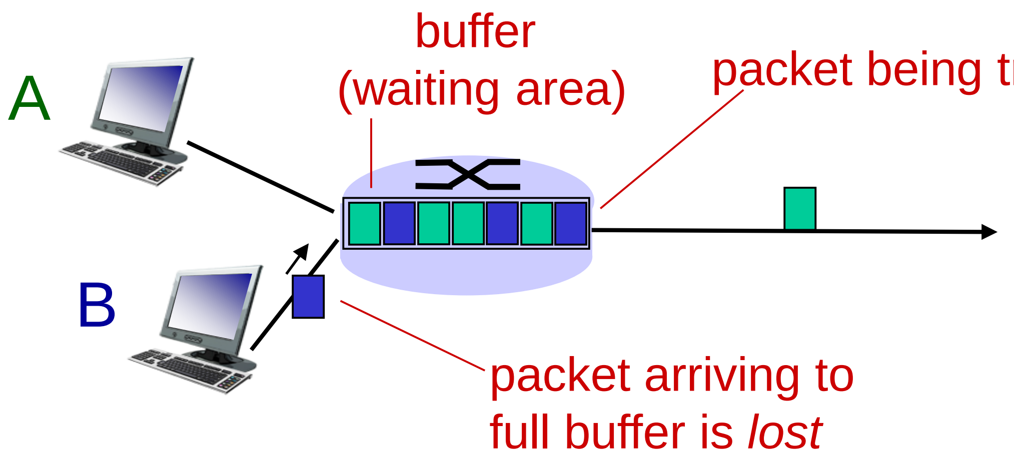
trans-oceanic link

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

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

Packet loss

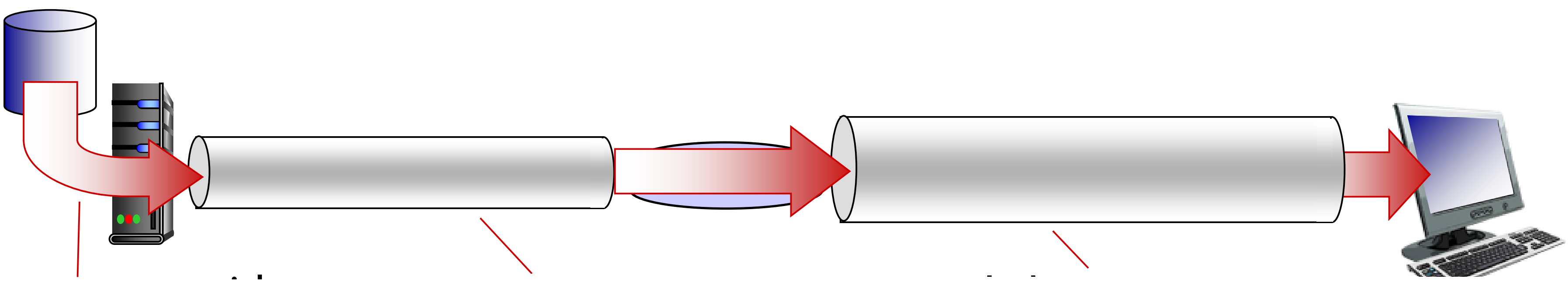
- queue (aka buffer) preceding link in buffer has **finite** capacity
- packet arriving to **full** queue dropped (aka lost)
- lost packet may be **retransmitted** by previous node, by source end system, or not at all



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

Throughput 單位時間內的有效傳輸量

- *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



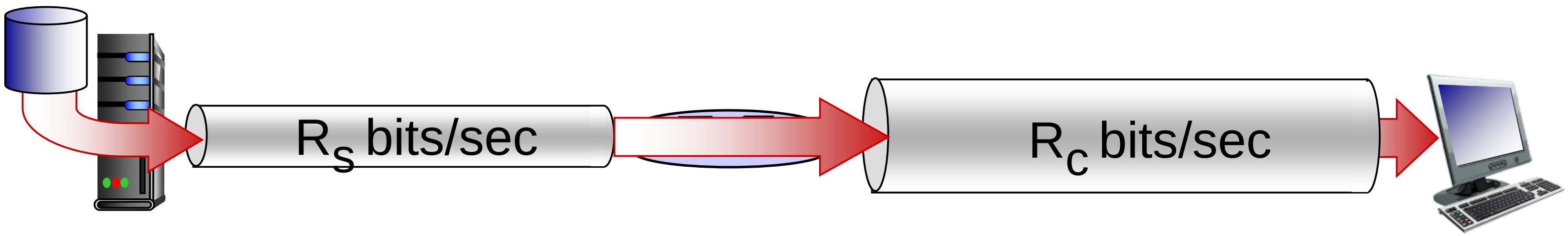
server sends bits (fluid) into pipe

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

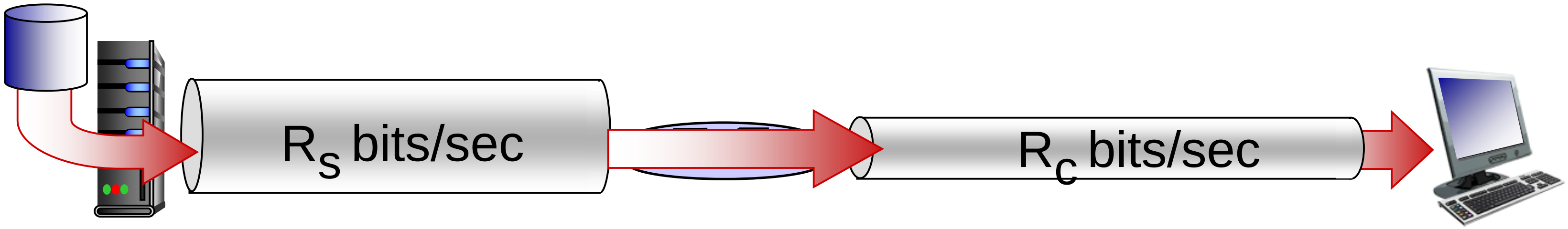
pipe that can carry fluid at rate R_c bits/sec)

Throughput (more)

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



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

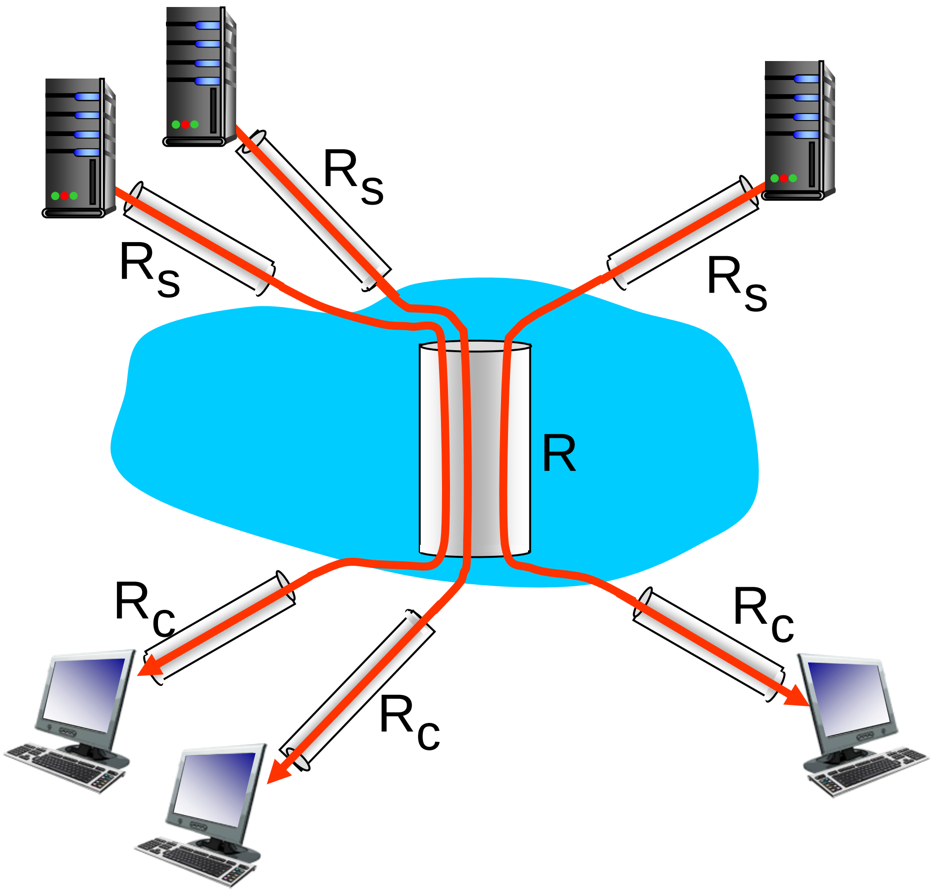


bottleneck link 造成delay最大的那條路徑

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



Client-server paradigm

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

- packet switching, circuit switching, network structure

1.4 delay, loss, throughput in networks

1.5 protocol layers, service models

1.6 networks under attack: security

1.7 history

Protocol Reference Model

Protocol Reference Model – why need it?

- Necessary communication functions are ***complex*** in modern communication systems (networks)
  - e.g., *addressing, routing, error handling, congestion control, access control or scheduling,* and application specific requirements
- *Layered* approach has been widely adopted for ***organizing*** communication functions.

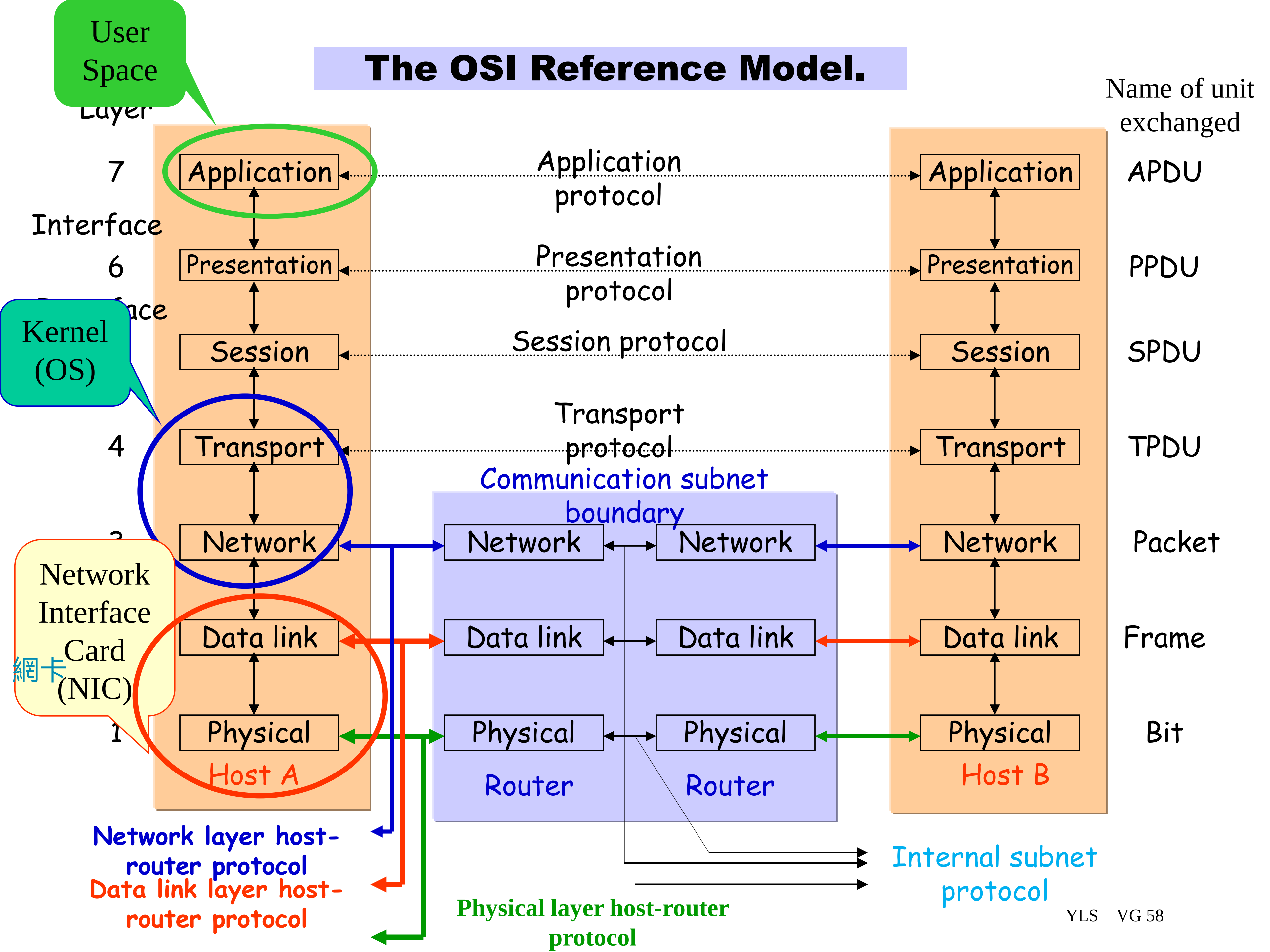
Protocol Reference Model – why need it? (cont'd)

- A *protocol reference model* (PRM) describes the *functions* of the layers and the *relations* of the layers with respect to each other.
- Each type of networks may have its own protocol reference model, e.g.,
  - ISO/OSI Seven Layer PRM
  - **Internet TCP/IP protocol suite**
  - Wireless network
  - Cable network

ISO/OSI Seven-Layer Protocol Reference Model

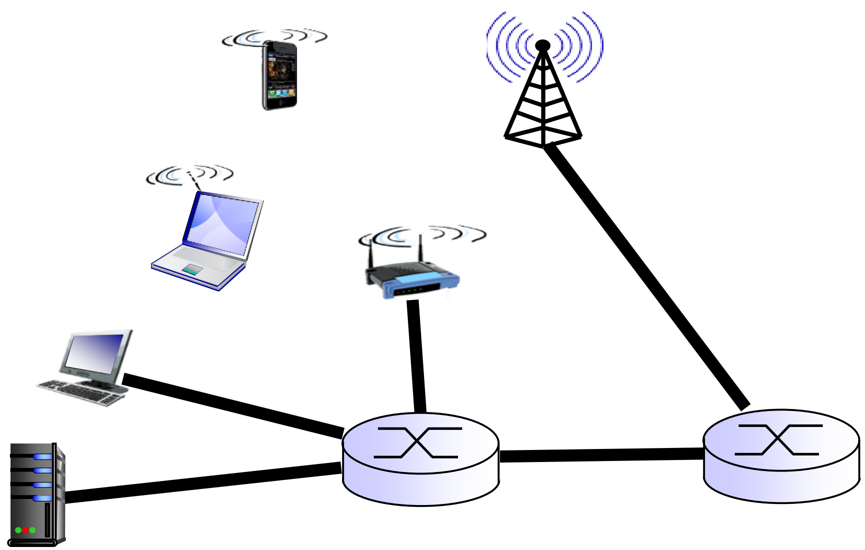
- Physical layer
- Data Link layer
- Network layer
- Transport layer
- Session layer
- Presentation layer
- Application layer

The OSI Reference Model.



Physical Layer

- To transmit raw bits over a communication channel
- Design issues
  - mechanical, electrical, and procedural interfaces and physical transmission medium



Data Link Layer

- Reliable and efficient transmission of raw bits between two machines
- Design issues
  - error detection and recovery, retransmission, frames in sequence, acknowledgment, etc.; frame boundaries; flow control.

=pocket

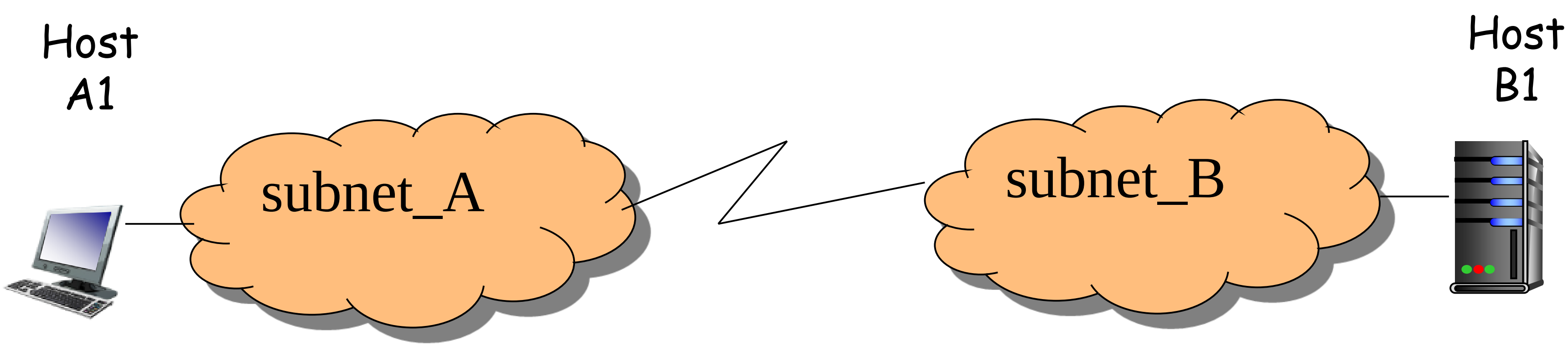
flow control 收、送入速度掌控

Network Node (router)

Network Node (router)

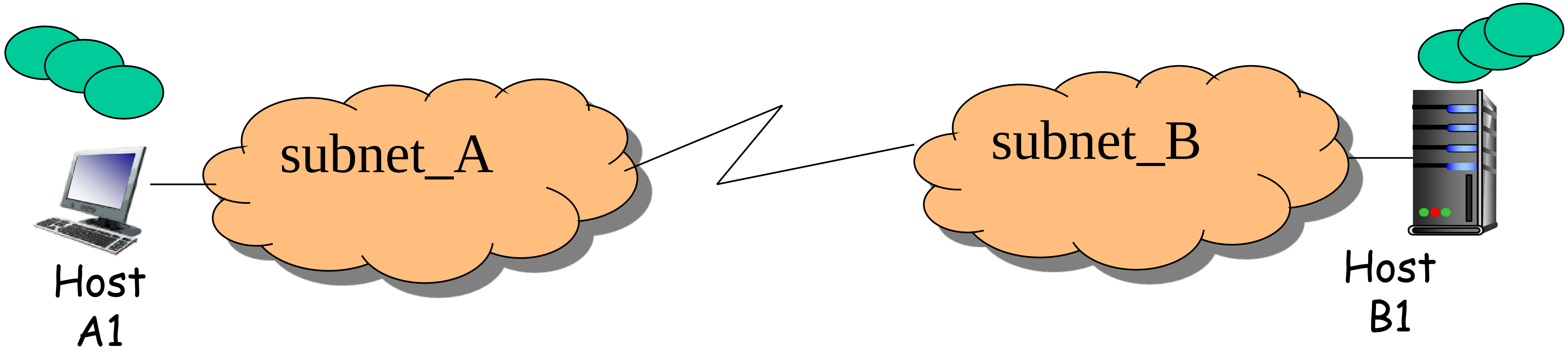
Network Layer

- To control the operation of the subnet/interconnection of networks
- Host-to-host
- Design issues
  - routing, addressing, congestion control, accounting



Transport Layer
TCP

- To control and manage messages exchange between communicating processes on machines
- Process-to-process
- Design issues
  - reliability, connecting services, efficiency, naming, flow control
  port



Session Layer

- To allow users on different machines to establish sessions between them.
- Design issues
  - dialogue control, token management, synchronization

SIP - Session Initiation Protocol
RTSP - Real-time Streaming Protocol
RSVP - Resource reSerVation Protocol
將來有可能用（e.g.遠端醫療）

…

Presentation Layer

- To manage the **syntax** and **semantics** (i.e. representation) of the information transmitted
- Design issues
  - abstract data types, encoding/decoding schemes, data compression, data encryption (security)
  - e.g., ASN.1 (Abstract Syntax Notation One) and BER (Basic Encoding Rule), XML, etc.

```
Contact ::= SEQUENCE {
    name VisibleString,
    phone NumericString
}
```

plore OSS'
SN.1 IO

So, a contact named John Smith with a phone number of (987) 654-3210 might be converted to 0x3018800A4A6F686E20536D697468810A39383736353433323130 using the ASN.1 Basic Encoding Rules or 0x0A4A6F686E20536D6974680AA987654321 using the Packed Encoding Rules or <?xml version="1.0" encoding="UTF-8"?> <name>John Smith</name> <phone>9876543210</phone> </Contact> if the XML Encoding Rules are

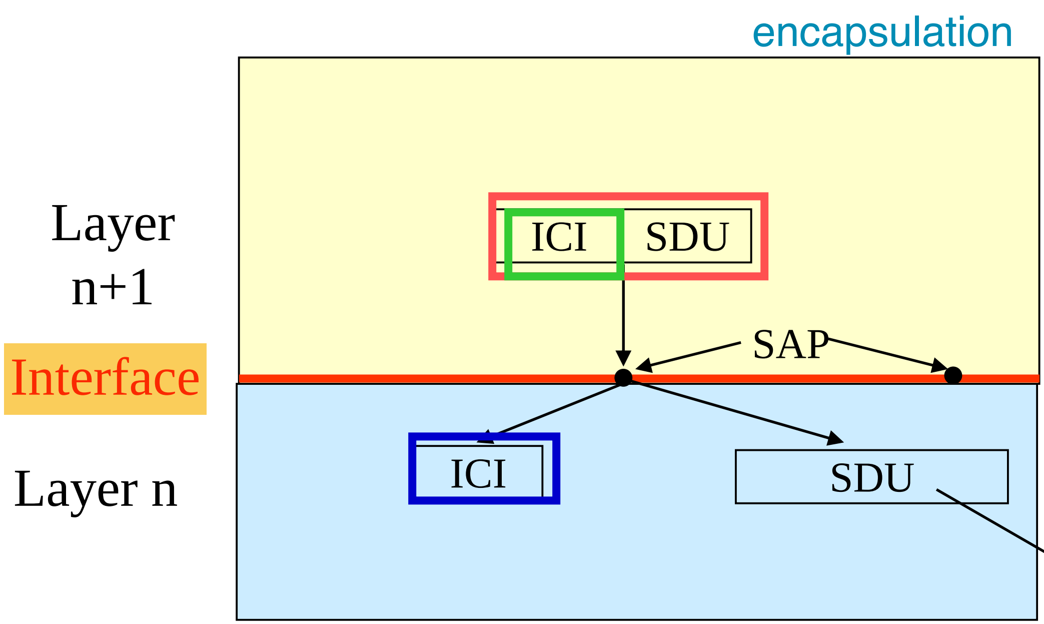
Type, Value

Application Layer

- To support *application specific* functions for information transfer
  - It contains a variety of protocols, e.g., FTAM, RDA, CMISE, telnet, ftp, e-mail, etc.

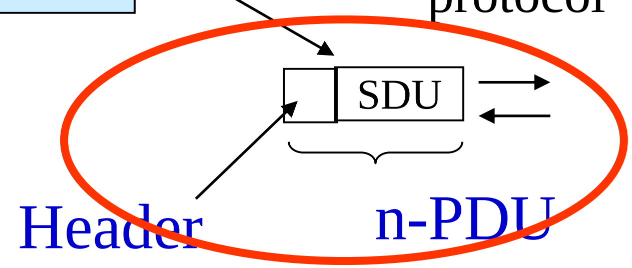
Relation between Layers at an Interface

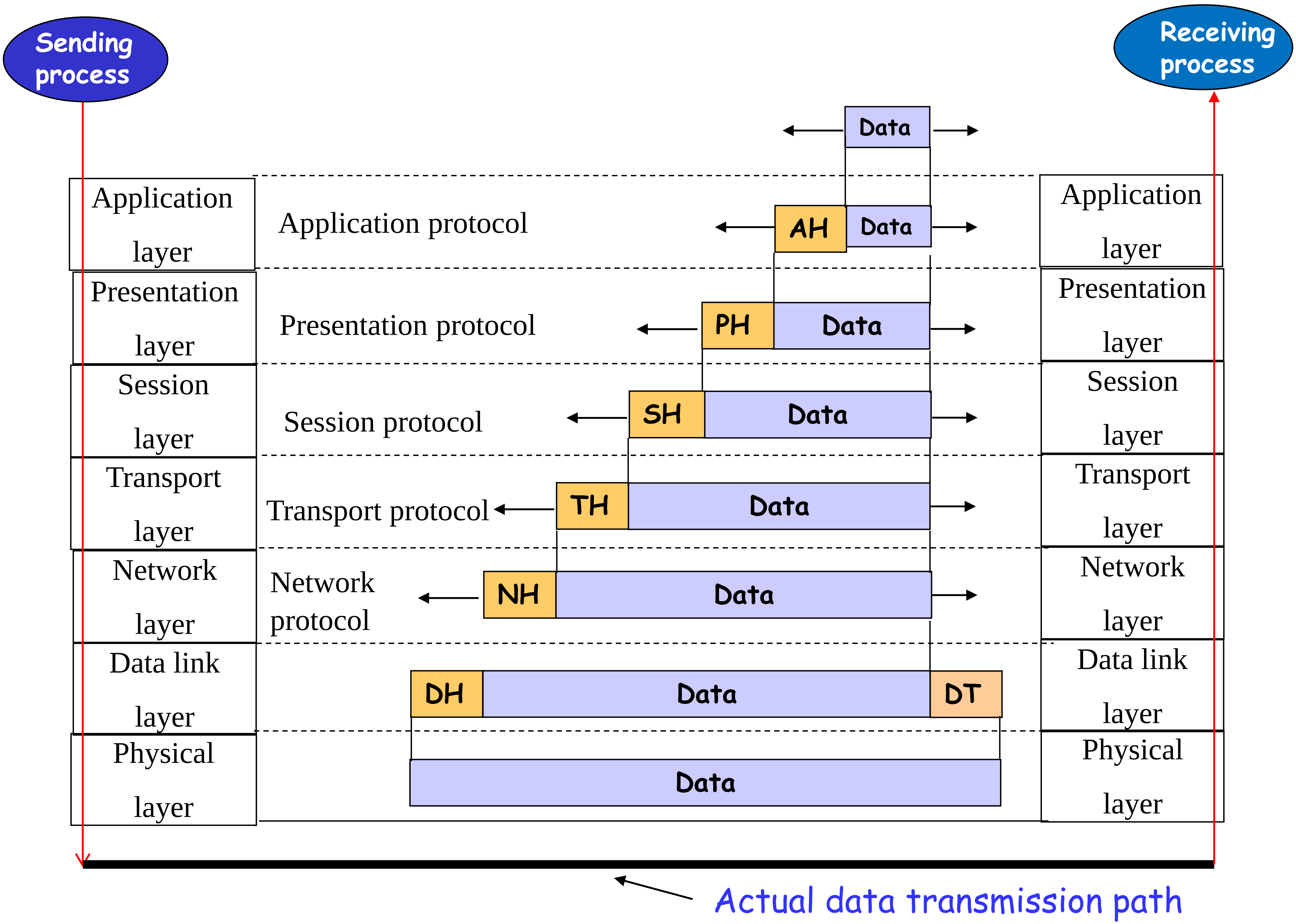
API
(Application Programming Interface)



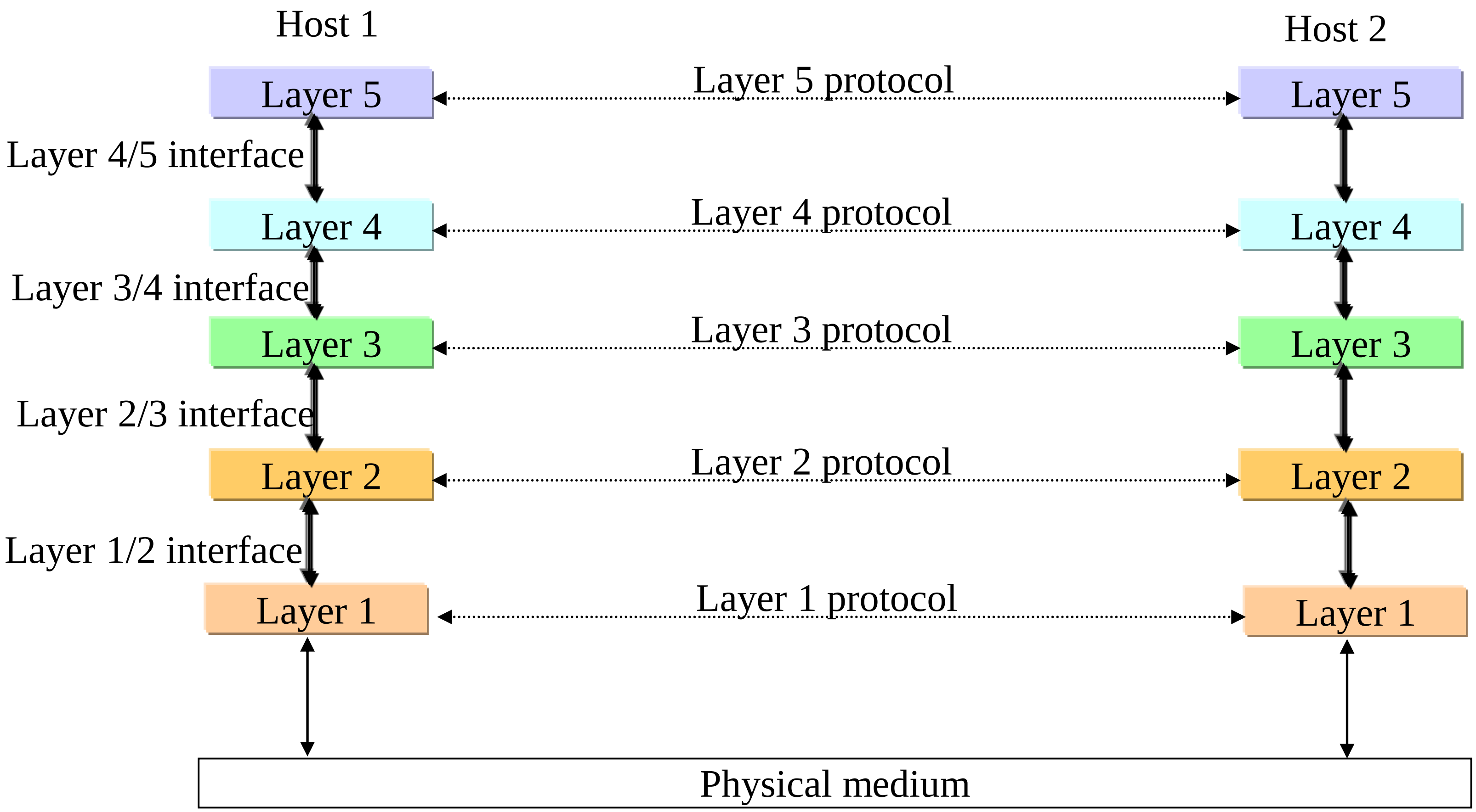
SAP =Service Access Point
SDU = Service Data Unit
PDU = Protocol Data Unit
ICI = Interface Control Information (header)

Layer n entities exchange n-**PDU**s in their layer n protocol

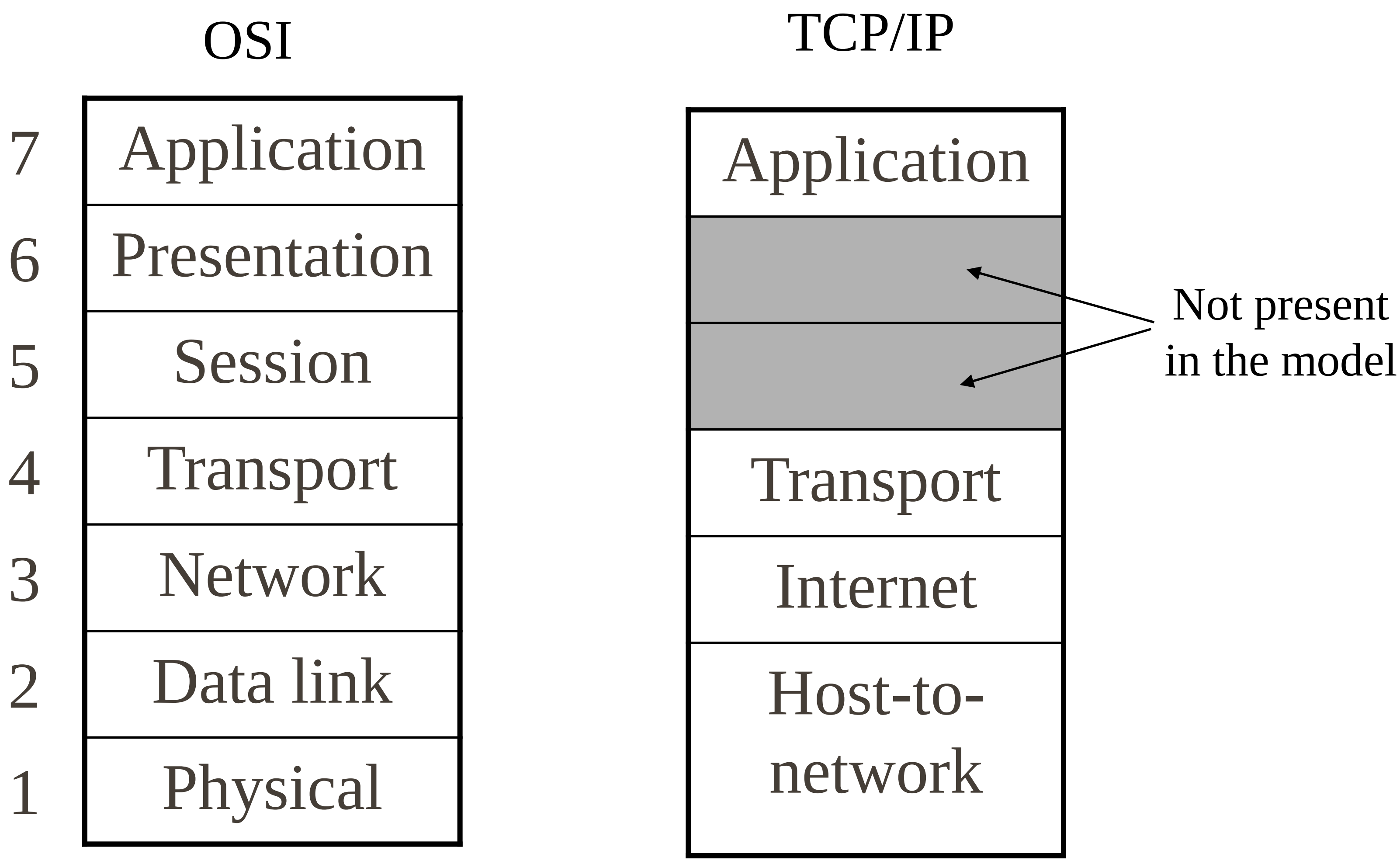



Sending process
Receiving process
Application layer
Presentation layer
Session layer
Transport layer
Network layer
Data link layer
Physical layer
Application protocol
Presentation protocol
Session protocol
Transport protocol
Network protocol
Data
AH
PH
SH
TH
NH
DH
DT
Actual data transmission path

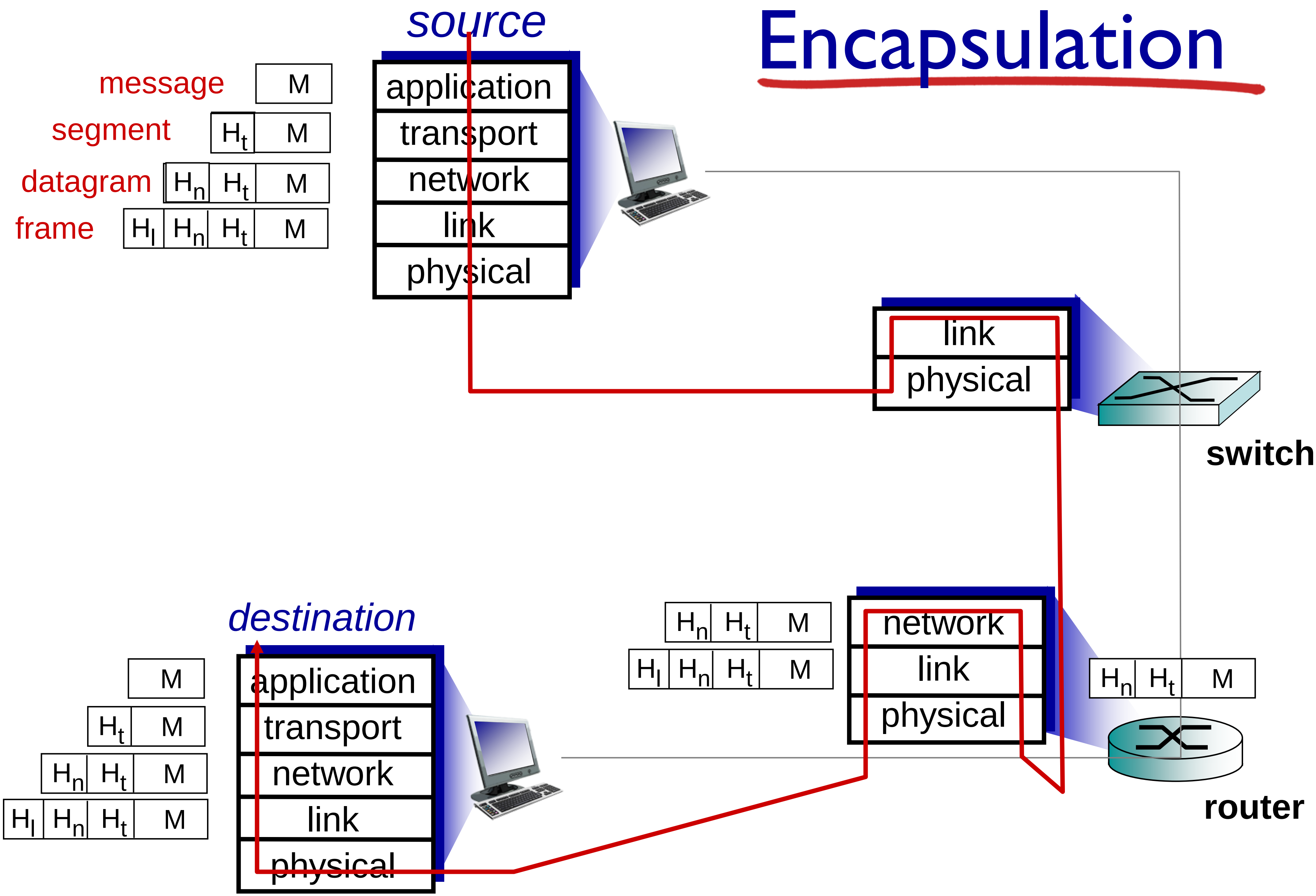
Layers, Protocol, and Interfaces



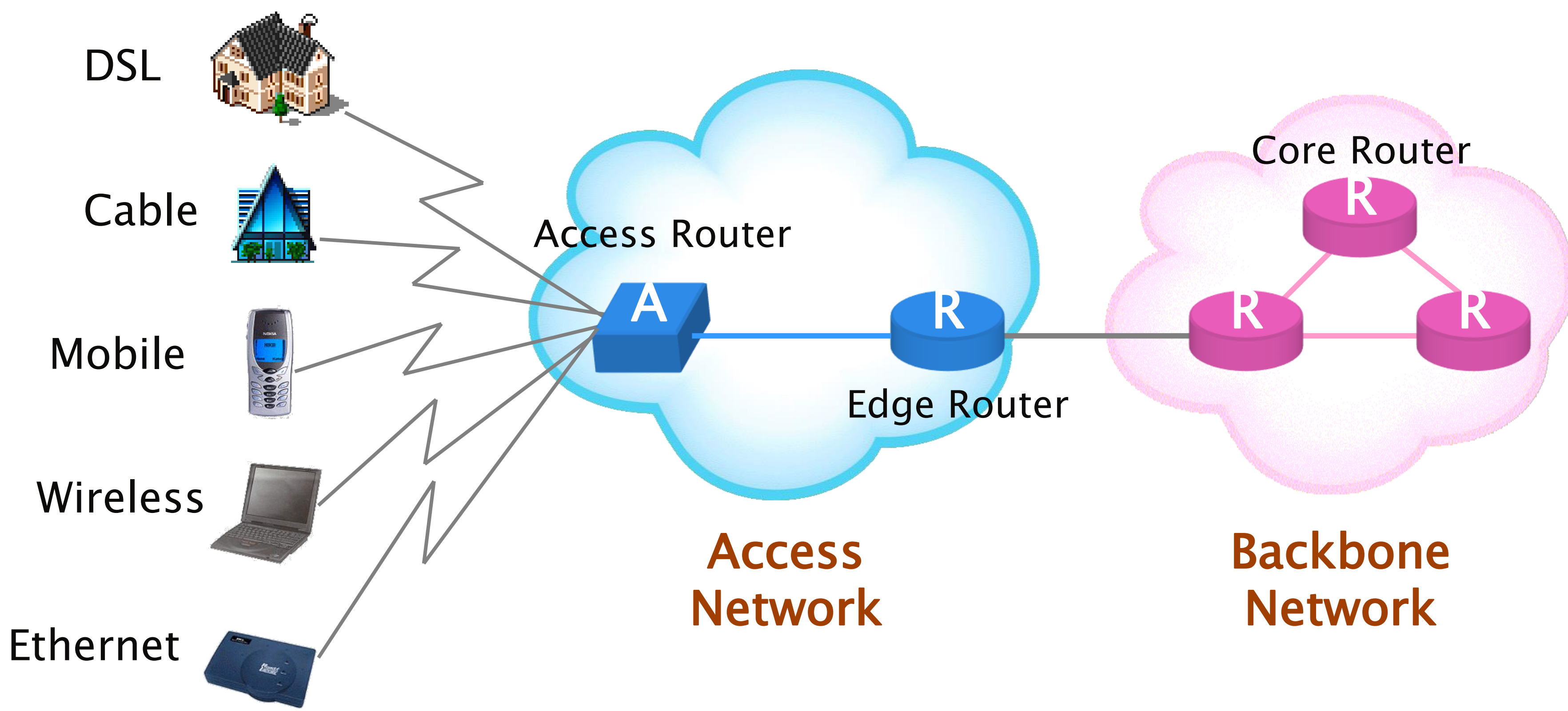
The TCP/IP reference model



Encapsulation



Access of the Future



Network Standards

- Scope
  - architecture, services, interfaces, protocols, etc.
- Why need standards
  - To achieve compatibility and interoperability between networking systems
    - proprietary, isolated subnetworks
  - To call for ***Open System Interconnection*** or ***Open Networking***

Two kinds of standards

- "*de facto*" standards
  - specifications that have happened without any formal plan.
  - examples: Internet protocols, IBM PC specifications, UNIX operating system
- "*de jure*" standards
  - formal, legal standards adopted by some authorized standardization body
  - examples: International Telecommunication Union (ITU) (formal CCITT), ISO, IEEE, ANSI, POSIX, etc.

Chapter 1: roadmap

1.1 what *is* the Internet?

1.2 network edge

- end systems, access networks, links

1.3 network core

- packet switching, circuit switching, network structure

1.4 delay, loss, throughput in networks

1.5 protocol layers, service models

1.6 networks under attack: security

1.7 history

Network security

- field of network security:
  - how bad guys can **attack computer networks**
  - how we can **defend** networks against attacks
  - **how to design architectures that are immune to attacks**
- Internet not originally designed with (much) security in mind
  - *original vision:* "a group of *mutually trusting* users attached to a transparent network" ☺
  - Internet protocol designers playing "catch-up"
  - security considerations in all layers!

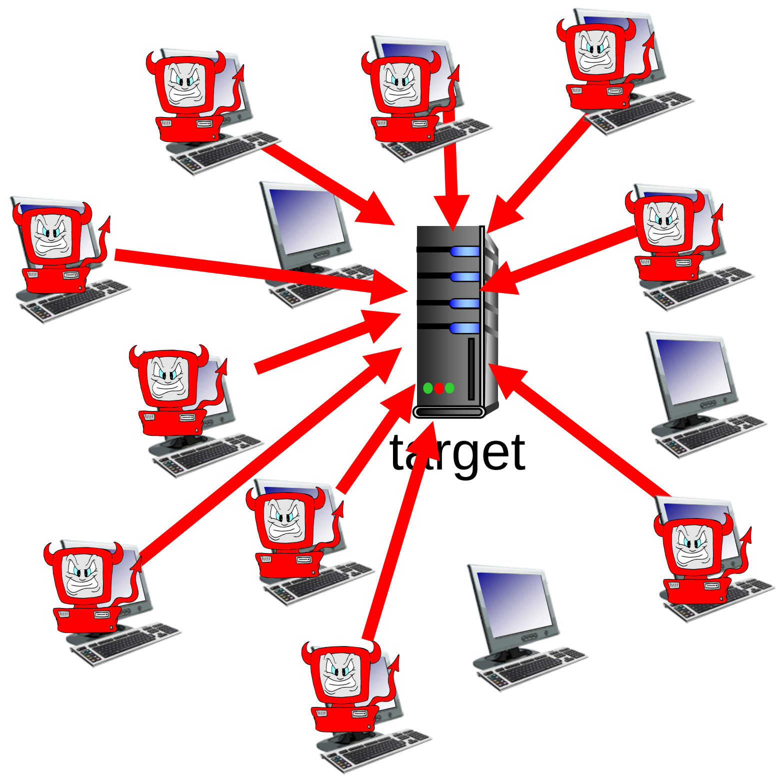
Bad guys: put malware into hosts via Internet

- malware can get in host from:
  - *virus:* self-replicating infection by receiving/executing object (e.g., e-mail attachment)
  - *worm:* self-replicating infection by passively receiving object that gets itself executed會擴散的感染疾病
- 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

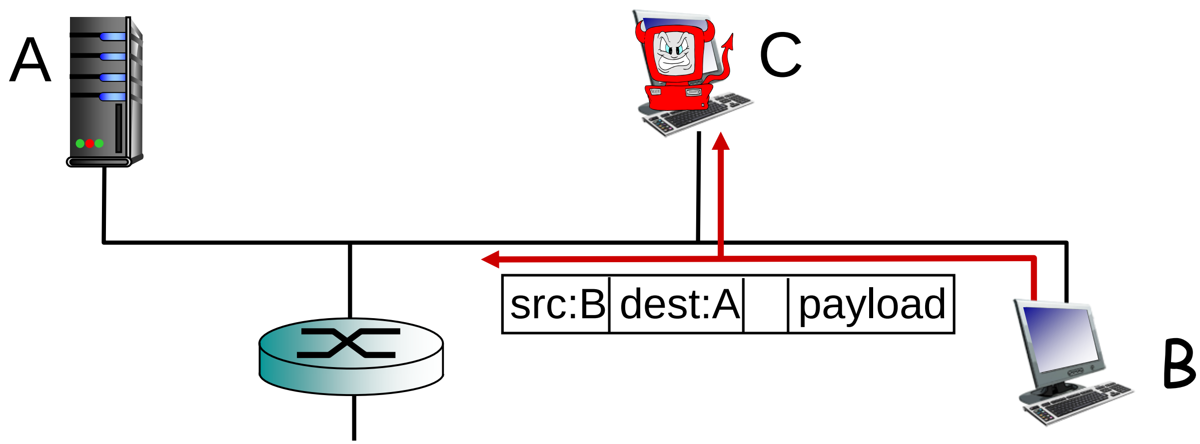
1. select target
2. break into hosts around the network (see botnet)
3. send packets to target from compromised hosts



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

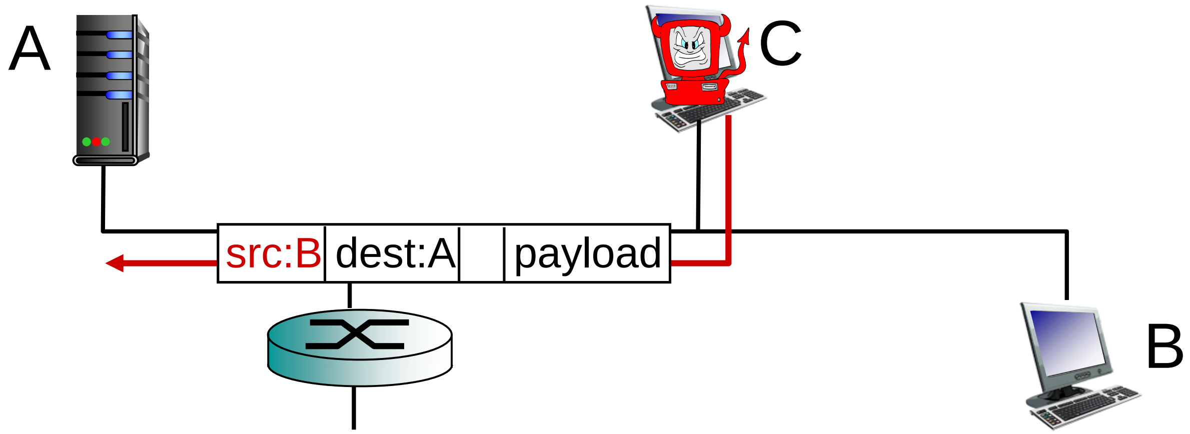


- wireshark software used for end-of-chapter labs is a (free) packet-sniffer

Bad guys can use fake addresses

IP spoofing: send packet with false source address

假冒



… lots more on security (throughout, Chapter 8)

Chapter 1: roadmap

1.1 what *is* the Internet?

1.2 network edge

- end systems, access networks, links

1.3 network core

- packet switching, circuit switching, network structure

1.4 delay, loss, throughput in networks

1.5 protocol layers, service models

1.6 networks under attack: security

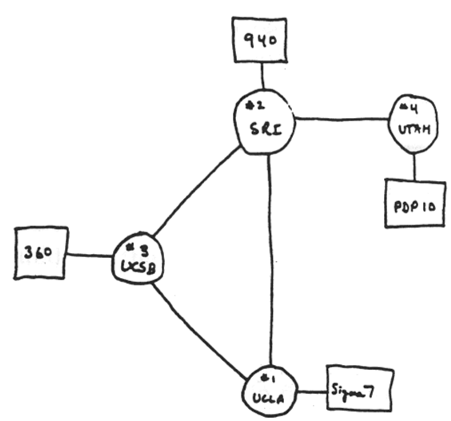
1.7 history

Internet history

1961-1972: Early packet-switching principles

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

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



THE ARPA NETWORK

Internet history

1972-1980: Internetworking, new and proprietary nets

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

Cerf and Kahn's internetworking principles:

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

define today's Internet architecture

Internet history

1980-1990: new protocols, a proliferation of networks

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

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

Internet history

1990, 2000's: commercialization, the Web, new apps

- early 1990's: **ARPAnet decommissioned**
- 1991: NSF lifts restrictions on commercial use of **NSFnet** (decommissioned, 1995)
- early 1990s: Web
  - hypertext [Bush 1945, Nelson 1960's]
  - HTML, HTTP: Berners-Lee
  - 1994: Mosaic, later Netscape
  - late 1990's: commercialization of the Web

late 1990's – 2000's:

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

Internet history

2005-present

- ~750 million hosts
  - Smartphones and tablets
- Aggressive deployment of broadband access
- Increasing ubiquity of high-speed wireless access (5G)
- Emergence of online social networks, IoT:
  - Facebook: soon one billion users
- Service providers (Google, Microsoft) create their own networks
  - Bypass Internet, providing "instantaneous" access to search, email, etc.
- E-commerce, universities, enterprises running their services in "cloud" (eg, Amazon EC2), edge computing

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!*