Chapter 1: Introduction – Part 1

Instructor: Zhuozhao Li

Lab: Qing 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

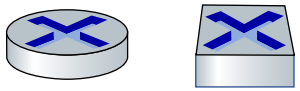
Overview/roadmap:

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



Packet switches: forward packets (chunks of data)

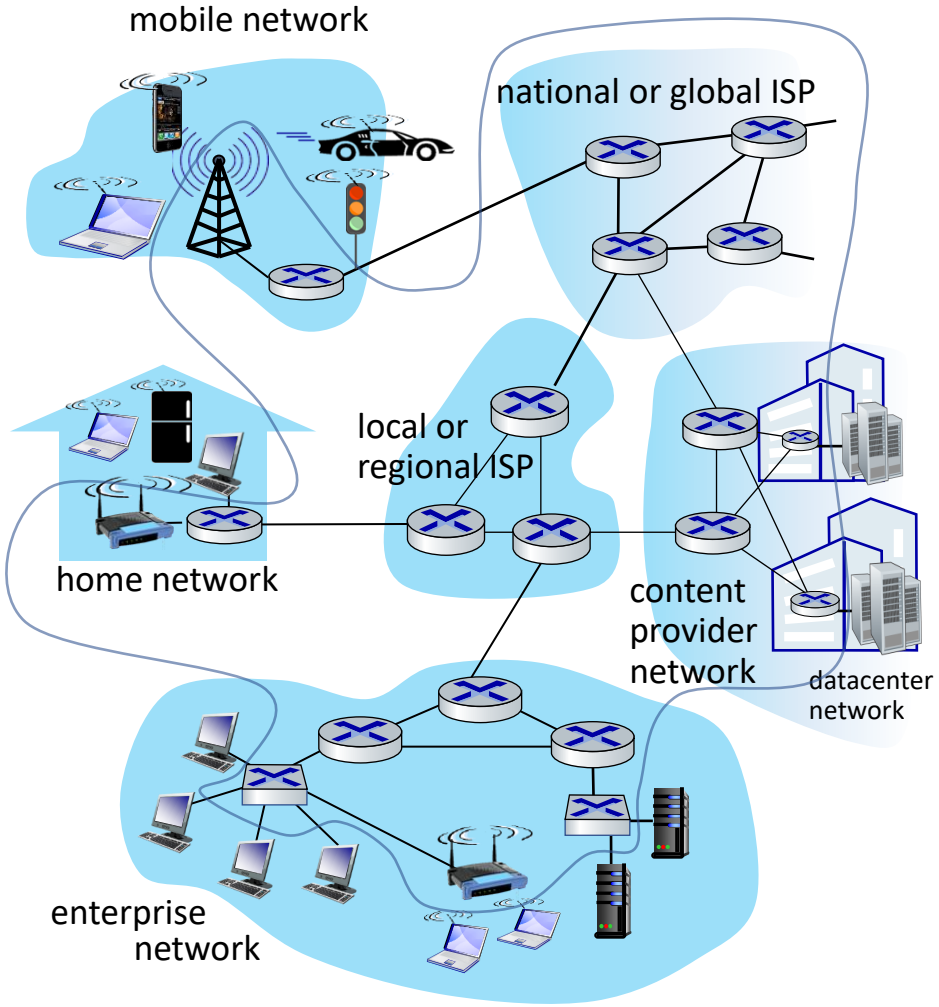
- *routers, switches*

Communication links

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

Networks

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



“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

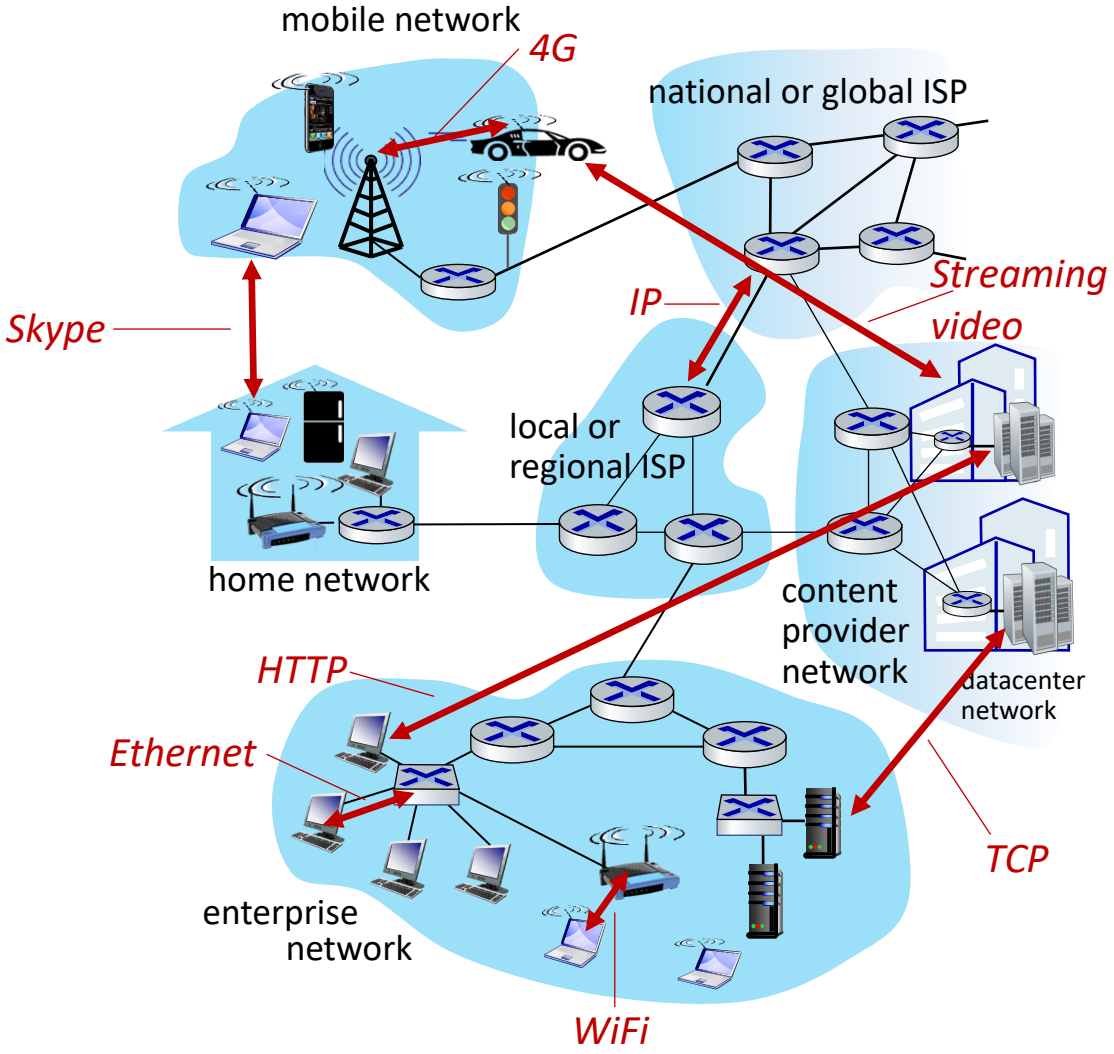
sensorized, bed mattress

Fitbit

Others?

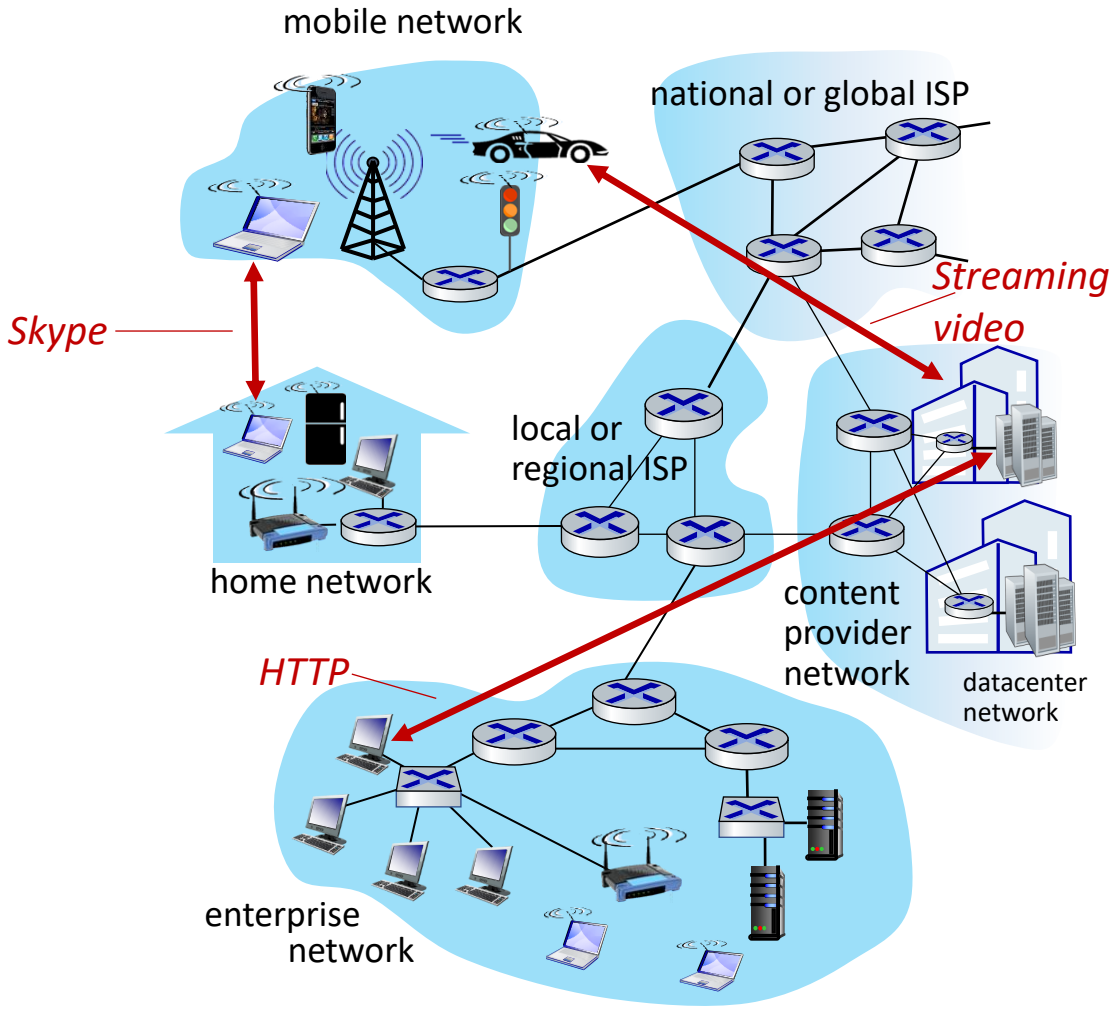
The Internet: a "nuts and bolts" view

- *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



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

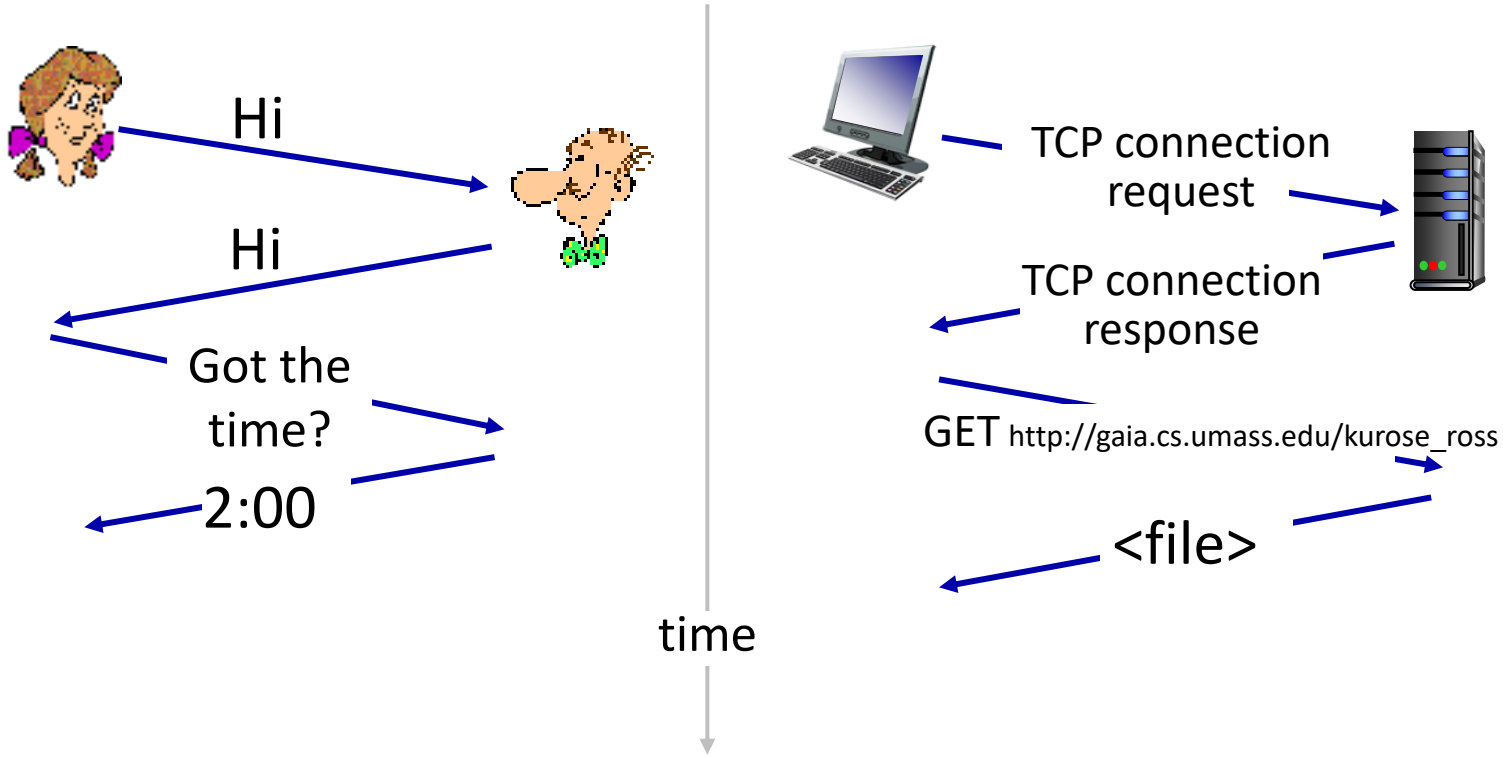
Network protocols:

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

Protocol “layers” and reference models

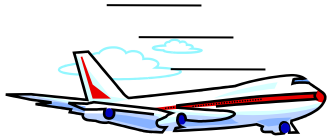
Networks are complex, with many “pieces”:

- 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

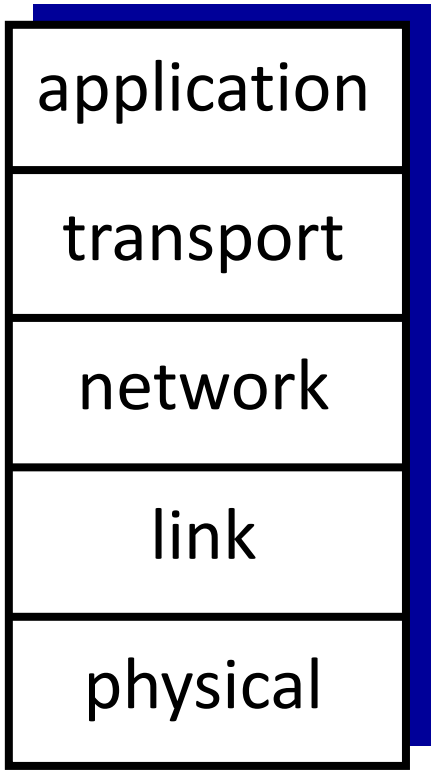
Why layering?

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”

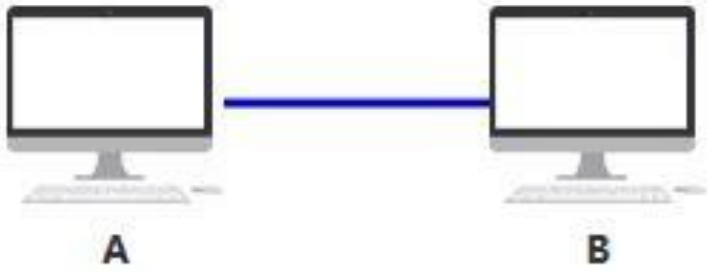


Internet protocol stack – an example

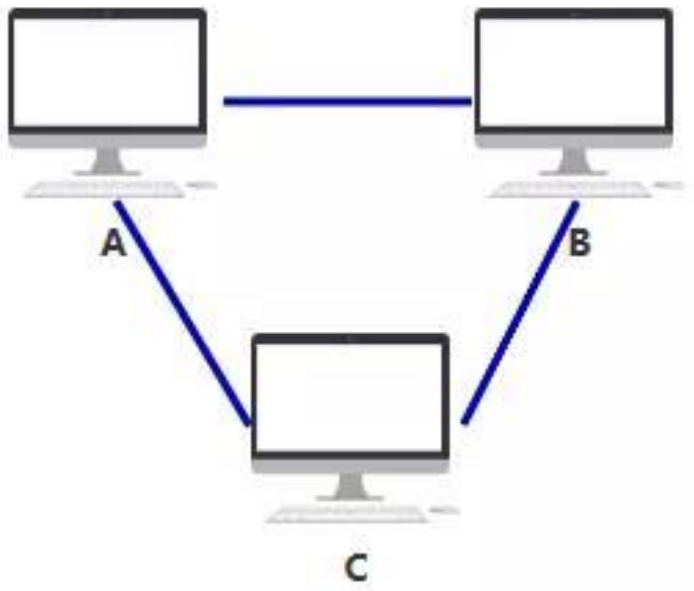
Source: 低并发编程 WeChat Channel
https://mp.weixin.qq.com/s/jiPMUk6zUdOY6eKxAjNDbQ

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

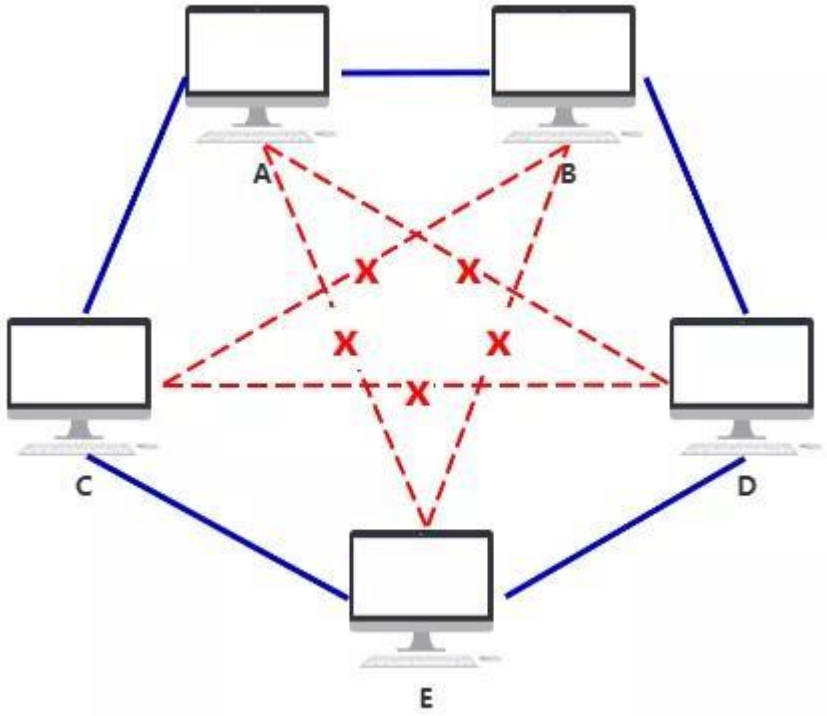
Internet protocol stack – an example



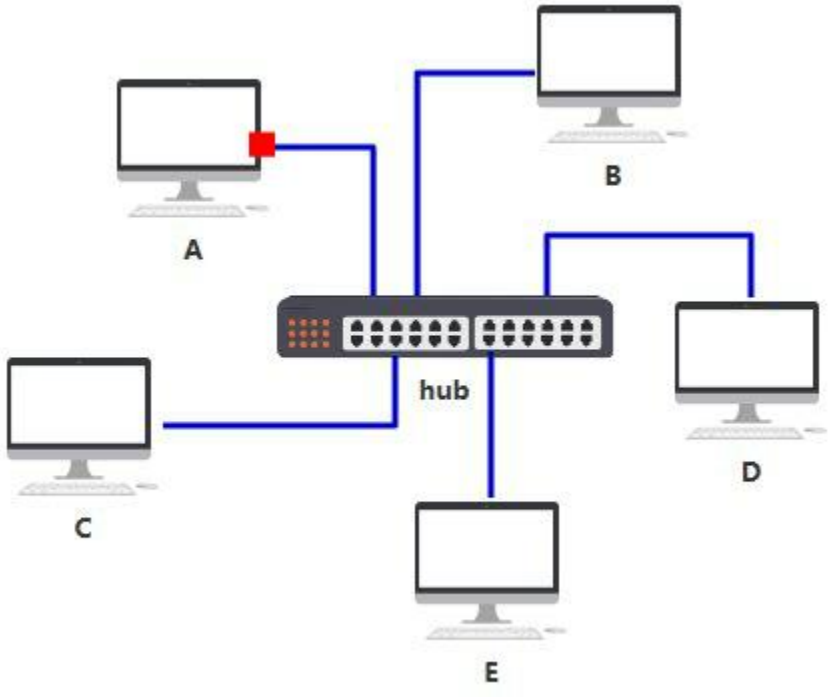
Internet protocol stack – an example



Internet protocol stack – an example



Internet protocol stack – an example



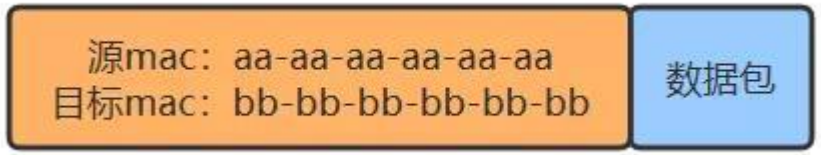
Hub (集线器)

- Broadcast
- Signal amplification and signal regeneration

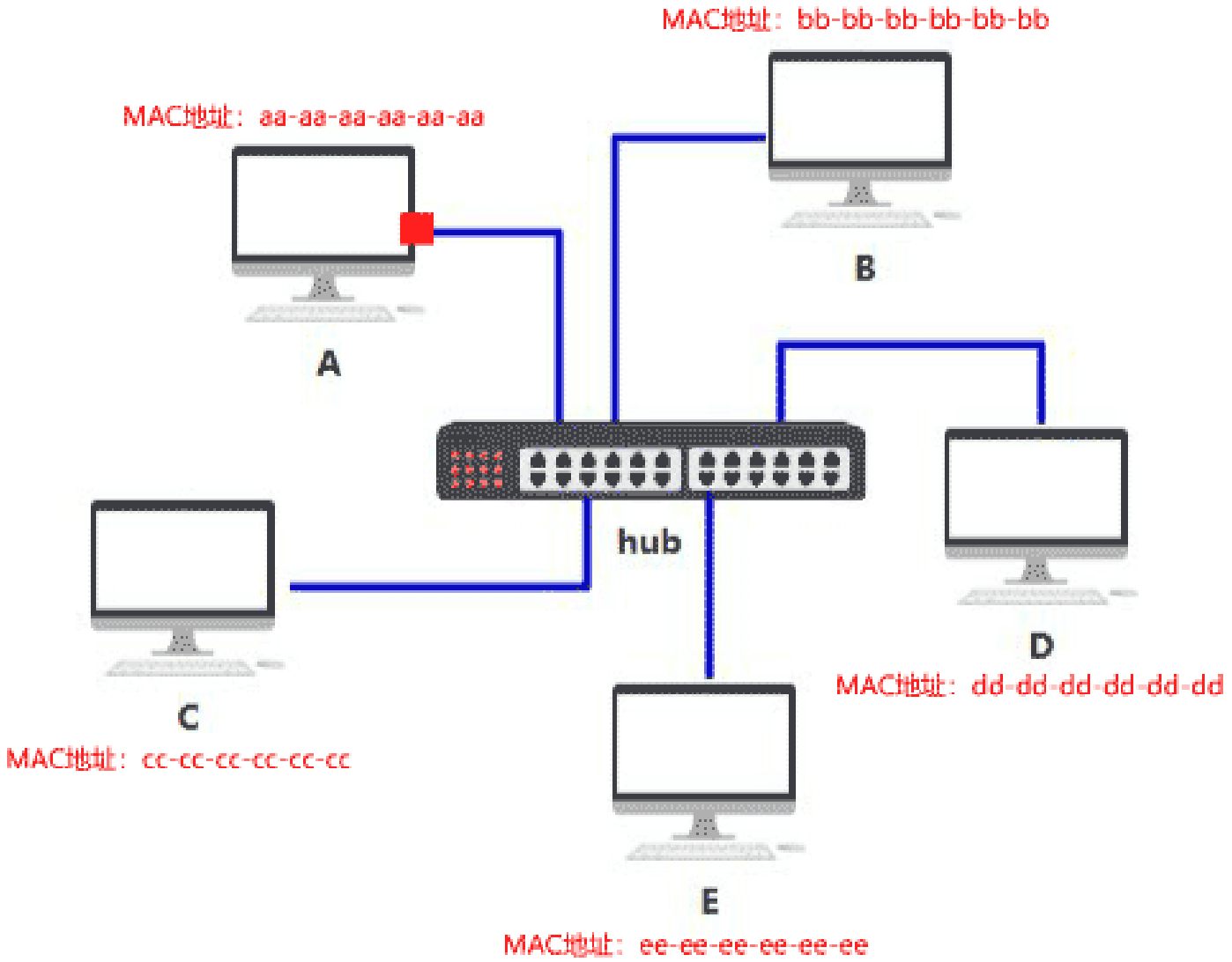
Summary of physical layer

- Bits “on the wire”

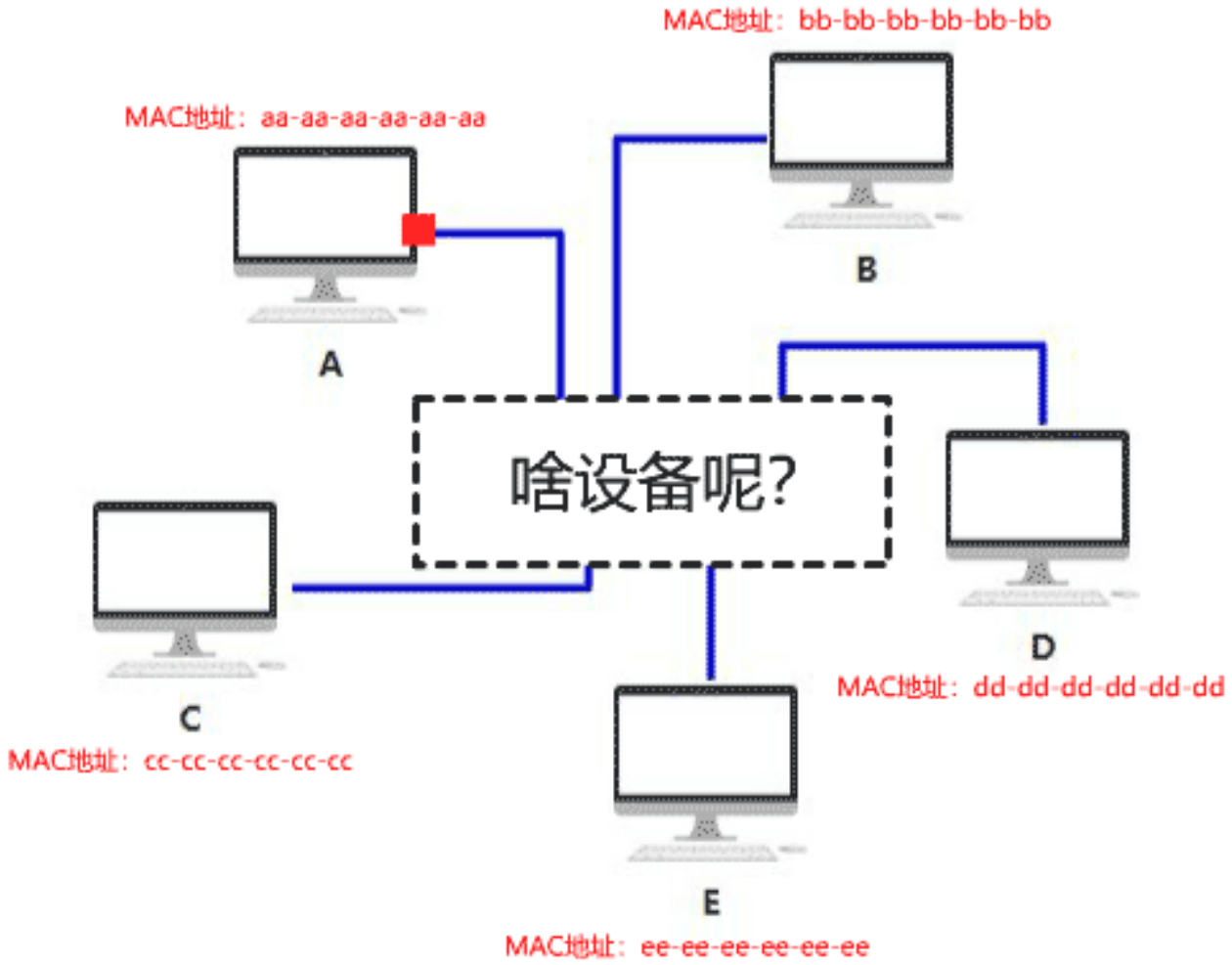
Internet protocol stack – an example



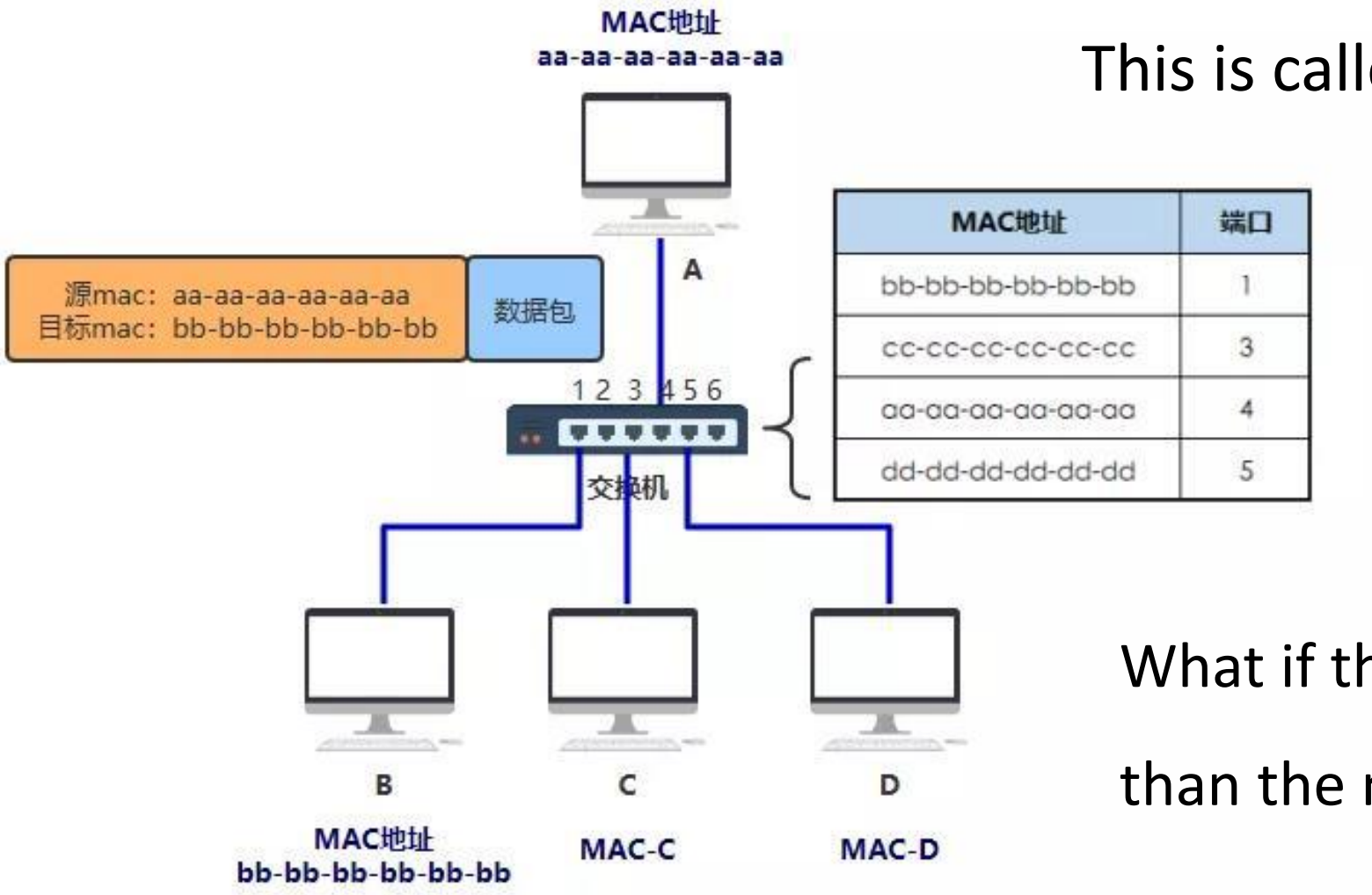
- Media Access Control (MAC) address
- Each network interface has a MAC address



Internet protocol stack – an example



Internet protocol stack – an example

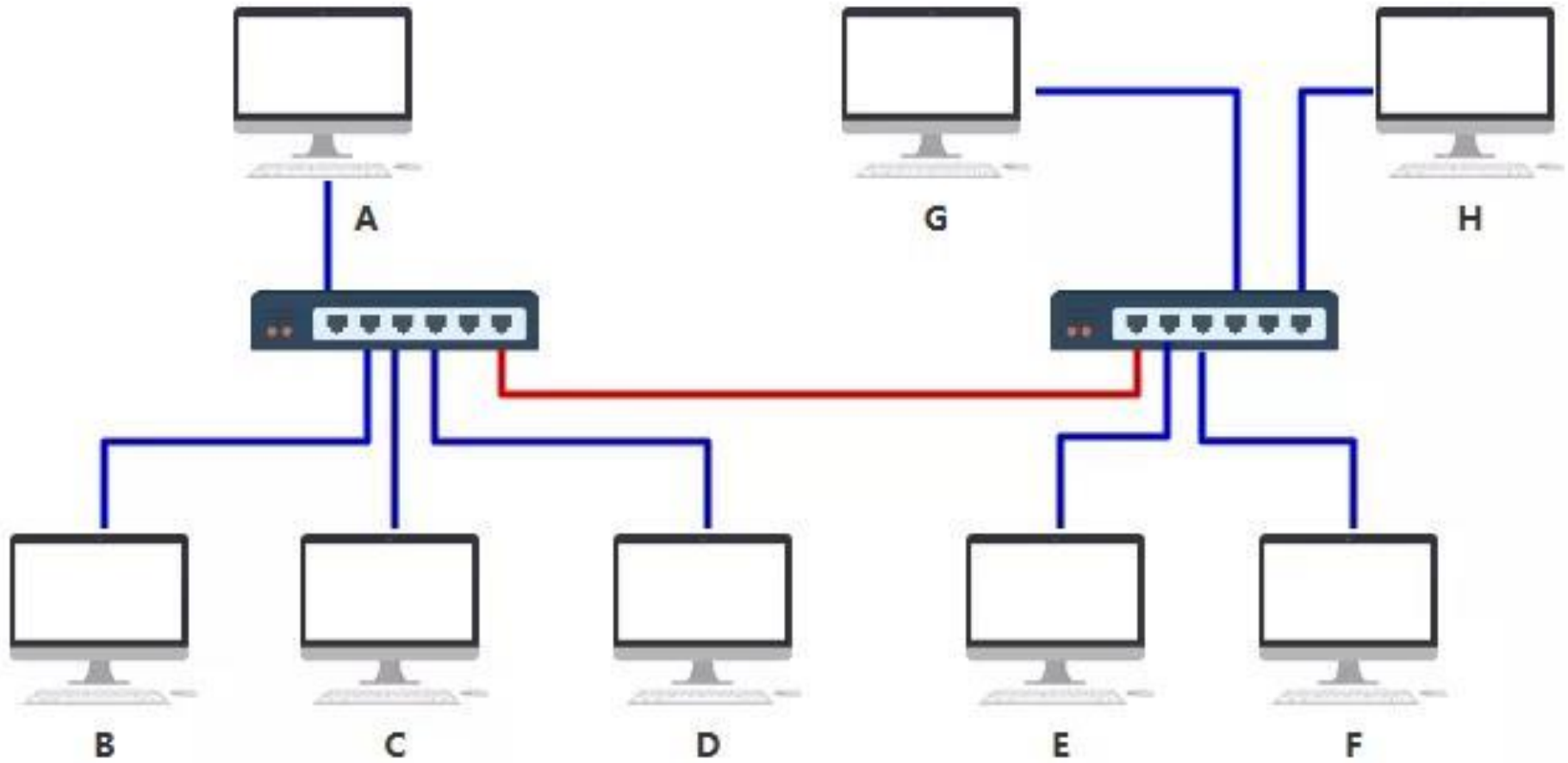


MAC地址	端口
bb-bb-bb-bb-bb-bb	1
cc-cc-cc-cc-cc-cc	3
aa-aa-aa-aa-aa-aa	4
dd-dd-dd-dd-dd-dd	5

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

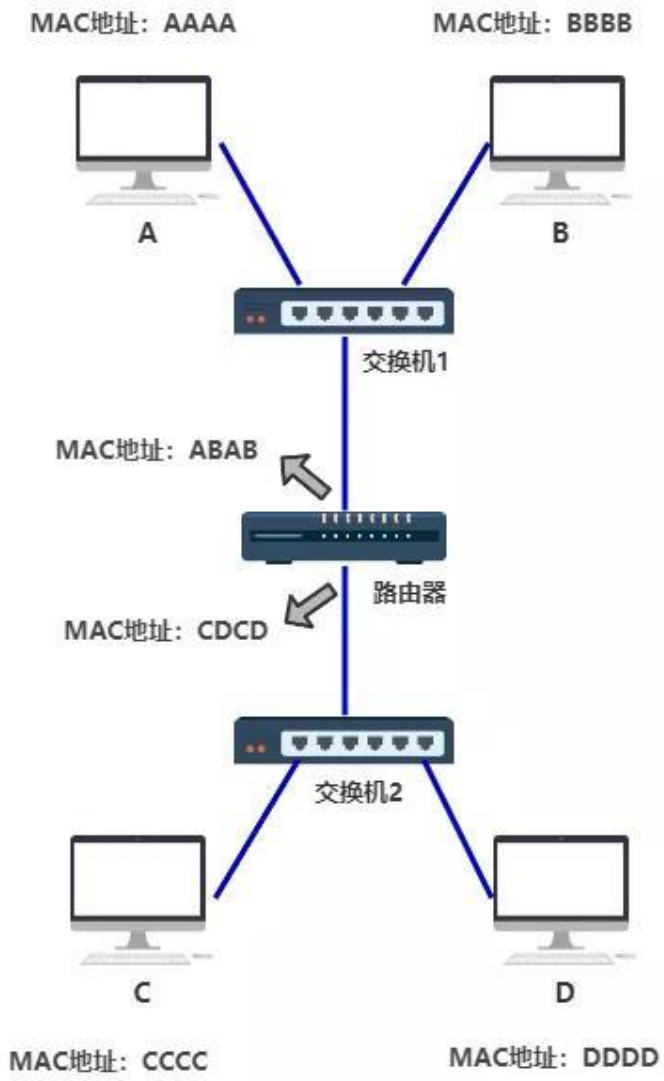
Summary of link layer

- 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

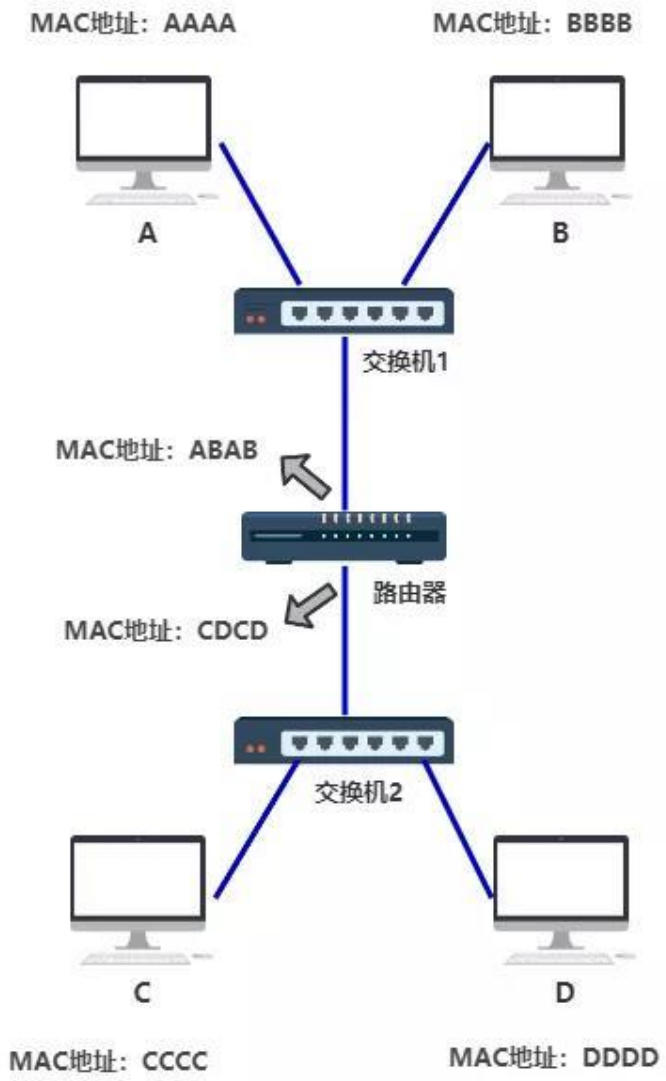
How about routing packets with the same MAC prefix the router?

For example:

C's MAC add: FF FFFF-CCCC

D's MAC add: FF FFFF-DDDD

It is hard and not quite practical



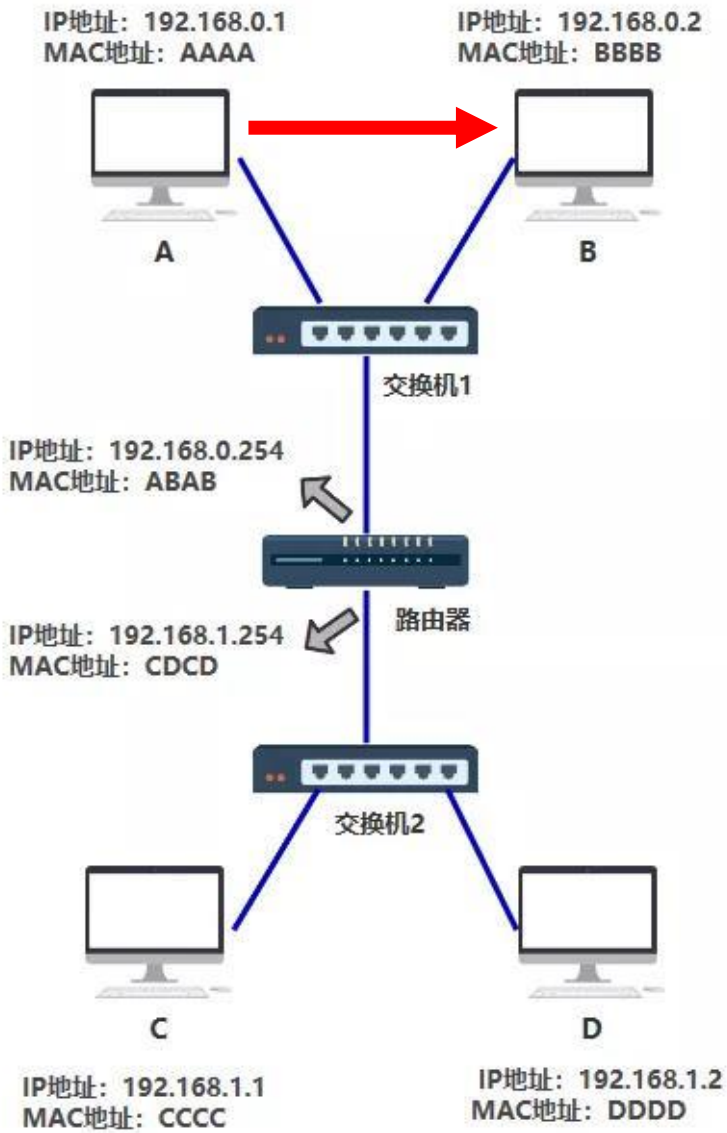
Internet protocol stack – an example

A new type of address

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

Internet protocol stack – an example

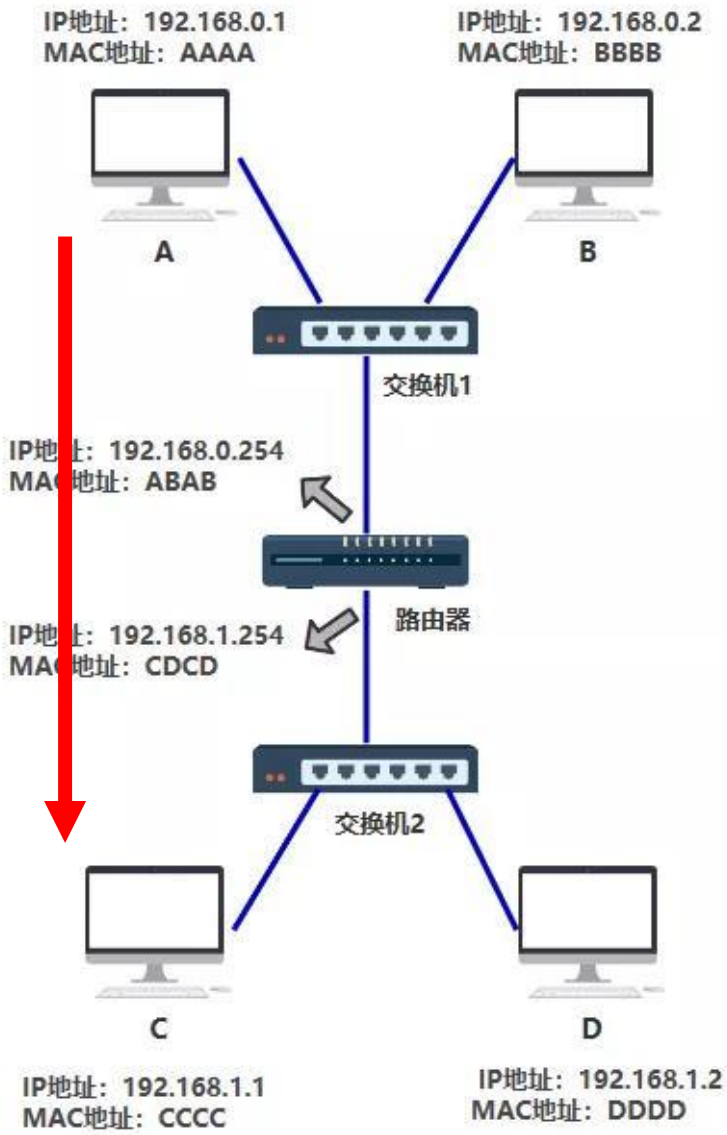
A to B:



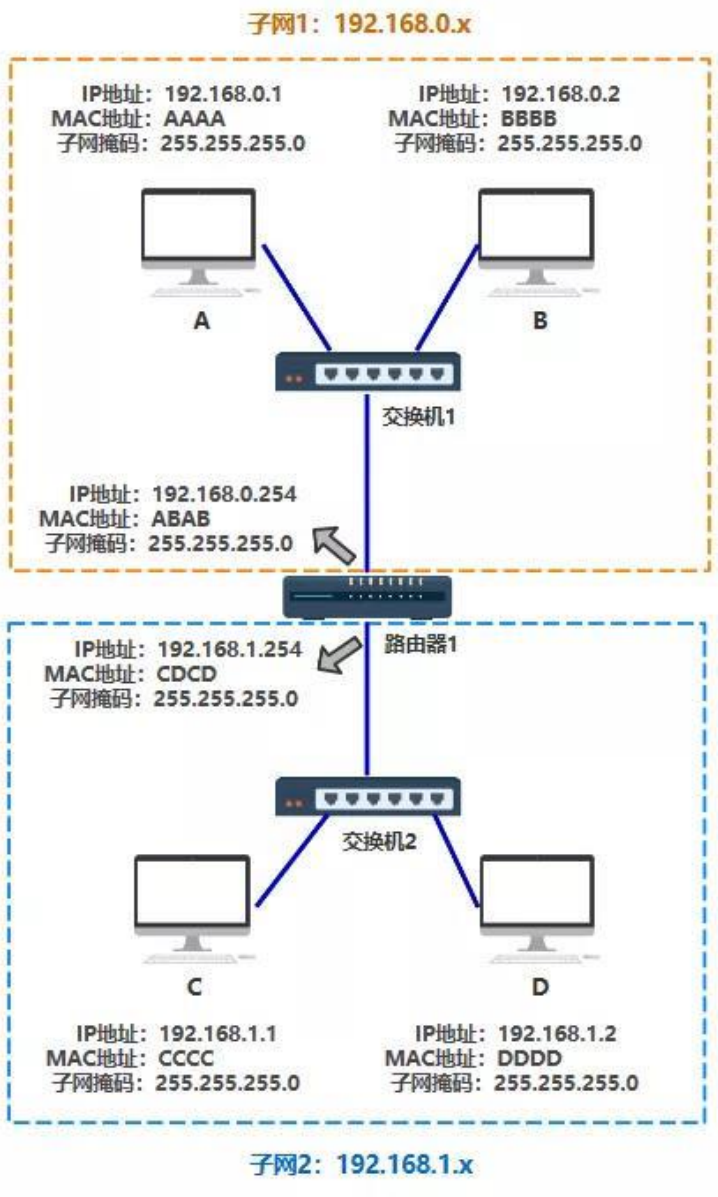
Internet protocol stack – an example

A to 路由器:

路由器 to C:



Internet protocol stack – an example



Subnet

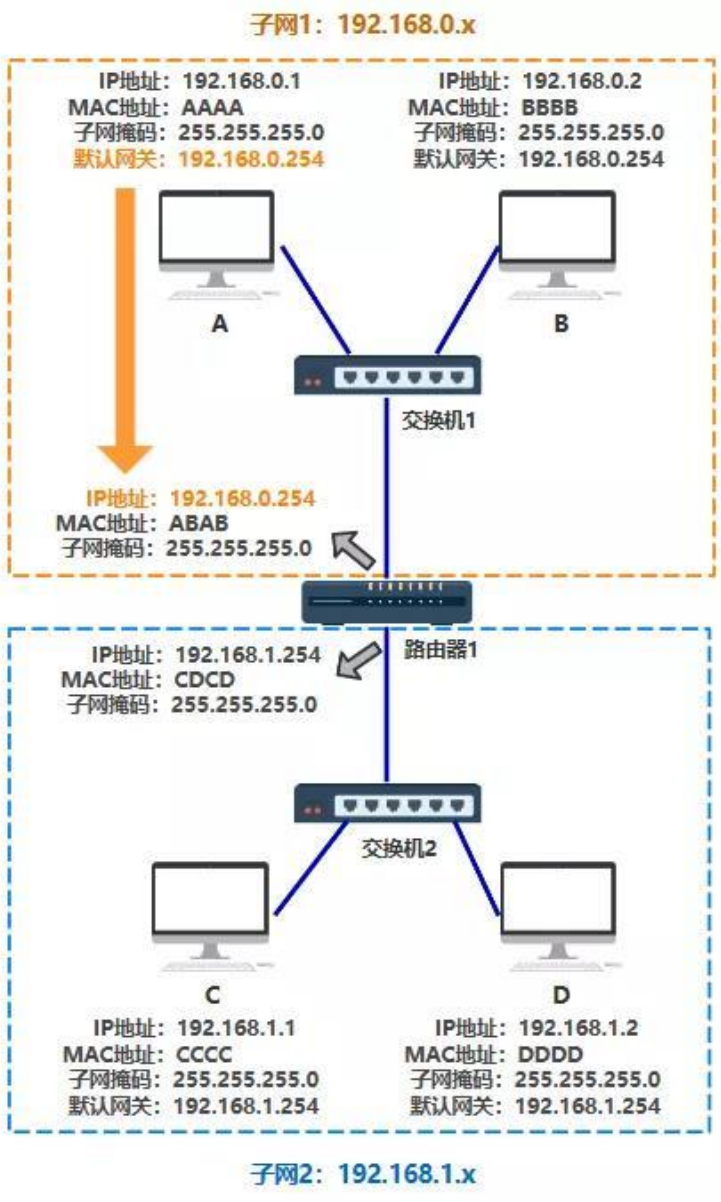
- 192.168.0.1 and 192.168.0.2: same subnet
- 192.168.0.1 and 192.168.1.1: different subnet

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



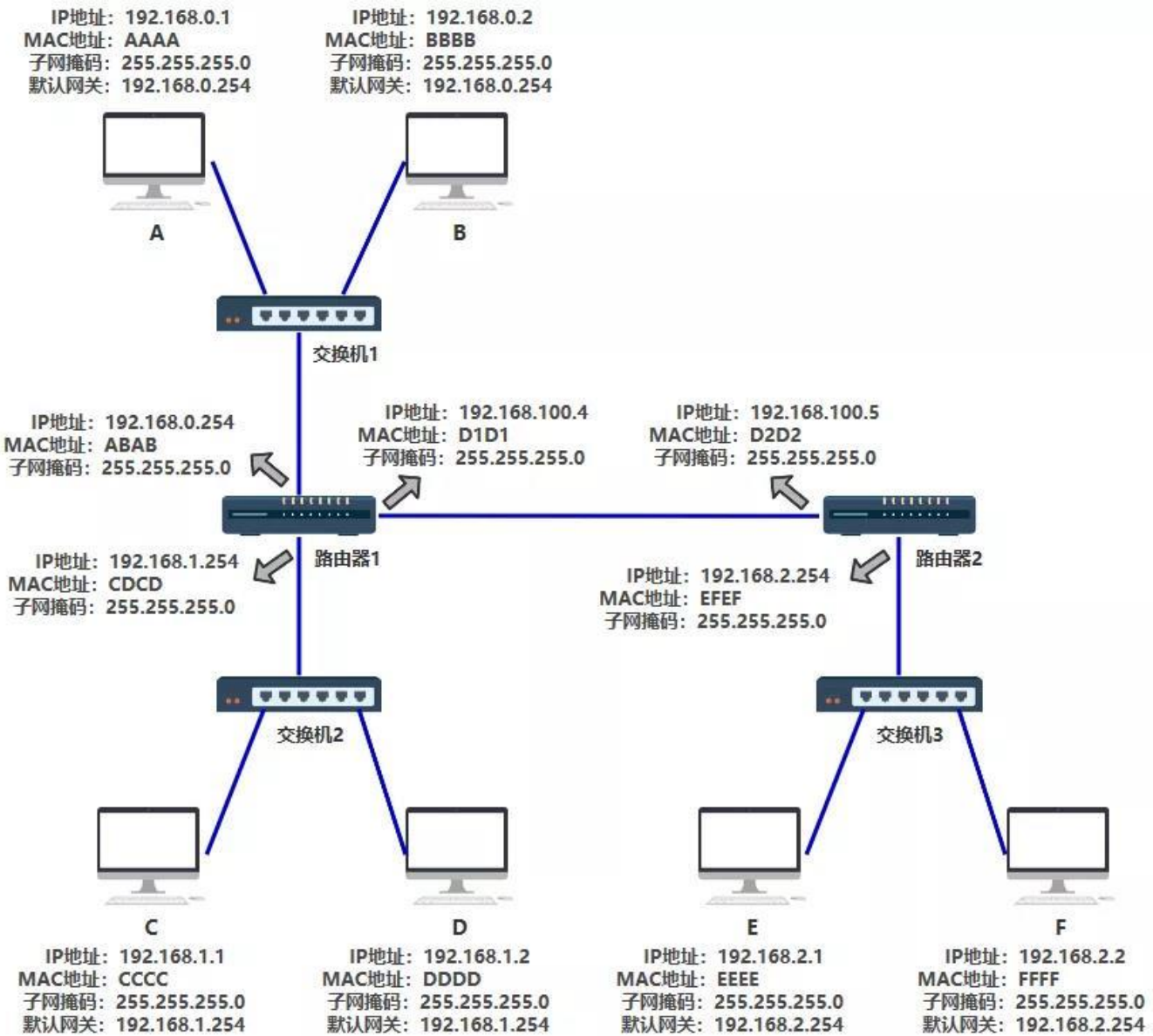
Gateway

- 192.168.0.254 is the router

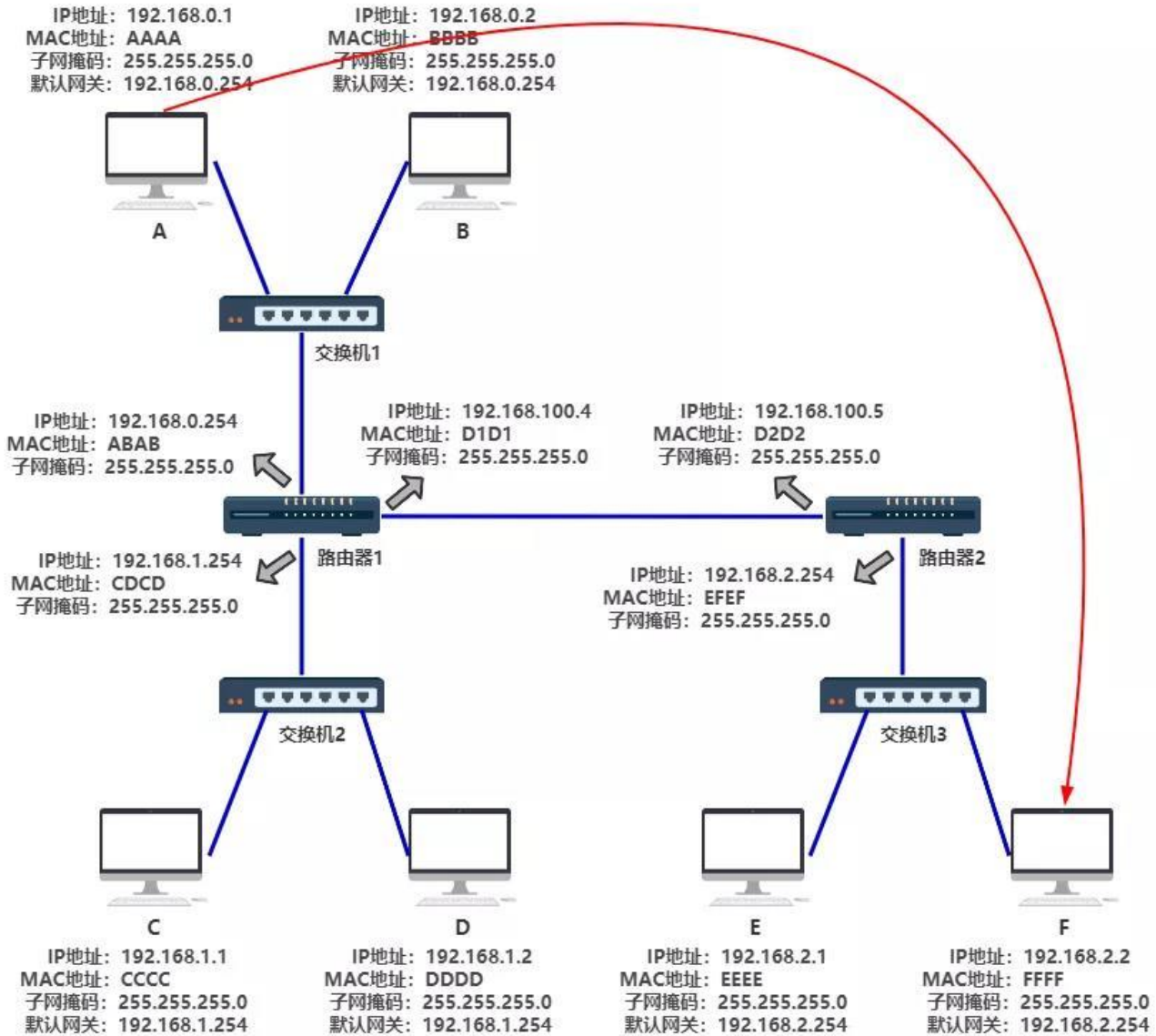
Summary of network layer

- Route datagram from source to destination

Internet protocol stack – an example



Internet protocol stack – an example



Transport and application layers

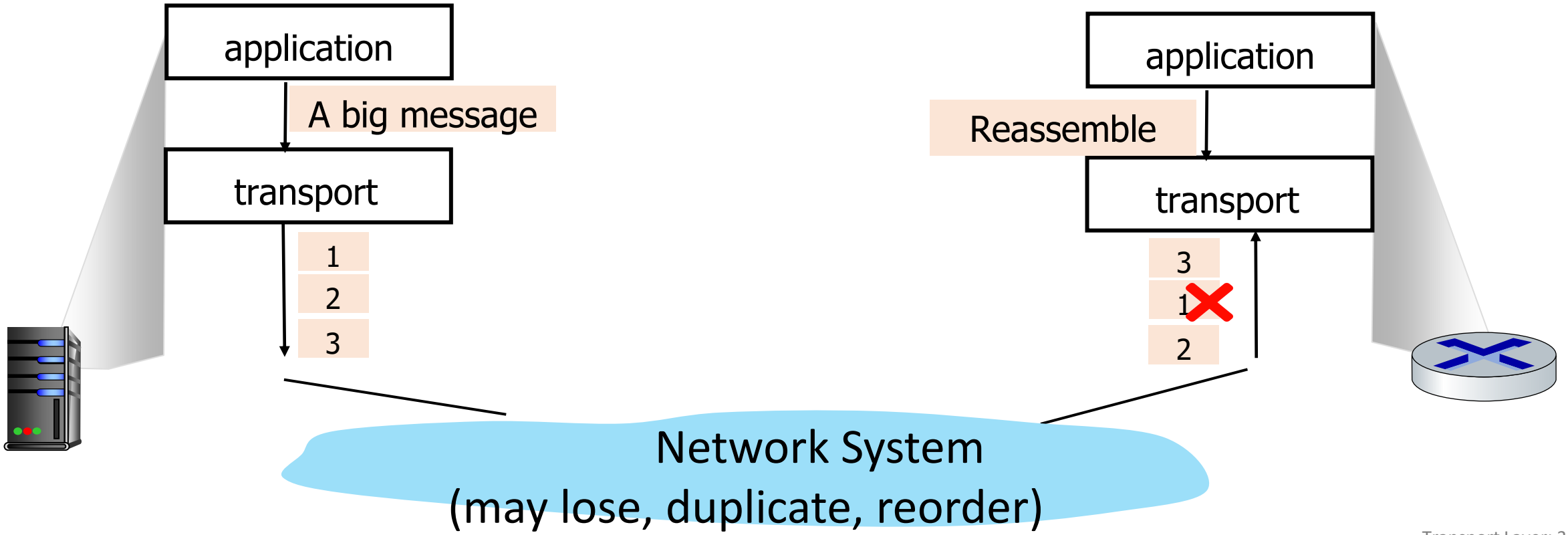
- Build on top of the bottom three layers
  - Use the functionalities of the bottom layers to provide services

- 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)

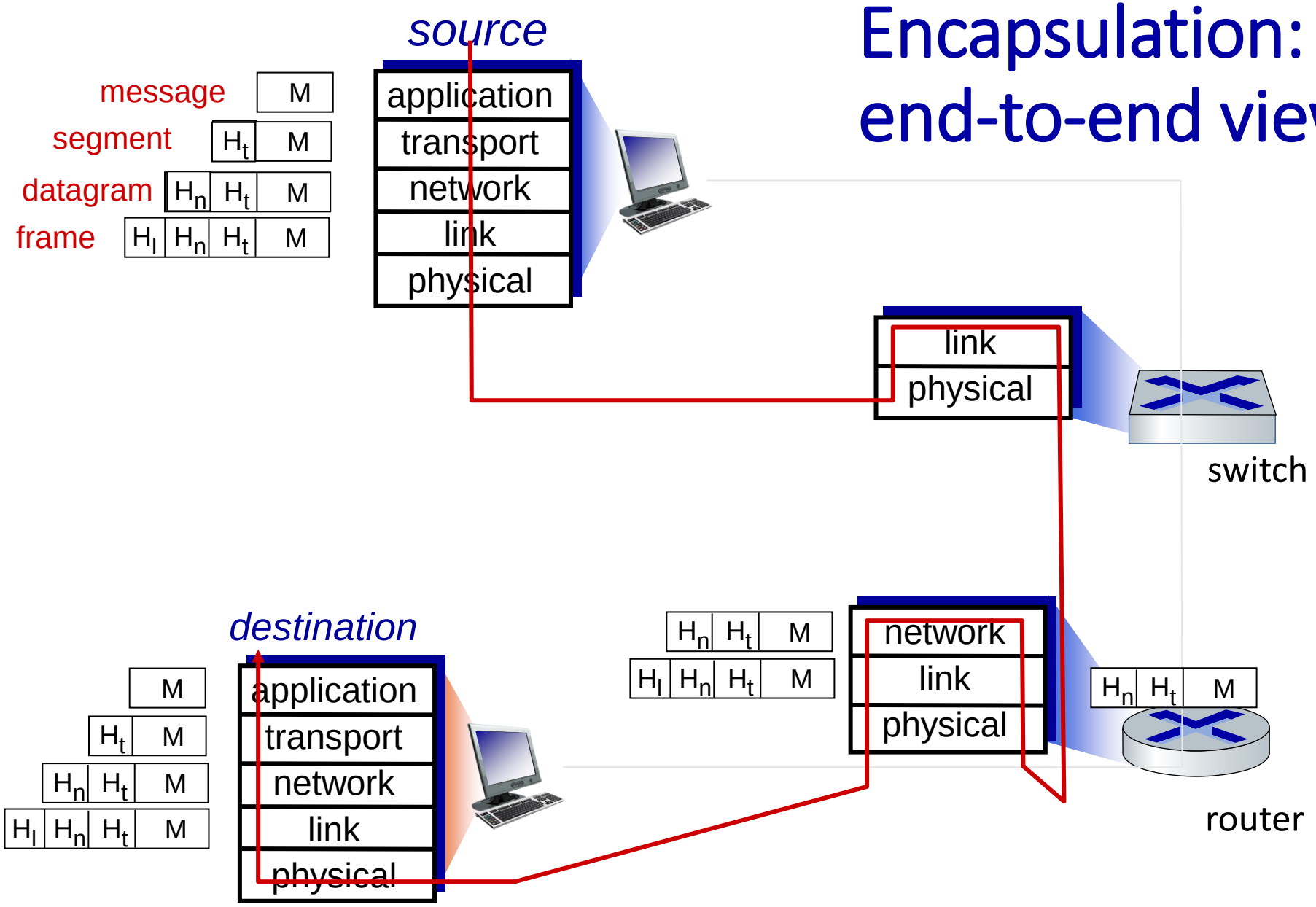


Notes on the example

- 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



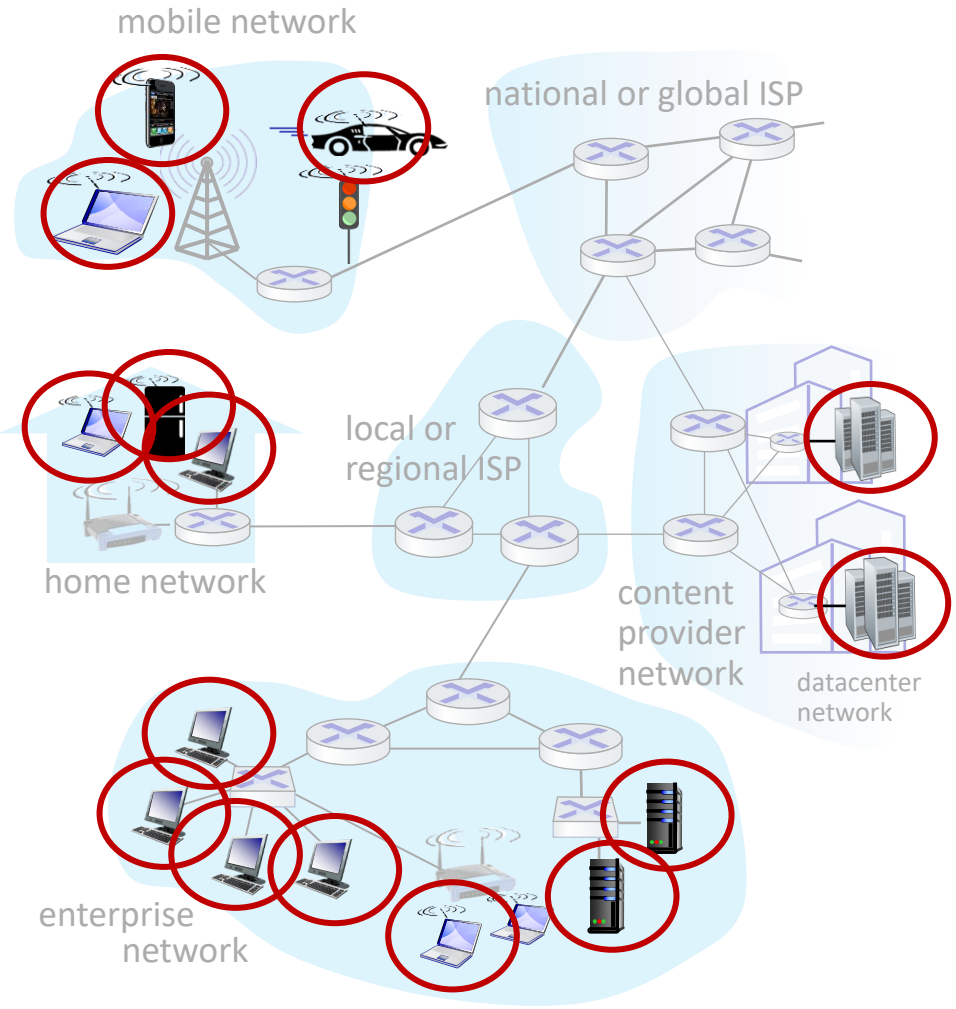
Chapter 1: roadmap

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

A closer look at Internet structure

Network edge:

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



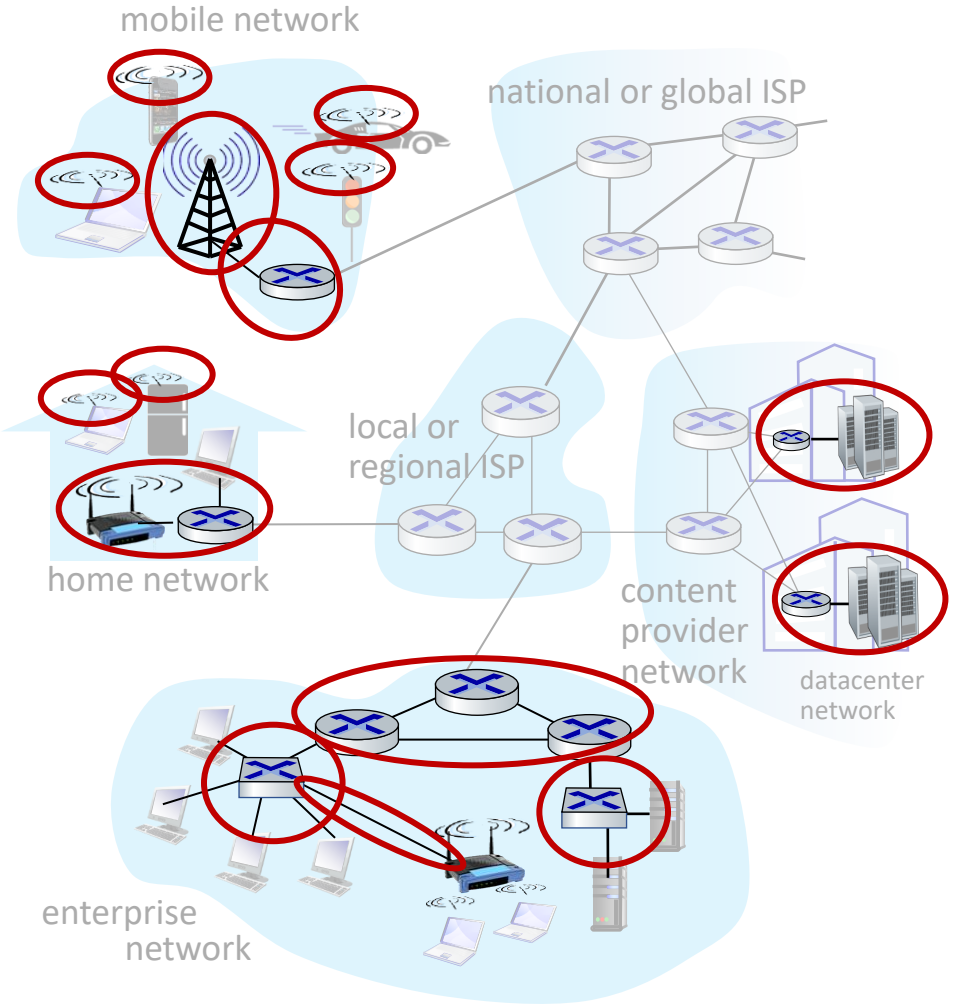
A closer look at Internet structure

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:

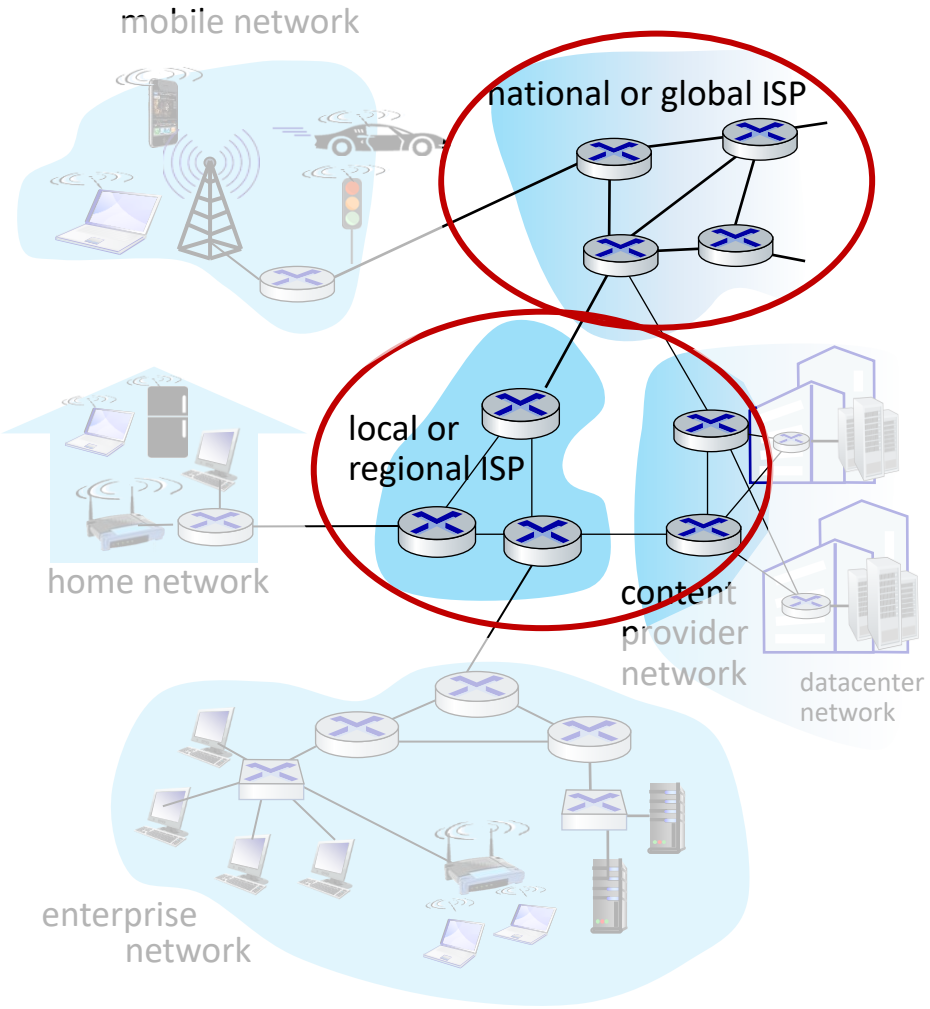
- hosts: clients and servers
- servers often in data centers

Access networks, physical media:

- wired, wireless communication links

Network core:

- interconnected routers
- network of networks



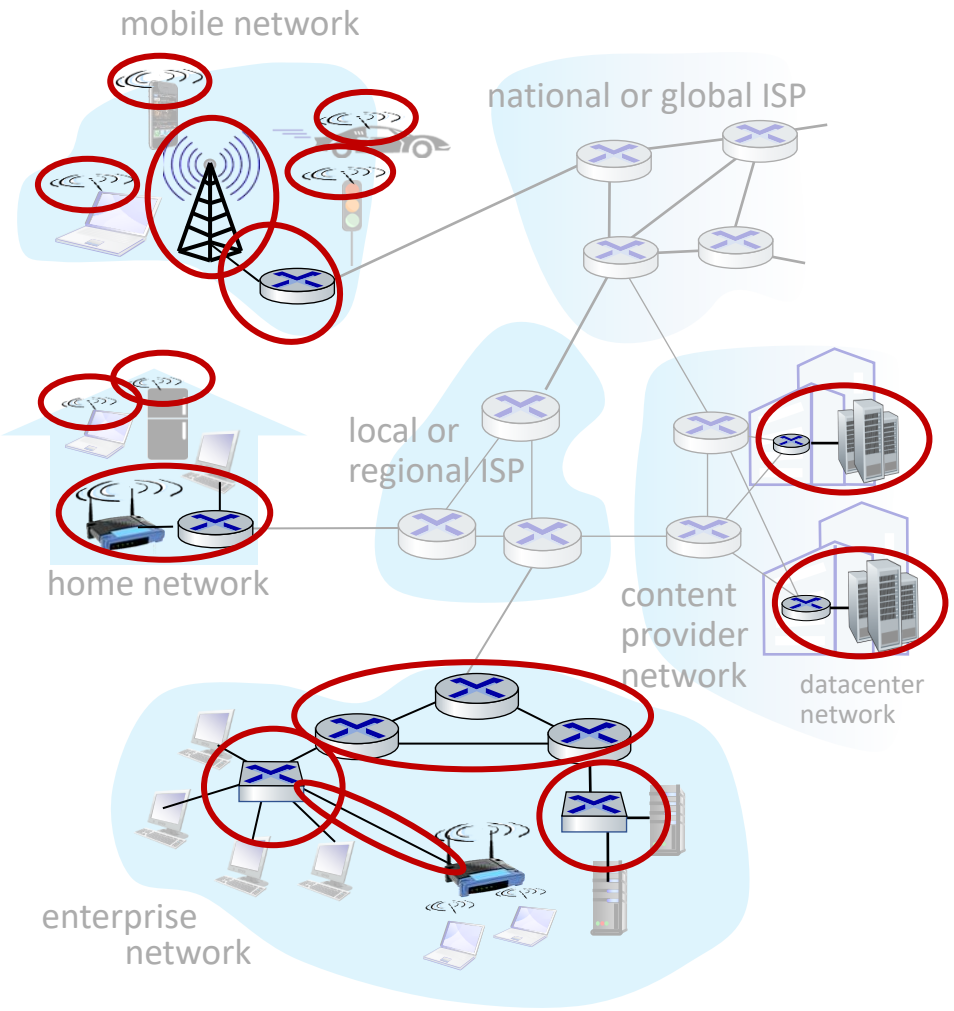
Access networks and physical media

Q: How to connect end systems to edge router?

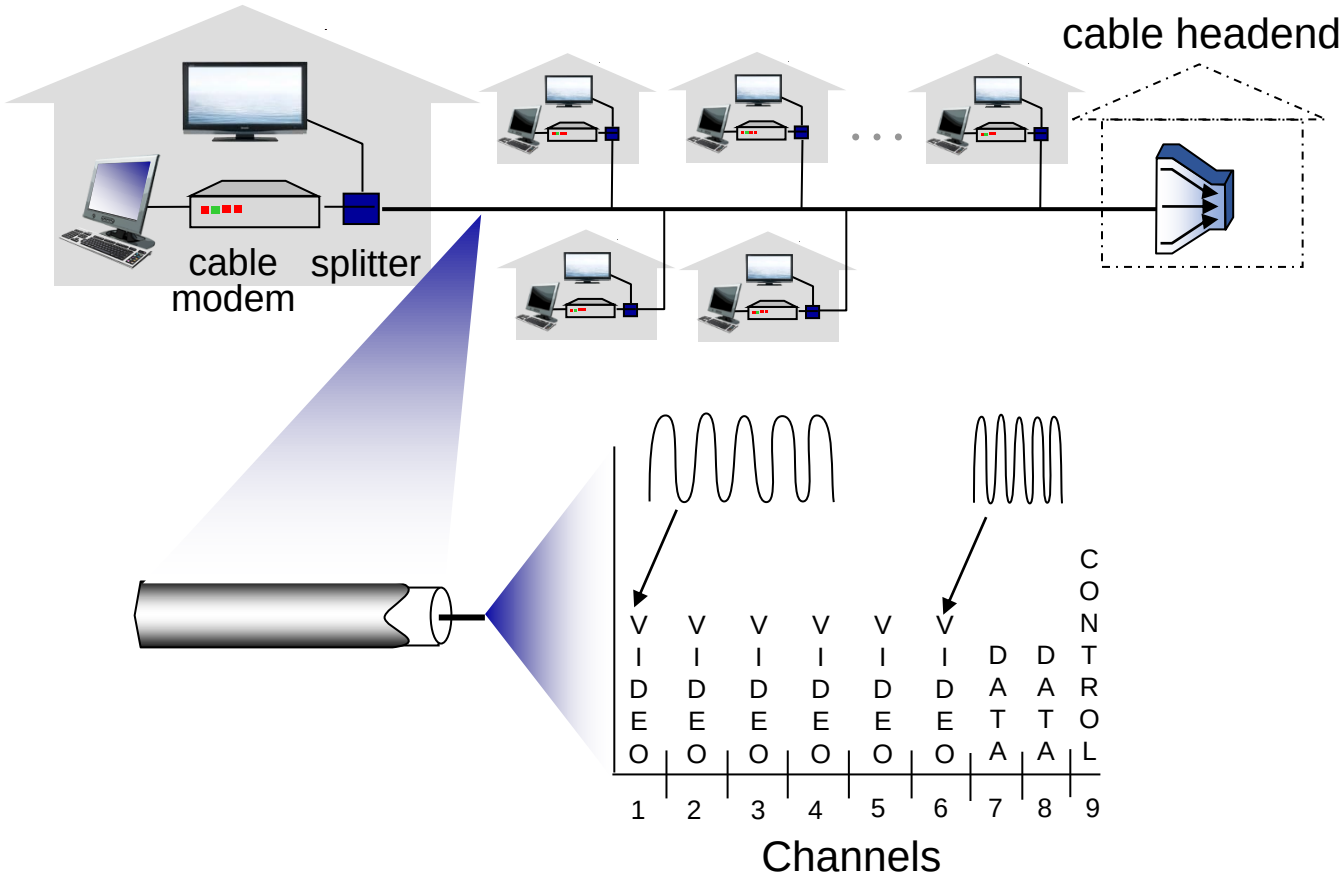
- residential access nets
- institutional access networks (school, company)
- mobile access networks (WiFi, 4G/5G)

What to look for:

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

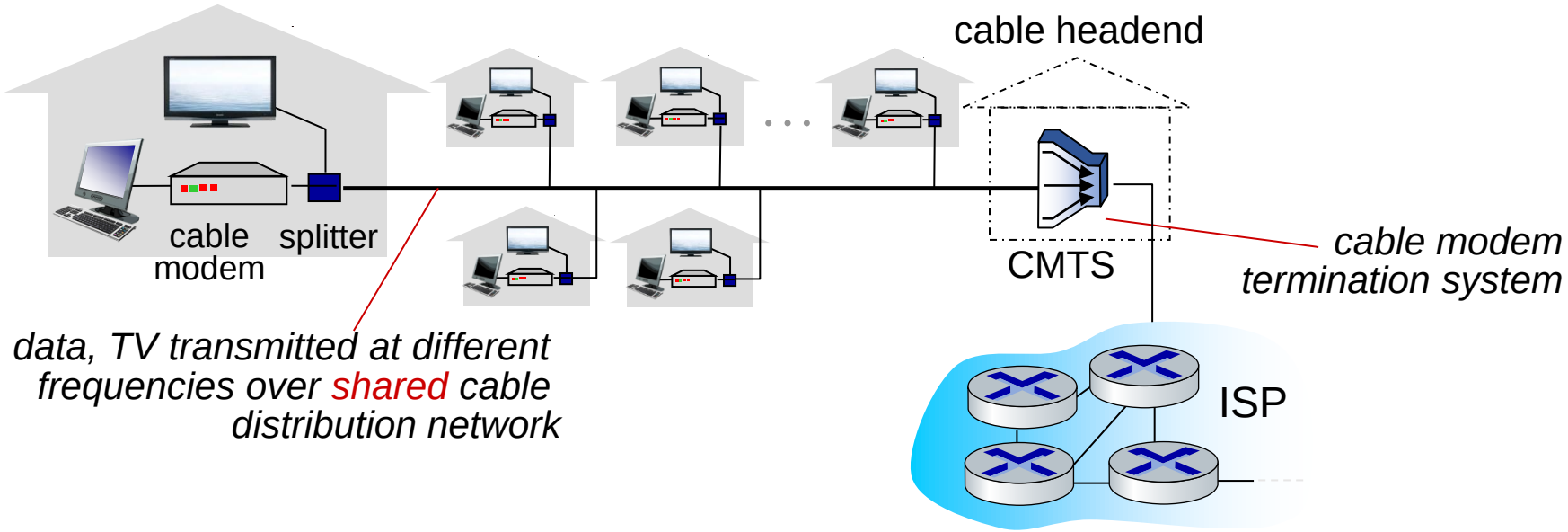


Access networks: cable-based access



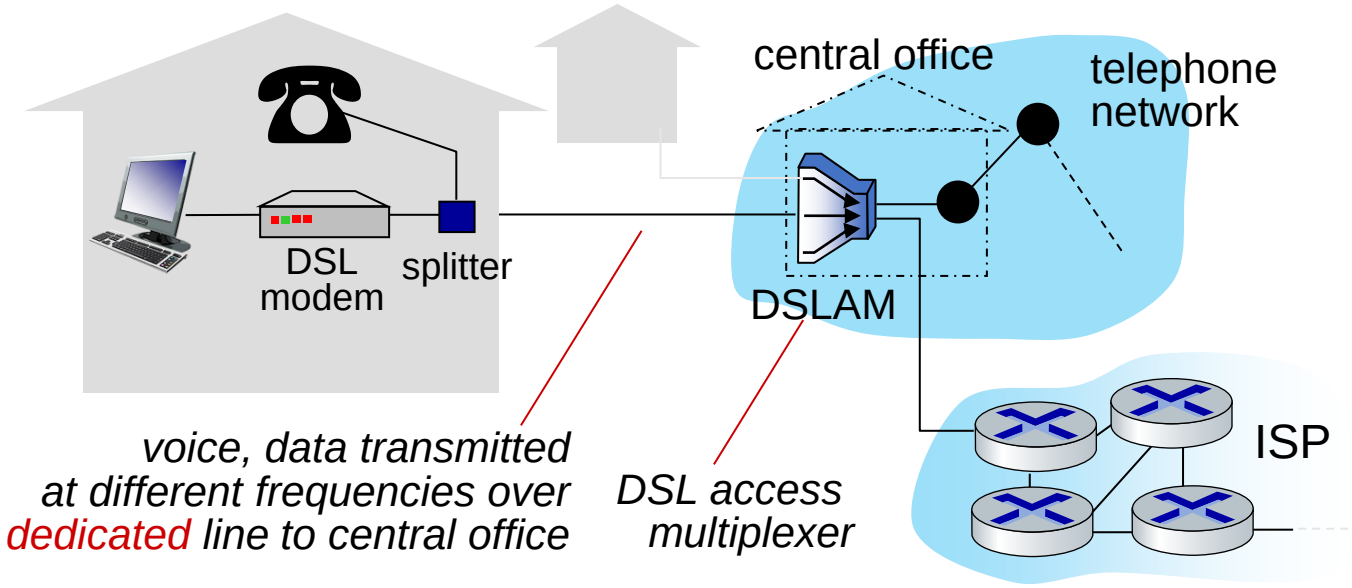
frequency division multiplexing (FDM): different channels transmitted in different frequency bands

Access networks: cable-based access



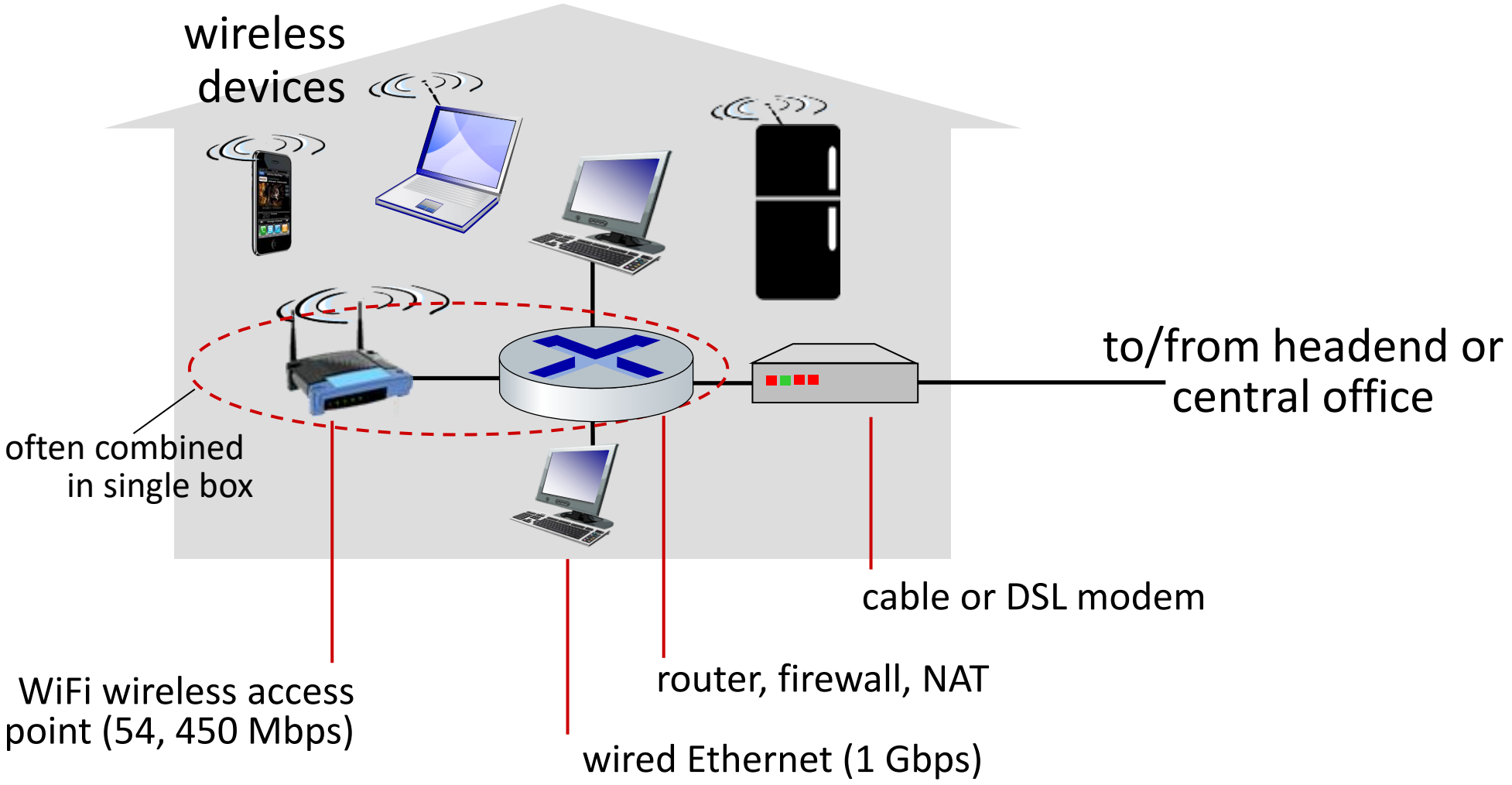
- HFC: hybrid fiber coax
  - *asymmetric*: up to 40 Mbps – 1.2 Gbps downstream transmission rate, 30-100 Mbps upstream transmission rate
- network of cable, fiber attaches homes to ISP router
  - homes *share access network* to cable headend

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



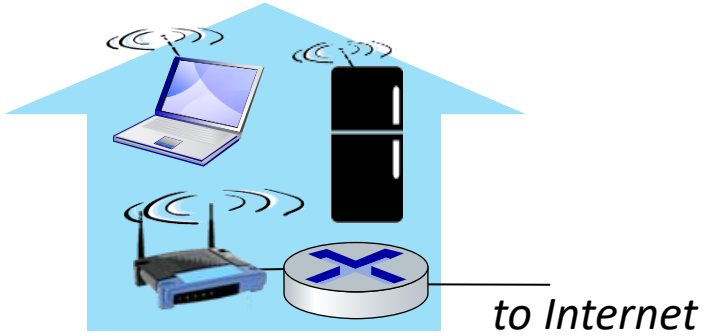
Wireless access networks

Shared *wireless* access network connects end system to router

- via base station aka “access point”

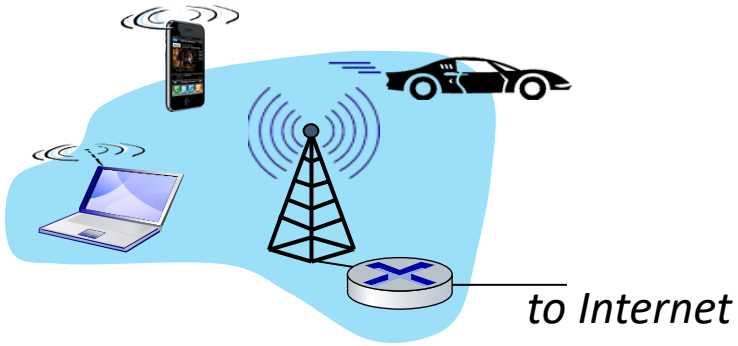
Wireless local area networks (WLANs)

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

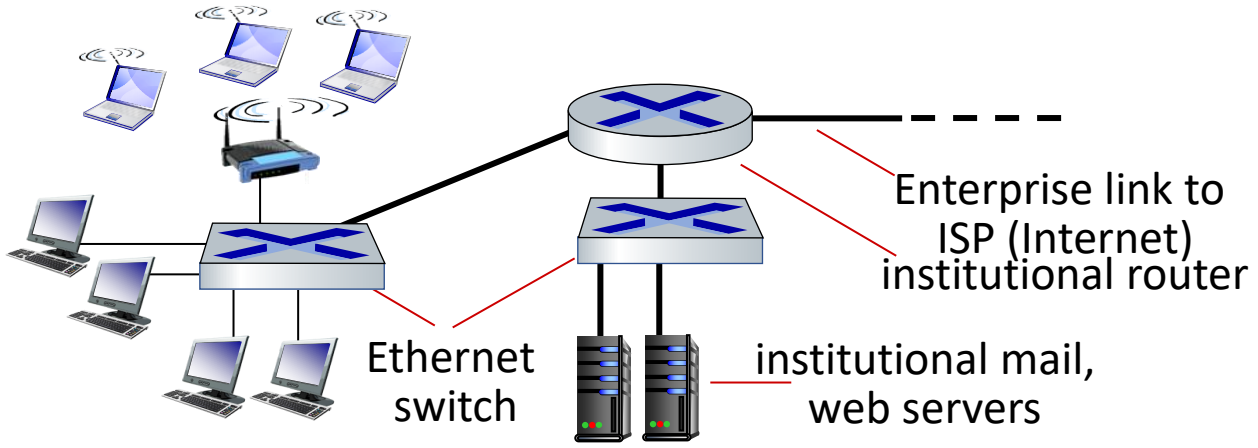


Wide-area cellular access networks

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



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