Chapter 1: Introduction

Instructor: Zhuozhao Li

Lab: Wei Wang

Department of Computer Science and Engineering

Chapter 1: introduction

Chapter goal:

- Get "feel," "big picture," introduction to terminology
  - more depth, detail *later* in course
- Approach:
  - use Internet as example

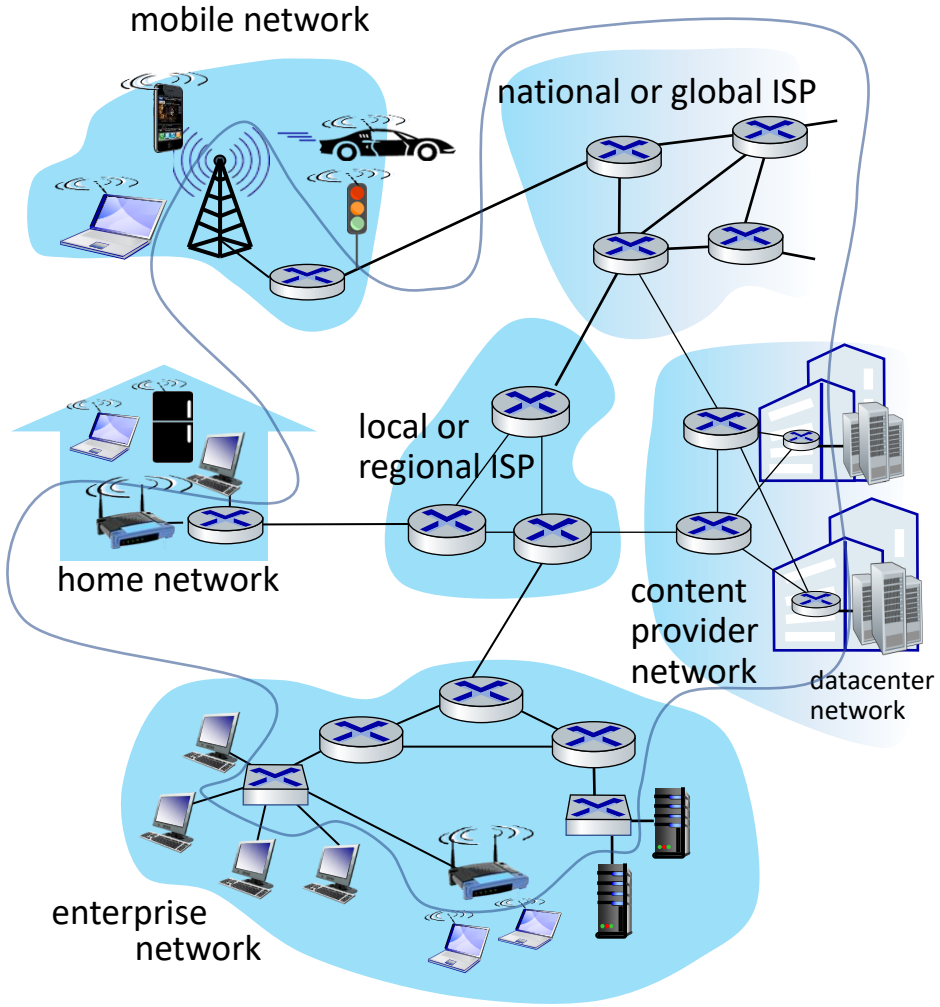
Overview/roadmap:

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

The Internet: a “nuts and bolts” view

Billions of connected computing *devices*:

- *hosts = end systems*
- running *network apps* at Internet’s “edge”



“Fun” Internet-connected devices

Amazon Echo

Internet refrigerator

IP picture frame

Pacemaker & Monitor

Tweet-a-watt: monitor energy use

Security Camera

Slingbox: remote control cable TV

Web-enabled toaster + weather forecaster

AR devices

Internet phones

sensorized, bed mattress

Fitbit

Others?

The Internet: a "nuts and bolts" view

Billions of connected computing *devices*:

- *hosts = end systems*
- running *network apps* at Internet's "edge"

Packet switches: forward packets (chunks of data)

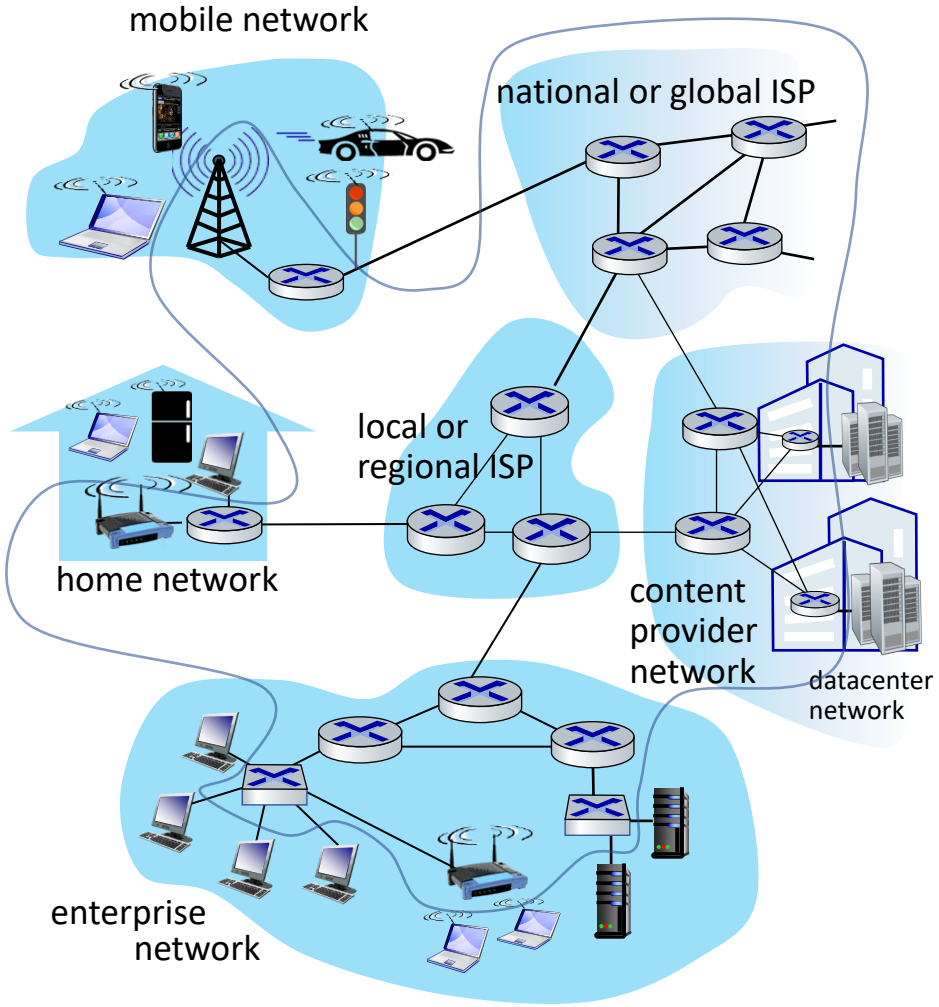
- *routers, switches*

Communication links

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

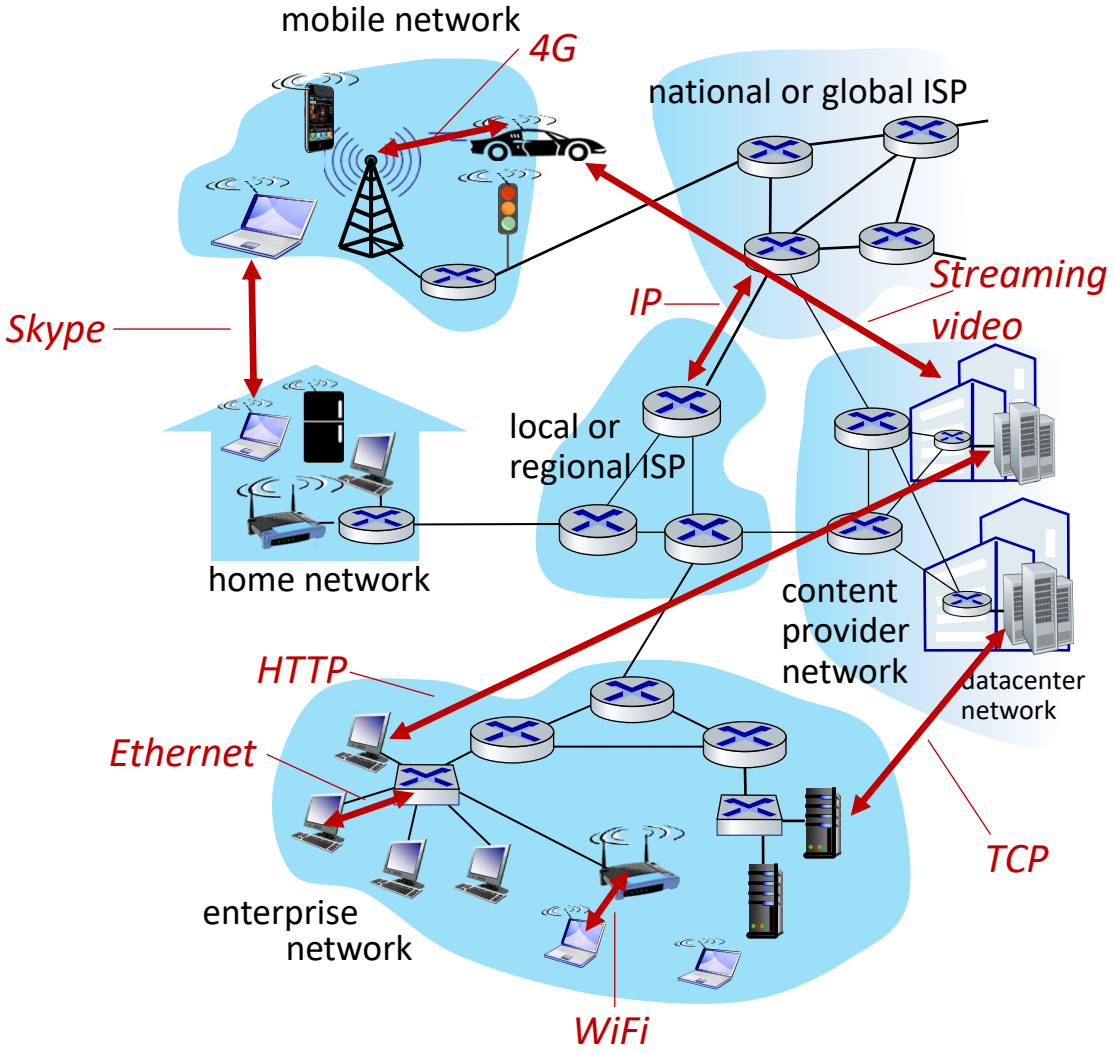
Networks

- collection of devices, routers, links: managed by an organization



The Internet: a "nuts and bolts" view

- *Internet:* "network of networks"
  - Interconnected ISPs
- *protocols* are *everywhere*
  - control sending, receiving of messages
  - e.g., HTTP (Web), streaming video, Skype, TCP, IP, WiFi, 4G, Ethernet
- *Internet standards*
  - RFC: Request for Comments
  - IETF: Internet Engineering Task Force



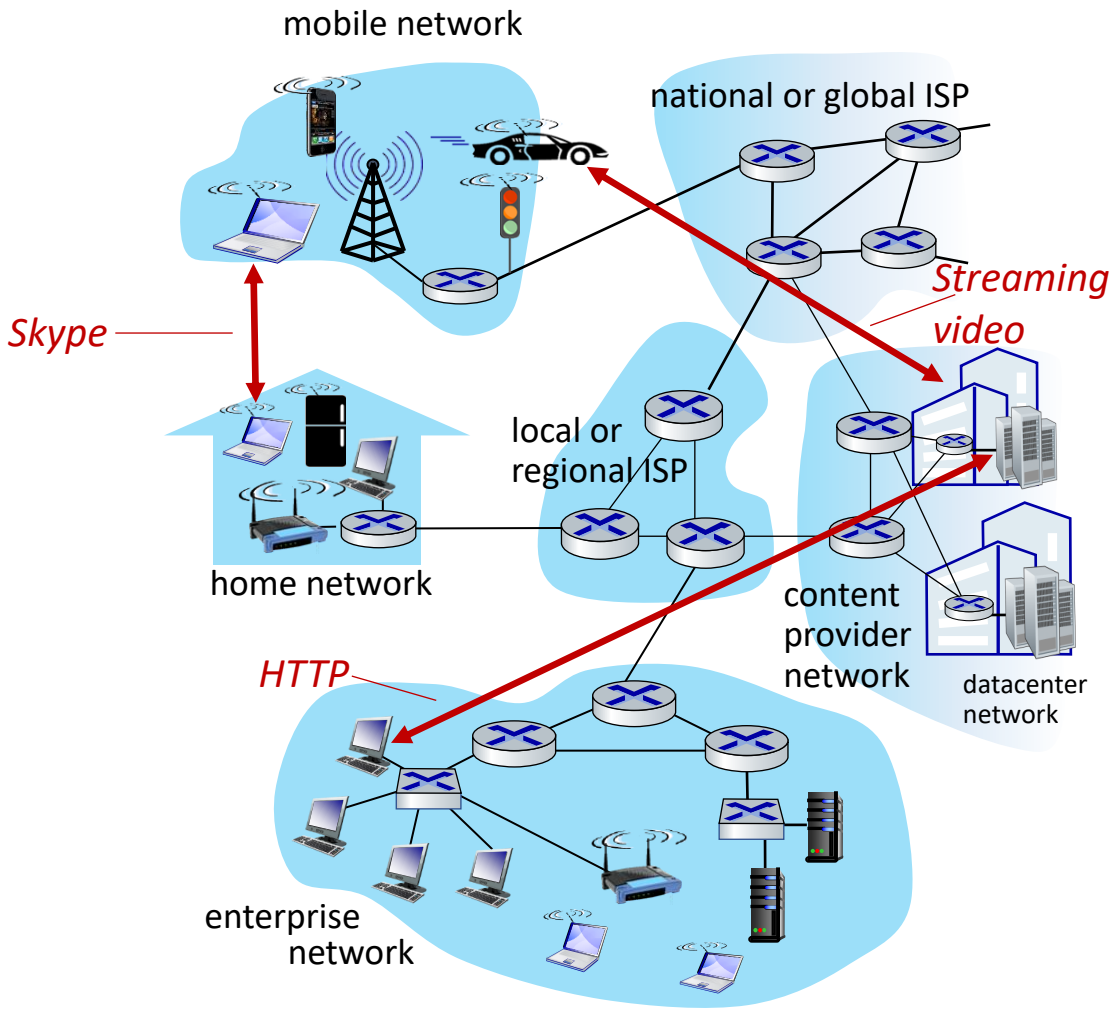
有多个标准组织参与数据通信领域的标准制定

标准组织		在数据通信领域的主要工作
三大官方国际标准组织		
ITU-T	国际电信联盟	电信业务 IP 化、物联网
ISO	国际标准组织	网络互联模型
IEC	国际电工委员会	机械电气接口和互换性
三大民间国际标准组织		
IETF	互联网工程任务组	网络互联协议，标准主导者
IEEE	电气与电子工程师协会	Ethernet、WLAN
3GPP	第三代合作伙伴计划	无线IP
三大区域性标准组织		
CCSA	中国通信标准协会	设备形态、接口、标准支持
ETSI	欧洲电信标准化协会	宽带接入、终端
ANSI	美国国家标准局	美国标准审批

Source: Huawei

The Internet: a "service" view

- *Infrastructure* that provides services to applications:
  - Web, streaming video, multimedia teleconferencing, email, games, e-commerce, social media, inter-connected appliances, …
- provides *programming interface* to distributed applications:
  - "hooks" allowing sending/receiving apps to "connect" to, use Internet transport service
  - provides service options, analogous to postal service



What's a protocol?

Human protocols:

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

… specific messages sent

… specific actions taken when message received, or other events

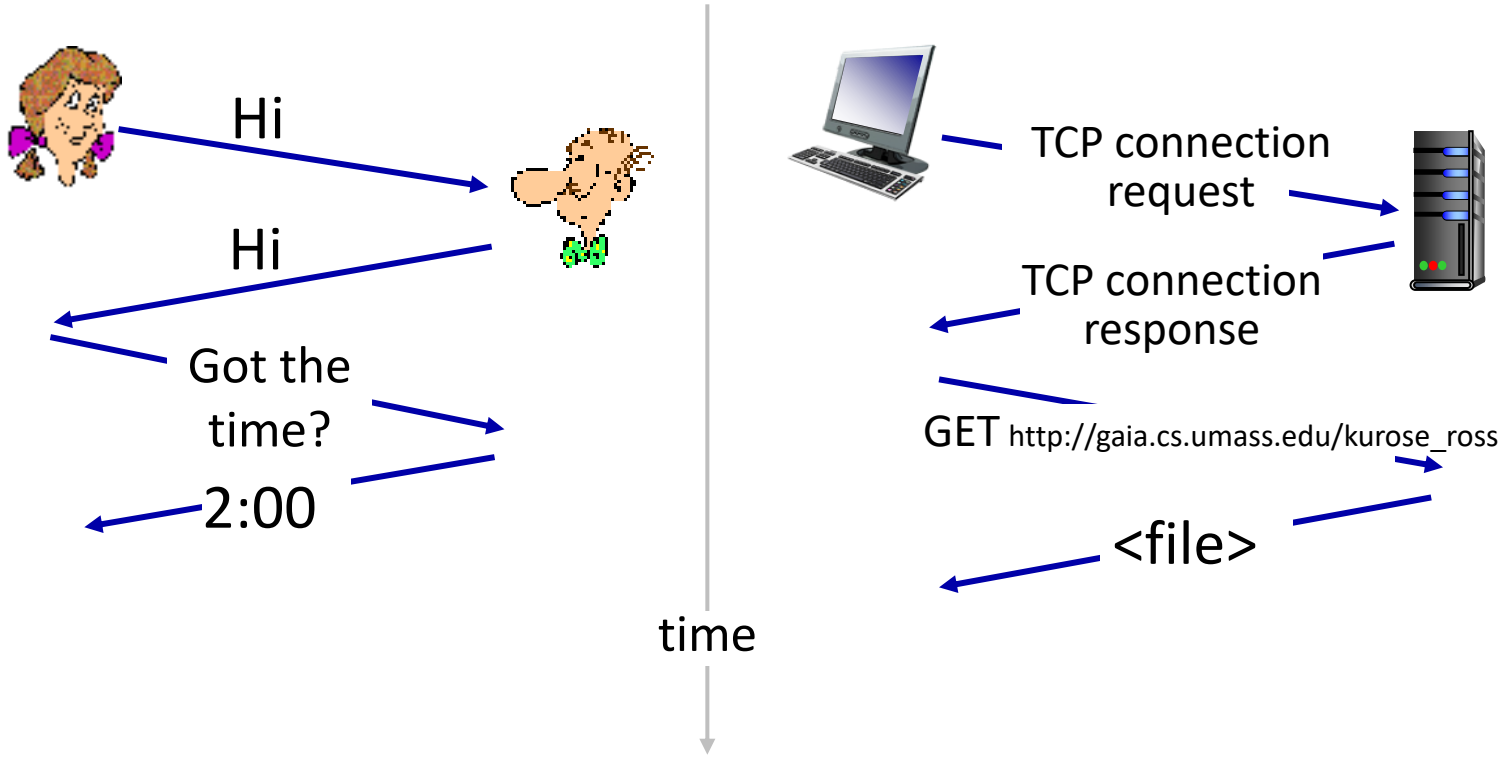
Network protocols:

- computers (devices) rather than humans
- all communication activity in Internet governed by protocols

Protocols define the format, order of messages sent and received among network entities, and actions taken on msg transmission, receipt

What's a protocol?

A human protocol and a computer network protocol:



Q: other human protocols?

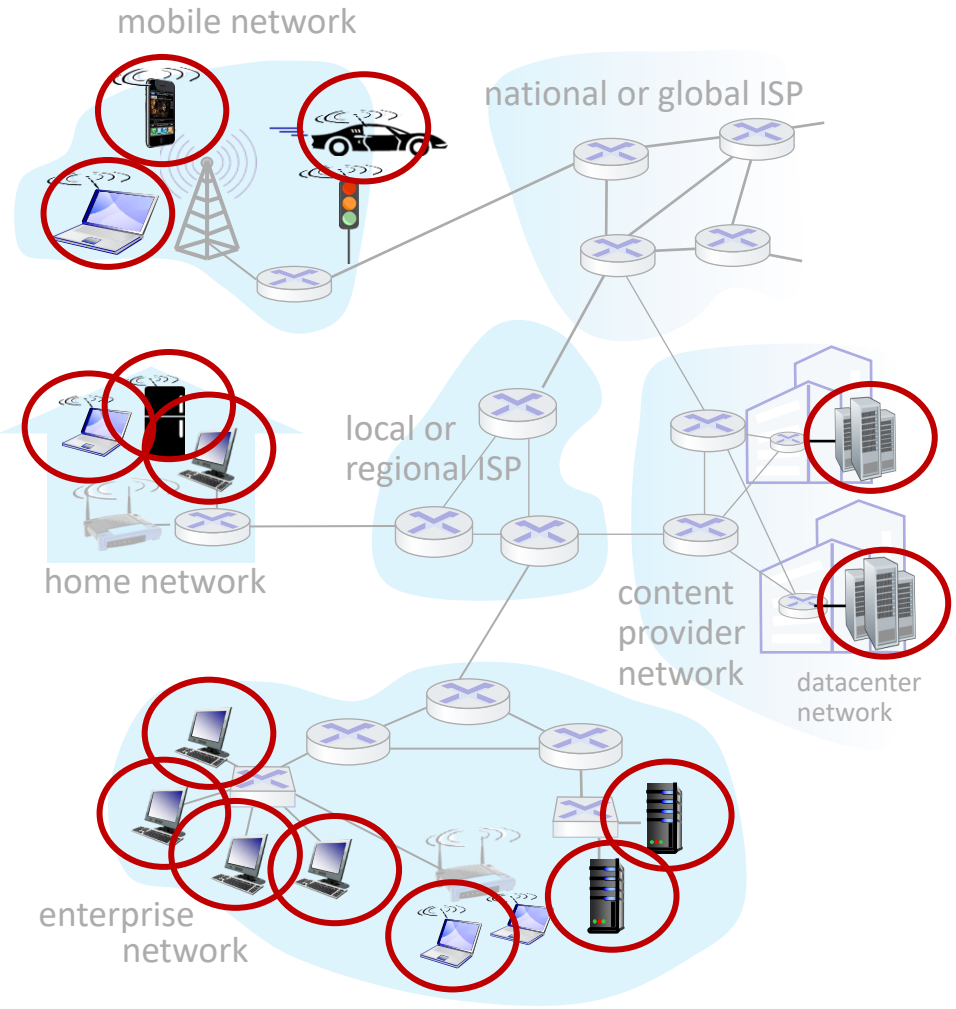
Chapter 1: roadmap

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

A closer look at Internet structure

Network edge:

- hosts: clients and servers
- servers often in data centers



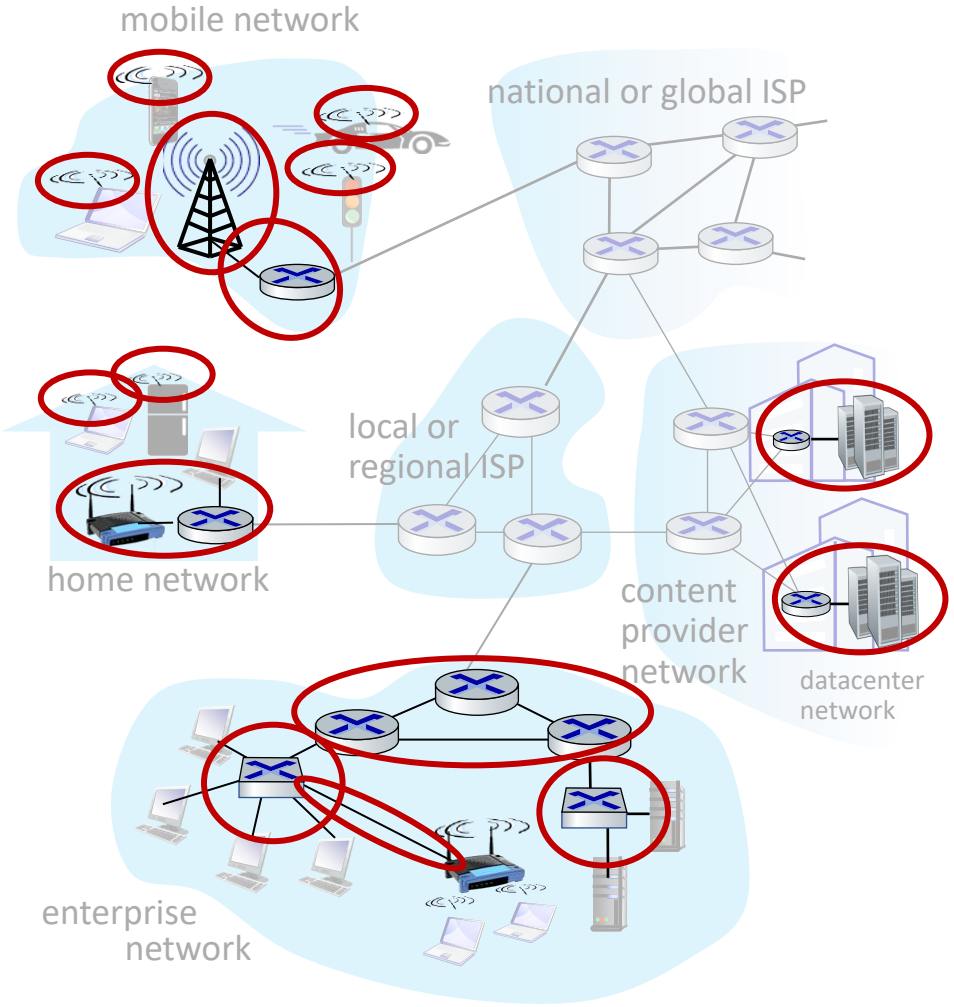
A closer look at Internet structure

Network edge:

- hosts: clients and servers
- servers often in data centers

Access networks, physical media:

- wired, wireless communication links



A closer look at Internet structure

Network edge:

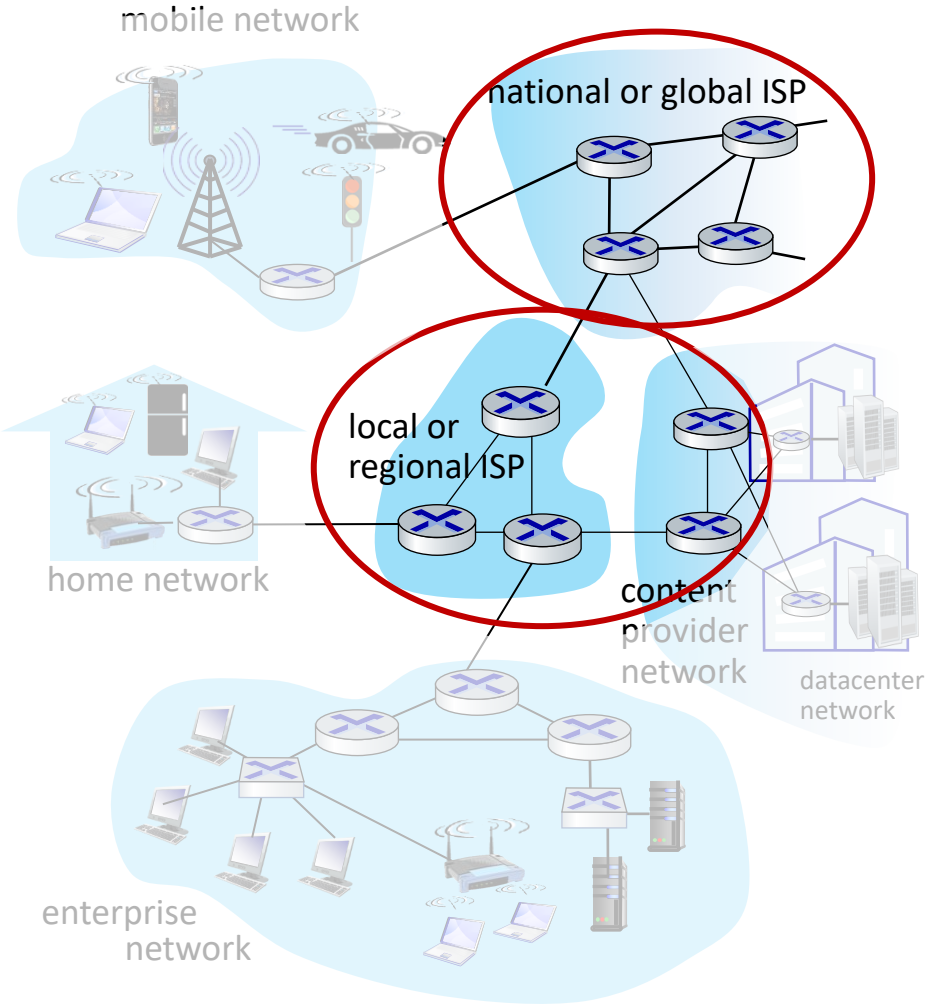
- hosts: clients and servers
- servers often in data centers

Access networks, physical media:

- wired, wireless communication links

Network core:

- interconnected routers
- network of networks



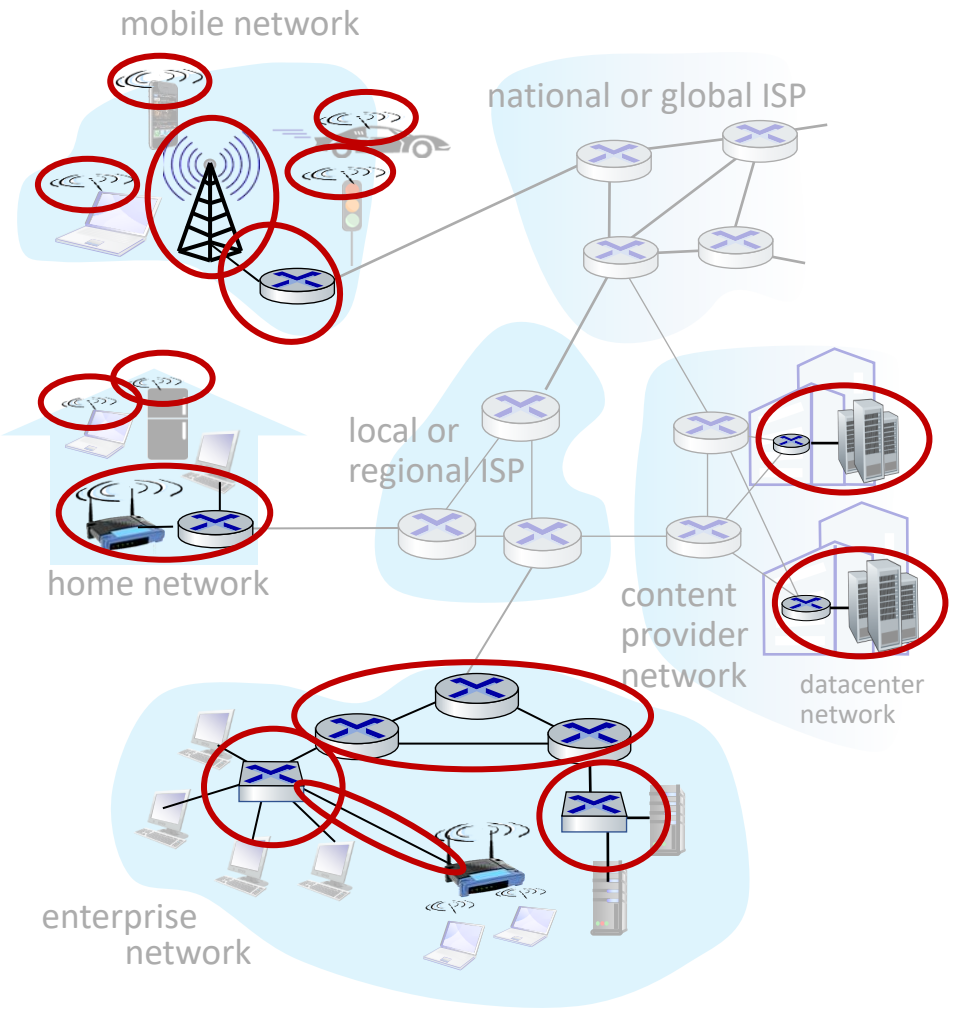
Access networks and physical media

Q: How to connect end systems to edge router?

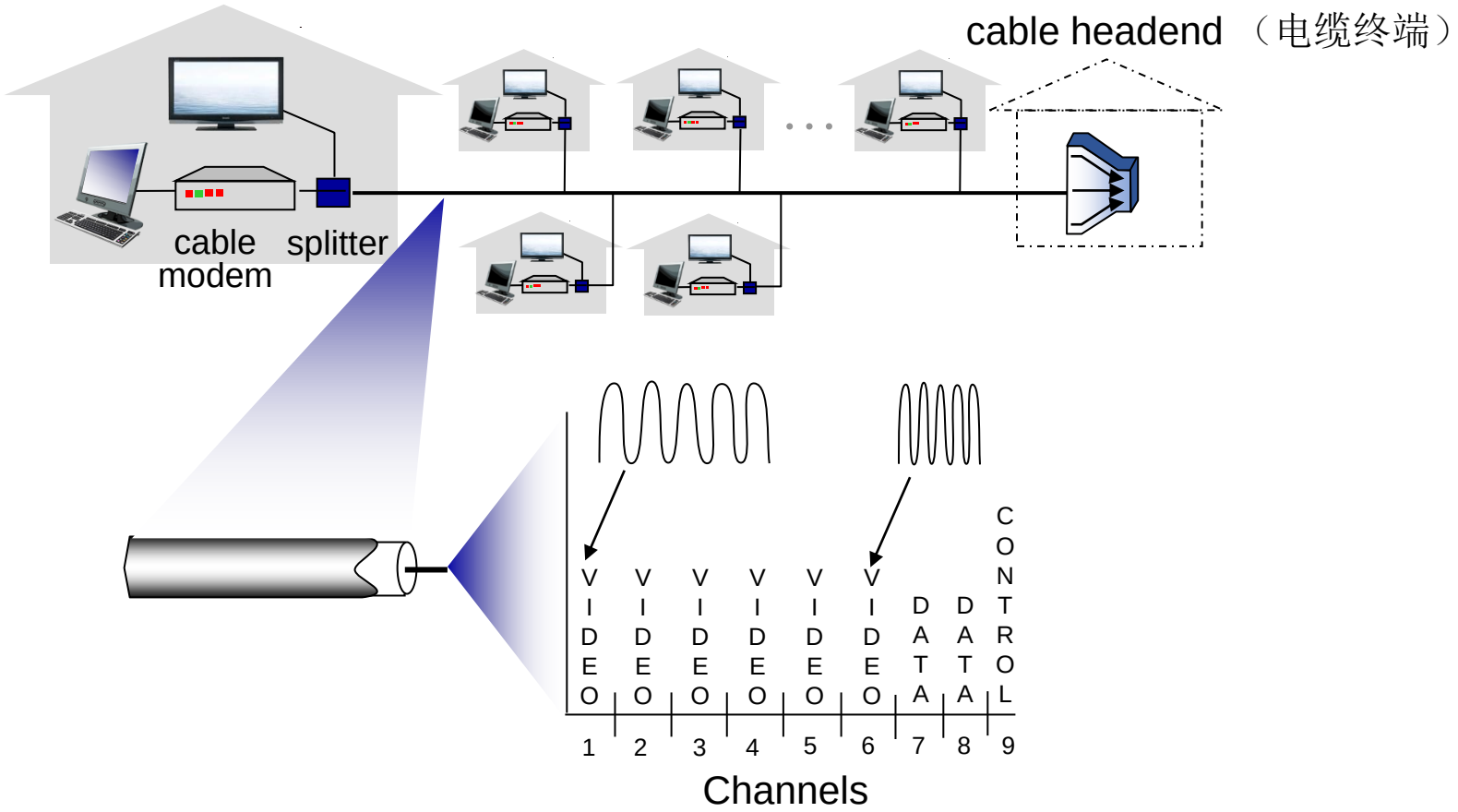
- residential access nets
- institutional access networks (school, company)
- mobile access networks (WiFi, 4G/5G)

What to look for:

- transmission rate (bits per second) of access network?
- shared or dedicated access among users?

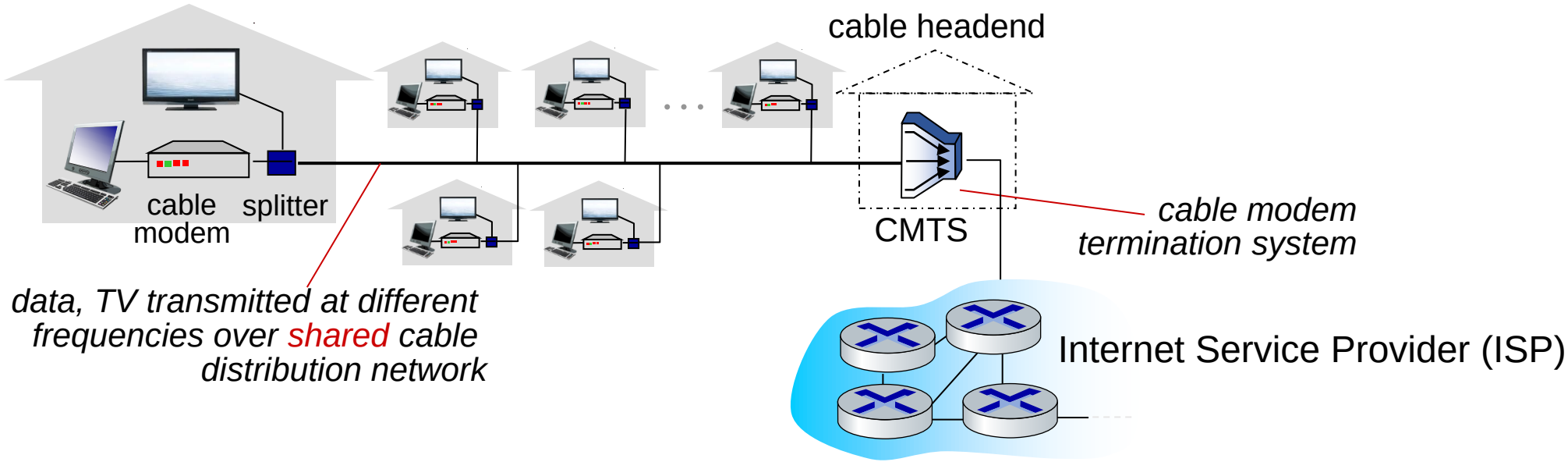


Access networks: cable-based access



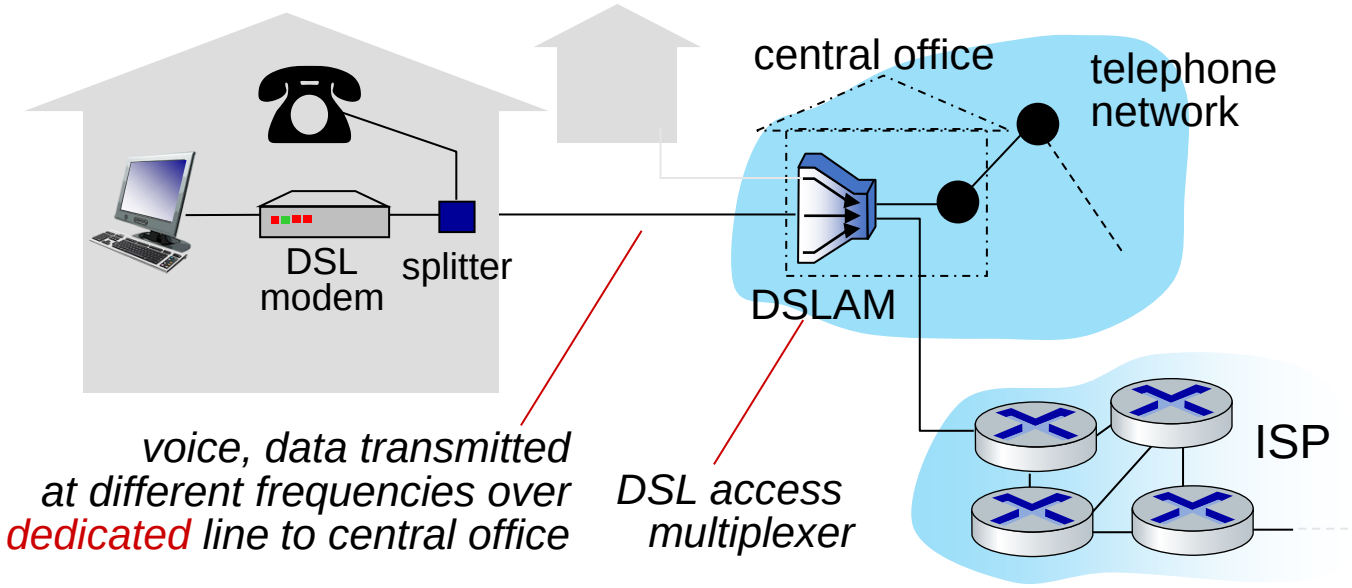
frequency division multiplexing (FDM): different channels transmitted in different frequency bands

Access networks: cable-based access



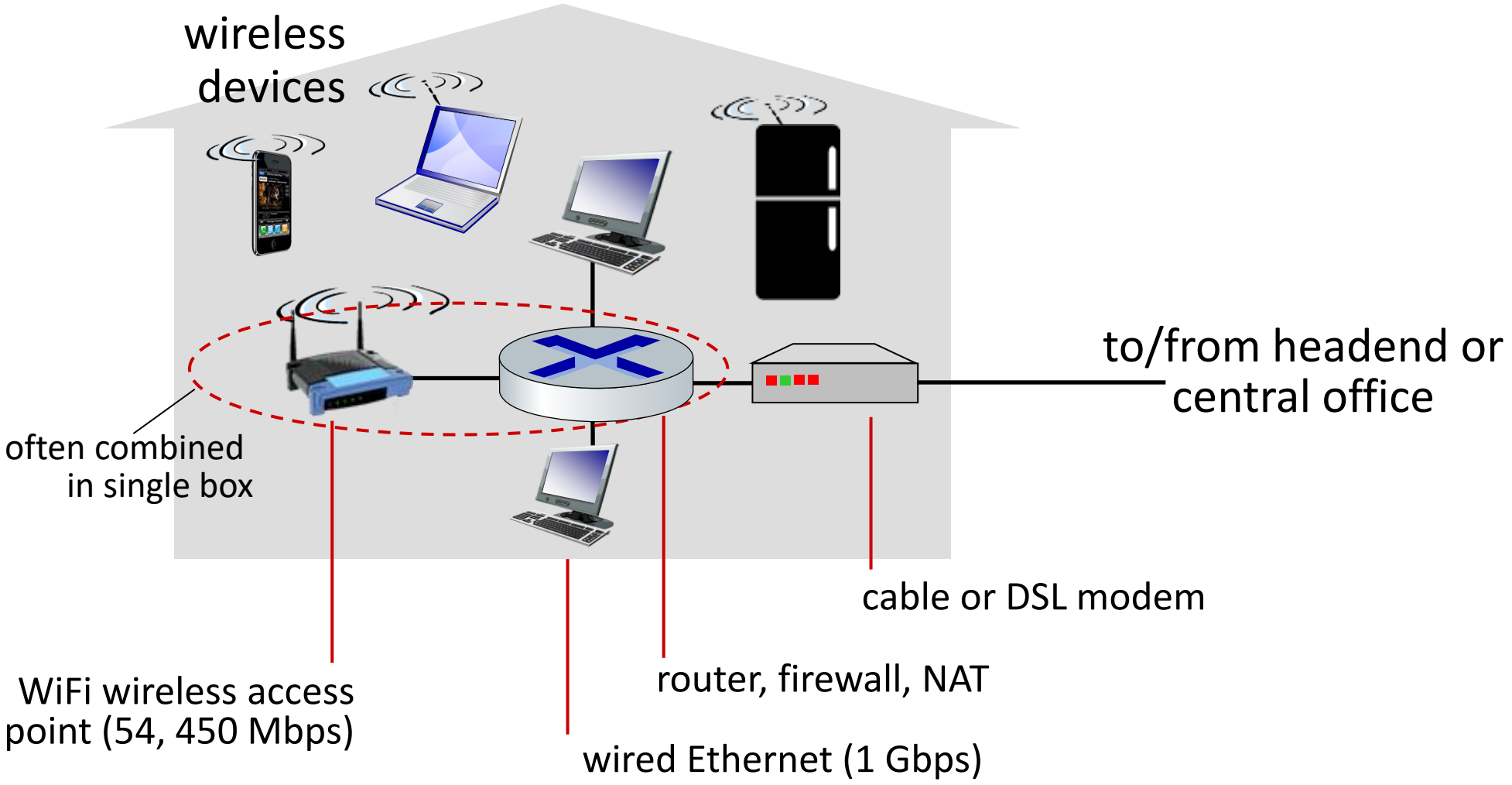
- HFC: hybrid fiber coax 混合光纤同轴电缆
  - *asymmetric*: up to 40 Mbps – 1.2 Gbps downstream transmission rate, 30-100 Mbps upstream transmission rate
- network of cable, fiber attaches homes to ISP router
  - homes *share access network* to cable headend

Access networks: digital subscriber line (DSL)



- use *existing* telephone line to central office DSLAM
  - data over DSL phone line goes to Internet
  - voice over DSL phone line goes to telephone net
- 24-52 Mbps dedicated downstream transmission rate
- 3.5-16 Mbps dedicated upstream transmission rate

Access networks: home networks



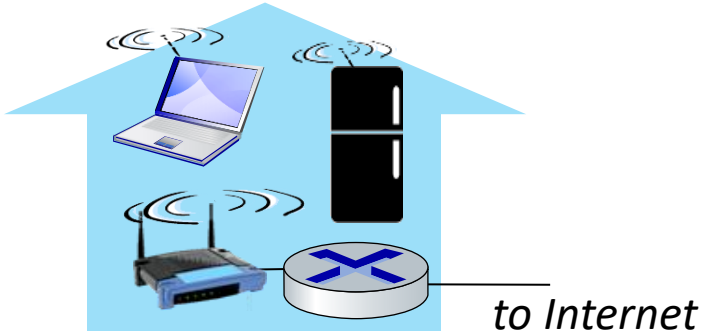
Wireless access networks

Shared *wireless* access network connects end system to router

- via base station aka "access point"

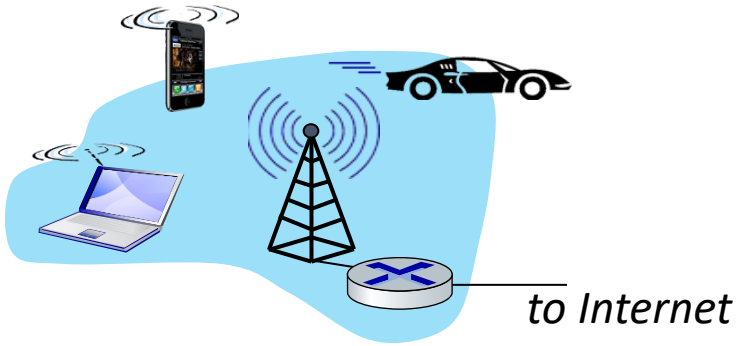
Wireless local area networks (WLANs)

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



Wide-area cellular access networks

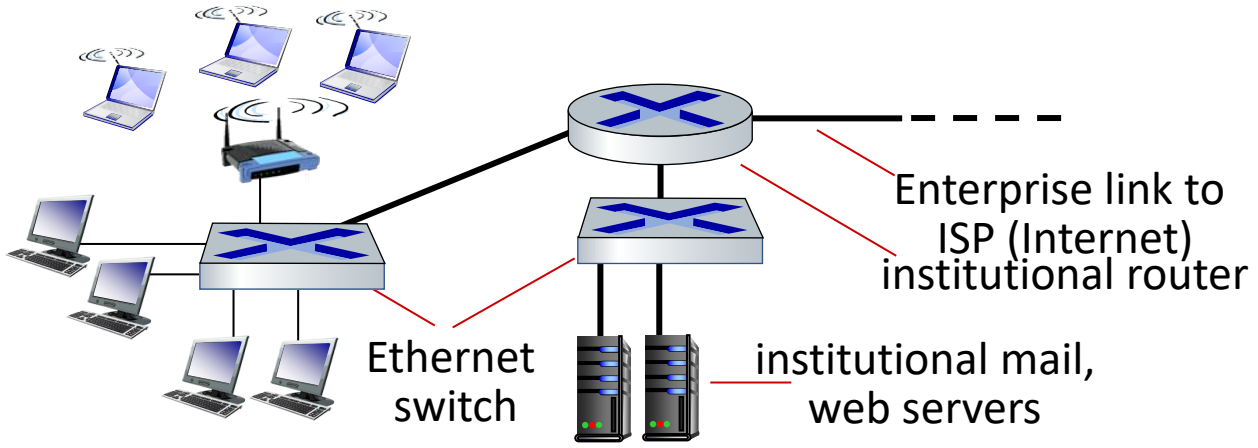
- provided by mobile, cellular network operator (10's km)
- 10's Mbps
- 4G/5G cellular networks



Wireless access networks

Protocol	Frequency	Channel Width	MIMO	Maximum data rate (theoretical)
802.11ax	2.4 or 5GHz	20, 40, 80, 160MHz	Multi User (MU-MIMO)	2.4 Gbps[1]
802.11ac wave2	5 GHz	20, 40, 80, 160MHz	Multi User (MU-MIMO)	1.73 Gbps[2]
802.11ac wave1	5 GHz	20, 40, 80MHz	Single User (SU-MIMO)	866.7 Mbps[2]
802.11n	2.4 or 5 GHz	20, 40MHz	Single User (SU-MIMO)	450 Mbps[3]
802.11g	2.4 GHz	20 MHz	N/A	54 Mbps
802.11a	5 GHz	20 MHz	N/A	54 Mbps
802.11b	2.4 GHz	20 MHz	N/A	11 Mbps
Legacy 802.11	2.4 GHz	20 MHz	N/A	2 Mbps

Access networks: enterprise networks

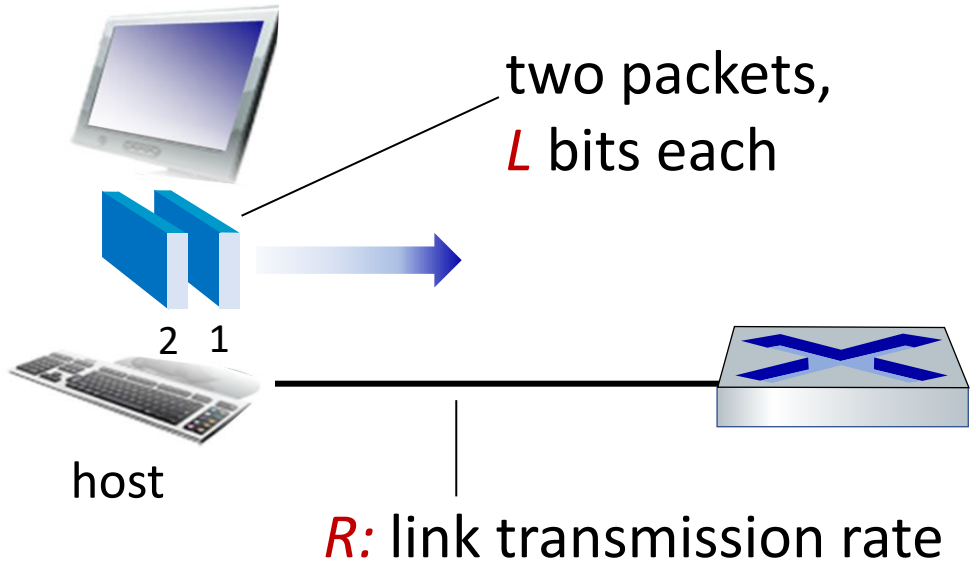


- companies, universities, etc.
- mix of wired, wireless link technologies, connecting a mix of switches and routers (we'll cover differences shortly)
  - Ethernet: wired access at 100Mbps, 1Gbps, 10Gbps
  - WiFi: wireless access points at 11, 54, 450 Mbps

Host: sends *packets* of data

host sending function:

- takes application message
- breaks into small chunks, known as *packets* (分组), of length *L* bits
- transmits packet into access network at *transmission rate R*
  - link transmission rate, aka link *capacity, aka link bandwidth*



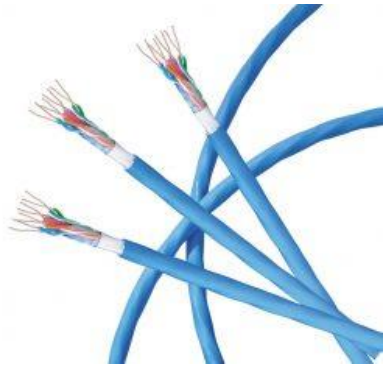
$$\text{packet transmission delay} = \text{time needed to transmit } L\text{-bit packet into link} = \frac{L \text{ (bits)}}{R \text{ (bits/sec)}}$$

Links: physical media

- bit: propagates between transmitter/receiver pairs
- physical link: what lies between transmitter & receiver
- guided media导向式介质:
  - signals propagate in solid media: copper, fiber, coax
- unguided media:
  - signals propagate freely, e.g., radio

Twisted pair (TP，双绞线)

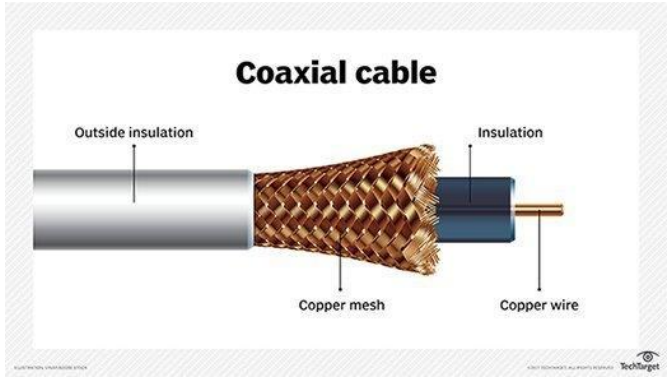
- two insulated copper wires
  - Category 5: 100 Mbps, 1 Gbps Ethernet
  - Category 6: 10 Gbps Ethernet



Links: physical media

Coaxial cable:

- two concentric copper conductors
- bidirectional
- broadband:
  - multiple frequency channels on cable
  - 100's Mbps per channel



Fiber optic cable:

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

Links: physical media

Wireless radio

- signal carried in electromagnetic spectrum
- no physical "wire"
- broadcast and "half-duplex" (sender to receiver)
- propagation environment effects:
  - reflection
  - obstruction by objects
  - interference

Radio link types:

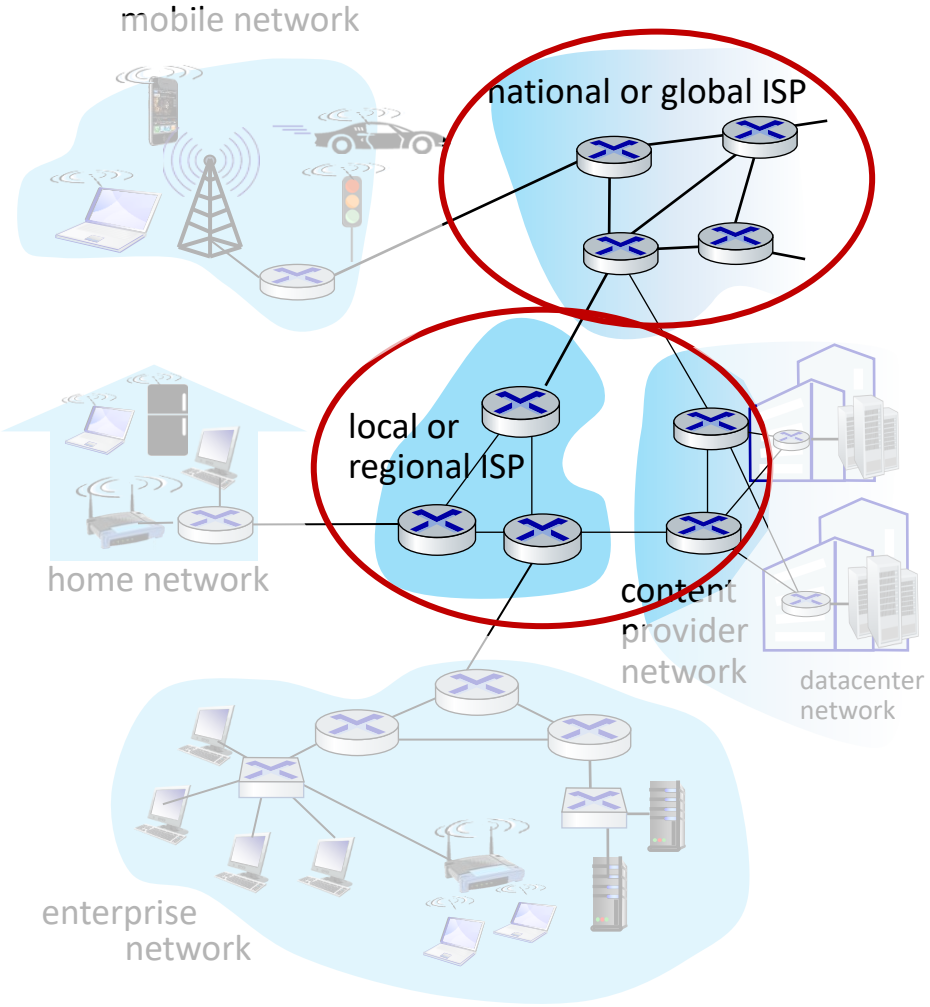
- terrestrial microwave
  - up to 45 Mbps channels
- Wireless LAN (WiFi)
  - Up to 100's Mbps
- wide-area (e.g., cellular)
  - 4G cellular: ~ 10's Mbps
- satellite
  - up to 45 Mbps per channel
  - 270 msec end-end delay
  - geosynchronous versus low-earth-orbit

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
- Protocol layers, service models
- Performance: loss, delay, throughput
- Security
- History

The network core

- mesh of interconnected routers



Routers

Home routers

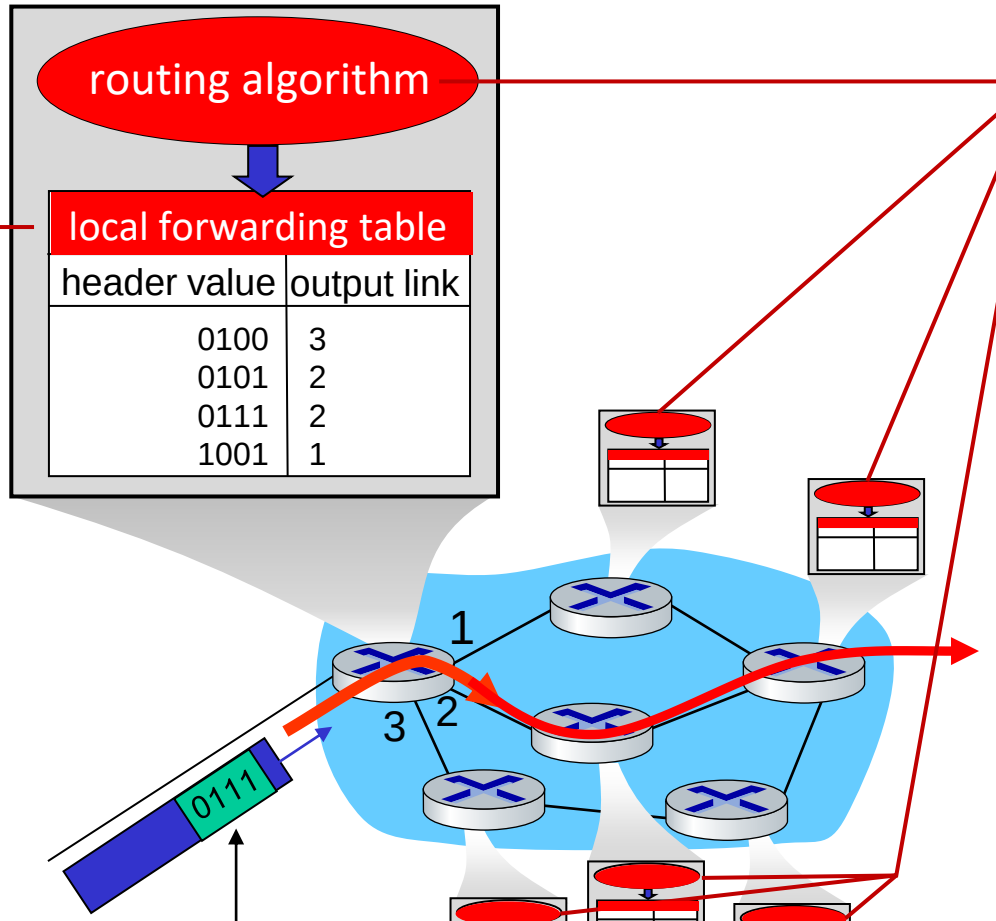
Edge routers

Core routers
Source: Wikipedia

Two key network-core functions

Forwarding:

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

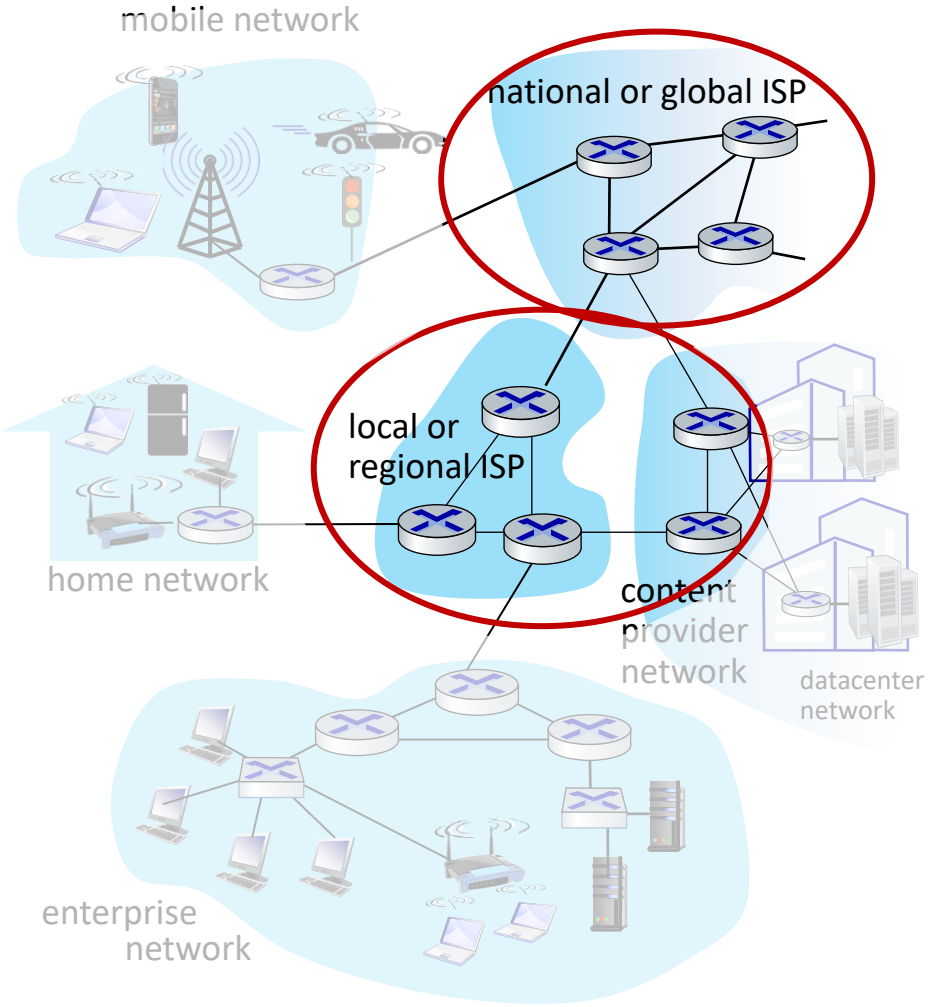


Routing:

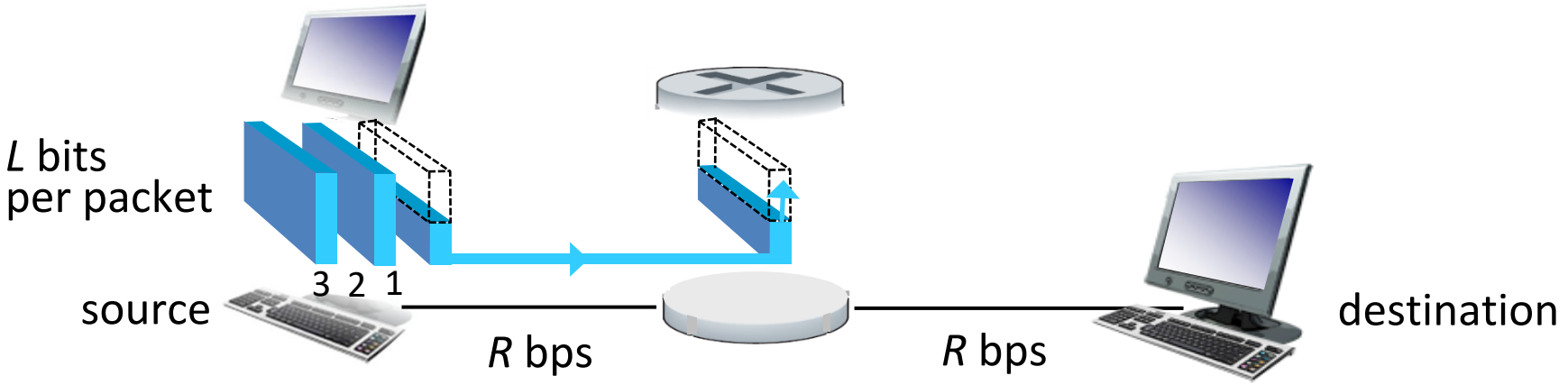
- *global* action: determine source-destination paths taken by packets
- routing algorithms

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 （分组交换）: store-and-forward

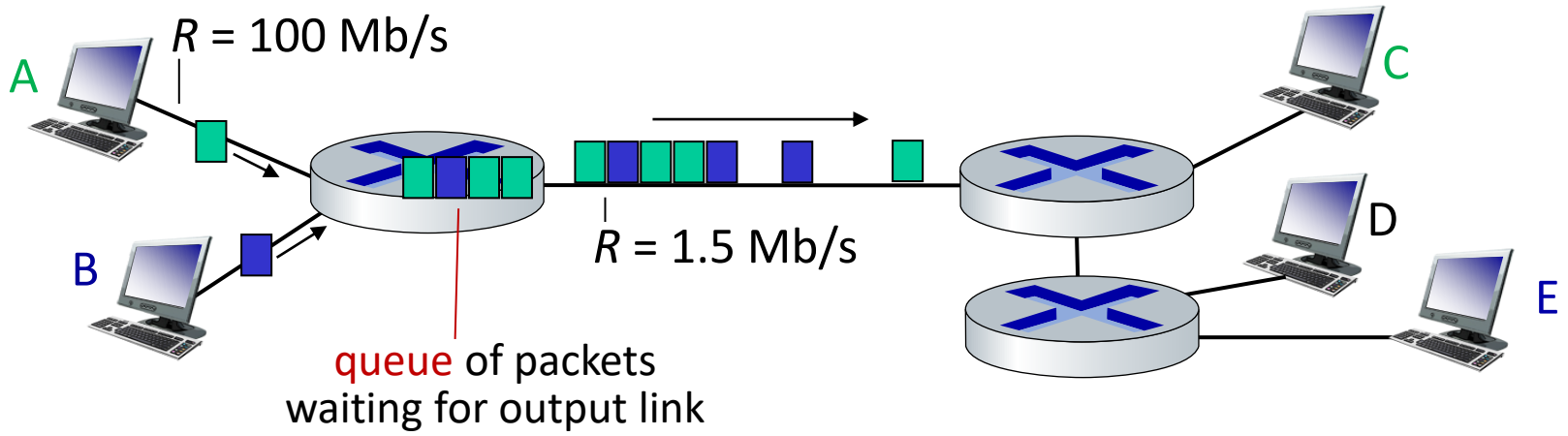


- Transmission delay: takes L/R seconds to transmit (push out) L-bit packet into link at R bps
- *Store and forward: entire* packet must arrive at router before it can be transmitted on next link
  - Store the bits on the router's buffer

One-hop numerical example:

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

Packet-switching: queueing delay, loss



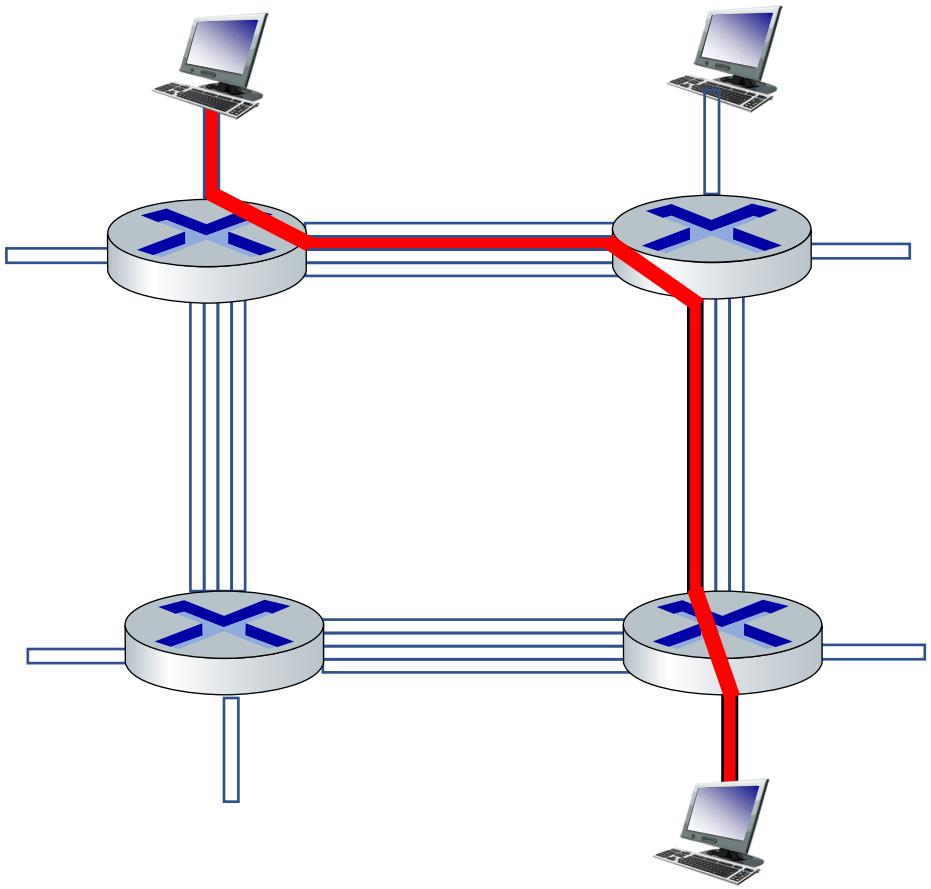
Packet queuing and loss: if arrival rate (in bps) to link exceeds transmission rate (bps) of link for a period of time:

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

Alternative to packet switching: circuit switching

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

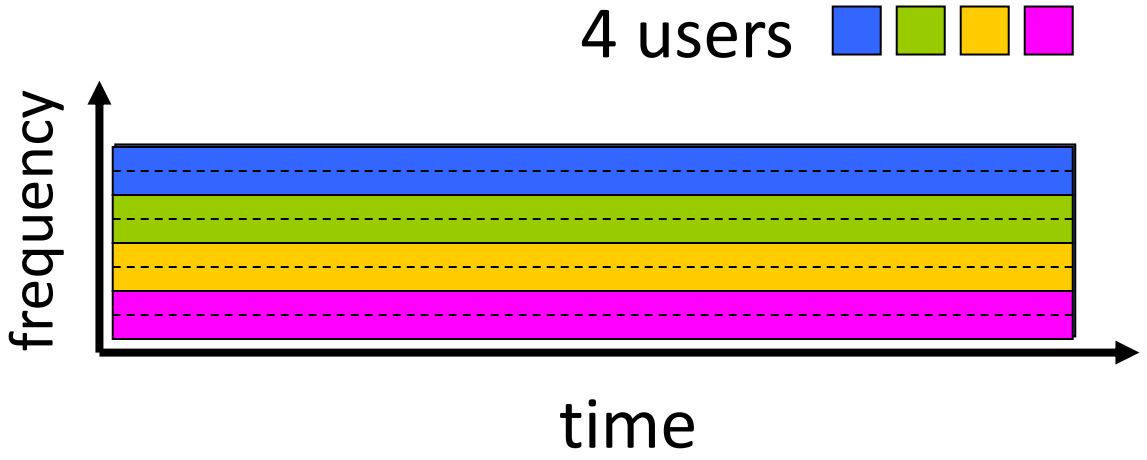
- in diagram, each link has four circuits.
  - call gets 2nd circuit in top link and 1st circuit in right link.
- dedicated resources: no sharing
  - circuit-like (guaranteed) performance
- circuit segment idle if not used by call (no sharing)
- commonly used in traditional telephone networks



Circuit switching: FDM and TDM

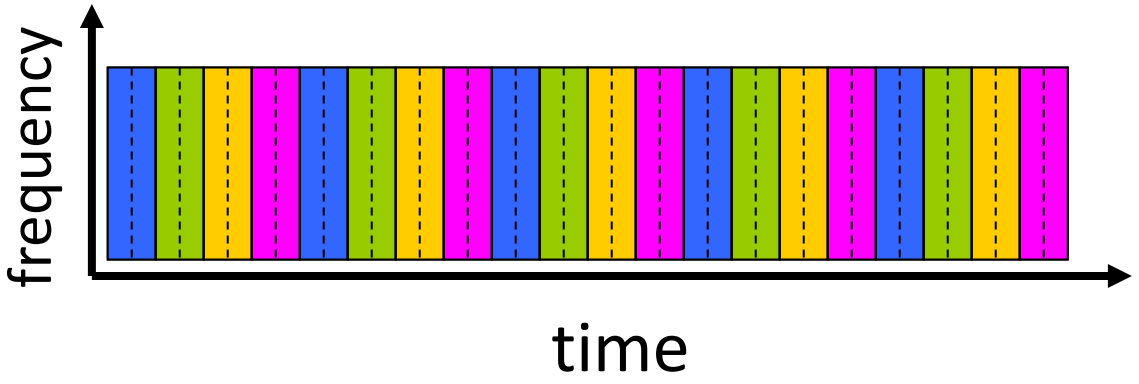
Frequency Division Multiplexing (FDM)

- 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)

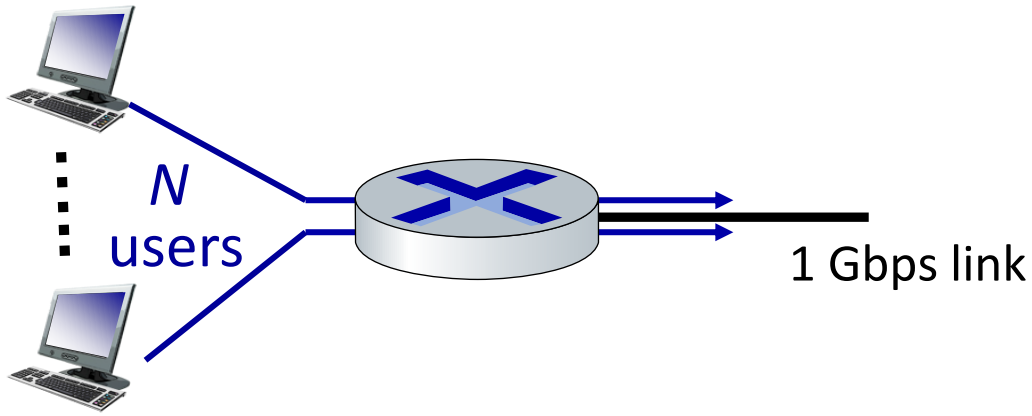


Packet switching versus circuit switching

packet switching allows more users to use network!

Example:

- Capacity: 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 ?

Packet switching versus circuit switching

Is packet switching a “slam dunk 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

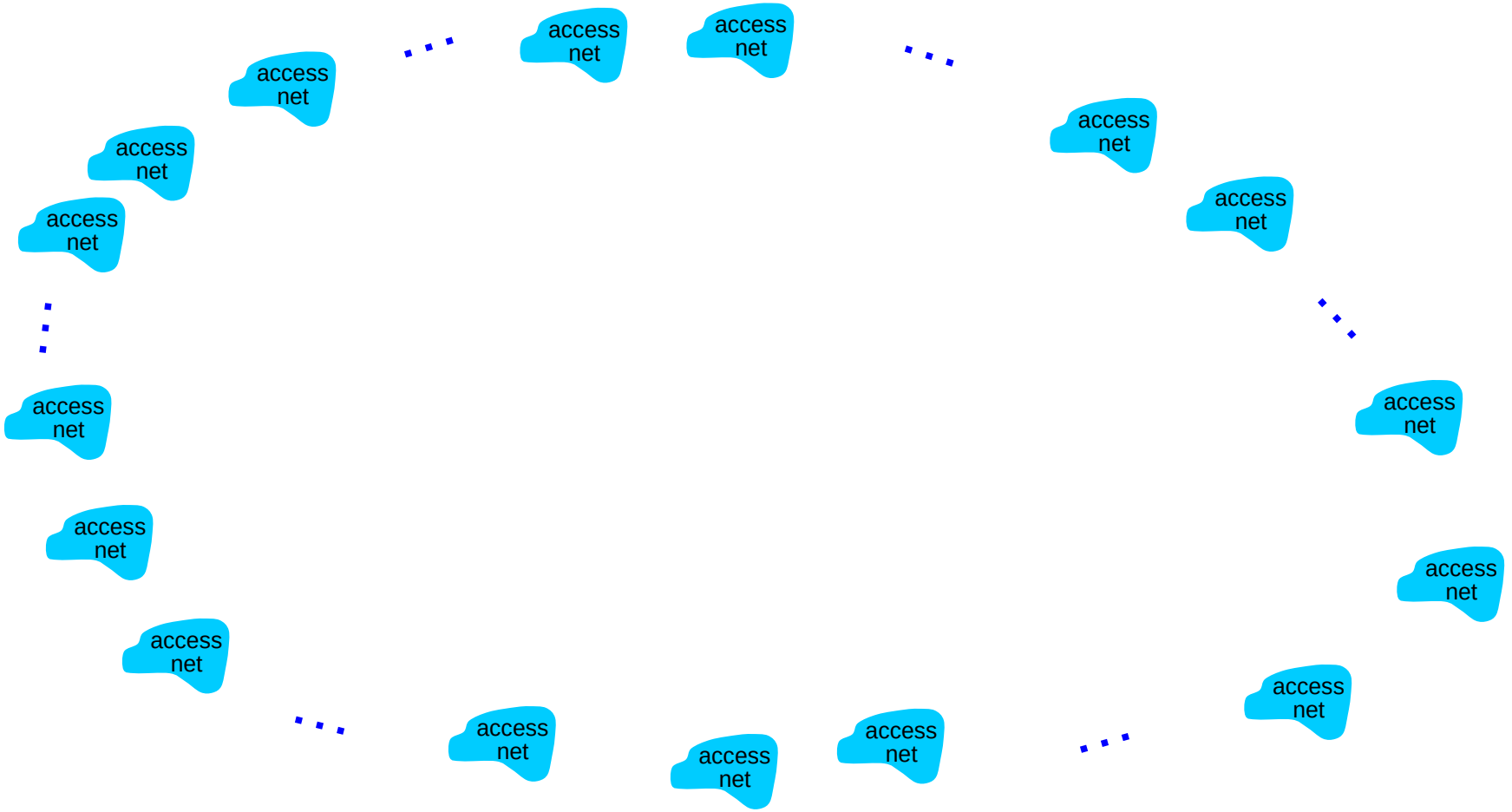
Q: human analogies of reserved resources (circuit switching) versus on-demand allocation (packet switching)?

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
  - evolution was driven by economics and national policies
- Let's take a stepwise approach to describe current Internet structure

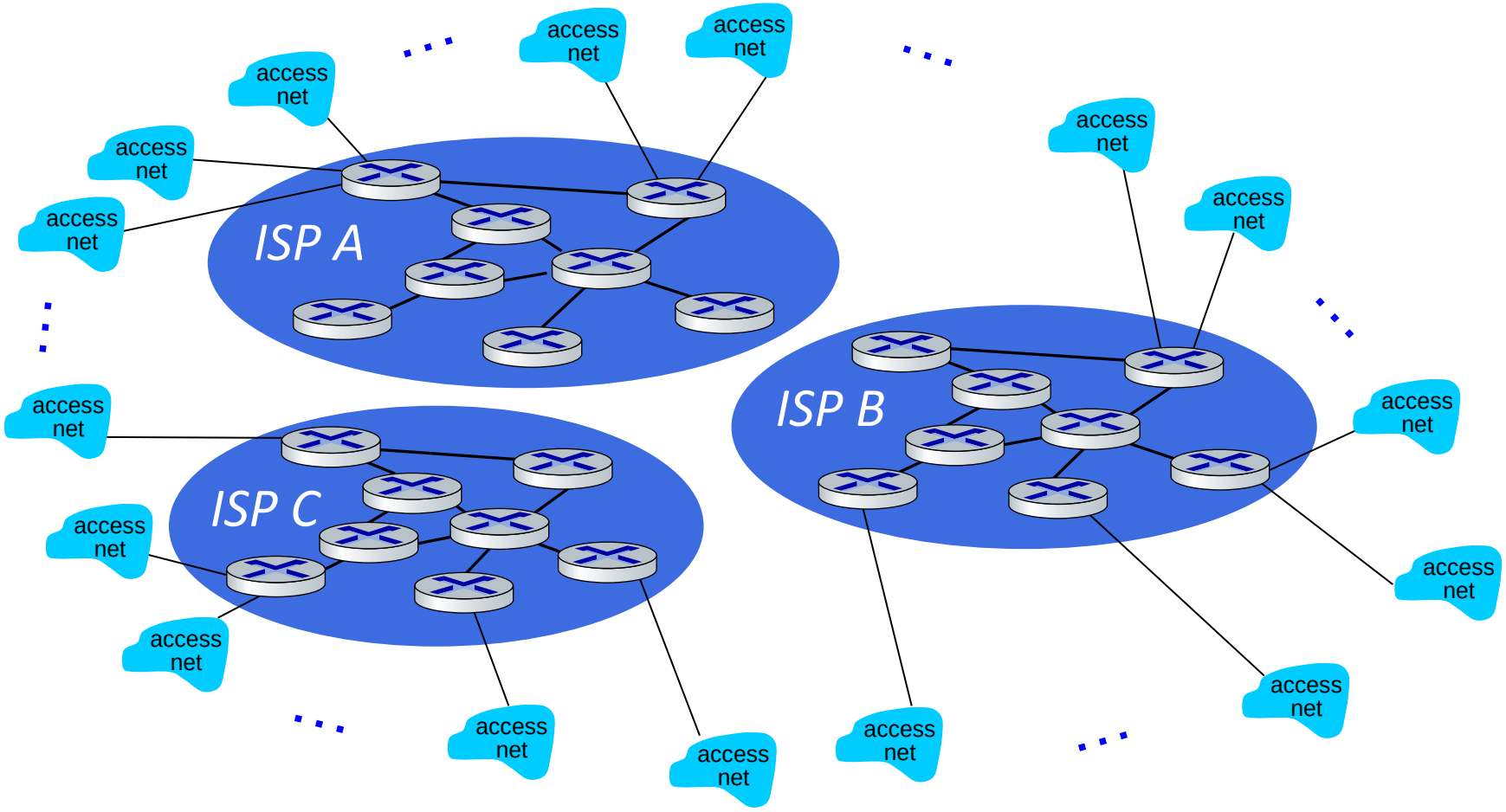
Internet structure: a “network of networks”

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



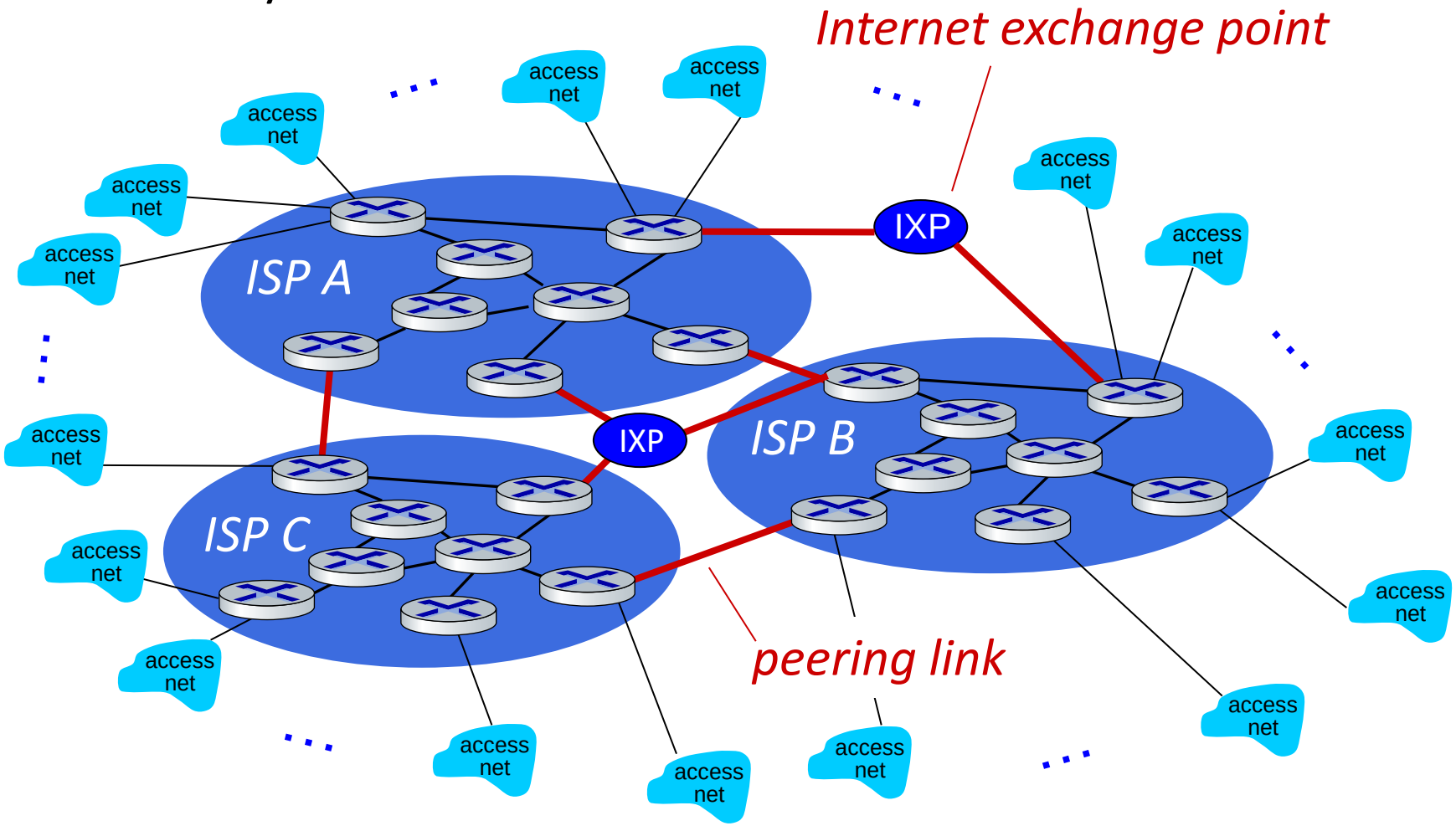
Internet structure: a “network of networks”

Connected via multiple global ISPs



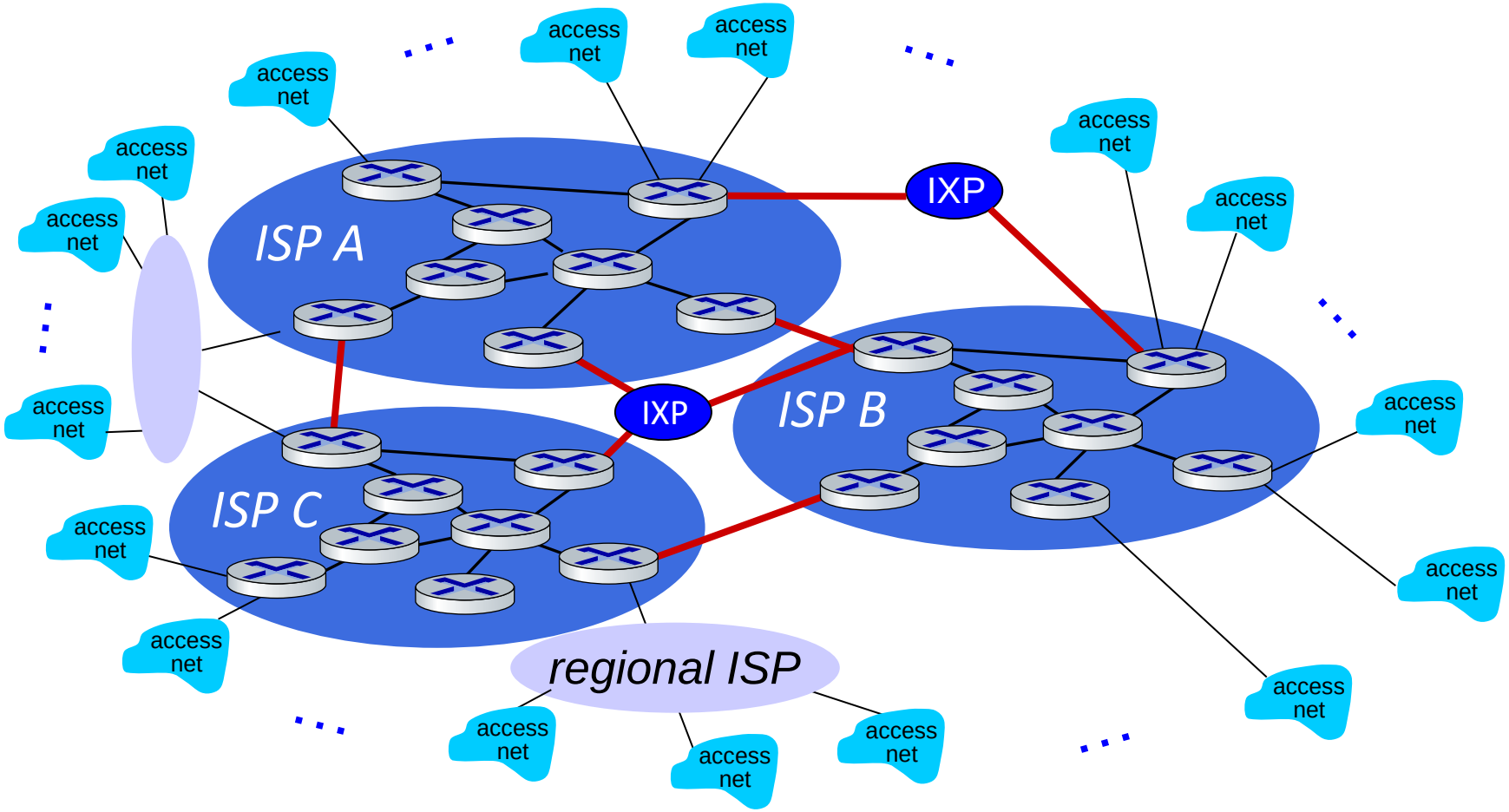
Internet structure: a "network of networks"

these global ISPs also need to be connected, so that each host can send packets to any hosts



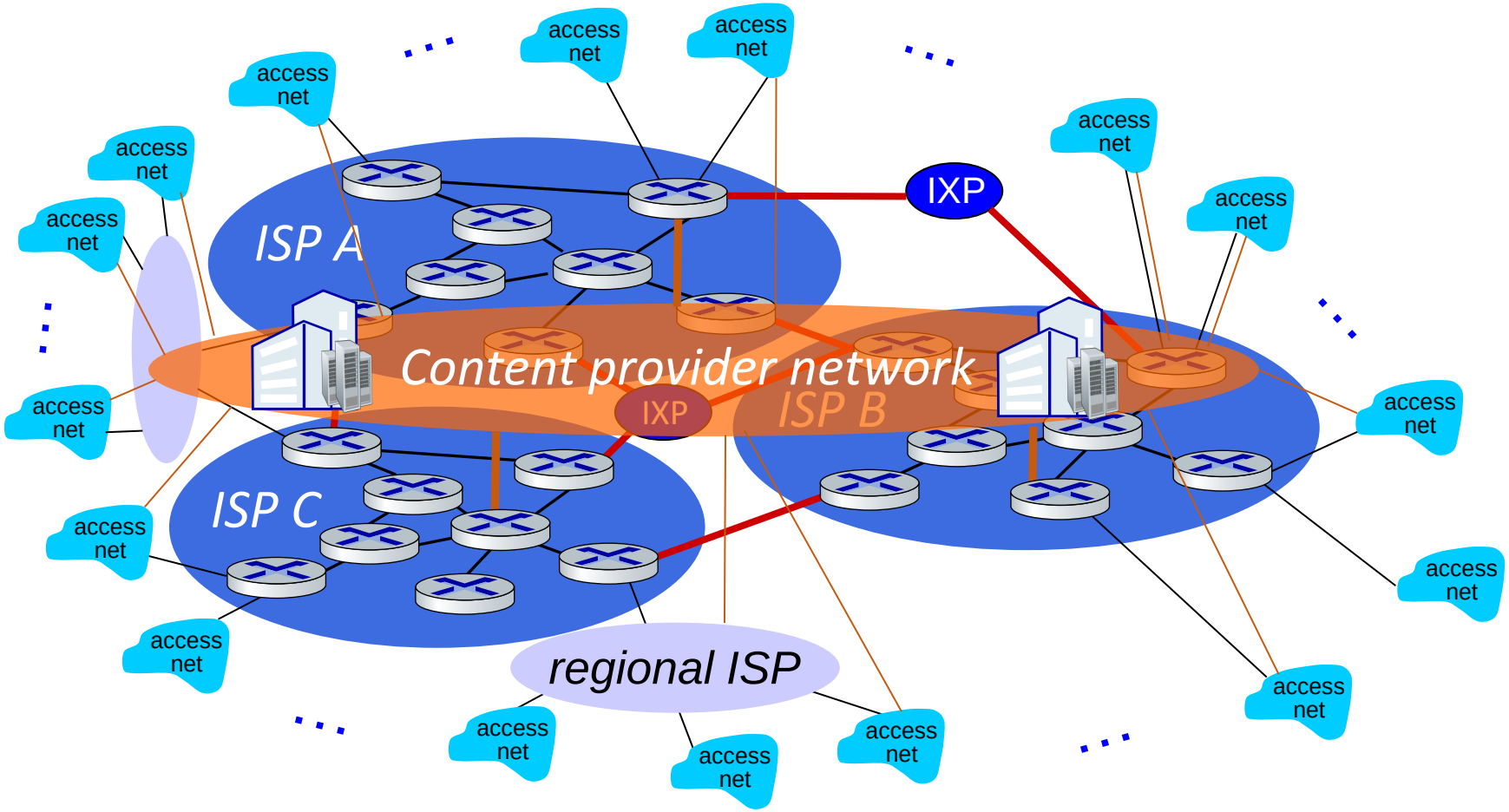
Internet structure: a "network of networks"

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

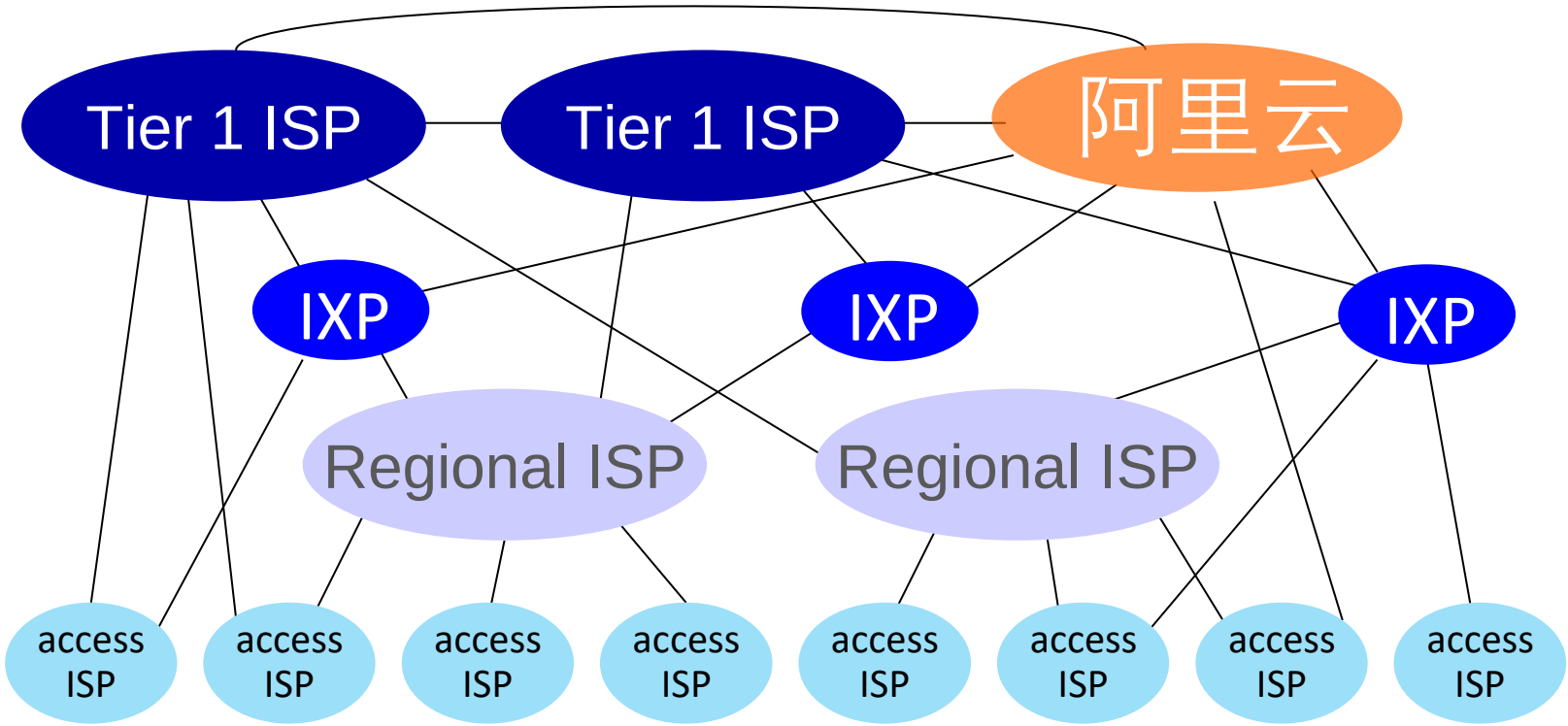


Internet structure: a "network of networks"

… 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., 移动、联通、电信), national & international coverage
- content provider networks (e.g., 阿里云、腾讯云): 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
- Network core: packet/circuit switching, internet structure
- Protocol layers, service models
- Performance: loss, delay, throughput
- Security
- History

Protocol “layers” and reference models

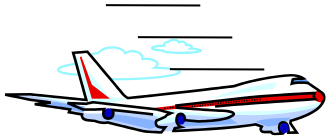
Networks are complex, with many “pieces”:

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

Question:

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

Example: organization of air travel



ticket (purchase)

baggage (check)

gates (load)

runway takeoff

airplane routing

ticket (complain)

baggage (claim)

gates (unload)

runway landing

airplane routing

airplane routing

airline travel: a series of steps, involving many services

Example: organization of air travel

layers: each layer implements a service

- via its own internal-layer actions
- relying on services provided by layer below

Q: describe in words the service provided in each layer above

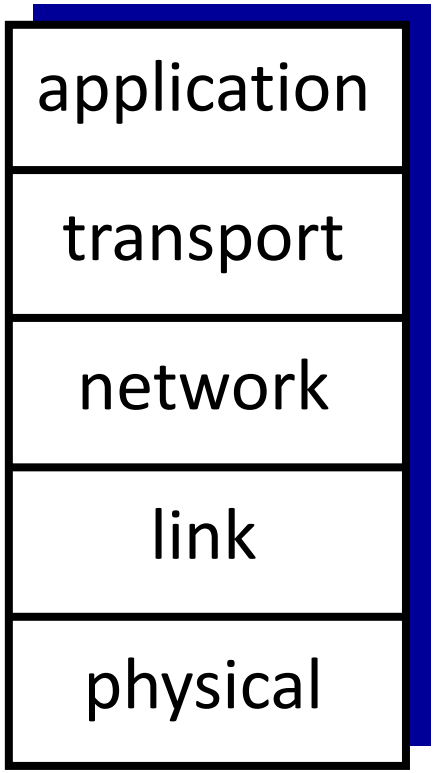
Why layering?

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 in layer's service *implementation*: transparent to rest of system
  - e.g., change in gate procedure doesn't affect rest of system

Internet protocol stack

- *application:* supporting network applications
  - IMAP, 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, 802.11 (WiFi), PPP
- *physical:* bits “on the wire”



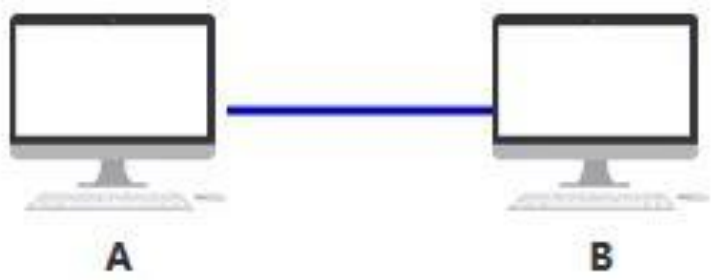
Internet protocol stack – an example

Source: 低并发编程 WeChat Channel
https://mp.weixin.qq.com/s/jiPMUk6zUdOY6eKxAjNDbQ

Use this example to understand why the layers are designed in this way!

Internet protocol stack – an example

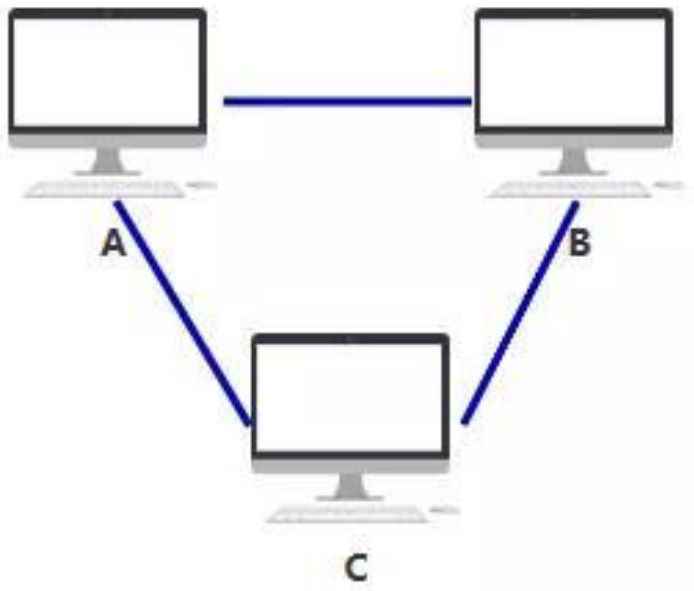
Network interface controller (NIC)



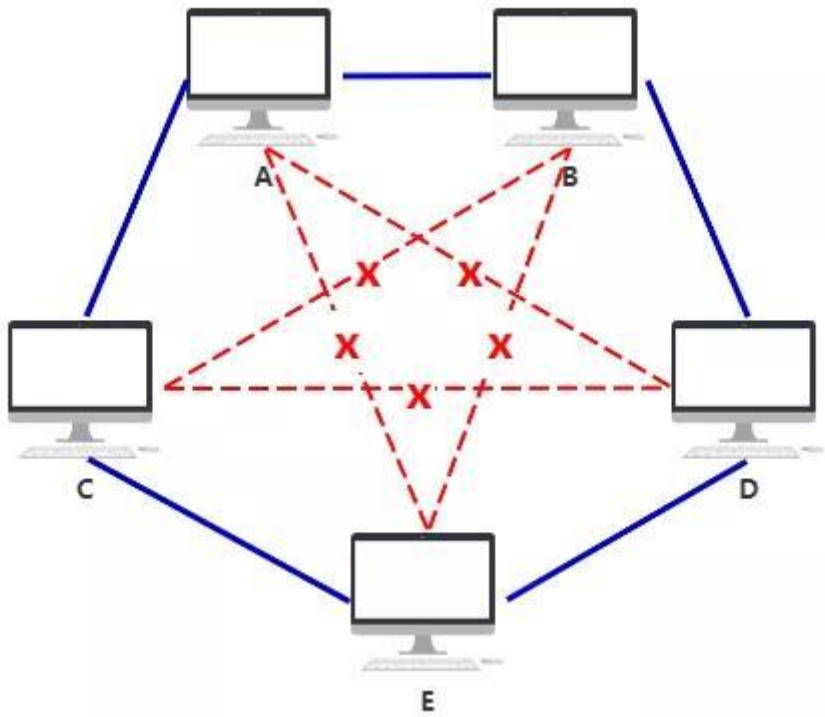
Physical layer

- provides an electrical, mechanical, and procedural interface to the transmission medium
- the hardware equipment, cabling, wiring, frequencies, pulses used to represent binary signals etc.
- NOT the physical medium

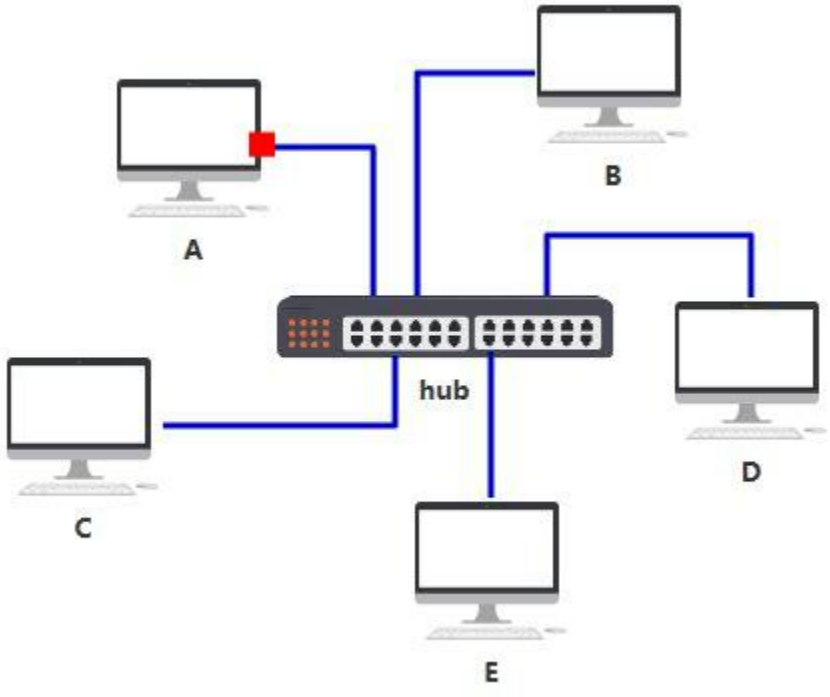
Internet protocol stack – an example



Internet protocol stack – an example



Internet protocol stack – an example



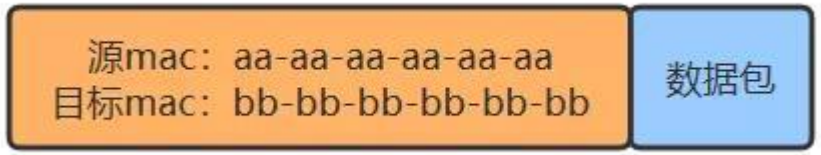
Hub (集线器)

- Broadcast
- Signal amplification and signal regeneration

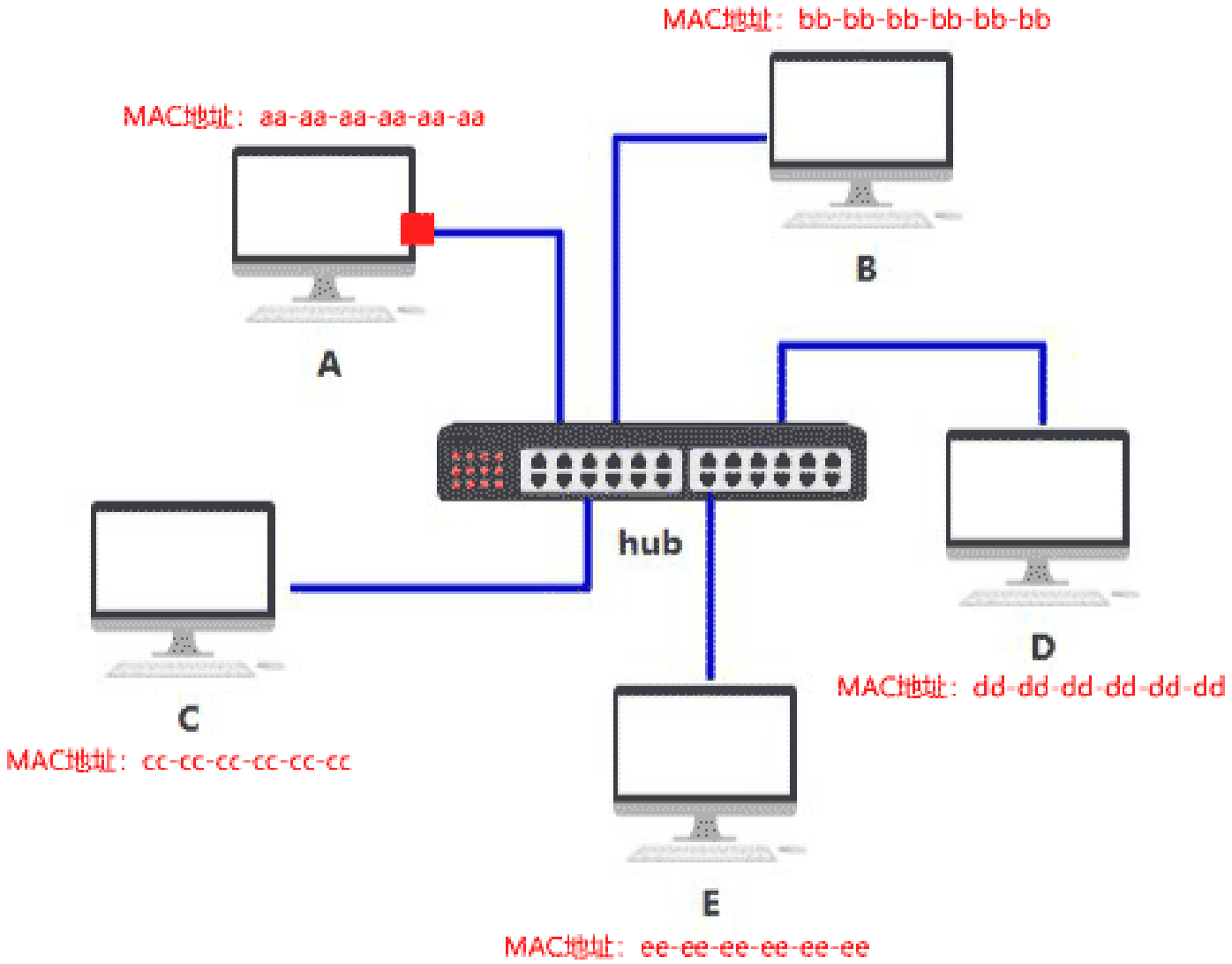
Summary of the physical layer

- Bits “on the wire”

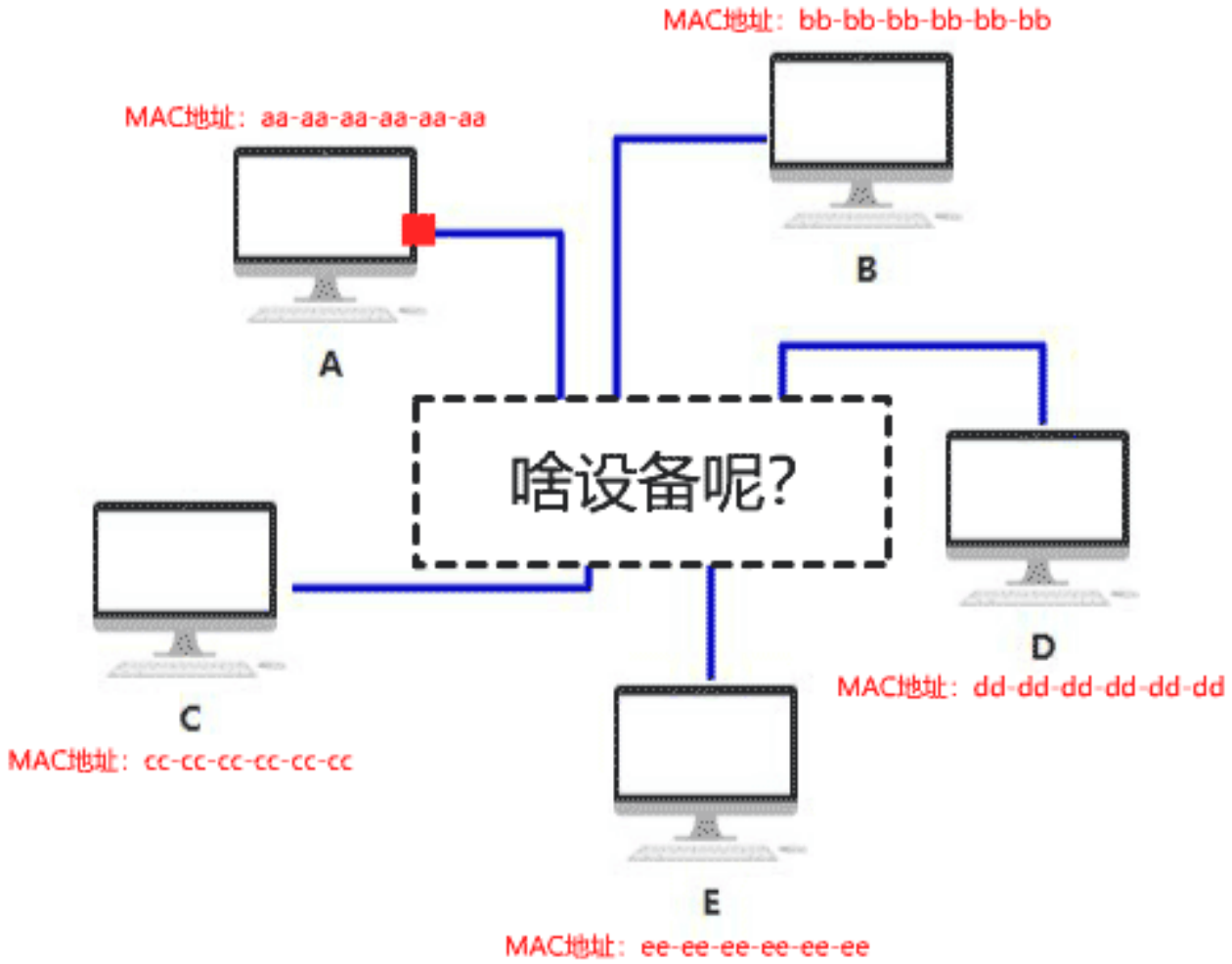
Internet protocol stack – an example



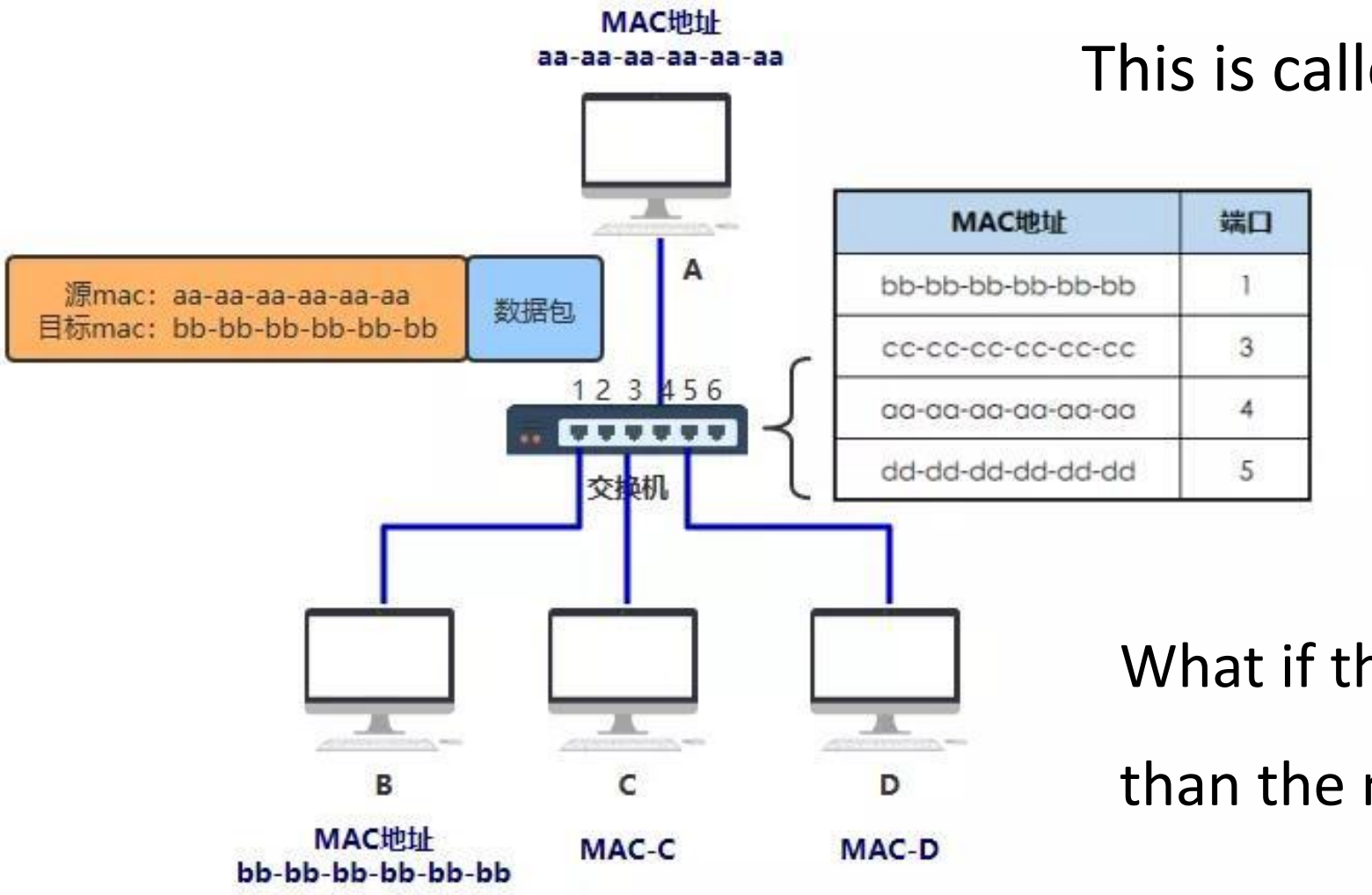
- Media Access Control (MAC) address
  - 48 bits
  - 00-B0-D0-63-C2-26
- Each network interface has a MAC address



Internet protocol stack – an example



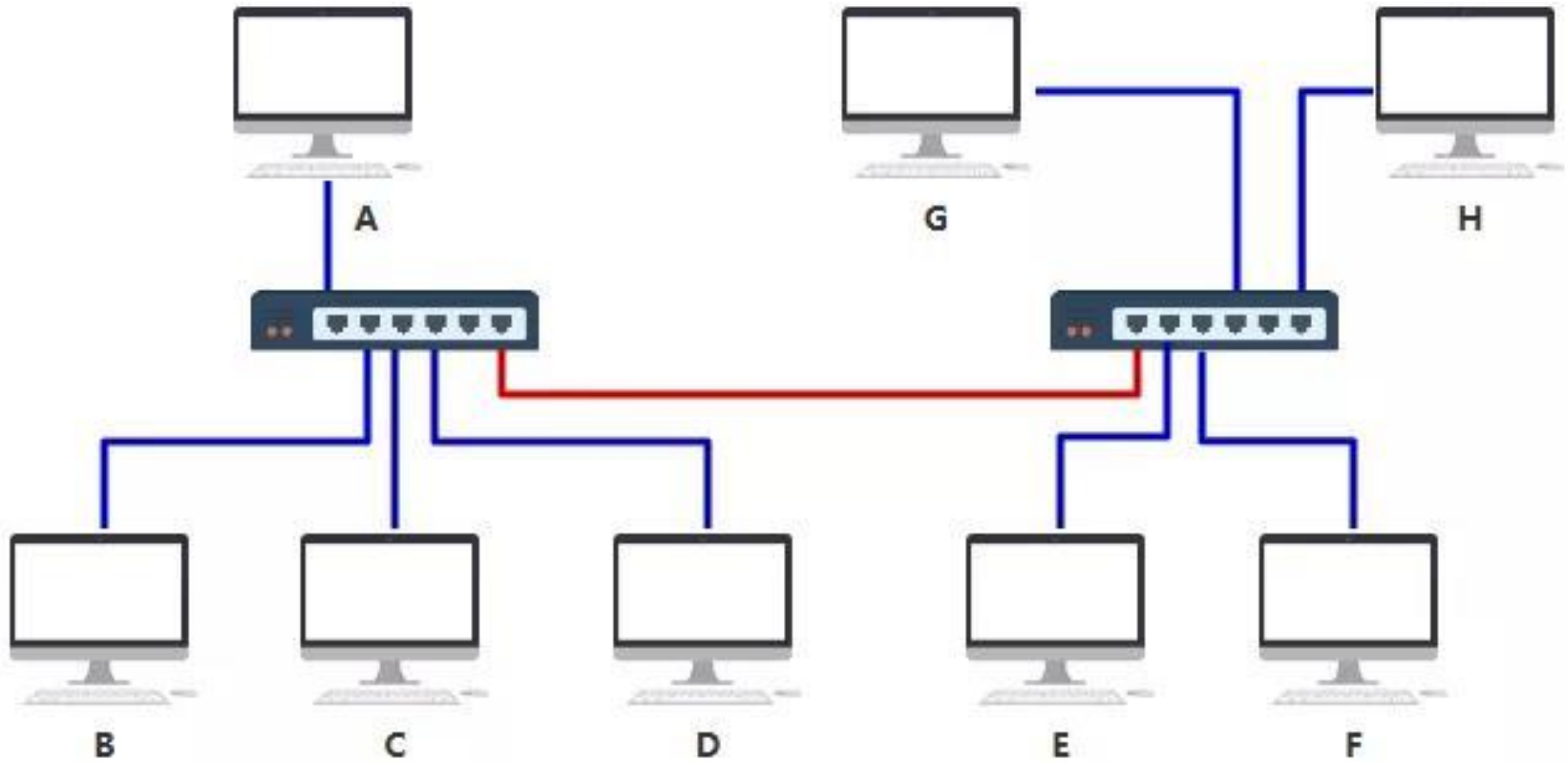
Internet protocol stack – an example



This is called Ethernet (以太网)

What if there are more machines than the number of ports?

Internet protocol stack – an example



Internet protocol stack – an example

MAC Address	Port
bb-bb-bb-bb-bb-bb	1
cc-cc-cc-cc-cc-cc	3
aa-aa-aa-aa-aa-aa	4
dd-dd-dd-dd-dd-dd	5
ee-ee-ee-ee-ee-ee	6
ff-ff-ff-ff-ff-ff	6
gg-gg-gg-gg-gg-gg	6
hh-hh-hh-hh-hh-hh	6

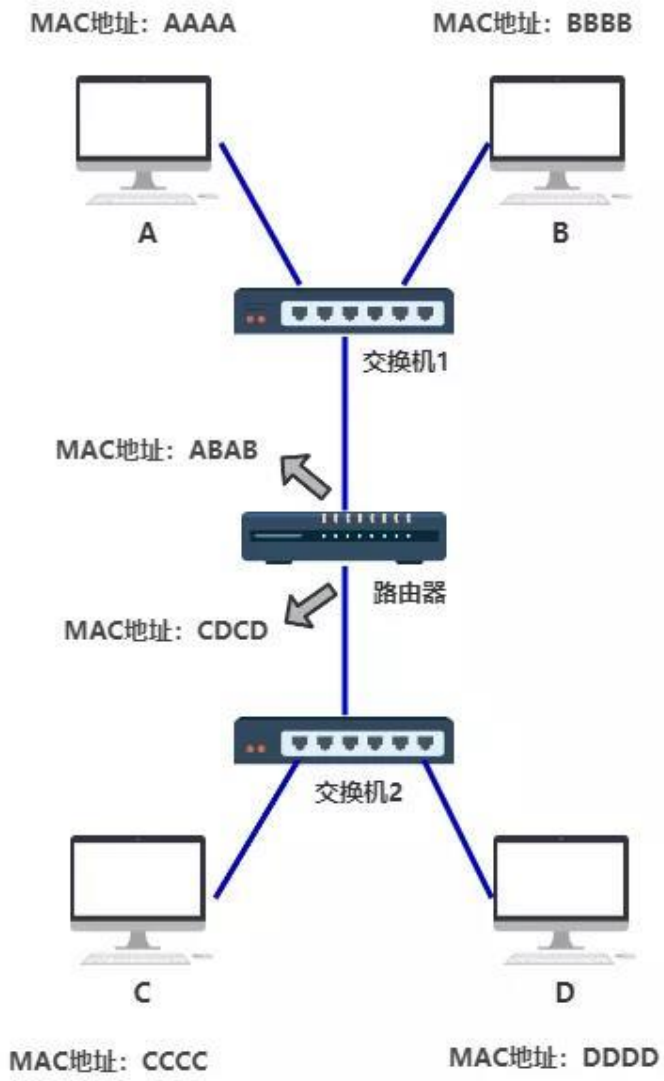
Summary of the link layer

- Data communication with neighbor

Internet protocol stack – an example

What if there are more local networks to interconnect?

Question: how does a host know when it should send a packet to the router?



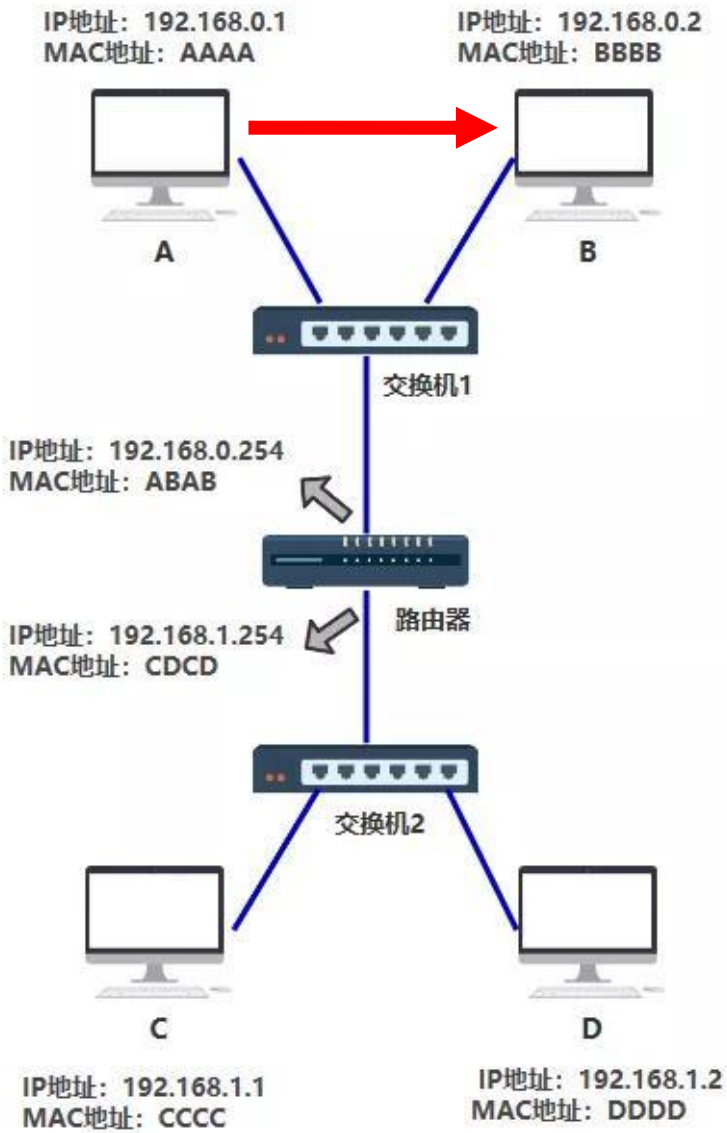
Internet protocol stack – an example

A new type of address

- Internet Protocol address (IP address)
- An example IP address: 11000000101010000000000000000001
- 192.168.0.1 (dotted-decimal format)
- 0.0.0.0 – 255.255.255.255

Internet protocol stack – an example

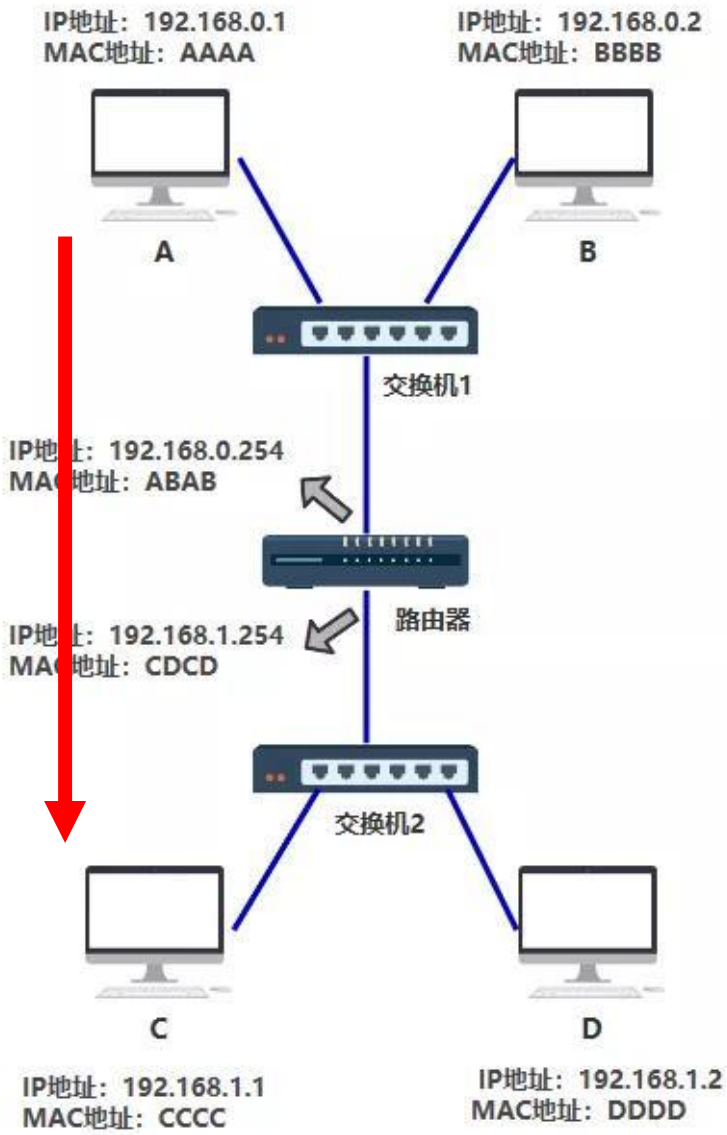
A to B:



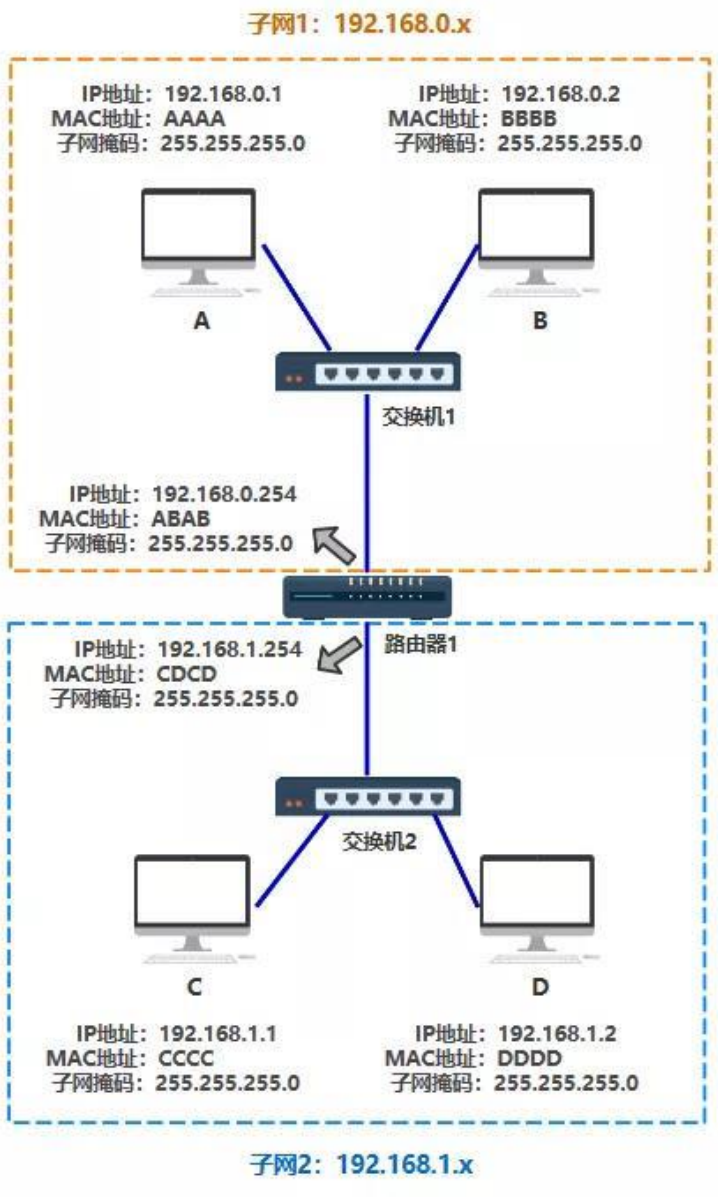
Internet protocol stack – an example

A to 路由器:

路由器 to C:



Internet protocol stack – an example



Subnet

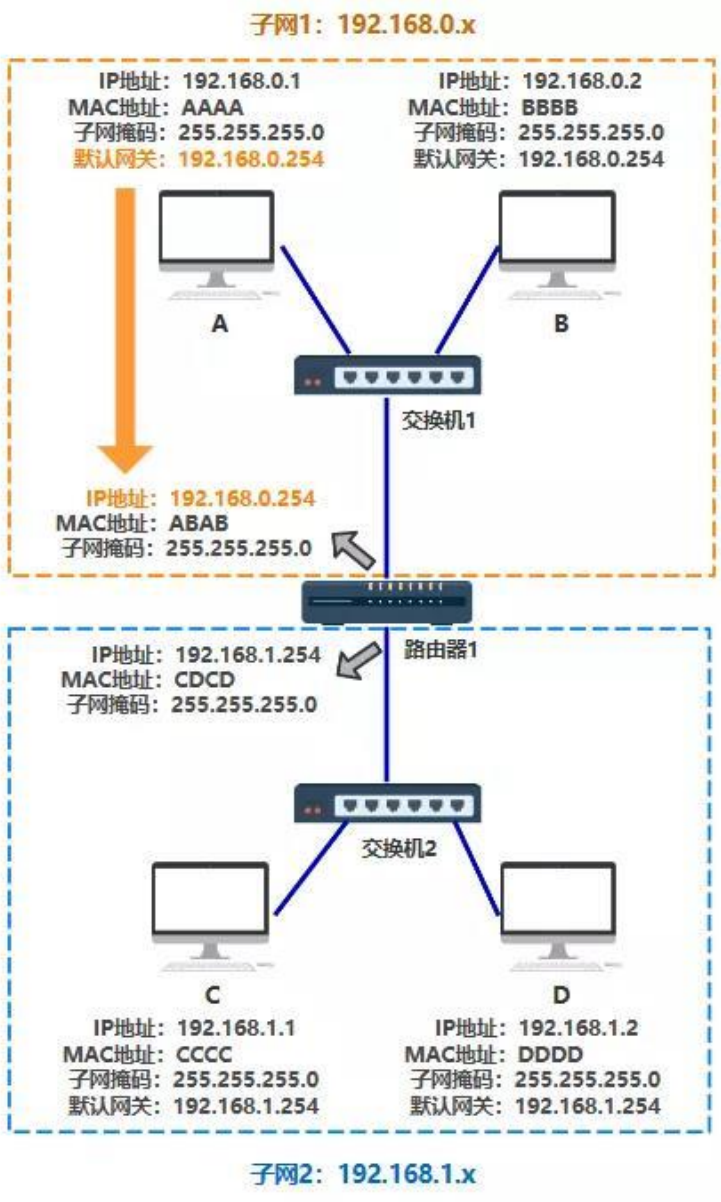
- 192.168.0.1 and 192.168.0.2: same subnet
- 192.168.0.1 and 192.168.1.1: different subnet

Subnet mask to find the same subnet

- **A**: 192.168.0.1 & 255.255.255.0 = 192.168.0.0
- **B**: 192.168.0.2 & 255.255.255.0 = 192.168.0.0
- **C**: 192.168.1.1 & 255.255.255.0 = 192.168.1.0
- **D**: 192.168.1.2 & 255.255.255.0 = 192.168.1.0

PS: 255 in binary: 11111111

Internet protocol stack – an example



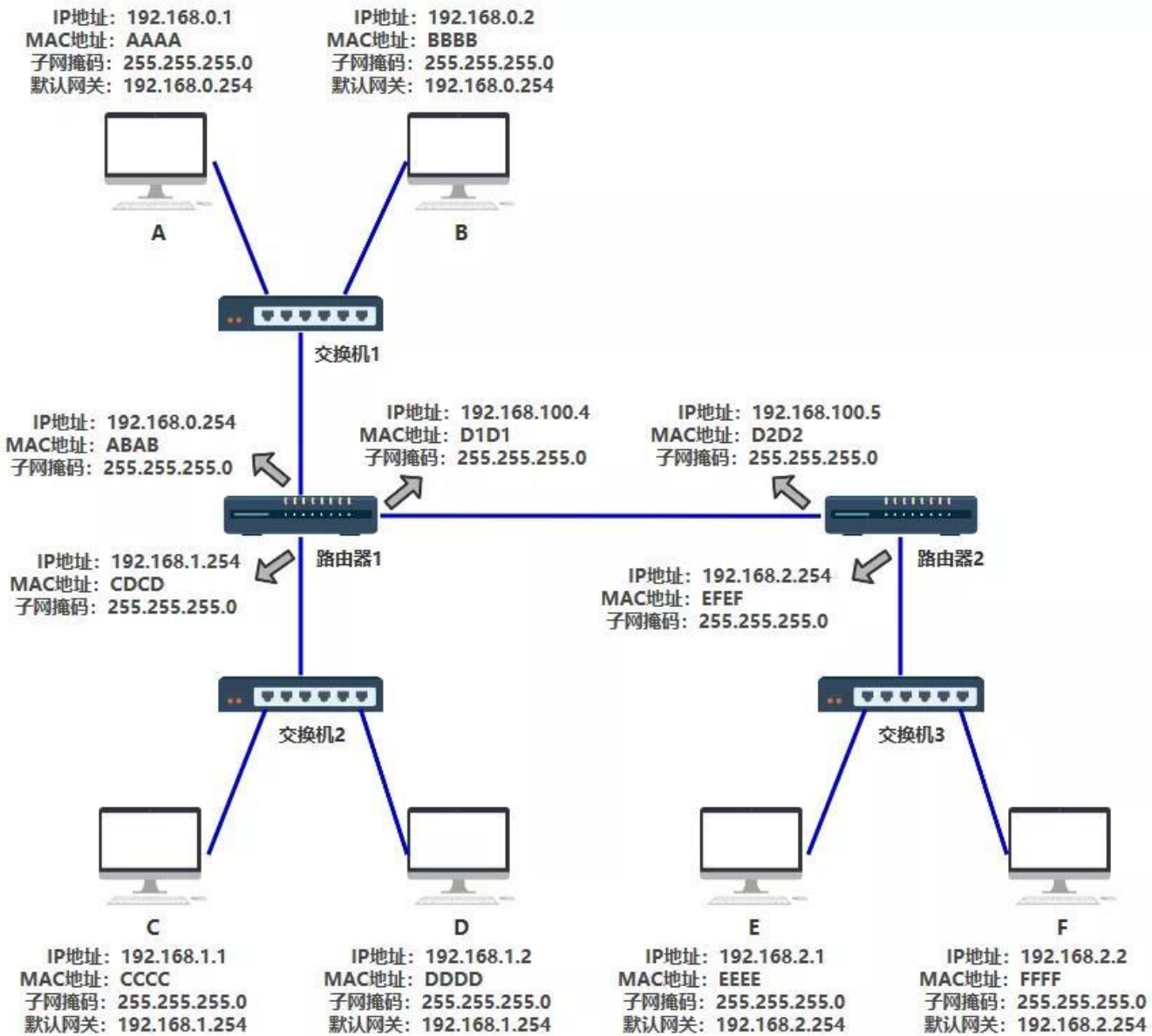
Gateway

- 192.168.0.254 is the router

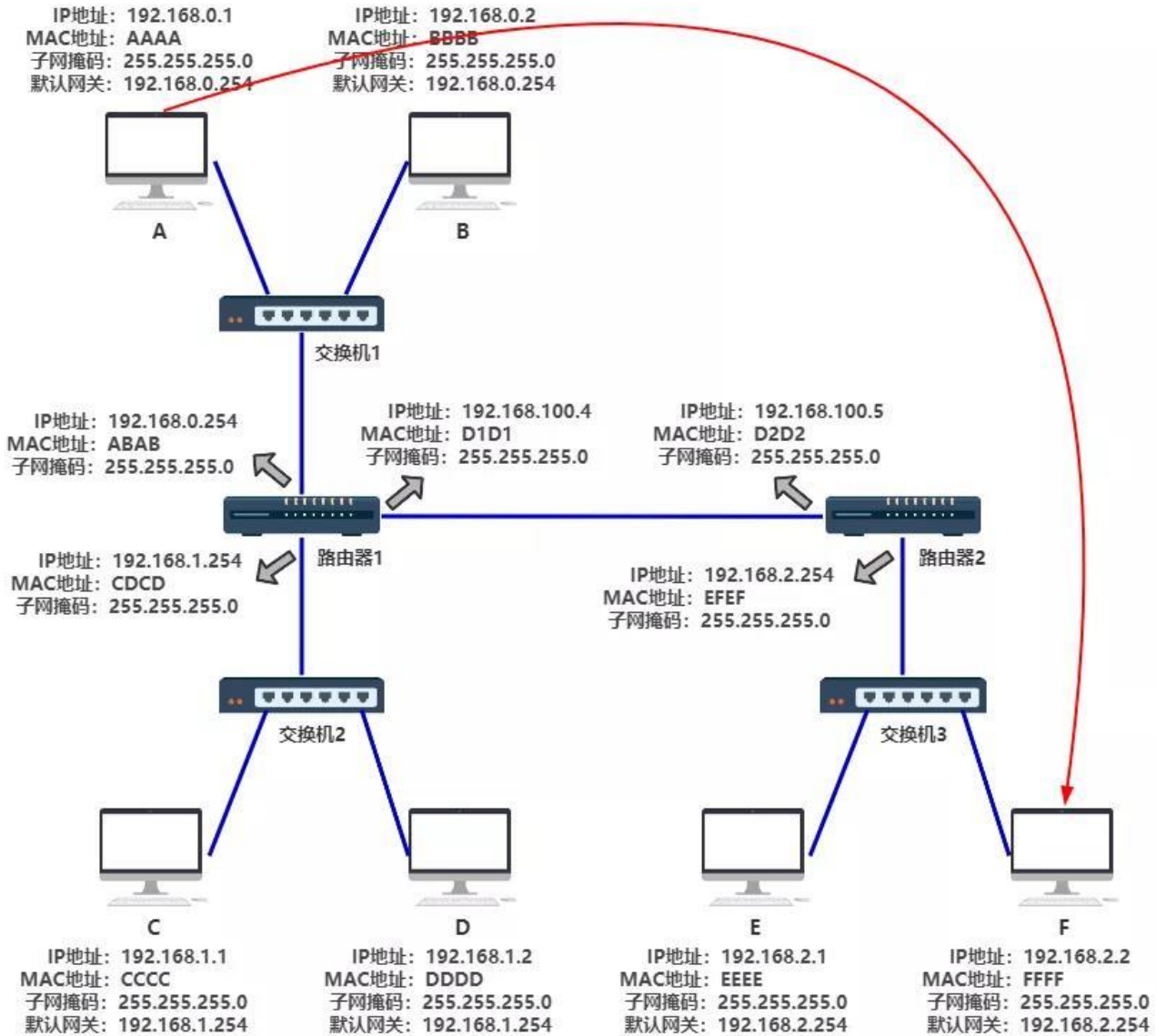
Summary of network layer

- Route datagram from source to destination

Internet protocol stack – an example



Internet protocol stack – an example



Transport and application layers

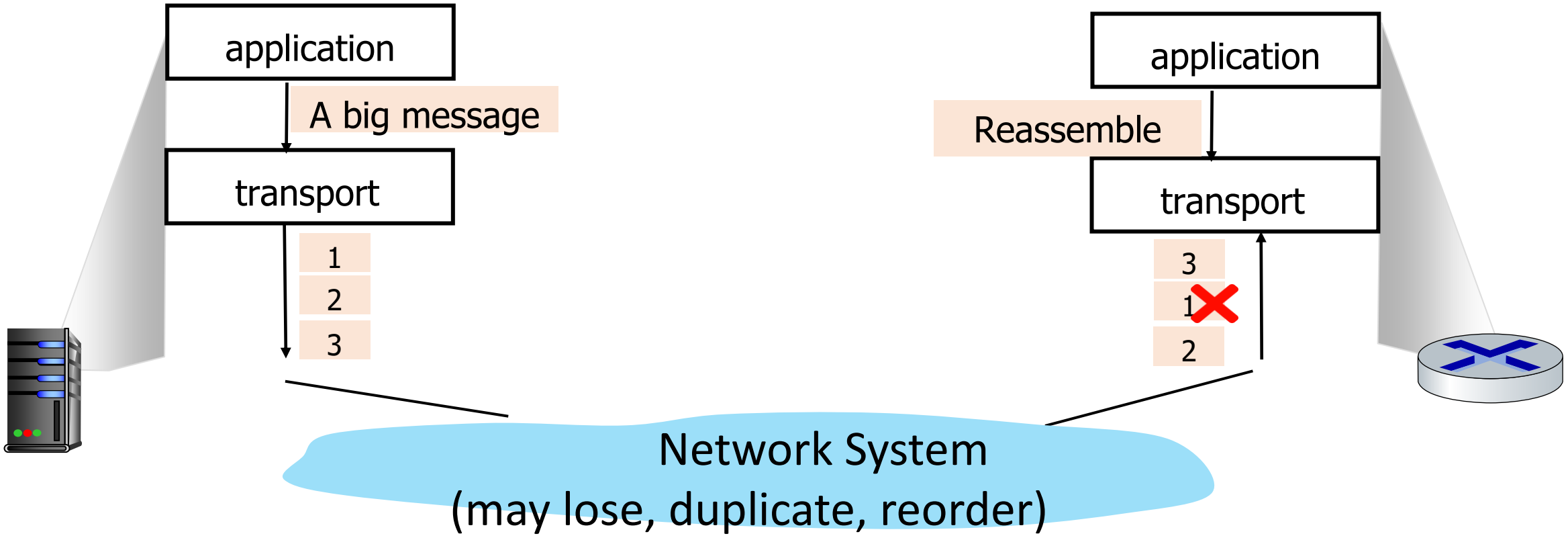
- Build on top of the bottom three layers
  - Use the functionalities of the bottom layers to provide services

- Transport layer
  - sender: breaks application messages into *segments*, passes to network layer
  - receiver: reassembles segments into messages, passes to application layer
  - E.g., TCP and UDP protocols

- Application layer
  - supporting network applications

Transport and application layers

- Transport layer: provide *logical communication* between application processes running on different hosts (e.g., TCP and UDP)

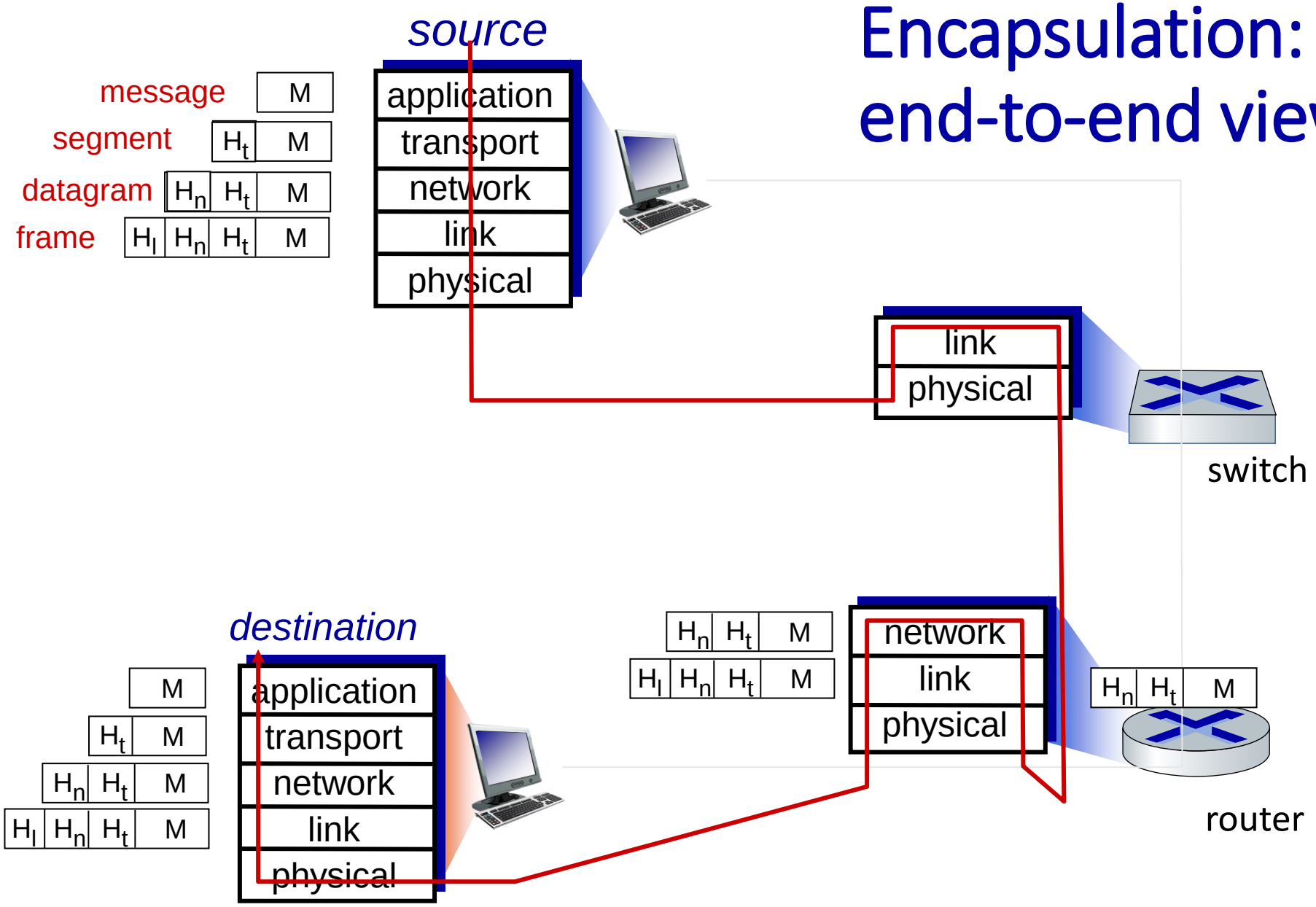


Notes on the example

- A high-level brief overview of the main functionalities of each layer
  - The example still misses many other functionalities
  - More details on other functionalities later in the course
  - E.g., how to create the MAC table and routing table

- Please keep this example in mind when you study the course later in this semester
  - Have a big picture on why we need to design the network like this

Encapsulation: an end-to-end view



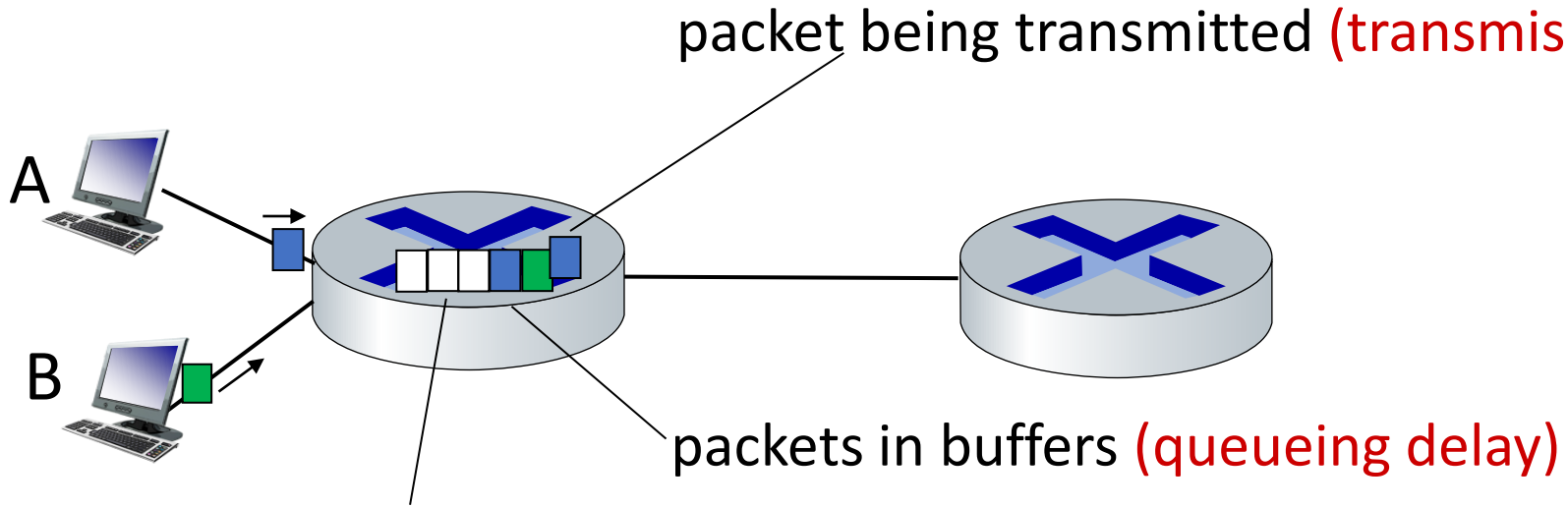
Chapter 1: roadmap

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

How do packet loss and delay occur?

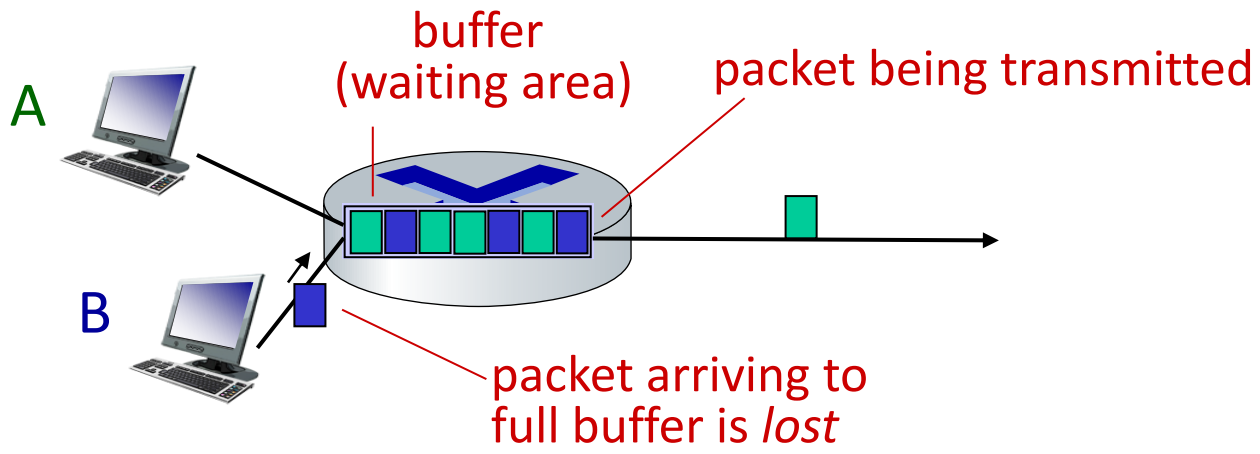
packets *queue* in router buffers

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



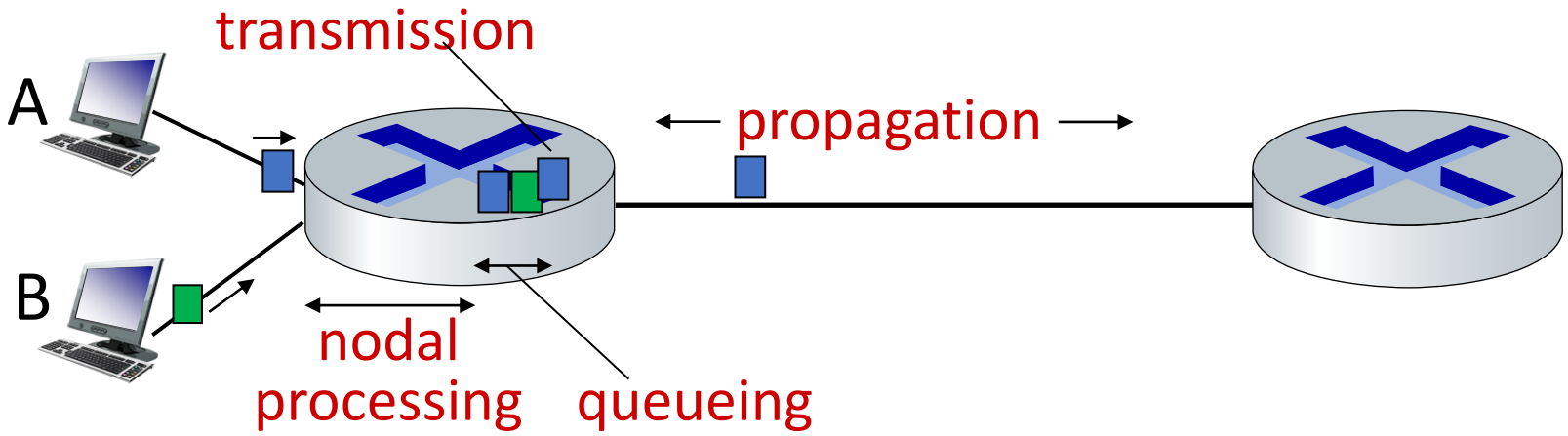
Packet loss

- 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

Packet delay at a router: four sources



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

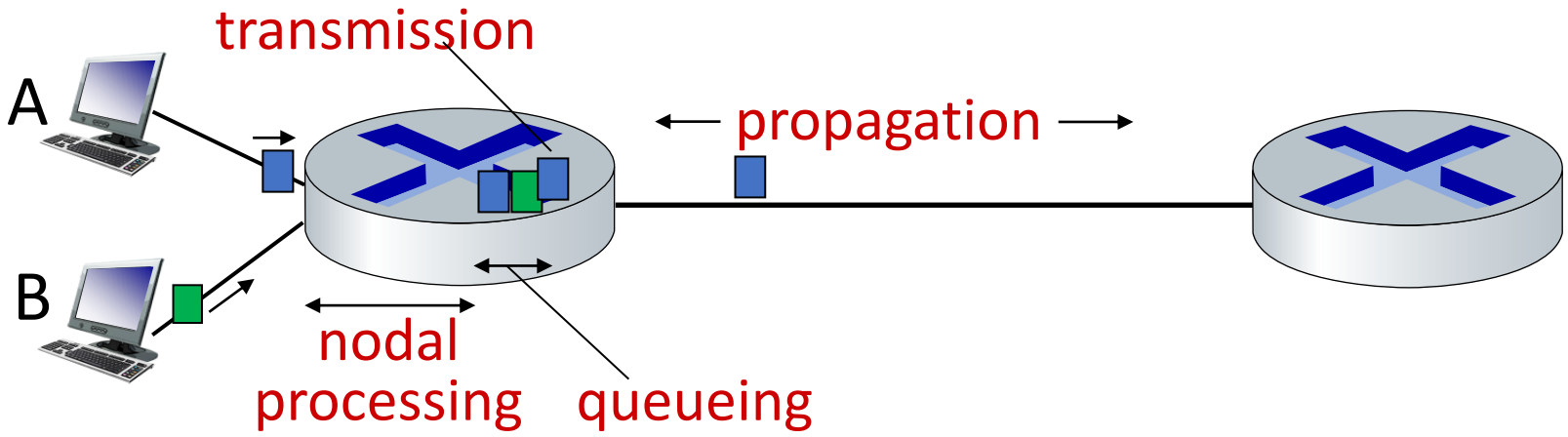
d_{proc}: nodal processing

- 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

Packet delay: four sources



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

d_{trans}: transmission delay:

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

d_{prop}: propagation delay:

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

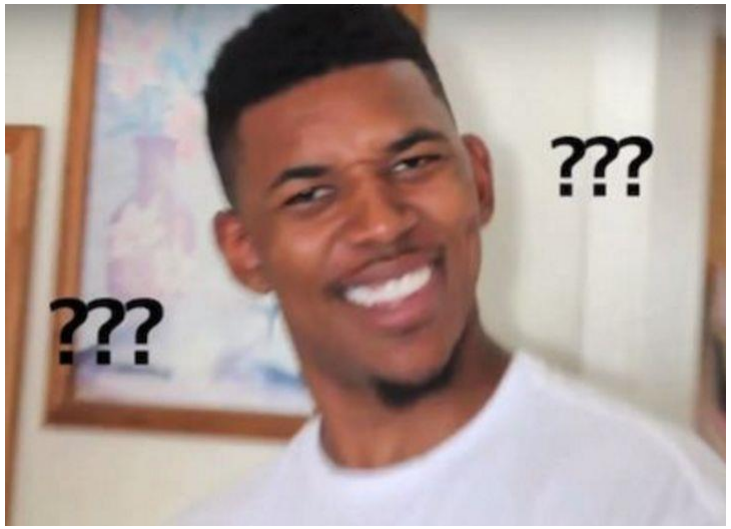
d_{trans} and d_{prop} *very* different

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

Transmission delay vs. propagation delay

d_{trans}: transmission delay (传输时延):

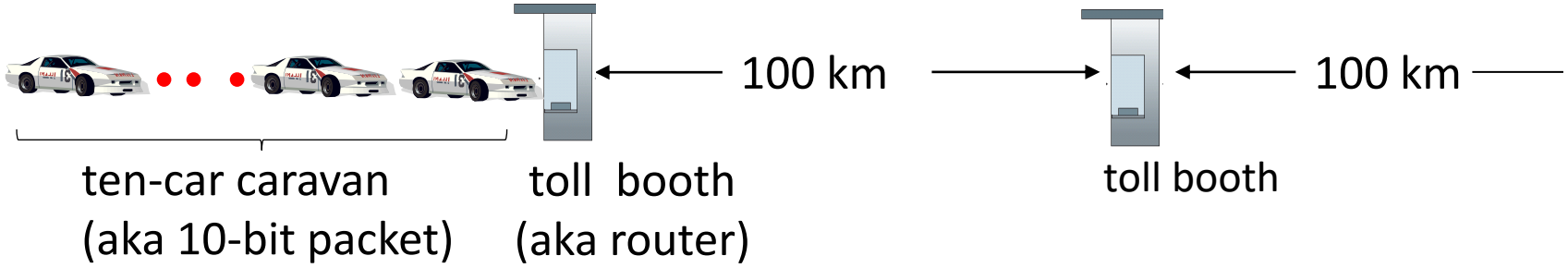
- L: packet length (bits)
- R: link *transmission rate (bps)*
- $d_{trans} = L/R$



d_{prop}: propagation delay (传播时延):

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

Caravan analogy

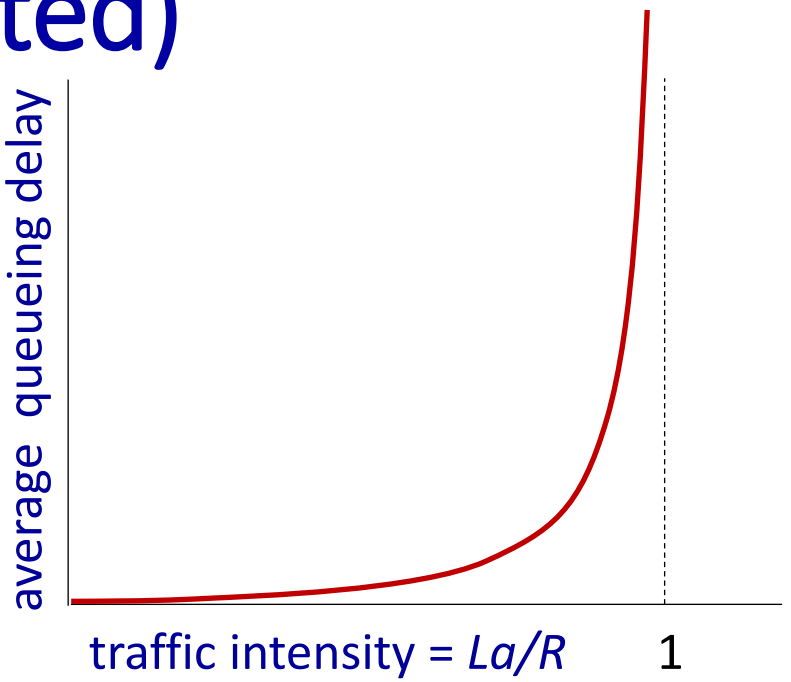


- cars "propagate" at 100 km/hr
- toll booth takes 12 sec to service car (bit transmission time)
- car ~ bit; caravan ~ packet
- *Q:* How long until caravan is lined up before 2nd toll booth?

- time to "push" entire caravan through toll booth onto highway = 12*10 = 120 sec
- time for last car to propagate from 1st to 2nd toll both: 100km/(100km/hr) = 1 hr
- *A:* 62 minutes

Packet queueing delay (revisited)

- *R:* link bandwidth (bps)
- *L:* packet length (bits)
- *a:* average packet arrival rate
- Total arrival of bits: L*a
- Traffic intensity: La/R

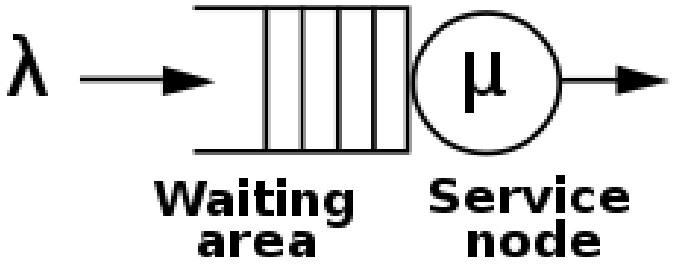


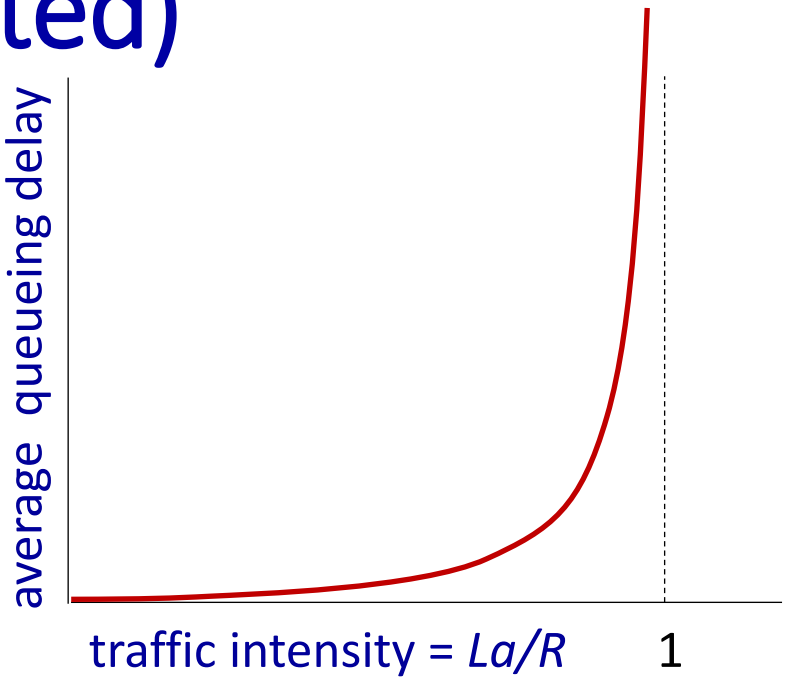
- *La/R* ~ 0: avg. queueing delay small
- *La/R* -> 1: avg. queueing delay large
- *La/R* > 1: more "work" arriving is more than can be serviced - average delay infinite!

La/R ~ 0

La/R -> 1

Packet queueing delay (revisited)





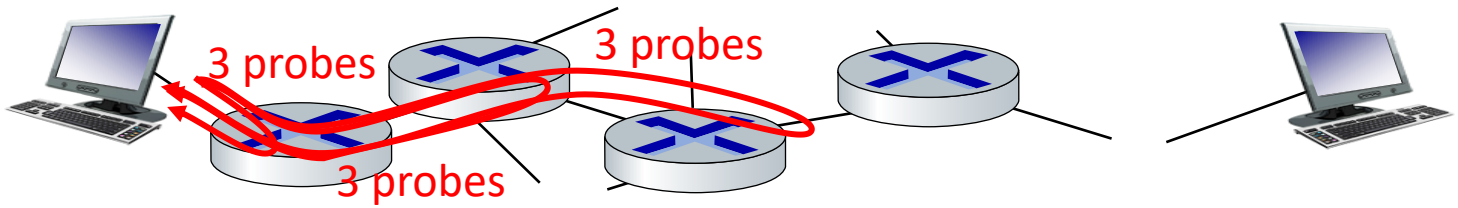
$$W(\lambda) = \frac{1}{\mu - \lambda}, 0 \leq \lambda \leq \mu$$

La/R ~ 0

La/R -> 1

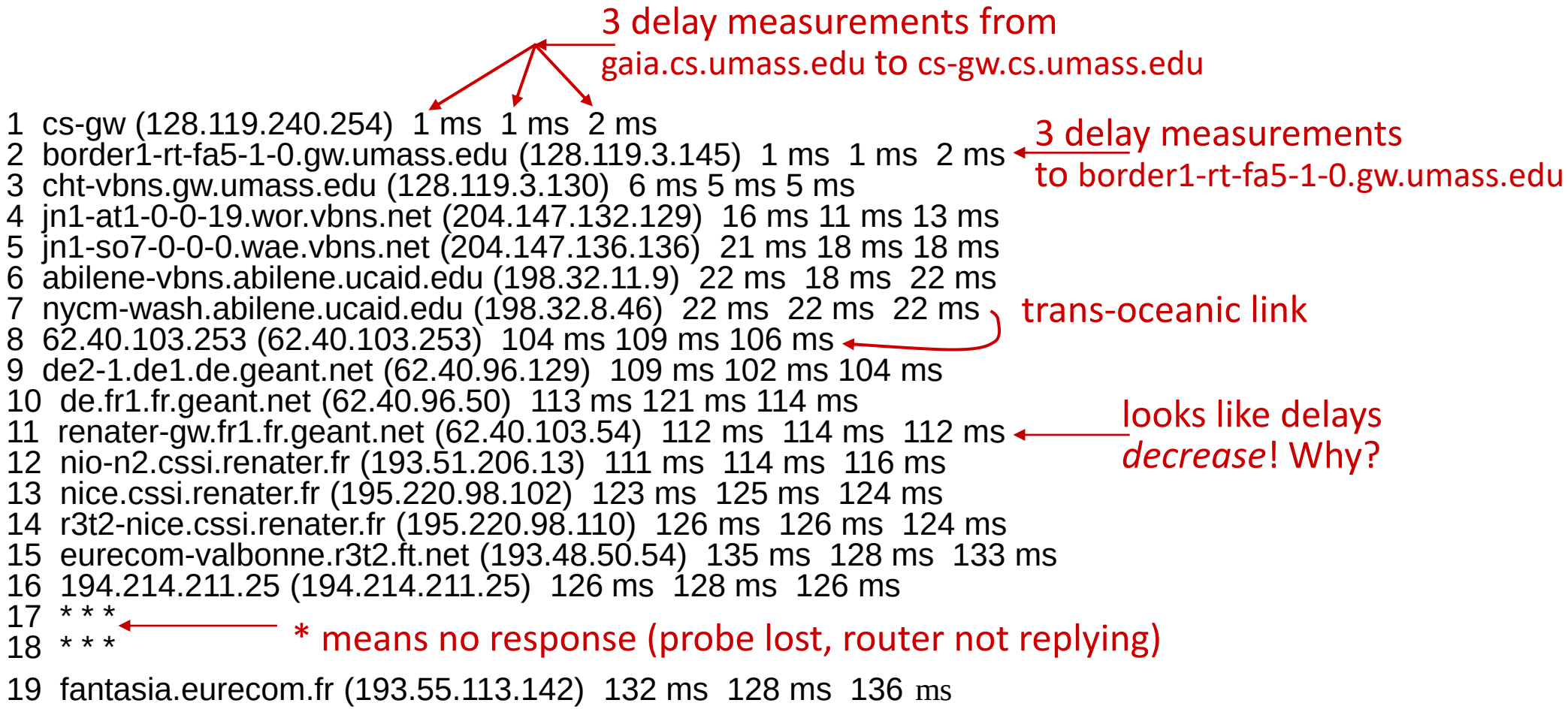
“Real” Internet delays and routes

- 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 (router):*
  - 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

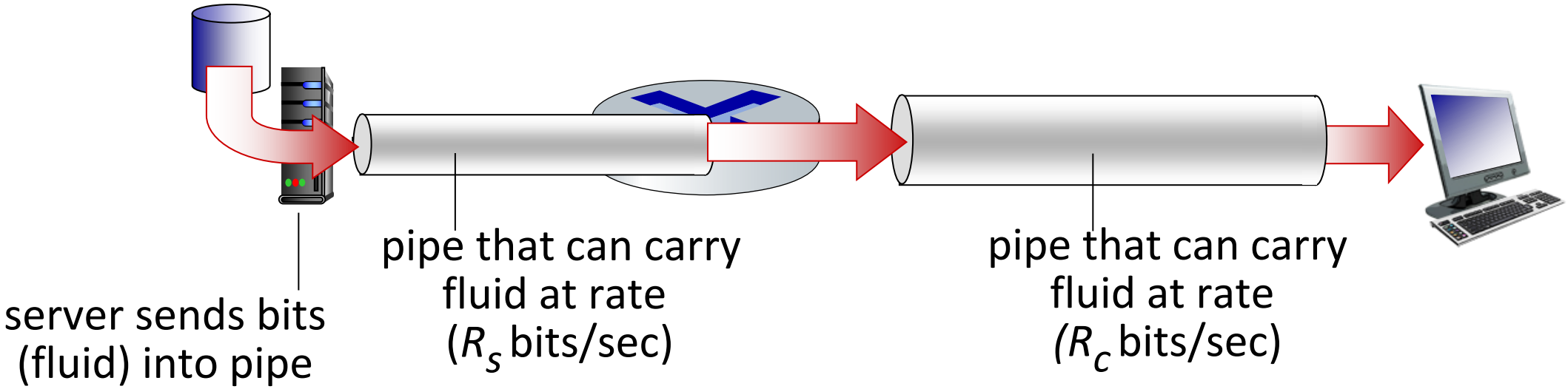
traceroute: gaia.cs.umass.edu to www.eurecom.fr



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

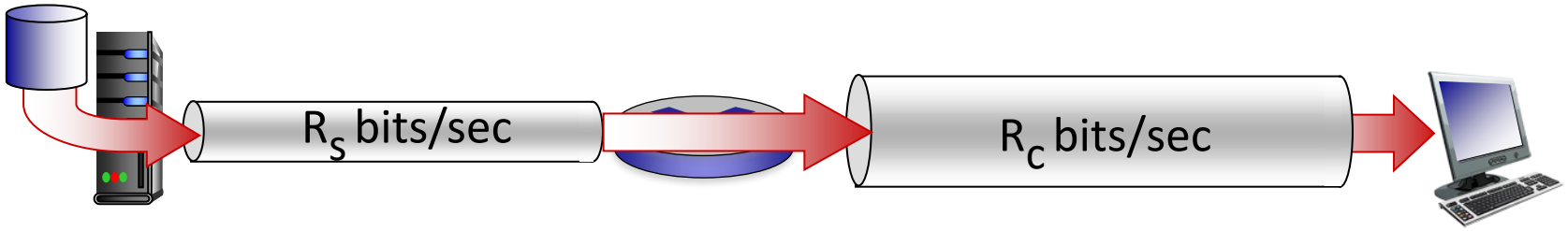
Throughput

- *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 a long period of time

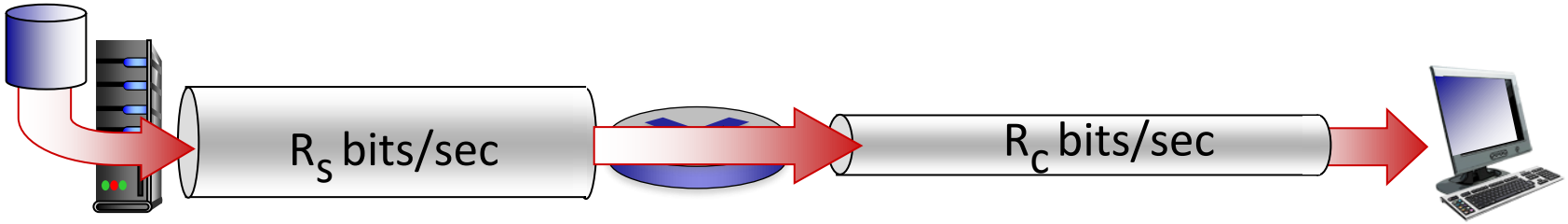


Throughput

$R_s < R_c$ 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

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
- Protocol layers, service models
- Performance: loss, delay, throughput
- Security
- History

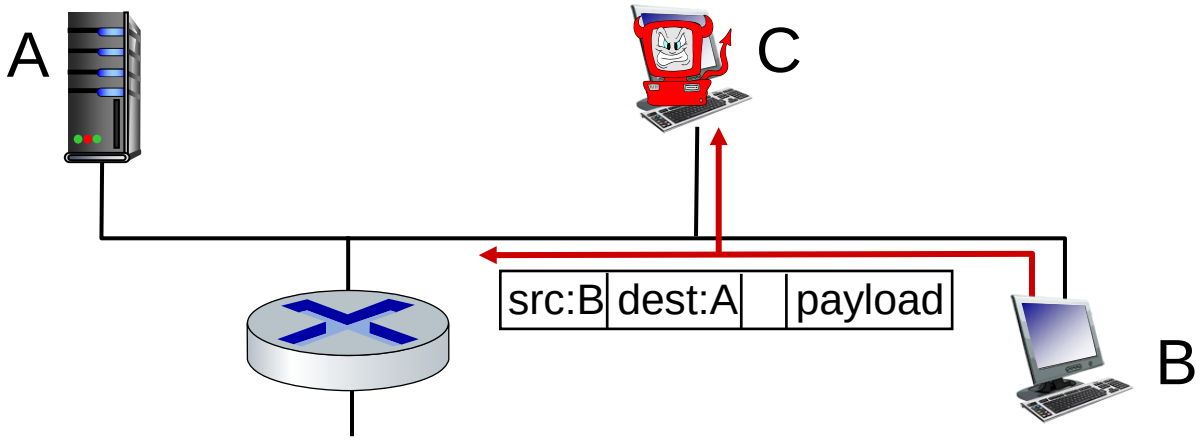
Network security

- 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: packet interception

packet "sniffing":

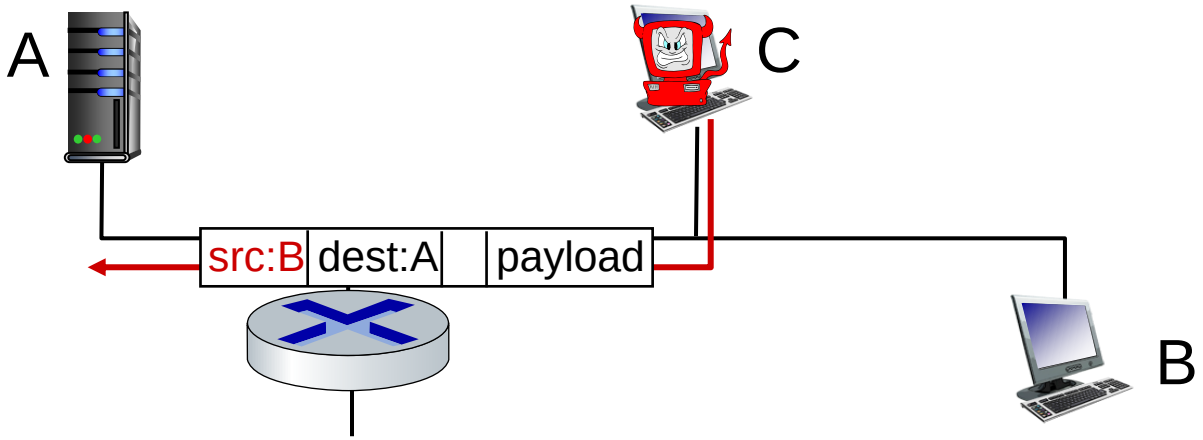
- broadcast media (shared Ethernet, wireless)
- promiscuous network interface reads/records all packets (e.g., including passwords!) passing by



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

Bad guys: fake identity

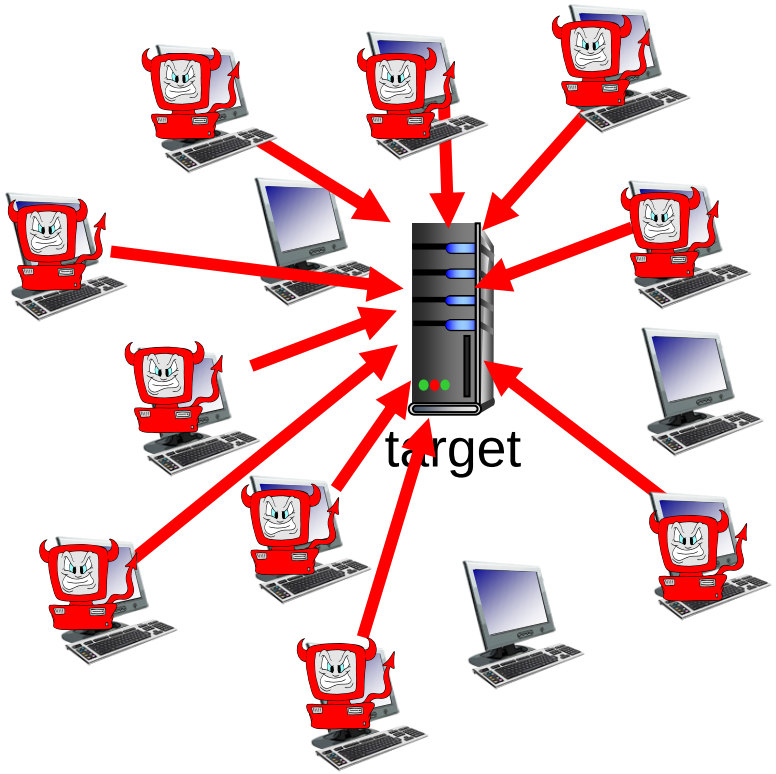
IP spoofing: send packet with false source address



Bad guys: denial of service

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



Lines of defense:

- authentication: proving you are who you say you are
  - cellular networks provides hardware identity via SIM card; no such hardware assist in traditional Internet
- confidentiality: via encryption
- integrity checks: digital signatures prevent/detect tampering
- access restrictions: password-protected VPNs
- firewalls: specialized "middleboxes" in access and core networks:
  - off-by-default: filter incoming packets to restrict senders, receivers, applications
  - detecting/reacting to DOS attacks

… lots more on security (throughout, Chapter 8)

Chapter 1: roadmap

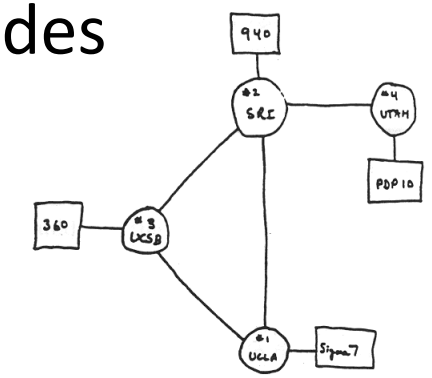
- What *is* the Internet?
- What *is* a protocol?
- Network edge: hosts, access network, physical media
- Network core: packet/circuit switching, internet structure
- Protocol layers, service models
- Performance: loss, delay, throughput
- Security
- History

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 routing
- decentralized control

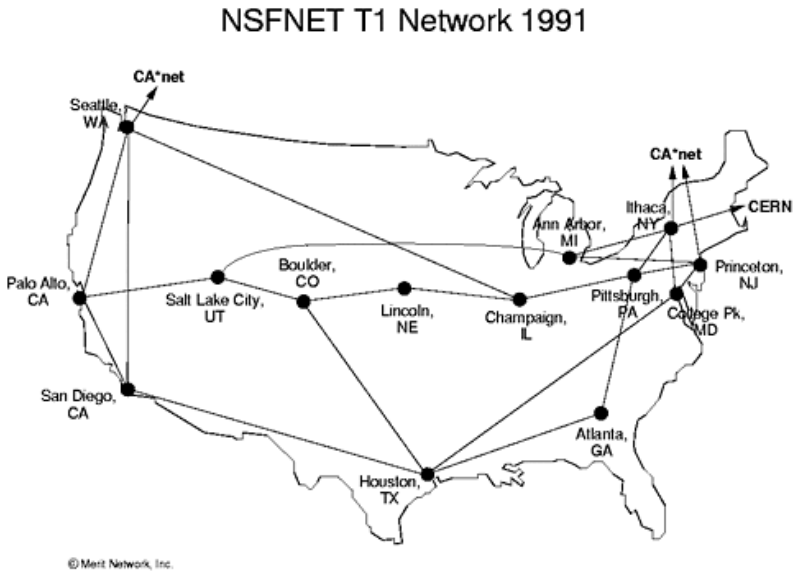
define today's Internet architecture

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, 2000s: commercialization, the Web, new applications

- early 1990s: 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 1990s: commercialization of the Web

late 1990s – 2000s:

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

Internet history

2005-present: more new applications, Internet is "everywhere"

- ~18B devices attached to Internet (2017)
  - rise of smartphones (iPhone: 2007)
- aggressive deployment of broadband access
- increasing ubiquity of high-speed wireless access: 4G/5G, WiFi
- emergence of online social networks:
- Facebook: ~ 2.5 billion users
- service providers (Google, FB, Microsoft) create their own networks
  - bypass commercial Internet to connect "close" to end user, providing "instantaneous" access to search, video content, …
- enterprises run their services in "cloud" (e.g., Amazon Web Services, Microsoft Azure)

Chapter 1: summary

We've covered a "ton" of material!

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

You now have:

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