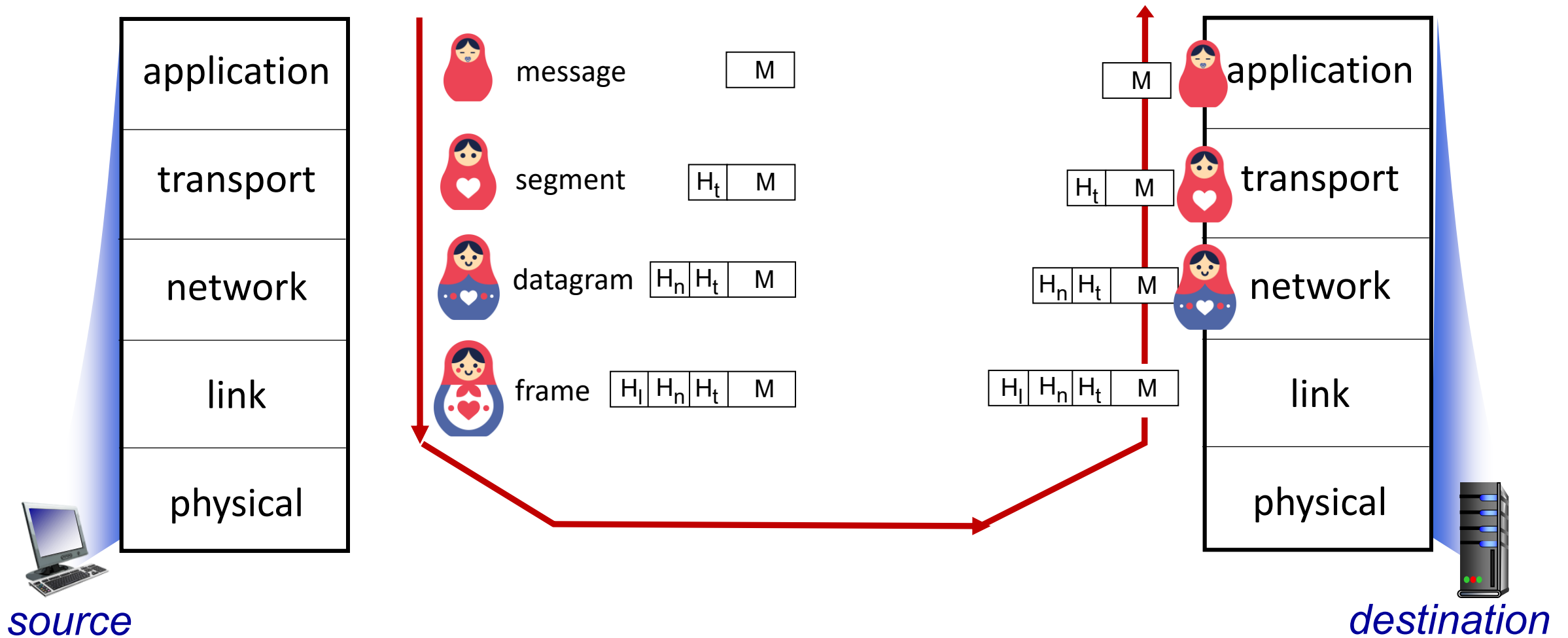


Dịch vụ, phân lớp, và đóng gói



Chapter 4

Network Layer:

Data plane

A note on the use of these PowerPoint slides:

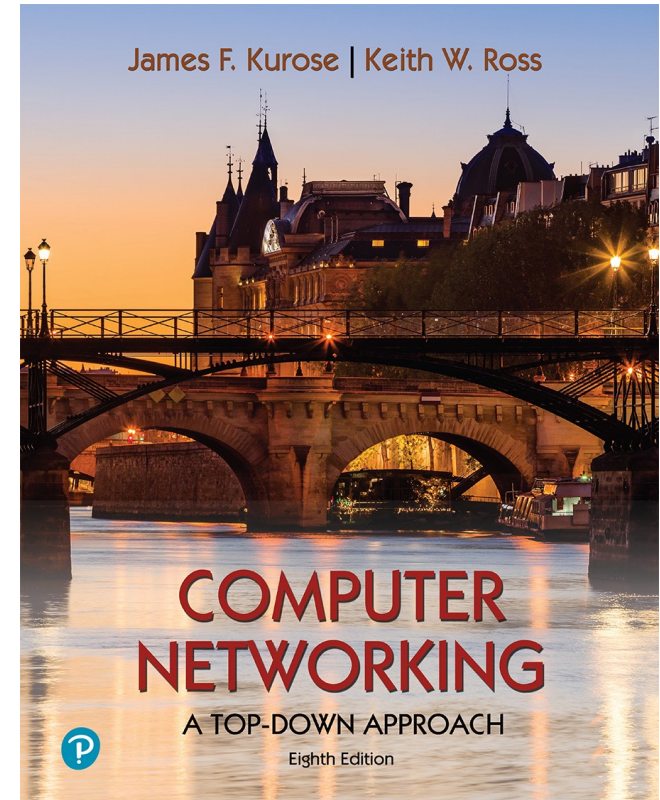
We're making these slides freely available to all (faculty, students, readers). They're in PowerPoint form so you see the animations; and can add, modify, and delete slides (including this one) and slide content to suit your needs. They obviously represent a *lot* of work on our part. In return for use,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

For a revision history, see the slide note for this page.

Thanks and enjoy! JFK/KWR

All material copyright 1996-2020
J.F Kurose and K.W. Ross, All Rights Reserved



*Computer Networking: A
Top-Down Approach*

8th edition

Jim Kurose, Keith Ross
Pearson, 2020

Lớp mạng: Mục tiêu

- Hiểu các nguyên tắc đằng sau các dịch vụ lớp mạng, tập trung vào **mặt dữ liệu (data plane)**:
 - Mô hình dịch vụ lớp mạng
 - Chuyển tiếp so với định tuyến
 - Cách bộ định tuyến hoạt động
 - Địa chỉ
 - Giao thức IP
 - NAT



Mặt dữ liệu (Data plane): Lộ trình

- Lớp mạng: Tổng quan
 - Mặt dữ liệu
 - Mặt điều khiển
- Có gì bên trong một router?
 - Cổng vào, kết cấu chuyển mạch, cổng ra
 - Quản lý bộ nhớ đệm
- IP: Giao thức Internet
 - Định dạng gói tin datagram
 - Địa chỉ
 - Dịch địa chỉ mạng
 - IPv6



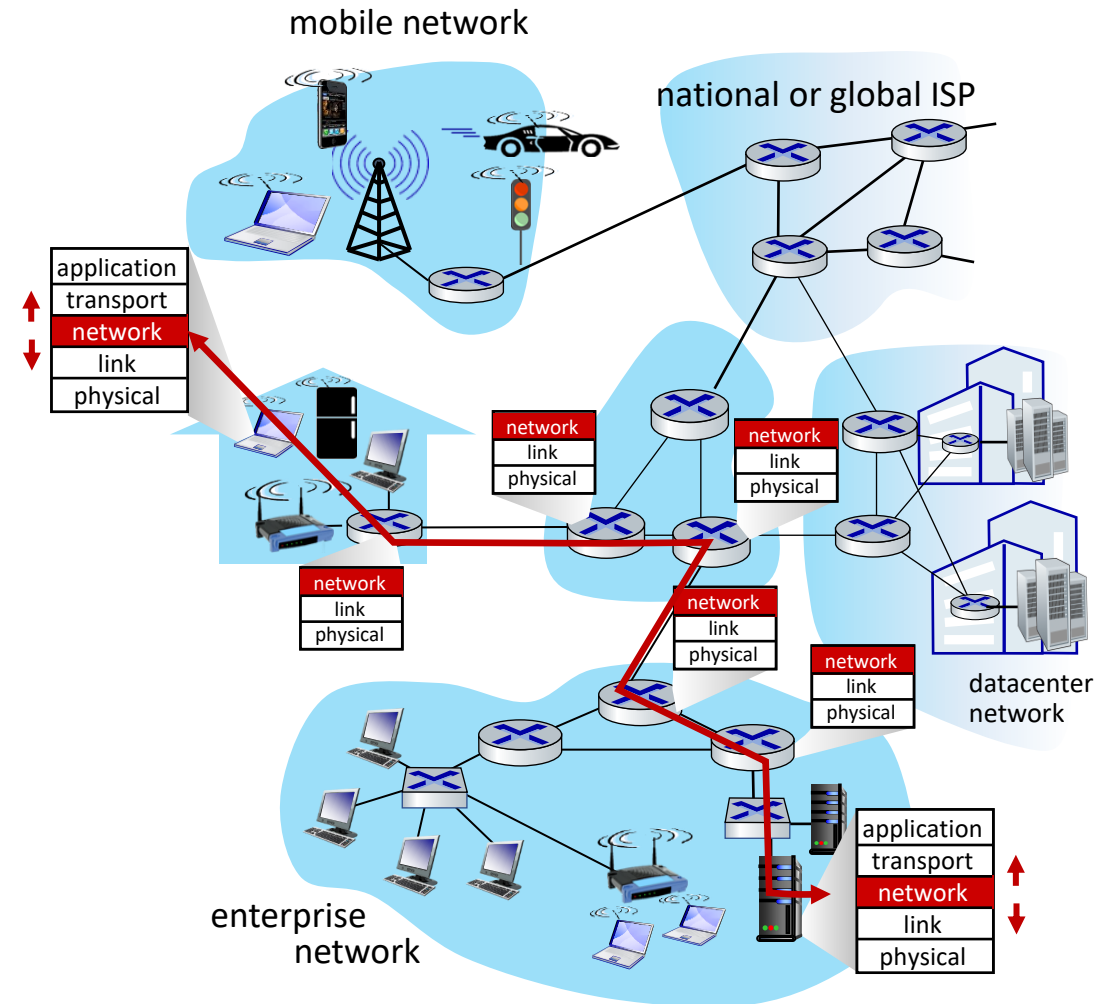
Mặt dữ liệu (Data plane): Lộ trình

- **Lớp mạng: Tổng quan**
 - Mặt dữ liệu
 - Mặt điều khiển
- Có gì bên trong một router?
 - Cổng vào, kết cấu chuyển mạch, cổng ra
 - Quản lý bộ nhớ đệm
- IP: Giao thức Internet
 - Định dạng gói tin datagram
 - Địa chỉ
 - Dịch địa chỉ mạng
 - IPv6



Giao thức và mô hình dịch vụ lớp mạng

- Truyền vận phân đoạn từ host gửi tới host nhận
 - Bên gửi: đóng gói phân đoạn vào gói tin datagrams, chuyển tới lớp liên kết dữ liệu
 - Bên nhận: chuyển phân đoạn tới lớp giao vận
- Giao thức lớp mạng ở **mọi thiết bị kết nối Internet**: hosts, routers
- **Bộ định tuyến (Routers)**:
 - Kiểm tra trường tiêu đề của tất cả gói tin datagrams đi qua nó
 - Chuyển gói tin datagrams từ cổng vào đến cổng ra để truyền gói tin datagrams theo đường đầu đến cuối



Hai chức năng chính của lớp mạng

Chức năng lớp mạng:

- *Chuyển tiếp (Forwarding)*: chuyển gói tin từ liên kết đầu vào của router đến liên kết đầu ra của router **thích hợp**
- *Định tuyến (Routing)*: xác định tuyến đường cho gói tin đi từ nguồn tới đích
 - *Thuật toán định tuyến*

Sự tương tự: taking a trip

- *Chuyển tiếp*: quá trình đi qua một nút giao thông duy nhất
- *Định tuyến*: quá trình lập kế hoạch chuyển đi từ điểm xuất phát đến điểm đích



forwarding



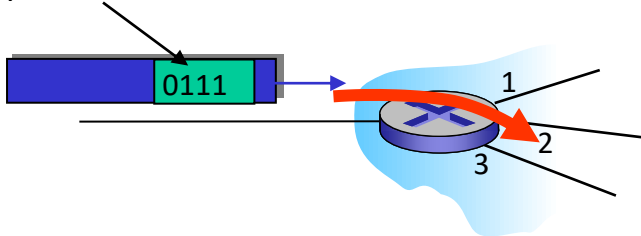
routing

Lớp mạng: Mặt dữ liệu và mặt điều khiển

Mặt dữ liệu (Data plane):

- *Cục bộ*, trên mỗi router
- Xác định cách gói tin datagram đến cổng vào của bộ định tuyến được chuyển tiếp đến cổng ra của bộ định tuyến

values in arriving packet header

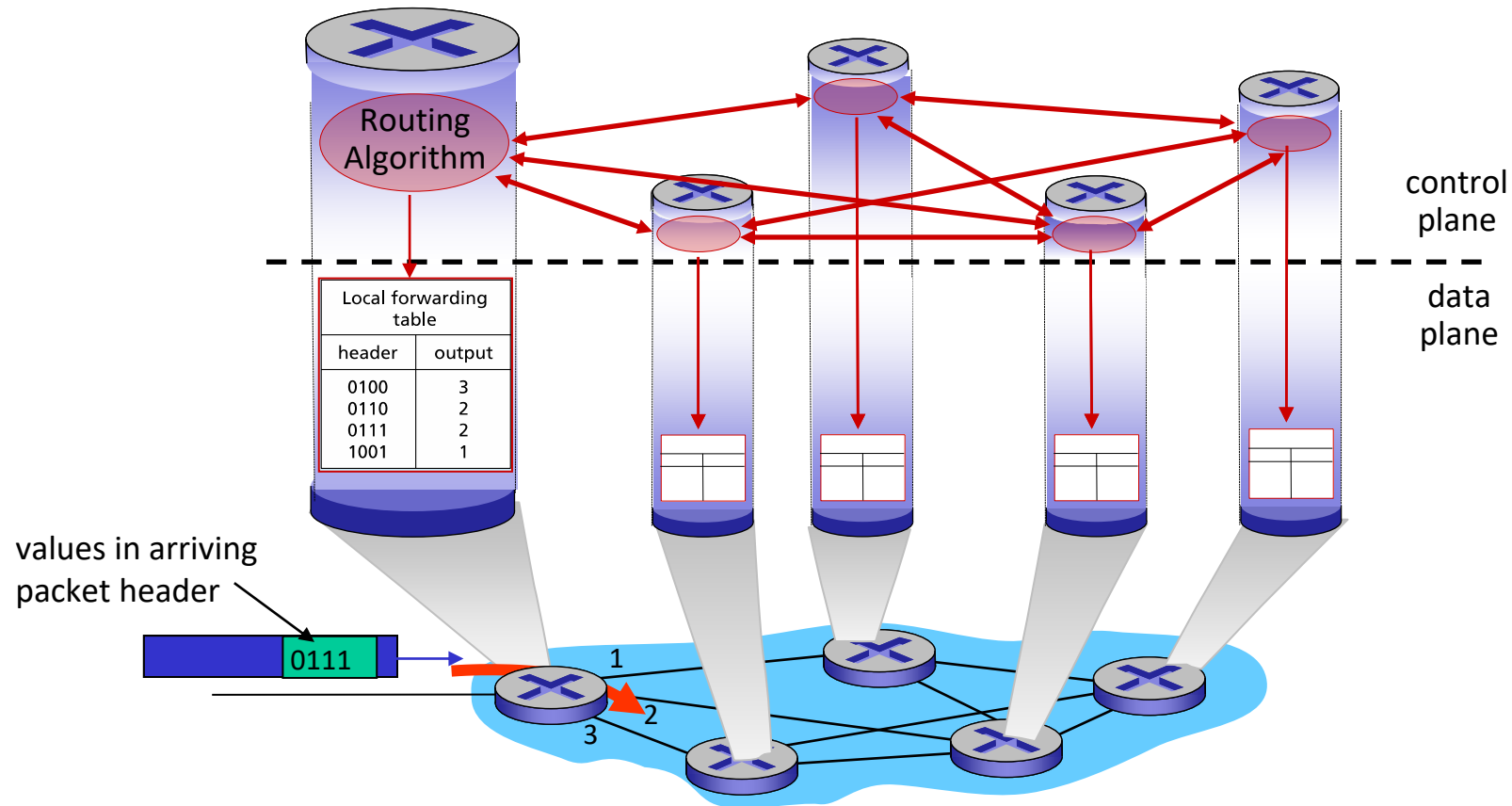


Mặt điều khiển (Control plane):

- Logic *toàn mạng*
- Xác định cách gói tin datagram được định tuyến giữa các bộ định tuyến theo đường đầu cuối từ host nguồn đến host đích
- Hai cách tiếp cận mặt điều khiển:
 - *Traditional routing algorithms*: được thực hiện trong bộ định tuyến
 - *Software-defined networking (SDN)*: được triển khai trong các máy chủ (từ xa)

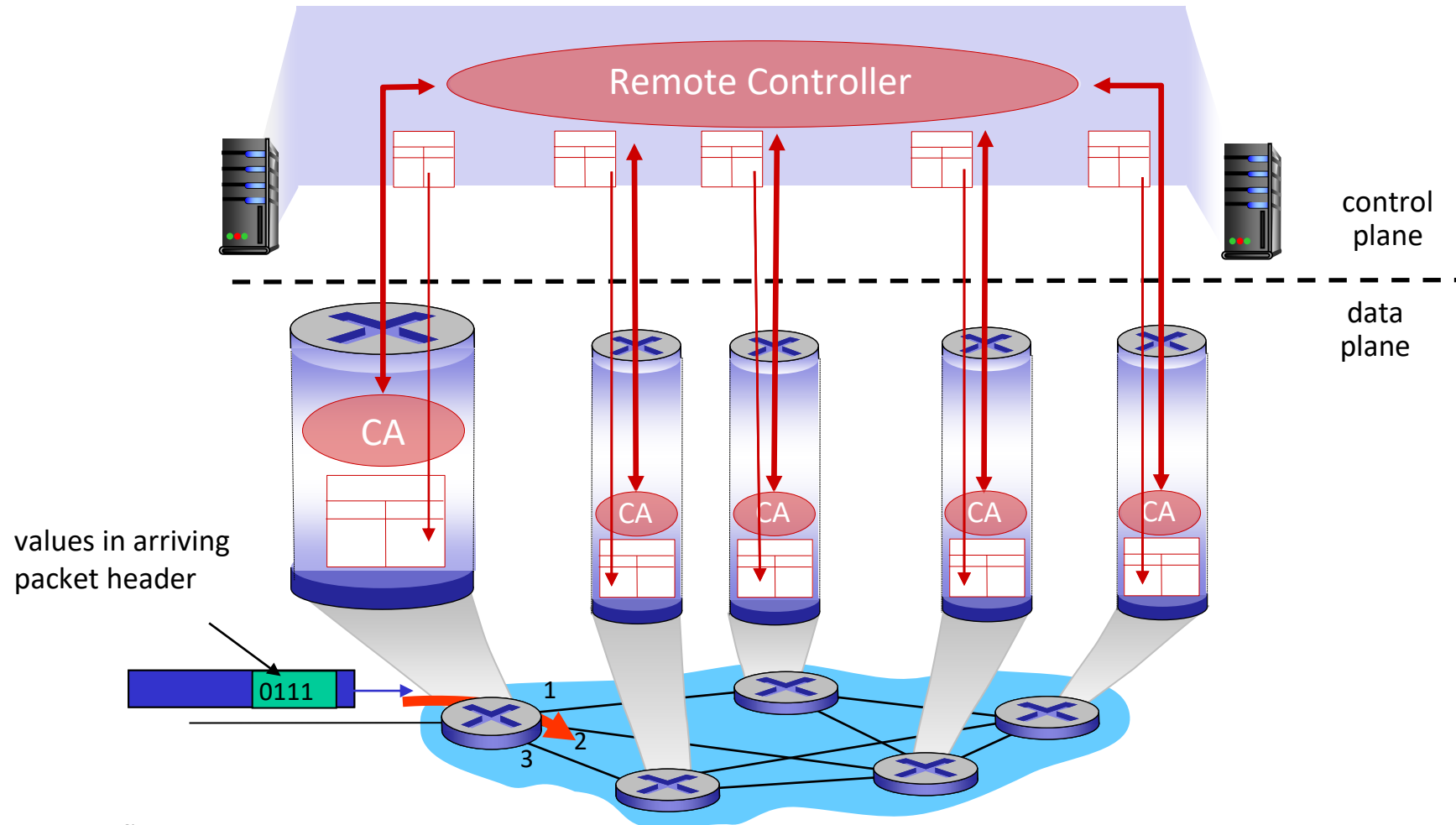
Mặt điều khiển trên bộ định tuyến

Các thành phần thuật toán định tuyến riêng lẻ *trong mỗi và mọi bộ định tuyến* tương tác trong mặt điều khiển



Mạng được xác định bởi phần mềm

Bộ điều khiển từ xa tính toán, thiết lập bảng chuyển tiếp trong các bộ định tuyến



Mô hình dịch vụ lớp mạng

Q: Mô hình dịch vụ nào cho “kênh” vận chuyển gói tin datagram từ bên gửi đến bên nhận?

Ví dụ dịch vụ cho các gói tin datagram *riêng lẻ* :

- Truyền gói tin đảm bảo
- Truyền gói tin đảm bảo với độ trễ dưới 40 msec

Ví dụ dịch vụ cho một *luồng* gói tin datagram:

- Truyền gói tin theo thứ tự
- Đảm bảo băng thông tối thiểu để lưu thông
- Hạn chế về những thay đổi trong khoảng cách giữa các gói tin

Mô hình dịch vụ lớp mạng

Network Architecture	Service Model	Quality of Service (QoS) Guarantees ?			
		Bandwidth	Loss	Order	Timing
Internet	best effort	none	no	no	no

Mô hình dịch vụ “nỗ lực tốt nhất” Internet

Không đảm bảo về :

- i. Truyền gói tin thành công tới đích
- ii. Thời gian hoặc thứ tự truyền gói tin
- iii. Băng thông có sẵn cho luồng dữ liệu đầu cuối

Mô hình dịch vụ lớp mạng

Network Architecture	Service Model	Quality of Service (QoS) Guarantees ?			
		Bandwidth	Loss	Order	Timing
Internet	best effort	none	no	no	no
ATM	Constant Bit Rate	Constant rate	yes	yes	yes
ATM	Available Bit Rate	Guaranteed min	no	yes	no
Internet	Intserv Guaranteed (RFC 1633)	yes	yes	yes	yes
Internet	Diffserv (RFC 2475)	possible	possibly	possibly	no

Phản ánh về dịch vụ nỗ lực tốt nhất

- Tính đơn giản của cơ chế đã cho phép Internet được triển khai và áp dụng rộng rãi
- Đủ **băng thông** cho phép hiệu suất của các ứng dụng thời gian thực (ví dụ: giao tiếp bằng giọng nói, video) ở mức “đủ tốt” trong “hầu hết thời gian”
- Các dịch vụ phân tán ở lớp ứng dụng, được sao chép (trung tâm dữ liệu, mạng phân phối nội dung) kết nối gần với mạng của Client, cho phép cung cấp dịch vụ từ nhiều địa điểm
- Kiểm soát tắc nghẽn của các dịch vụ “linh hoạt” giúp ích rất nhiều

Thật khó để tranh luận về sự thành công của mô hình dịch vụ nỗ lực tốt nhất

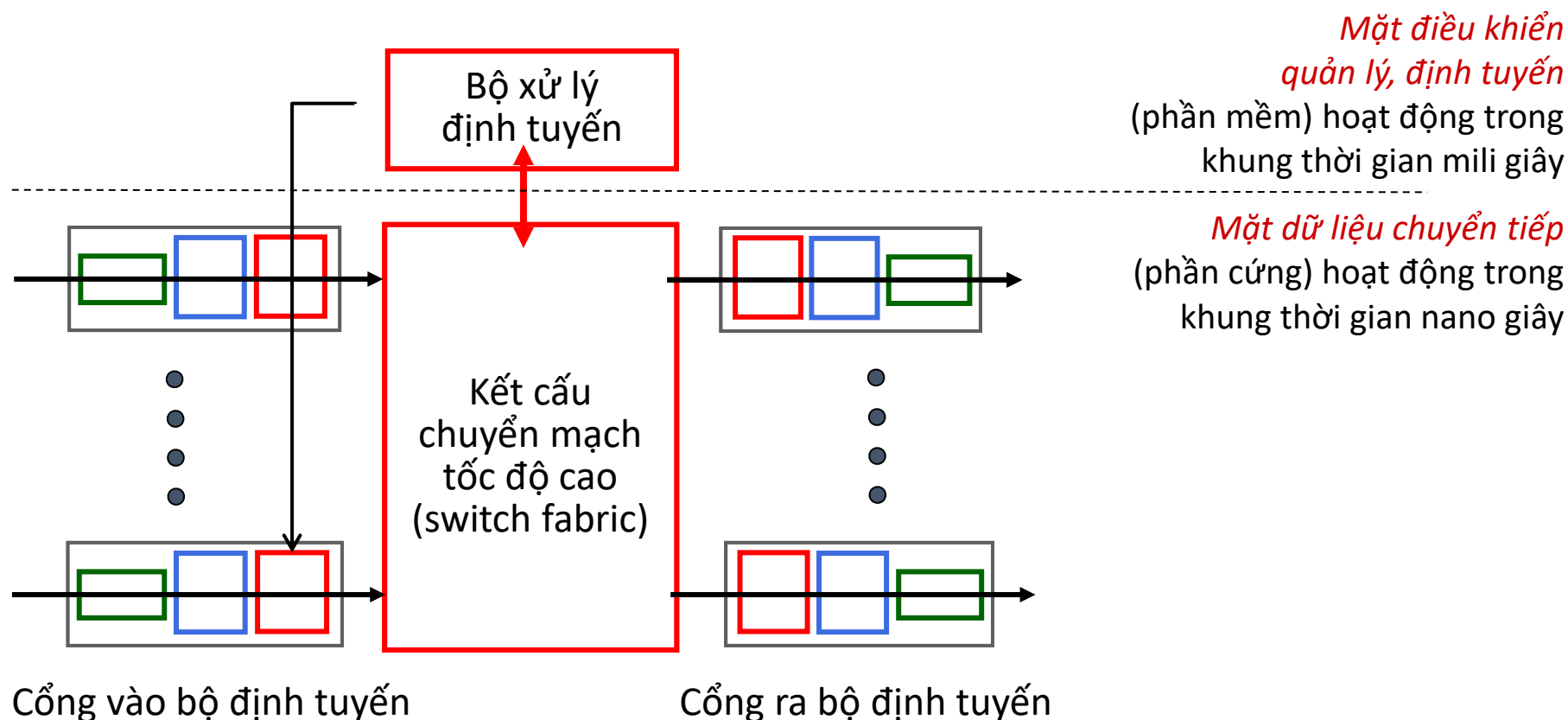
Mặt dữ liệu (Data plane): Lộ trình

- Lớp mạng: Tổng quan
 - Mặt dữ liệu
 - Mặt điều khiển
- Có gì bên trong một router?
 - Cổng vào, kết cấu chuyển mạch, cổng ra
 - Quản lý bộ nhớ đệm
- IP: Giao thức Internet
 - Định dạng gói tin datagram
 - Địa chỉ
 - Dịch địa chỉ mạng
 - IPv6

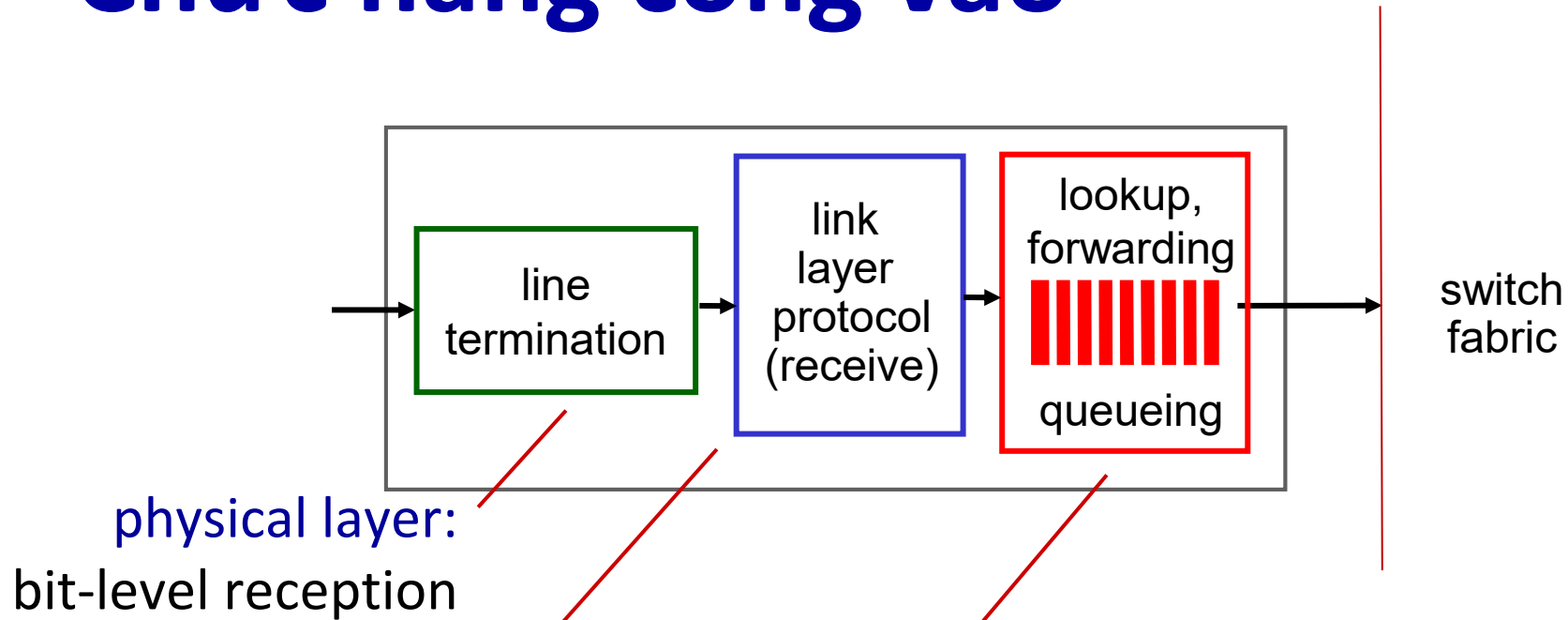


Tổng quan kiến trúc bộ định tuyến

Góc nhìn bậc cao của kiến trúc bộ định tuyến nói chung:



Chức năng cổng vào



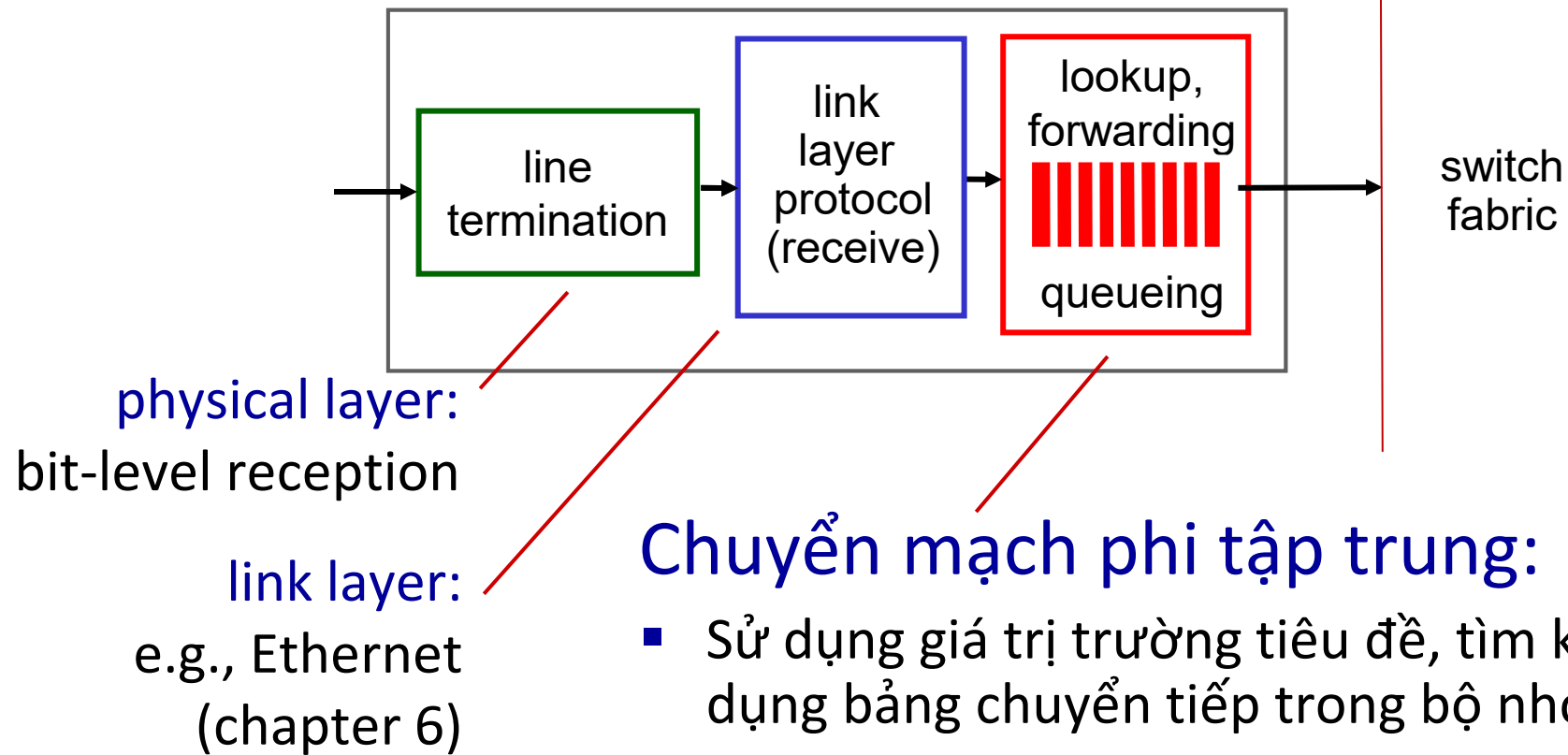
physical layer:
bit-level reception

link layer:
e.g., Ethernet
(chapter 6)

Chuyển mạch phi tập trung:

- Sử dụng giá trị trường tiêu đề, tìm kiếm cổng ra bằng cách sử dụng bảng chuyển tiếp trong bộ nhớ cổng vào
- Mục tiêu: hoàn thành xử lý cổng vào ở “tốc độ đường truyền”
- **Hàng đợi cổng vào:** nếu gói tin datagrams đến nhanh hơn tốc độ chuyển tiếp đến kết cấu chuyển mạch

Chức năng cổng vào



Chuyển mạch phi tập trung:

- Sử dụng giá trị trường tiêu đề, tìm kiếm cổng ra bằng cách sử dụng bảng chuyển tiếp trong bộ nhớ cổng vào
- **Chuyển tiếp dựa vào đích đến (destination-based forwarding):** chuyển tiếp chỉ dựa vào địa chỉ IP đích (truyền thống)
- **Generalized Forwarding:** chuyển tiếp dựa vào bất kỳ tập giá trị trường tiêu đề nào

Chuyển tiếp dựa vào đích đến

Q: Điều gì sẽ xảy ra nếu các dải địa chỉ IP không được chia đều?

<i>forwarding table</i>	
Destination Address Range	Link Interface
11001000 00010111 00010000 00000000 through 11001000 00010111 00010000 00000100 through 11001000 00010111 00010000 00000111	n 3
11001000 00010111 00011000 11111111 through 11001000 00010111 00011001 00000000 through 11001000 00010111 00011111 11111111	2
otherwise	3

Khớp tiền tố dài nhất (Longest prefix matching)

Khớp tiền tố dài nhất

Khi tìm đầu vào bảng chuyển tiếp cho địa chỉ đích nhất định, hãy sử dụng tiền tố địa chỉ **dài nhất** khớp với địa chỉ đích

Destination Address Range	Link interface
11001000 00010111 00010*** *****	0
11001000 00010111 00011000 *****	1
11001000 00010111 00011*** *****	2
otherwise	3

examples:

11001000 00010111 00010110 10100001 which interface?

11001000 00010111 00011000 10101010 which interface?

Khớp tiền tố dài nhất (Longest prefix matching)

Khớp tiền tố dài nhất

Khi tìm đầu vào bảng chuyển tiếp cho địa chỉ đích nhất định, hãy sử dụng tiền tố địa chỉ **dài nhất** khớp với địa chỉ đích

Destination Address Range	Link interface
11001000 00010111 00010*** *****	0
11001000 00010111 00011000 *****	1
11001000 match! 1 00011*** *****	2
otherwise	3

examples:

11001000 00010111 00010110 10100001 which interface?
11001000 00010111 00011000 10101010 which interface?

Khớp tiền tố dài nhất (Longest prefix matching)

Khớp tiền tố dài nhất

Khi tìm đầu vào bảng chuyển tiếp cho địa chỉ đích nhất định, hãy sử dụng tiền tố địa chỉ **dài nhất** khớp với địa chỉ đích

Destination Address Range	Link interface
11001000 00010111 00010*** *****	0
11001000 00010111 00011000 *****	1
11001000 00010111 00011*** *****	2
otherwise	3

match!

examples:

11001000 00010111 00010110 10100001	which interface?
11001000 00010111 00011000 10101010	which interface?

Khớp tiền tố dài nhất (Longest prefix matching)

Khớp tiền tố dài nhất

Khi tìm đầu vào bảng chuyển tiếp cho địa chỉ đích nhất định, hãy sử dụng tiền tố địa chỉ **dài nhất** khớp với địa chỉ đích

Destination Address Range	Link interface
11001000 00010111 00010*** *****	0
11001000 00010111 00011000 *****	1
11001000 00010111 00011*** *****	2
otherwise	3

match!

examples:

11001000 00010111 00010110 10100001 which interface?

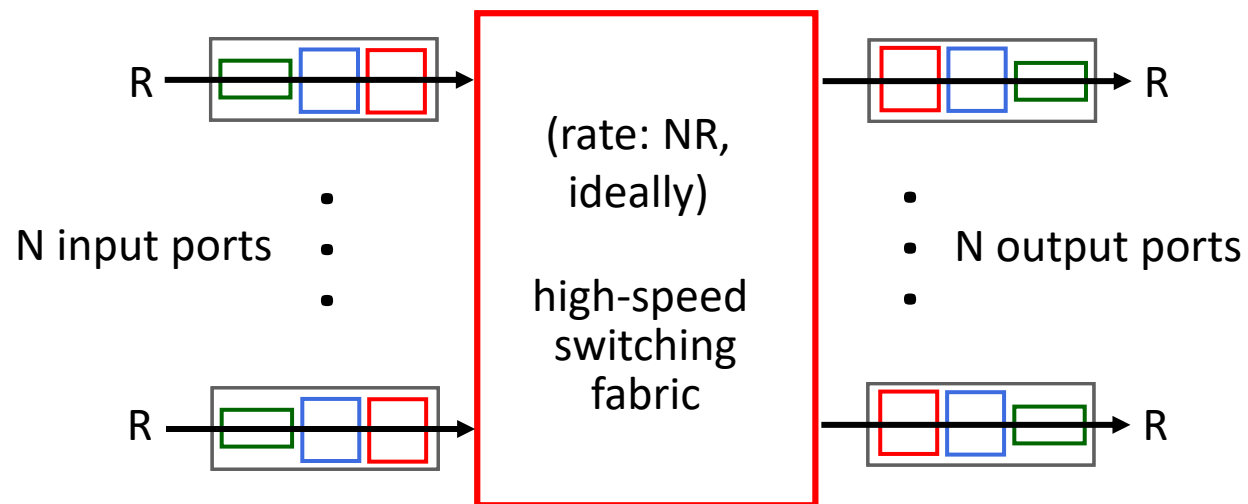
11001000 00010111 00011000 10101010 which interface?

Khớp tiền tố dài nhất (Longest prefix matching)

- Chúng ta sẽ thấy *tại sao* kết hợp tiền tố dài nhất được sử dụng ngay khi chúng ta nghiên cứu địa chỉ
- Khớp tiền tố dài nhất: thường được thực hiện bằng cách sử dụng bộ nhớ có thể định địa chỉ nội dung ternary (TCAM)
 - *Nội dung có thể định địa chỉ*: biểu diễn địa chỉ cho TCAM: truy xuất địa chỉ trong một chu kỳ xung, bất kể kích thước bảng chuyển tiếp
 - Cisco catalyst: ~1 triệu đầu mục trong bảng routing trong TCAM

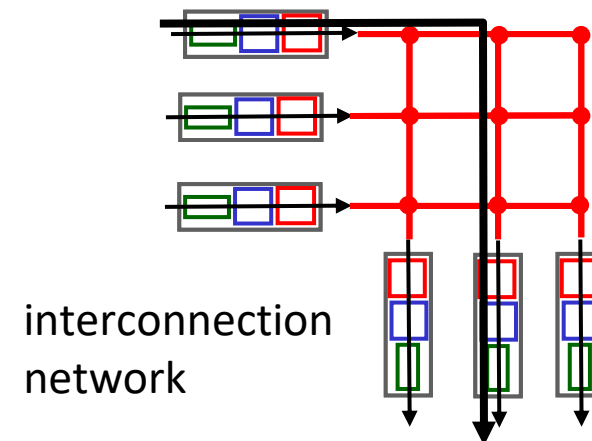
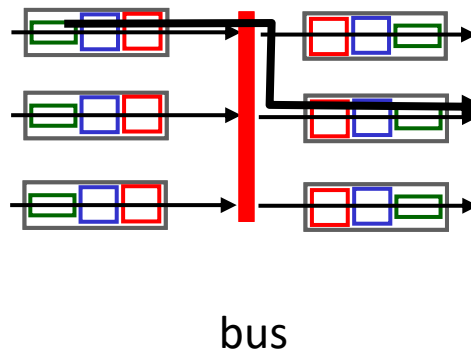
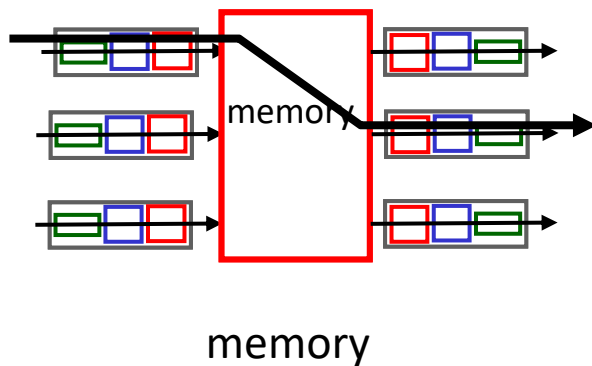
Kết cấu chuyển mạch

- Chuyển gói tin từ liên kết đầu vào tới liên kết đầu ra thích hợp
- **Tốc độ chuyển mạch:** tốc độ mà các gói tin có thể được chuyển từ đầu vào tới đầu ra
 - Thường được đo bằng bội số của tốc độ đường đầu vào/đầu ra
 - N đầu vào: tốc độ chuyển mạch R lần tốc độ dòng mong muốn



Kết cấu chuyển mạch

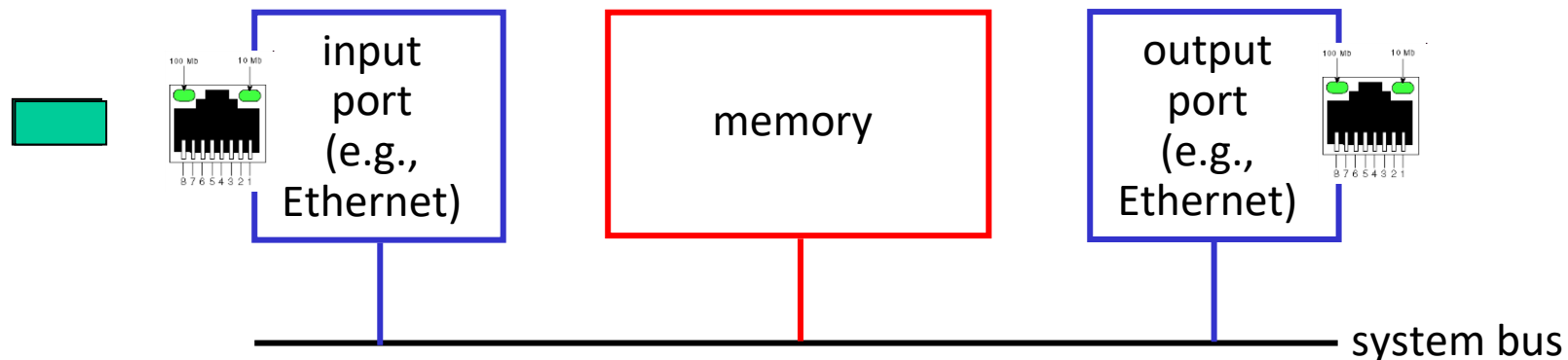
- Chuyển gói tin từ liên kết đầu vào tới liên kết đầu ra thích hợp
- **Tốc độ chuyển mạch:** tốc độ mà các gói tin có thể được chuyển từ đầu vào tới đầu ra
 - Thường được đo bằng bội số của tốc độ đường đầu vào/đầu ra
 - N đầu vào: tốc độ chuyển mạch R lần tốc độ dòng mong muốn
- Ba loại chuyển mạch chính:



Switching via memory

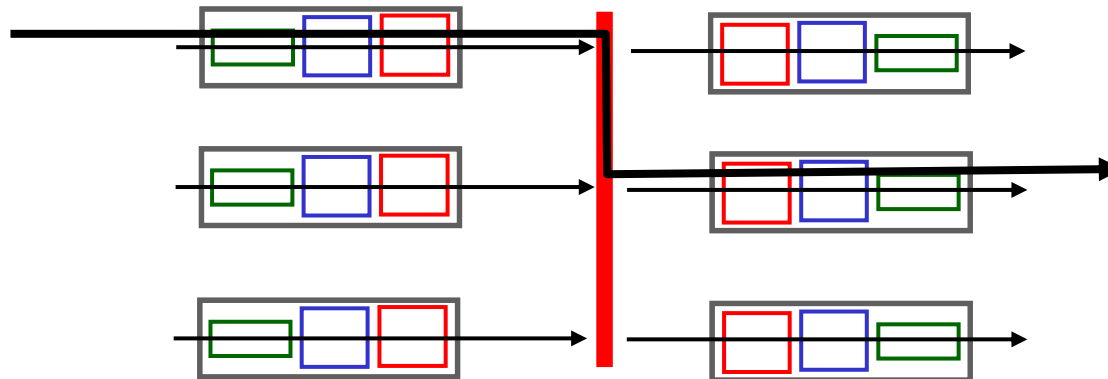
Bộ định tuyến thế hệ đầu tiên:

- Máy tính truyền thống với chuyển mạch dưới sự kiểm soát trực tiếp của CPU
- Gói tin được sao chép vào bộ nhớ của hệ thống
- Tốc độ bị giới hạn bởi băng thông bộ nhớ (2 bus crossings per datagram)



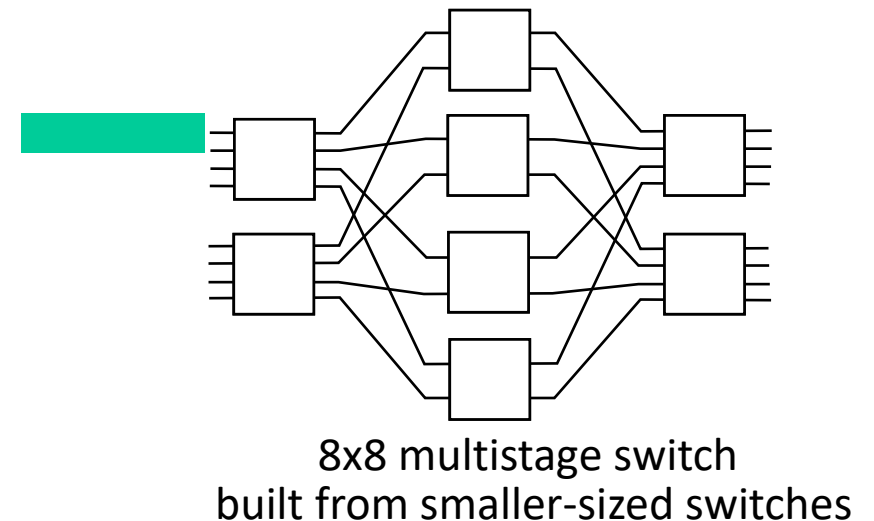
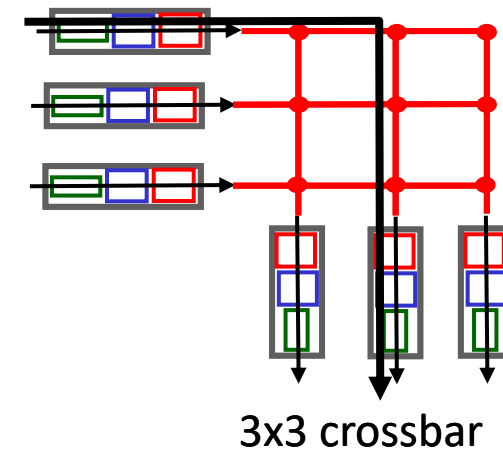
Switching via a bus

- Datagram từ bộ nhớ cổng vào đến bộ nhớ cổng ra thông qua một bus chia sẻ chung
- *Bus contention*: Tốc độ chuyển mạch bị giới hạn bởi băng thông của bus
- Bus 32 gbps, cisco 5600: đủ tốc độ cho các bộ định tuyến truy cập



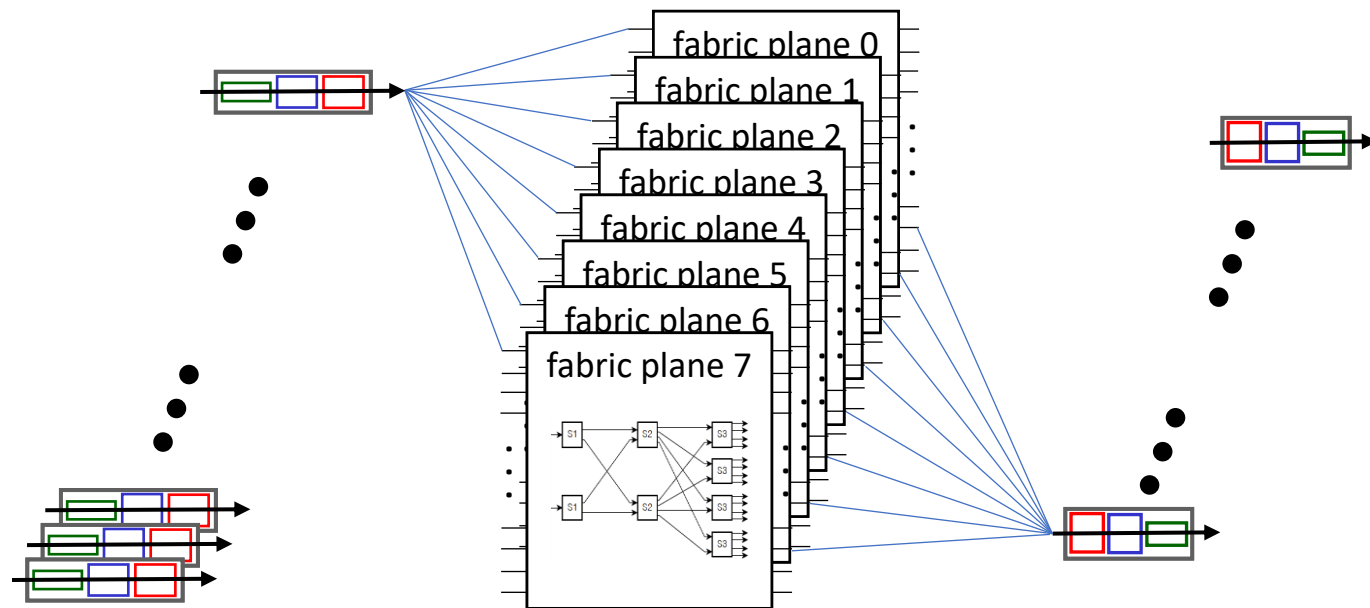
Switching via interconnection network

- Mạng Crossbar, Clos, các mạng kết nối khác ban đầu được phát triển để kết nối các bộ xử lý trong bộ đa xử lý
- Chuyển mạch nhiều giai đoạn: $n \times n$
chuyển mạch từ nhiều giai đoạn của các chuyển mạch nhỏ hơn
- Khai thác cơ chế song song:
 - Phân mảnh datagram thành các ô có độ dài cố định tại đầu vào kết cấu chuyển mạch
 - Chuyển mạch các ô thông qua kết cấu, tập hợp lại datagram tại đầu ra kết cấu chuyển mạch



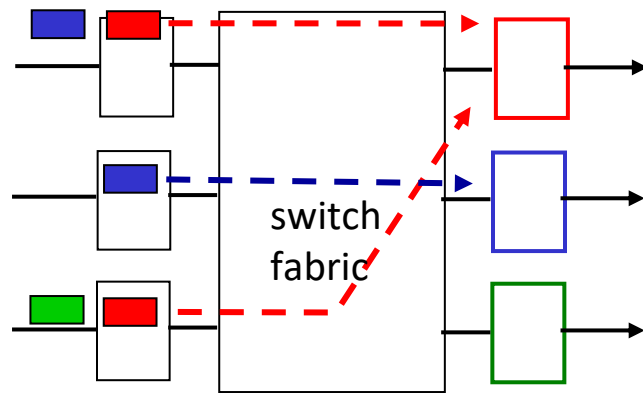
Switching via interconnection network

- Chia tỷ lệ, sử dụng song song nhiều “mặt” chuyển mạch:
 - Tăng tốc, mở rộng quy mô thông qua song song
- Router CRS của Cisco:
 - Đơn vị cơ bản: 8 mặt chuyển mạch
 - Mỗi mặt: mạng kết nối 3 tầng
 - Khả năng chuyển mạch lên đến 100 Tbps

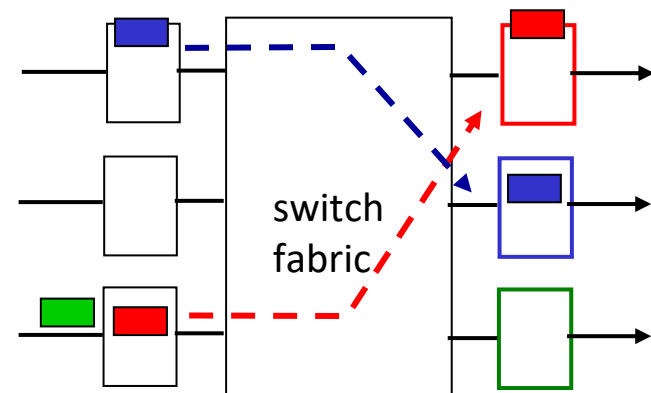


Hàng đợi cổng vào

- Nếu cấu trúc chuyển mạch chậm hơn so với cổng vào kết hợp -> hàng đợi có thể xảy ra ở hàng đợi đầu vào
 - Độ trễ và mất hàng đợi do tràn bộ đệm đầu vào!
- **Khoá Head-of-the-Line (HOL):** datagram xếp hàng đợi ở phía trước hàng đợi ngăn datagram khác trong hàng đợi di chuyển về phía trước

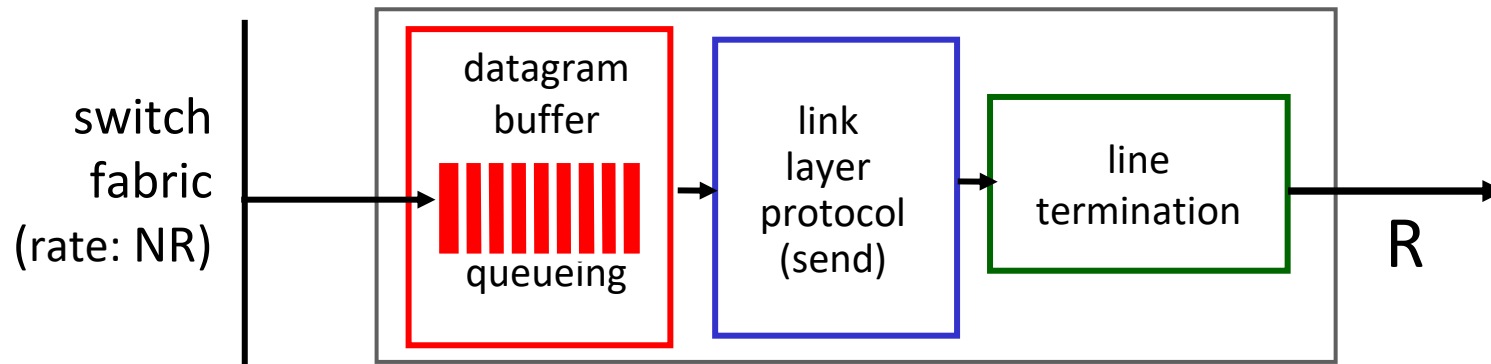


Tranh chấp cổng ra: chỉ có thể truyền một gói tin dữ liệu màu đỏ trên, gói tin màu đỏ dưới **bị khoá**



Một thời gian sau gói tin: gói tin màu xanh lá cây được qua khoá HOL

Hàng đợi công ra



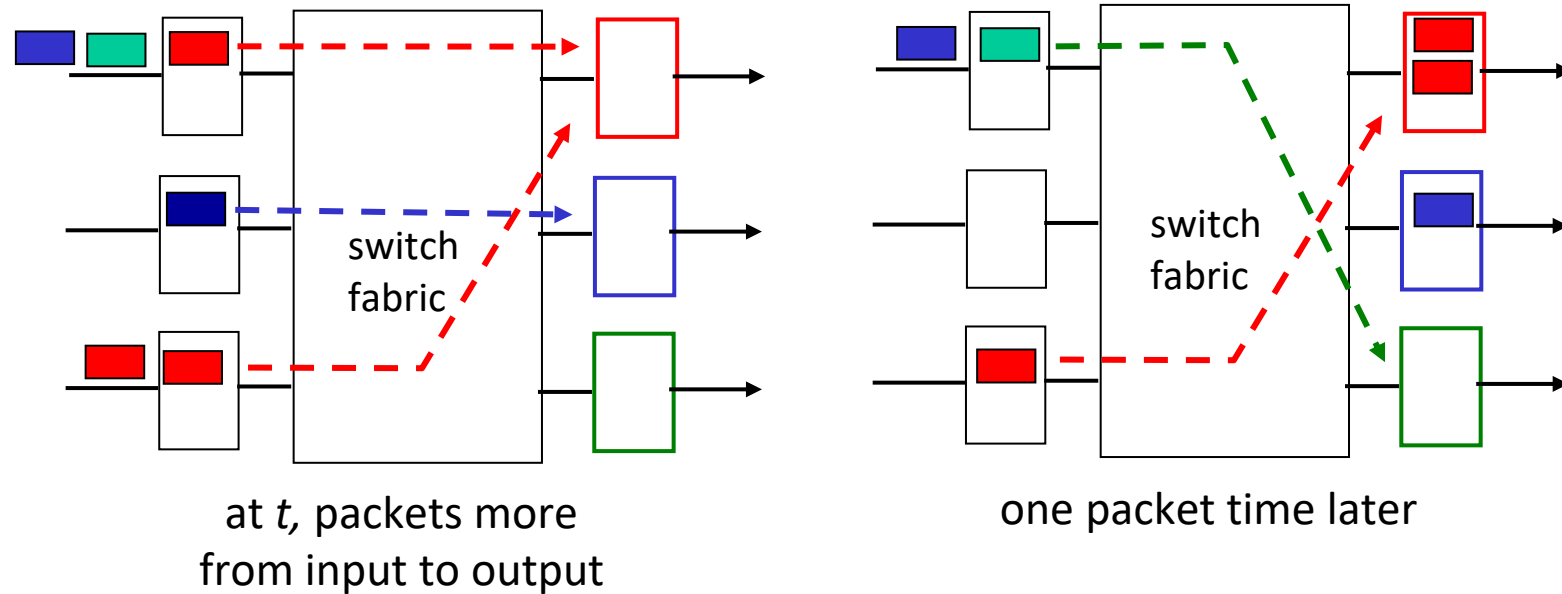
Đây là một slide thực sự quan trọng

- **Bộ đệm** được yêu cầu khi các gói tin datagrams đến từ kết cấu nhanh hơn tốc độ truyền tại lớp liên kết. Chính sách **loại bỏ**: gói tin datagrams nào sẽ loại bỏ nếu bộ đệm đầy?
- **Nguyên tắc lên lịch** chọn trong số các gói tin datagram được xếp hàng đợi để truyền

Gói tin datagram có thể bị mất do tắc nghẽn, thiếu bộ đệm

Lập kế hoạch ưu tiên – ai có hiệu suất tốt nhất, tính trung lập của mạng

Hàng đợi công ra



- Đệm khi tốc độ đến qua kết cấu chuyển mạch vượt quá tốc độ đường ra
- *Hàng đợi (độ trễ) và mất mát do tràn bộ đệm công ra!*

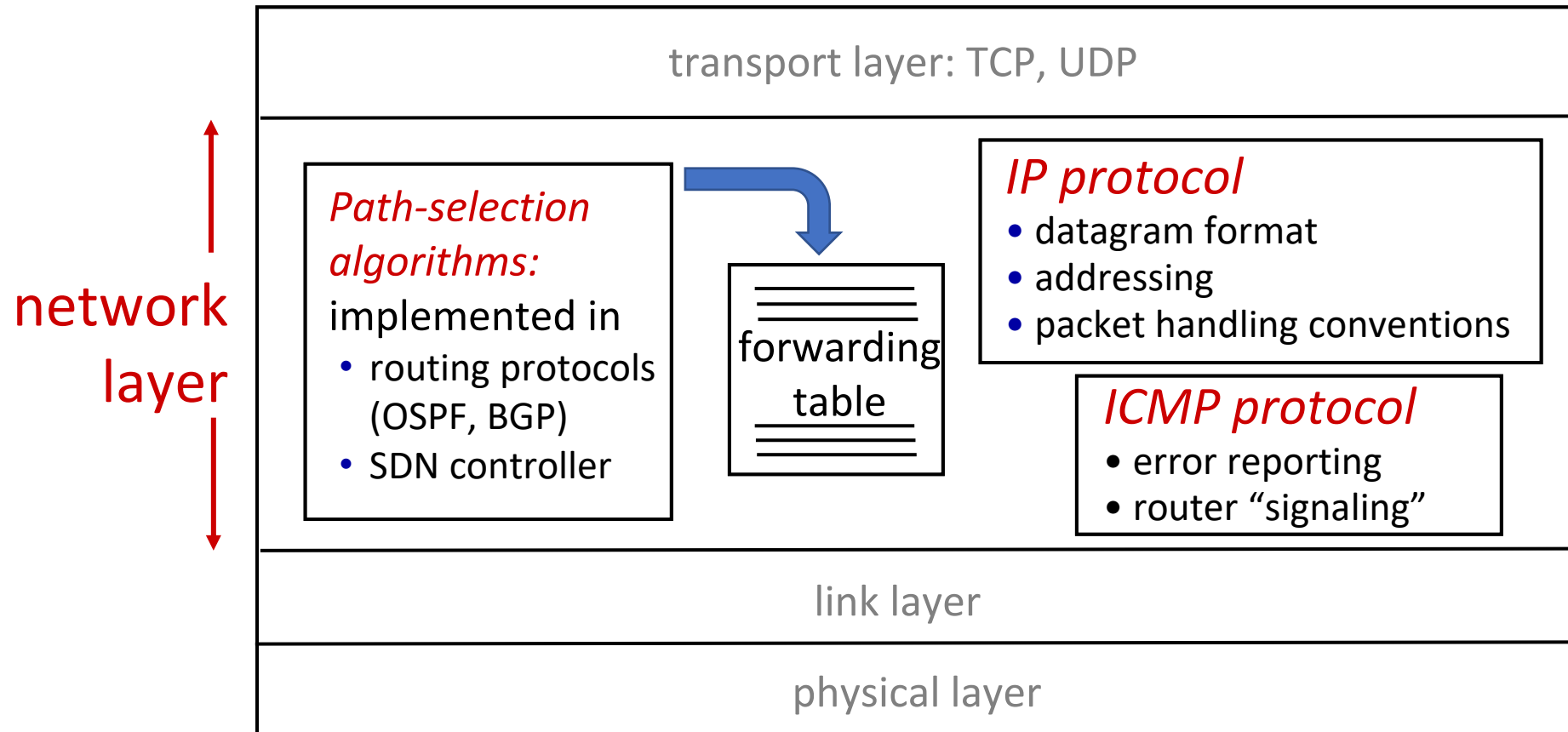
Mặt dữ liệu (Data plane): Lộ trình

- Lớp mạng: Tổng quan
 - Mặt dữ liệu
 - Mặt điều khiển
- Có gì bên trong một router?
 - Cổng vào, kết cấu chuyển mạch, cổng ra
 - Quản lý bộ nhớ đệm
- IP: Giao thức Internet
 - Định dạng gói tin datagram
 - Địa chỉ
 - Dịch địa chỉ mạng
 - IPv6

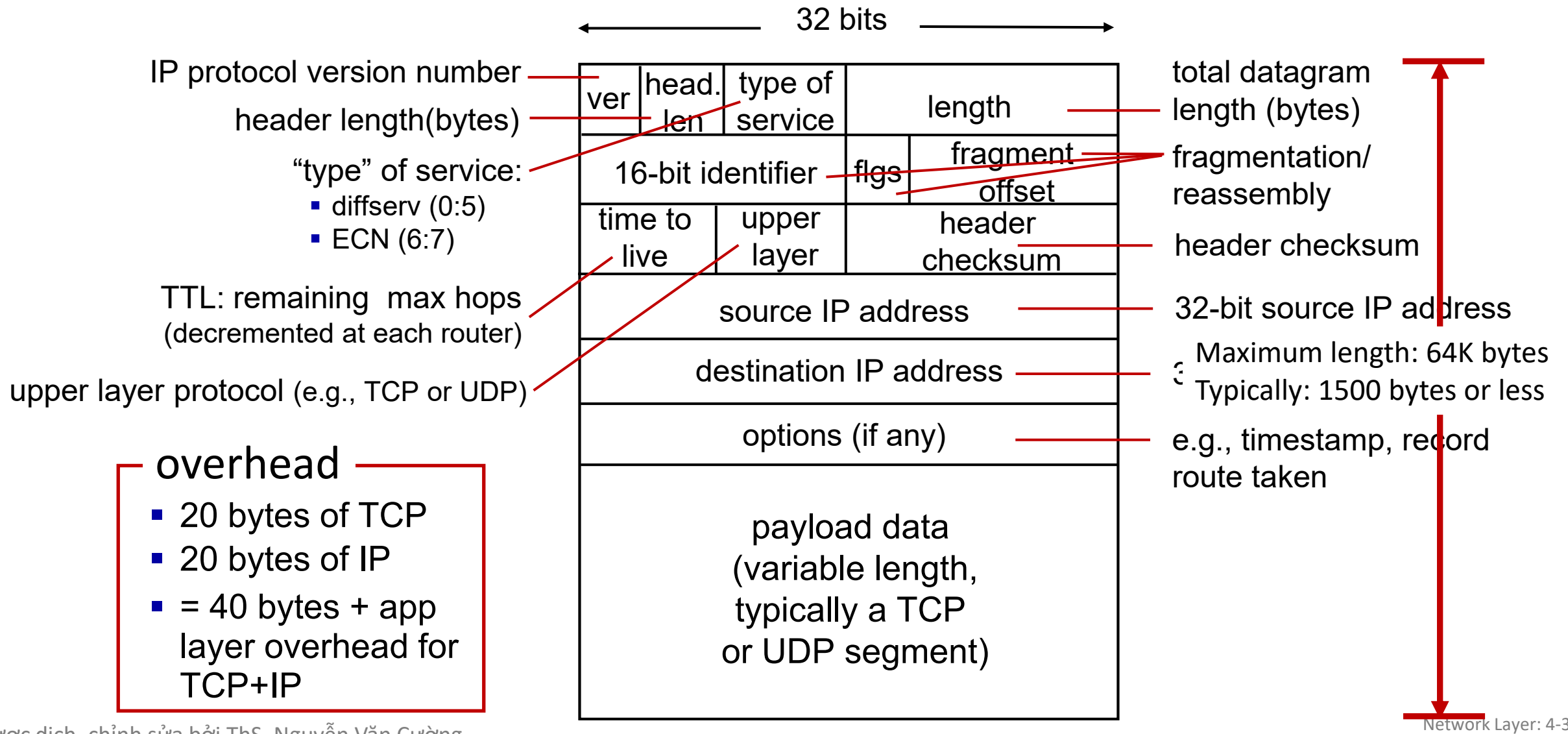


Lớp mạng: Internet

Chức năng lớp mạng trên host, router:

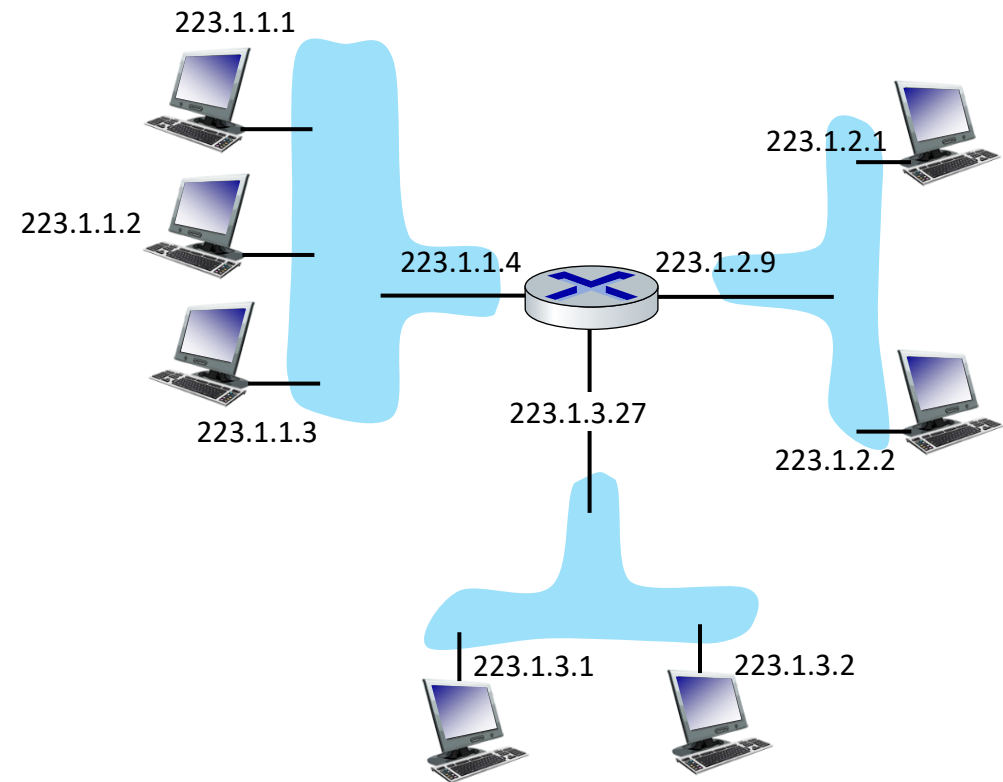


Định dạng gói tin IP



IP addressing: introduction

- **IP address** : Mã định danh 32 bit được liên kết với từng giao diện host hoặc *router*
- **Interface** : kết nối giữa host/router và liên kết vật lý
 - Router thường có nhiều giao diện
 - Host thường có một hoặc hai giao diện (ví dụ: Ethernet có dây, 802.11 không dây)



Ký hiệu địa chỉ IP dạng thập phân :

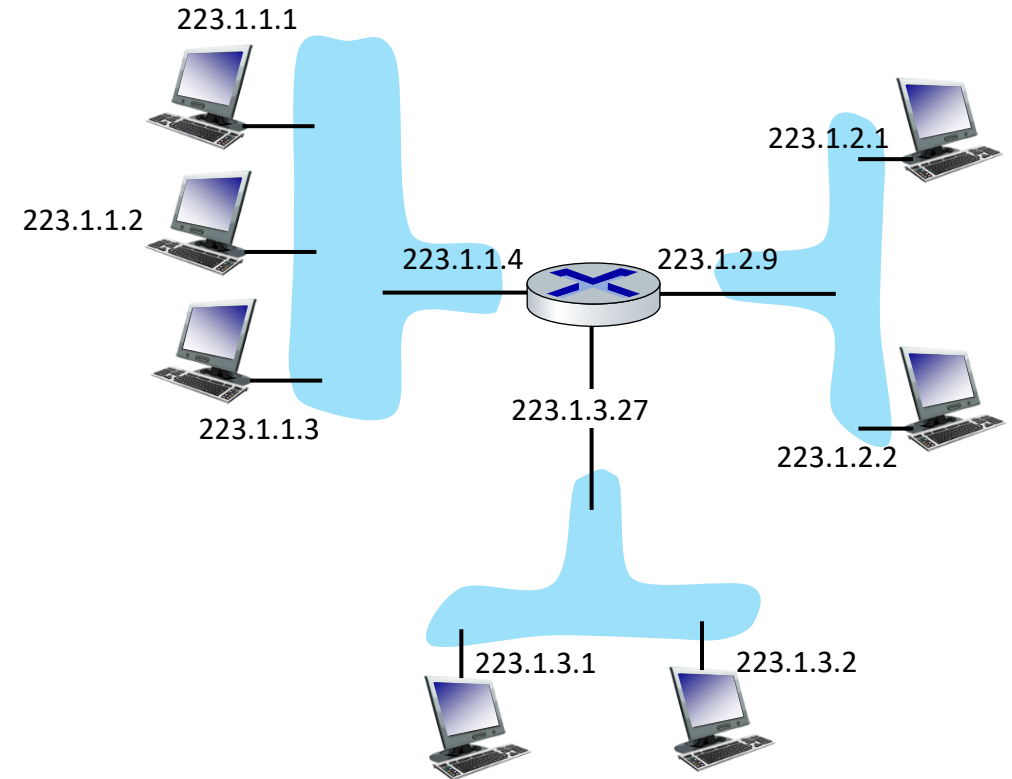
223.1.1.1 = 11011111 00000001 00000001 00000001

223 1 1 1

Network Layer: 4-37

IP addressing: introduction

- **IP address:** mã định danh 32 bit được liên kết với từng giao diện host hoặc *router*
- **Interface:** kết nối giữa host/router và liên kết vật lý
 - Router thường có nhiều giao diện
 - Host thường có một hoặc hai giao diện (ví dụ: ethernet có dây, 802.11 không dây)



Ký hiệu địa chỉ IP dạng thập phân :

223.1.1.1 = 11011111 00000001 00000001 00000001

223 1 1 1

Network Layer: 4-38

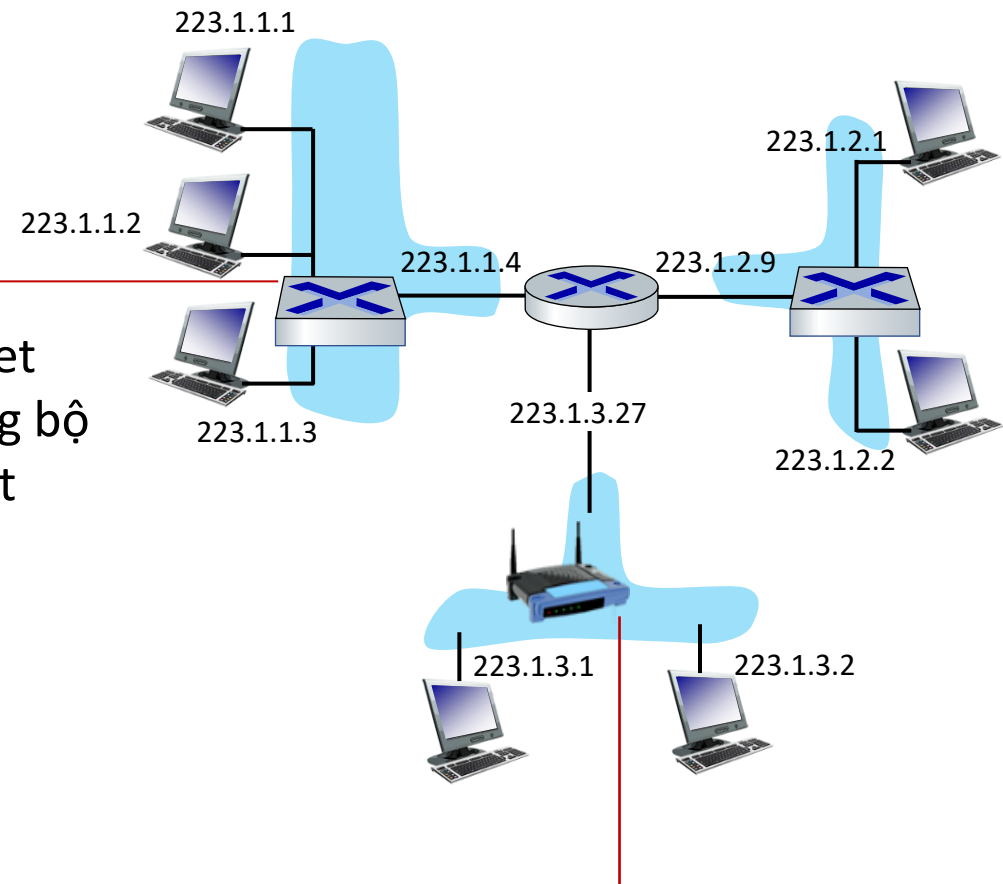
IP addressing: introduction

Q: Các giao diện thực sự được kết nối như thế nào?

A: Chúng ta sẽ tìm hiểu về điều đó trong chương 6, 7

Hiện tại: không cần lo lắng về cách một giao diện được kết nối với giao diện khác (không có router can thiệp)

A: có dây
Giao diện Ethernet được kết nối bằng bộ switches Ethernet



A: giao diện WiFi không dây được kết nối bởi trạm gốc WiFi

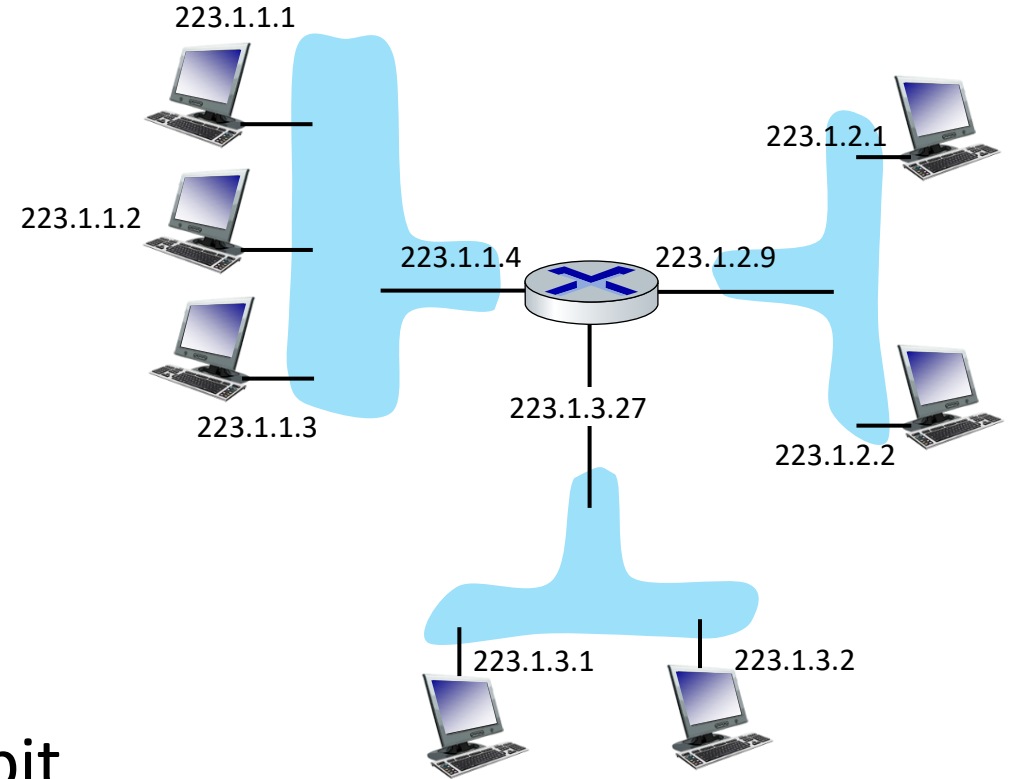
Subnets

■ Là gì ?

- Giao diện thiết bị có thể tiếp cận vật lý với nhau **mà không cần thông qua router can thiệp**

■ Địa chỉ IP có cấu trúc:

- **Phần mạng con:** các thiết bị trong cùng một mạng con có chung các bit bậc cao
- **Phần host:** các bit thứ tự thấp còn lại

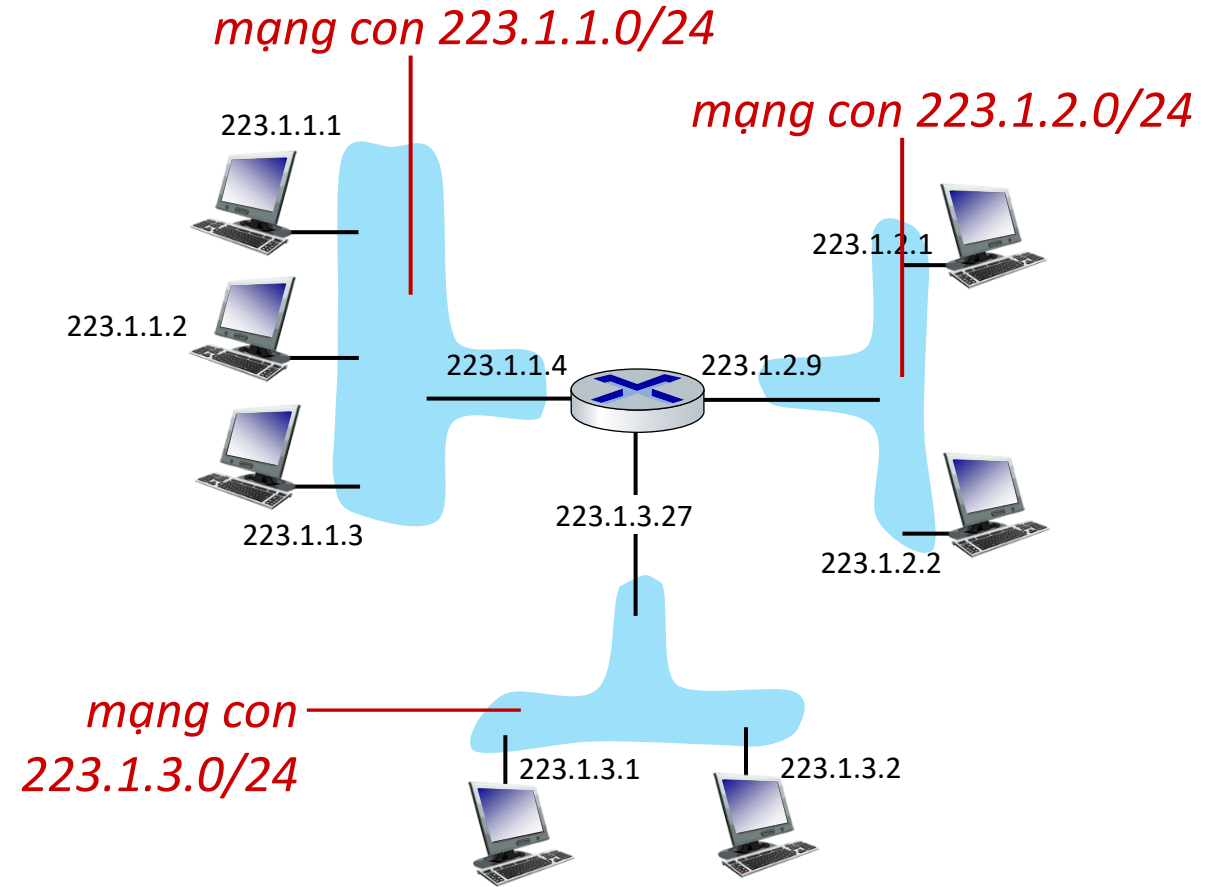


Mạng: gồm 3 mạng con

Subnets

Công thức xác định mạng con:

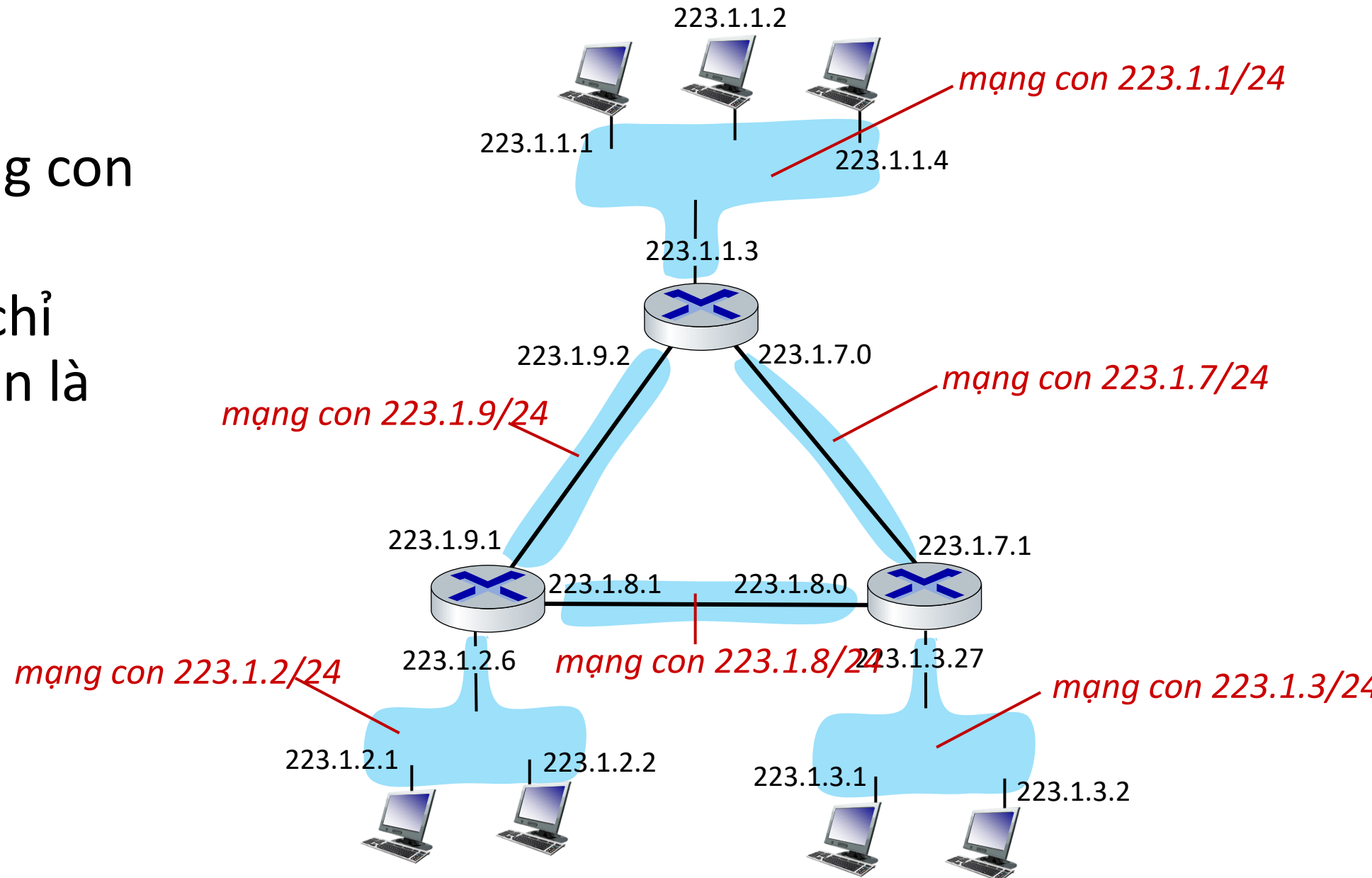
- Tách từng giao diện khỏi host hoặc router của nó, tạo ra các “islands” của các mạng bị cô lập
- Mỗi mạng bị cô lập được gọi là *mạng con*



Mặt nạ mạng con: /24
(24 bit bậc cao: phần mạng con của địa chỉ IP)

Subnets

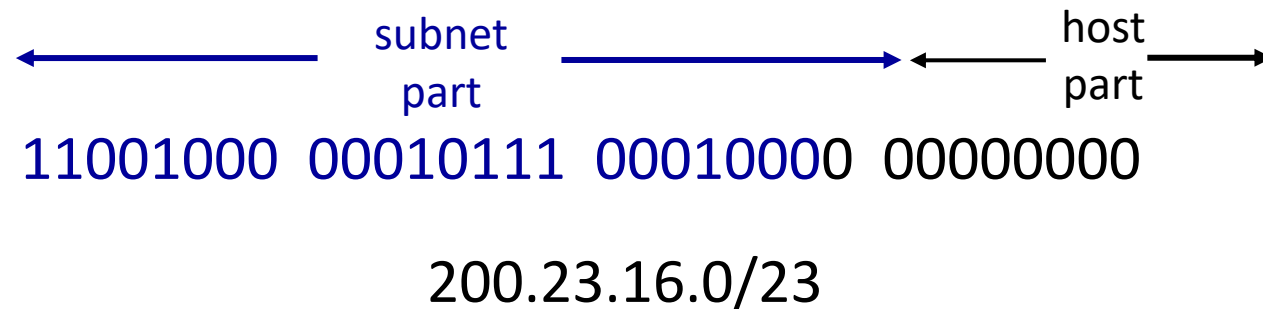
- Các mạng con ở đâu?
- /24 địa chỉ mạng con là gì?



IP addressing: CIDR

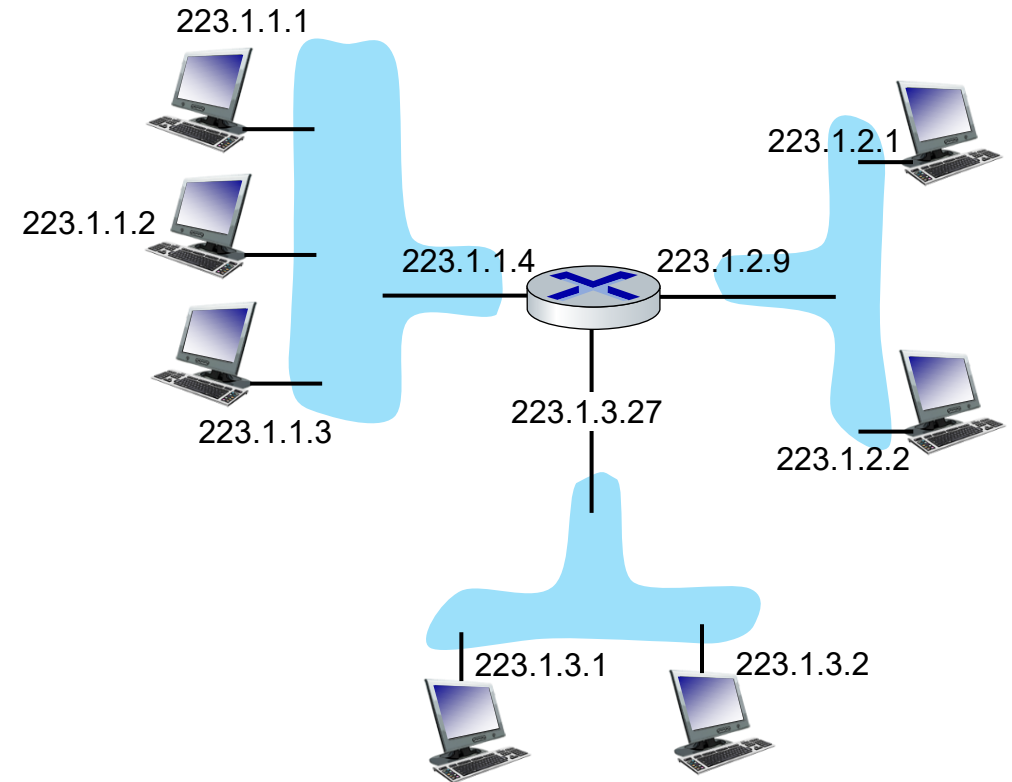
CIDR: Classless InterDomain Routing (pronounced “cider”)

- Phần mạng con của địa chỉ có độ dài tùy ý
- Định dạng địa chỉ: **a.b.c.d/x**, trong đó x là # bit trong phần mạng con của địa chỉ



IP addressing: introduction

- **IP address:** 32-bit identifier associated with each host or router *interface*
- **interface:** connection between host/router and physical link
 - router's typically have multiple interfaces
 - host typically has one or two interfaces (e.g., wired Ethernet, wireless 802.11)



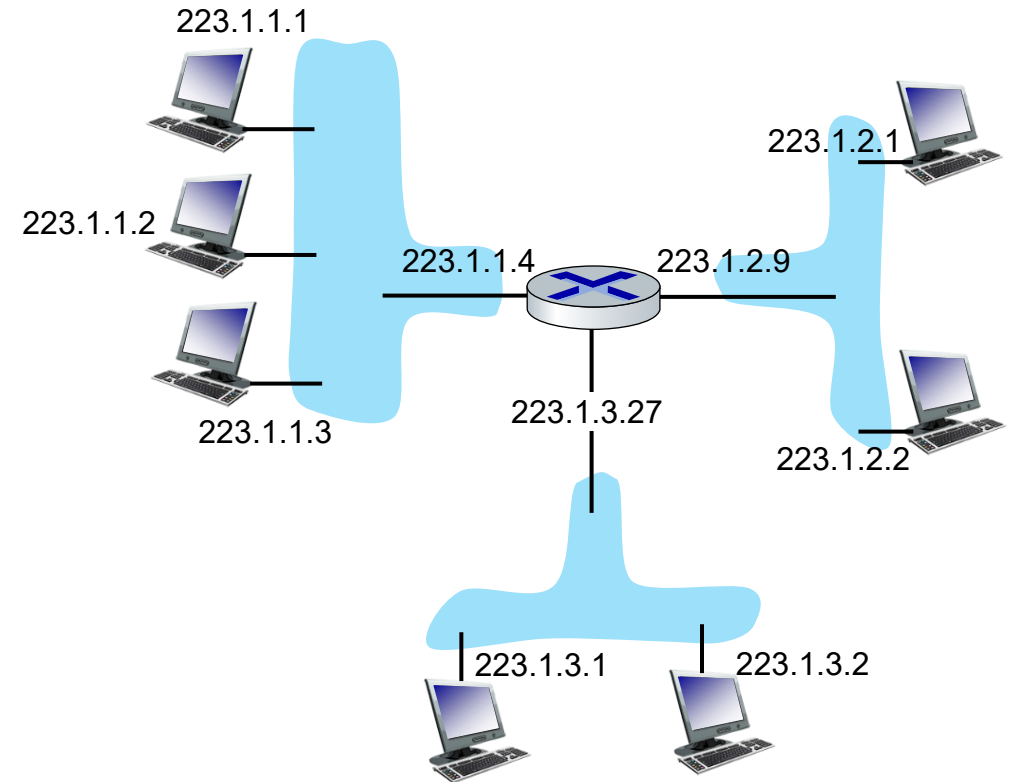
dotted-decimal IP address notation:

223.1.1.1 = 11011111 00000001 00000001 00000001

223 1 1 1

IP addressing: introduction

- **IP address:** 32-bit identifier associated with each host or router *interface*
- **interface:** connection between host/router and physical link
 - router's typically have multiple interfaces
 - host typically has one or two interfaces (e.g., wired Ethernet, wireless 802.11)



dotted-decimal IP address notation:

223.1.1.1 = 11011111 00000001 00000001 00000001

223 1 1 1

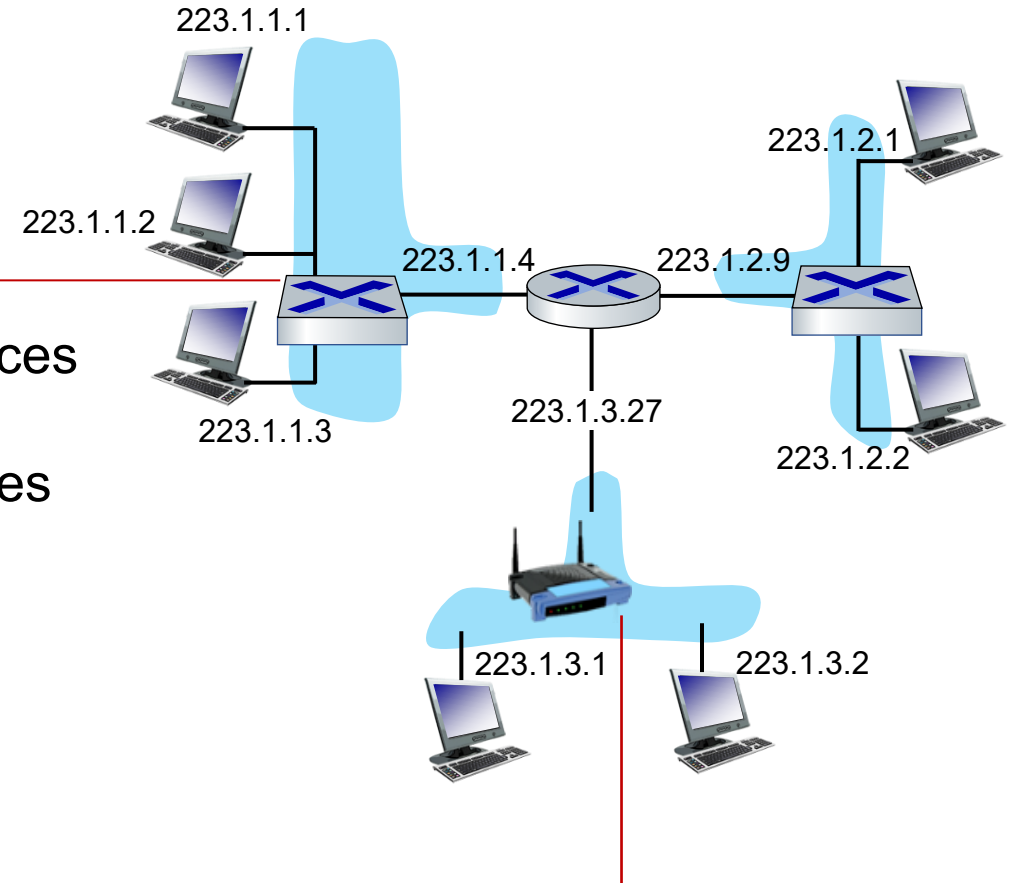
IP addressing: introduction

Q: how are interfaces actually connected?

A: we'll learn about that in chapters 6, 7

For now: don't need to worry about how one interface is connected to another (with no intervening router)

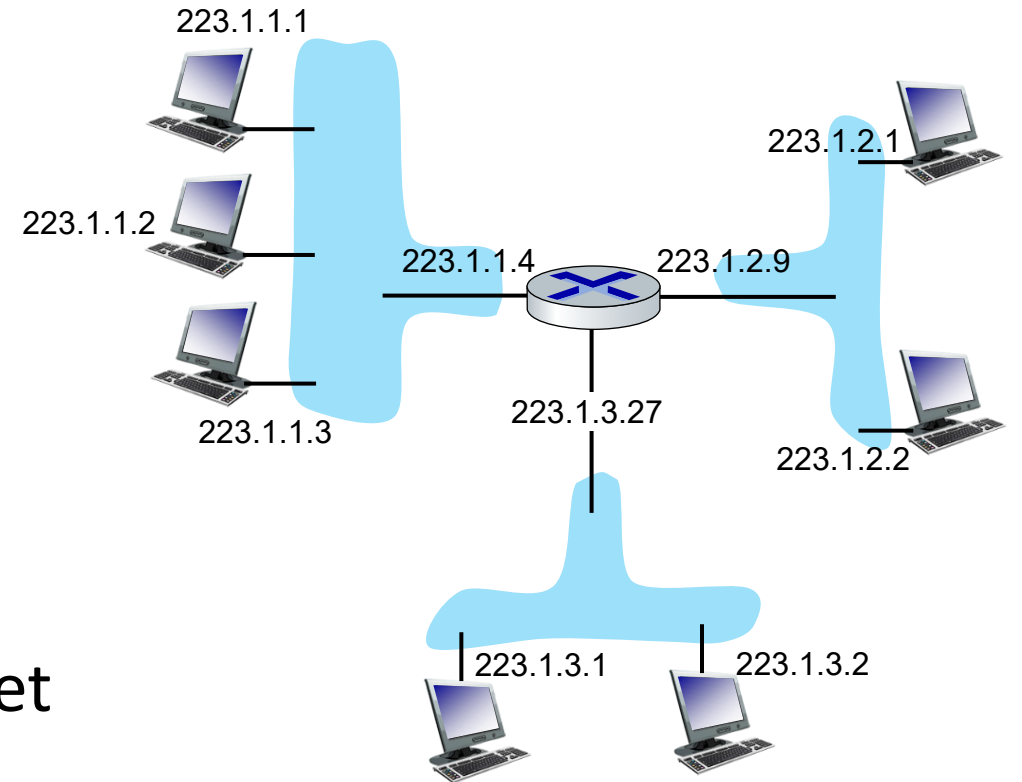
A: wired Ethernet interfaces connected by Ethernet switches



A: wireless WiFi interfaces connected by WiFi base station

Subnets

- *What's a subnet ?*
 - device interfaces that can physically reach each other **without passing through an intervening router**
- IP addresses have structure:
 - **subnet part**: devices in same subnet have common high order bits
 - **host part**: **remaining** low order bits

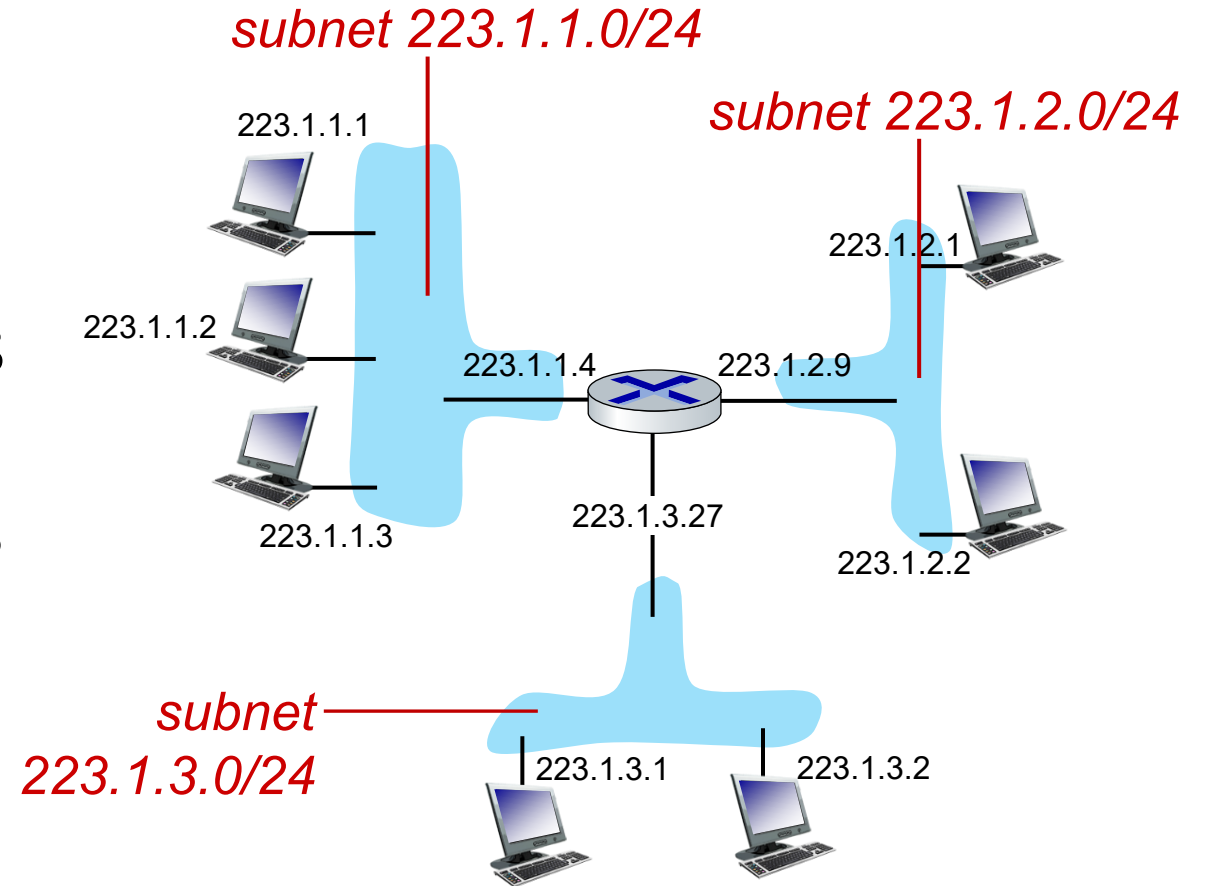


network consisting of 3 subnets

Subnets

Recipe for defining subnets:

- detach each interface from its host or router, creating “islands” of isolated networks
- each isolated network is called a *subnet*

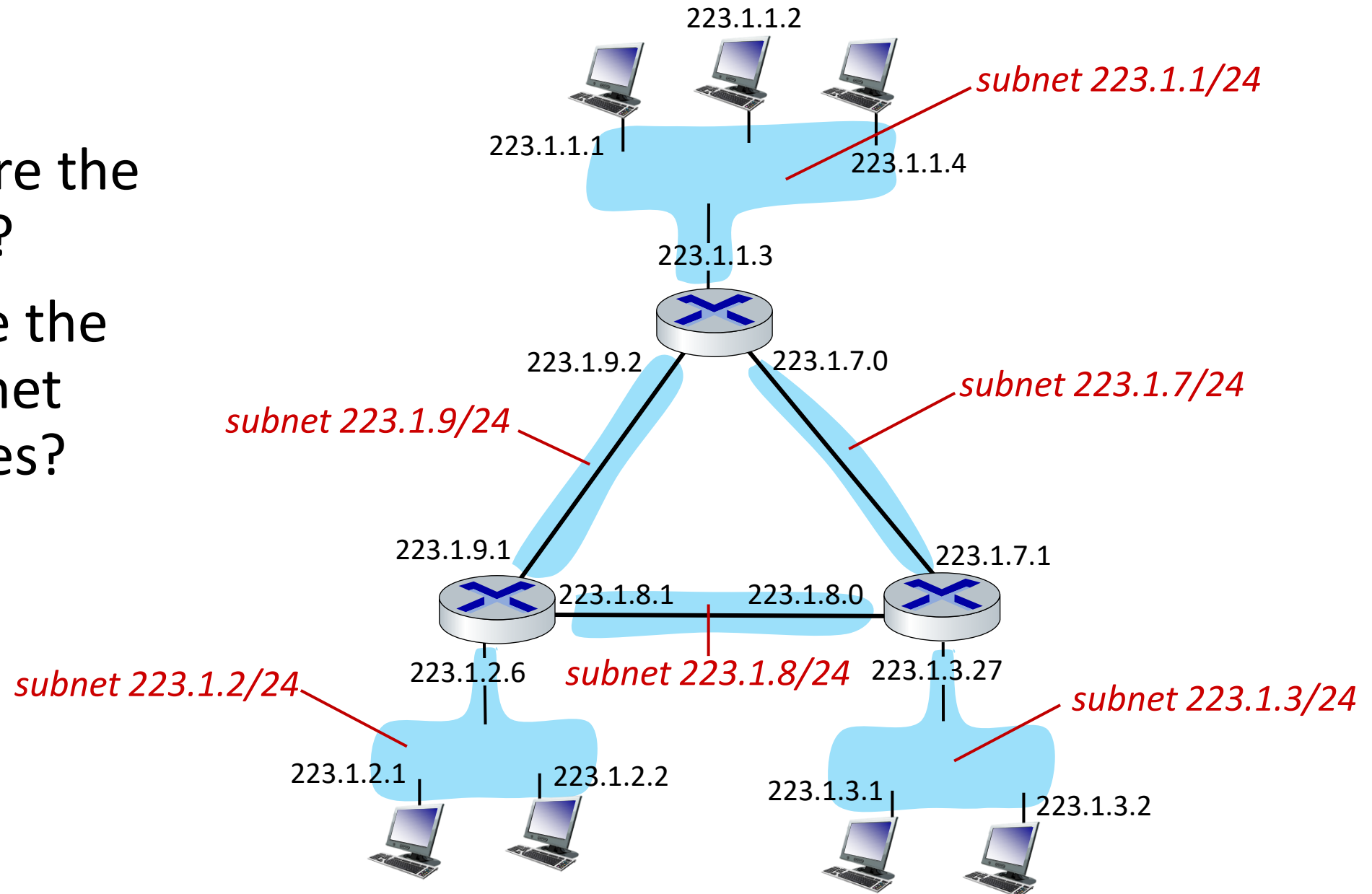


subnet mask: /24

(high-order 24 bits: subnet part of IP address)

Subnets

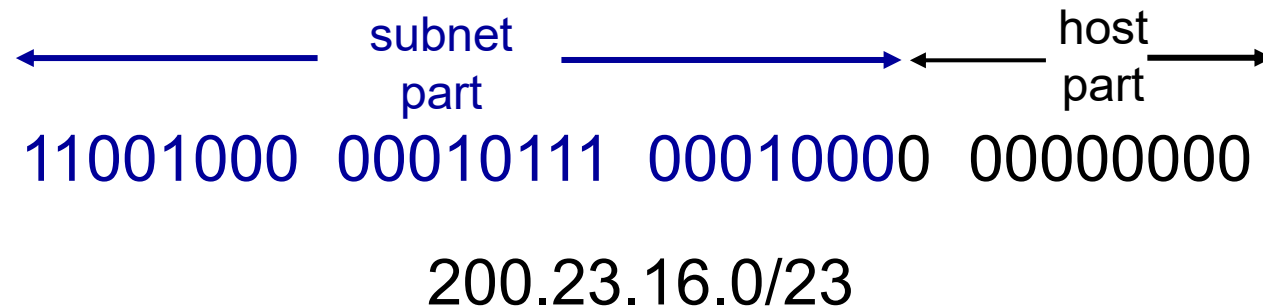
- where are the subnets?
- what are the /24 subnet addresses?



IP addressing: CIDR

CIDR: Classless **I**nter**D**omain **R**outing (pronounced “cider”)

- subnet portion of address of arbitrary length
- address format: **a.b.c.d/x**, where x is # bits in subnet portion of address



IP addresses: how to get one?

That's actually **two** questions:

1. Q: How does a *host* get IP address within its network (host part of address)?
2. Q: How does a *network* get IP address for itself (network part of address)?

How does *host* get IP address?

- hard-coded by sysadmin in config file (e.g., /etc/rc.config in UNIX)
- **DHCP**: Dynamic Host Configuration Protocol: dynamically get address from as server
 - “plug-and-play”

DHCP: Dynamic Host Configuration Protocol

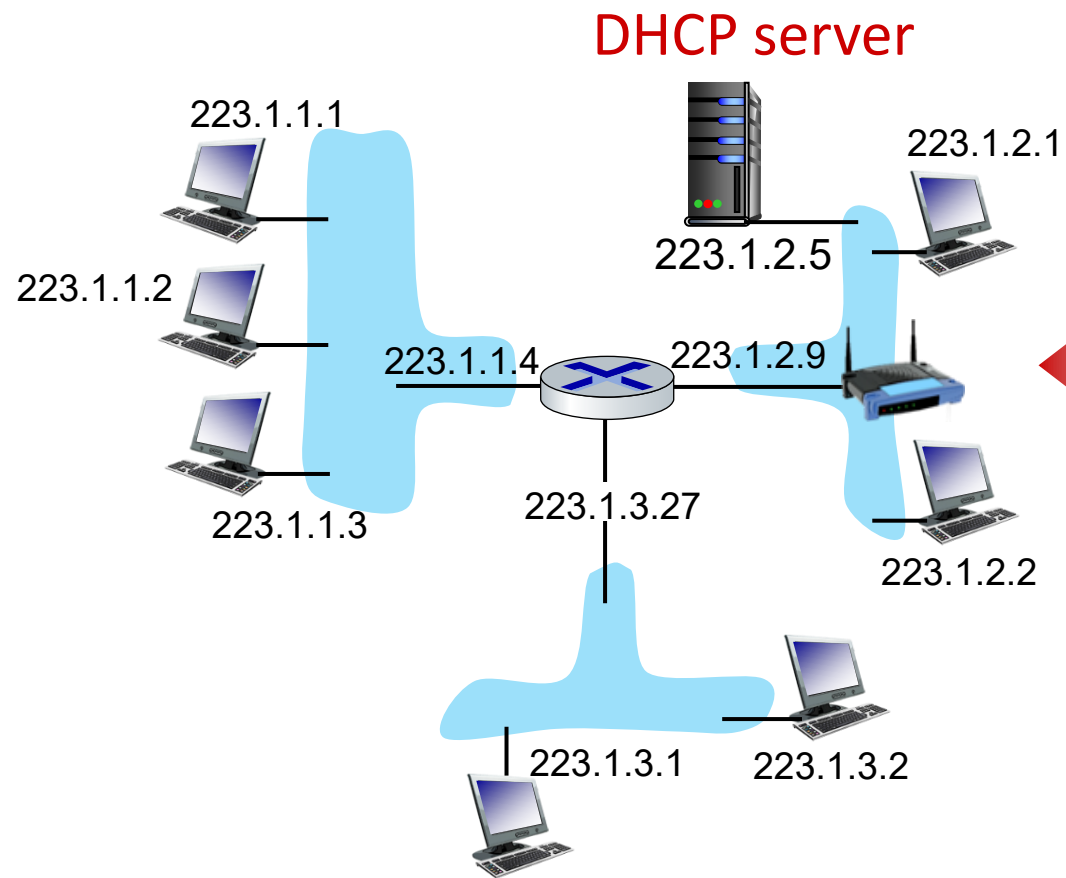
goal: host *dynamically* obtains IP address from network server when it “joins” network

- can renew its lease on address in use
- allows reuse of addresses (only hold address while connected/on)
- support for mobile users who join/leave network

DHCP overview:

- host broadcasts **DHCP discover** msg [optional]
- DHCP server responds with **DHCP offer** msg [optional]
- host requests IP address: **DHCP request** msg
- DHCP server sends address: **DHCP ack** msg

DHCP client-server scenario



Typically, DHCP server will be co-located in router, serving all subnets to which router is attached



arriving **DHCP client** needs address in this network

DHCP client-server scenario

DHCP server: 223.1.2.5



DHCP discover

Broadcast: is there a
DHCP server out there?

Arriving client



DHCP offer

Broadcast: I'm a DHCP
server! Here's an IP
address you can use
...

DHCP request

Broadcast: OK. I would
like to use this IP address!

DHCP ACK

Broadcast: OK. You've
got that IP address!

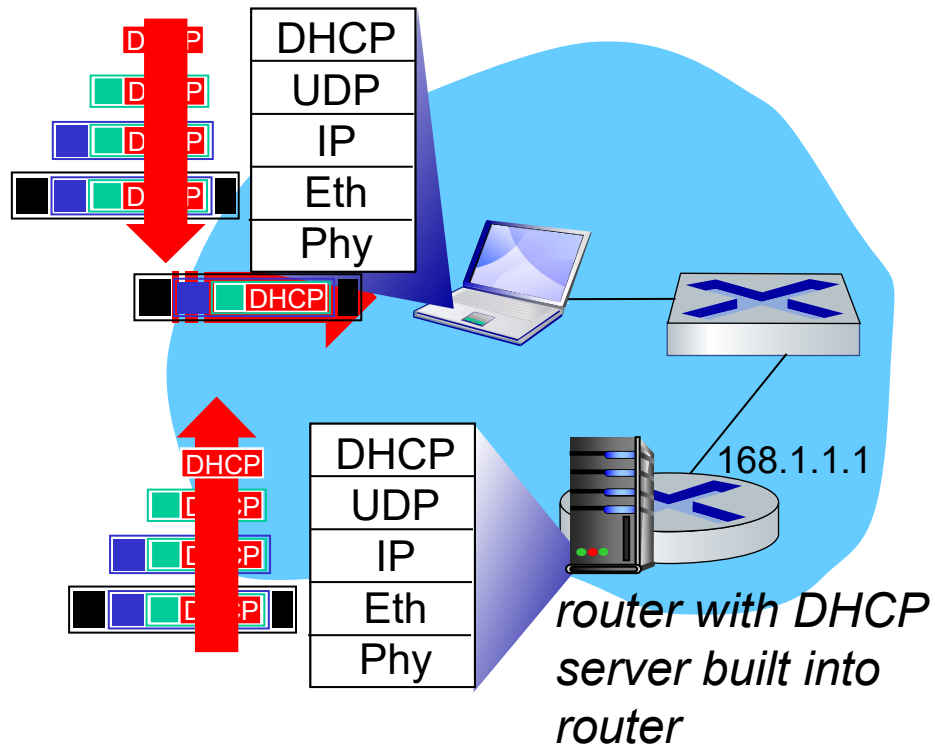
The two steps above can
be skipped "if a client
remembers and wishes to
reuse a previously
allocated network address"
[RFC 2131]

DHCP: more than IP addresses

DHCP can return more than just allocated IP address on subnet:

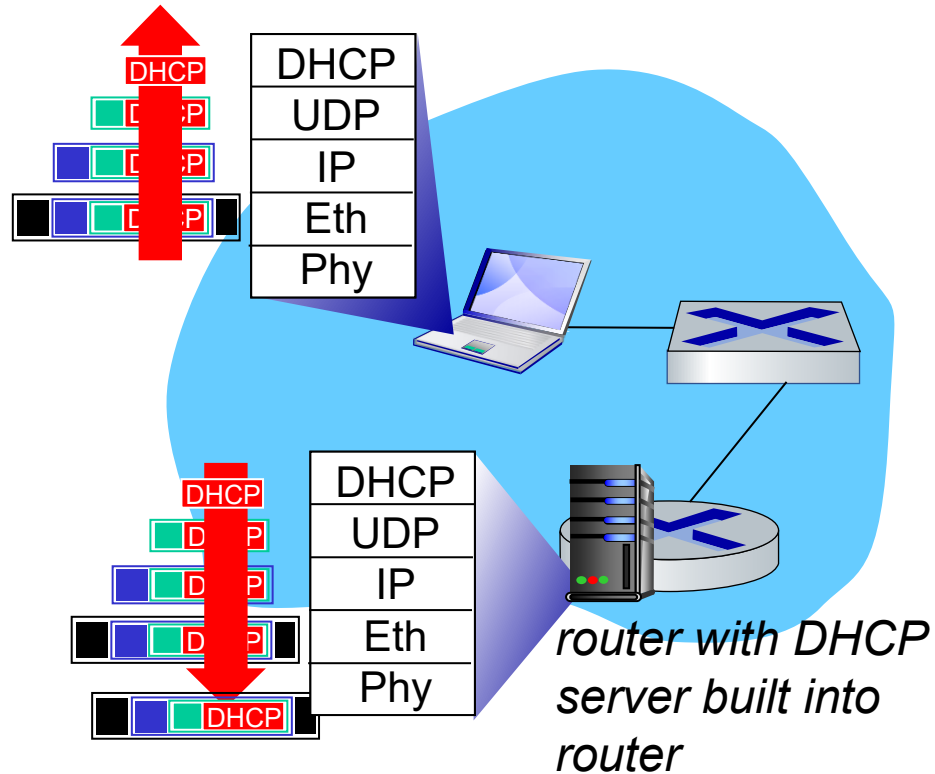
- address of first-hop router for client
- name and IP address of DNS sever
- network mask (indicating network versus host portion of address)

DHCP: example



- Connecting laptop will use DHCP to get IP address, address of first-hop router, address of DNS server.
- DHCP REQUEST message encapsulated in UDP, encapsulated in IP, encapsulated in Ethernet
- Ethernet frame broadcast (dest: FFFFFFFF) on LAN, received at router running DHCP server
- Ethernet demux'ed to IP demux'ed, UDP demux'ed to DHCP

DHCP: example



- DCP server formulates DHCP ACK containing client's IP address, IP address of first-hop router for client, name & IP address of DNS server
- encapsulated DHCP server reply forwarded to client, demuxing up to DHCP at client
- client now knows its IP address, name and IP address of DNS server, IP address of its first-hop router

IP addresses: how to get one?

Q: how does *network* get subnet part of IP address?

A: gets allocated portion of its provider ISP's address space

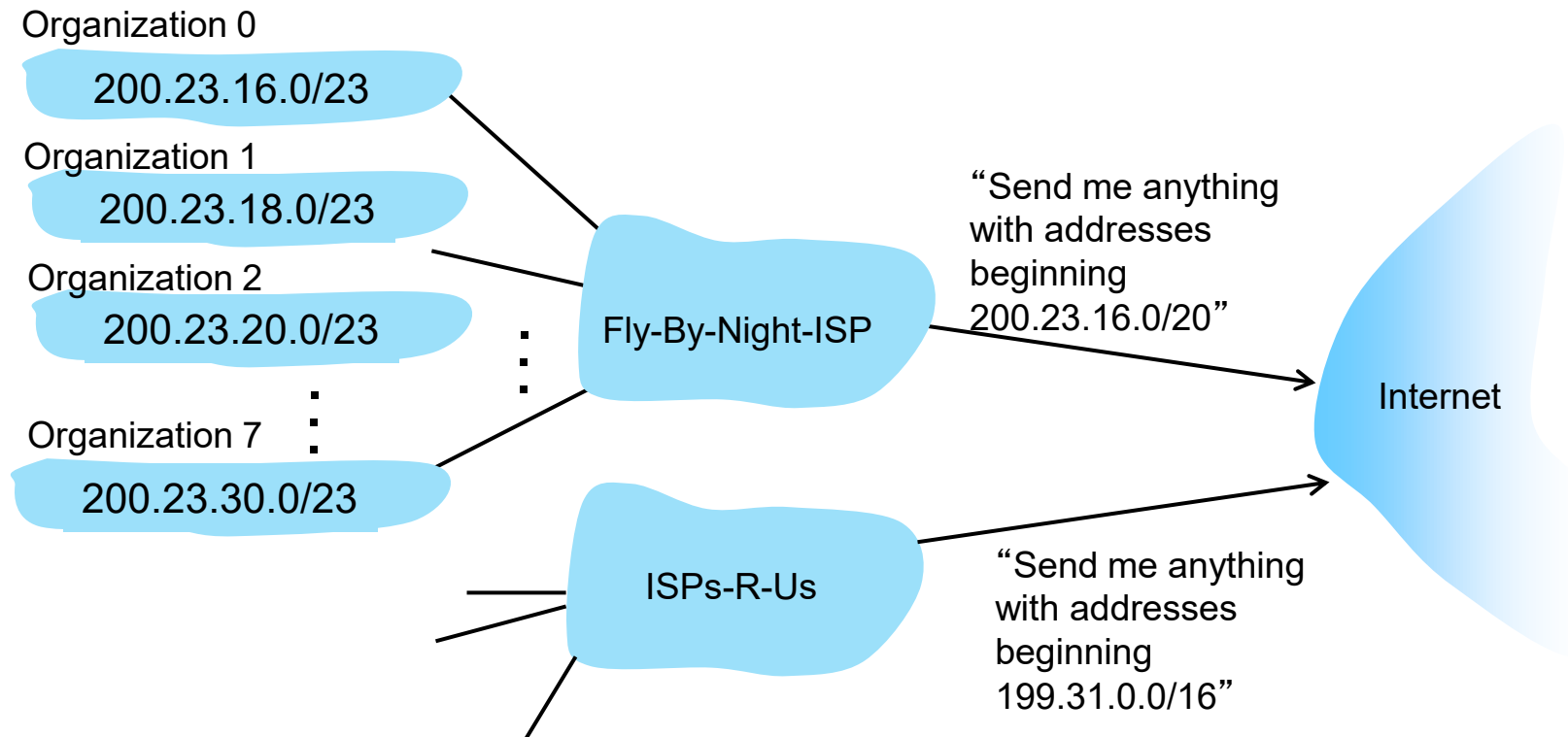
ISP's block 11001000 00010111 00010000 00000000 200.23.16.0/20

ISP can then allocate out its address space in 8 blocks:

Organization 0	<u>11001000 00010111 00010000</u>	00000000	200.23.16.0/23
Organization 1	<u>11001000 00010111 00010010</u>	00000000	200.23.18.0/23
Organization 2	<u>11001000 00010111 00010100</u>	00000000	200.23.20.0/23
...
Organization 7	<u>11001000 00010111 00011110</u>	00000000	200.23.30.0/23

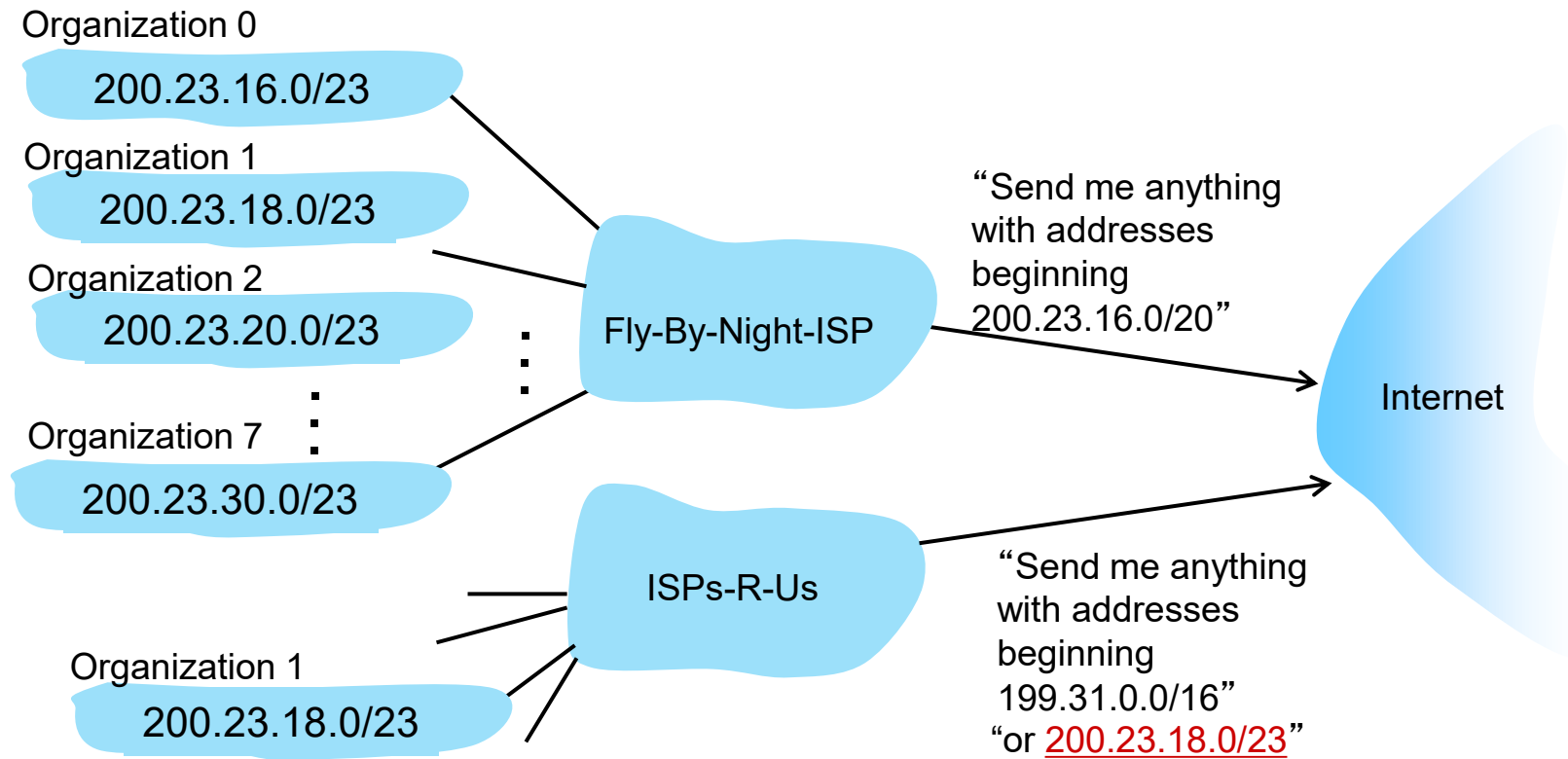
Hierarchical addressing: route aggregation

hierarchical addressing allows efficient advertisement of routing information:



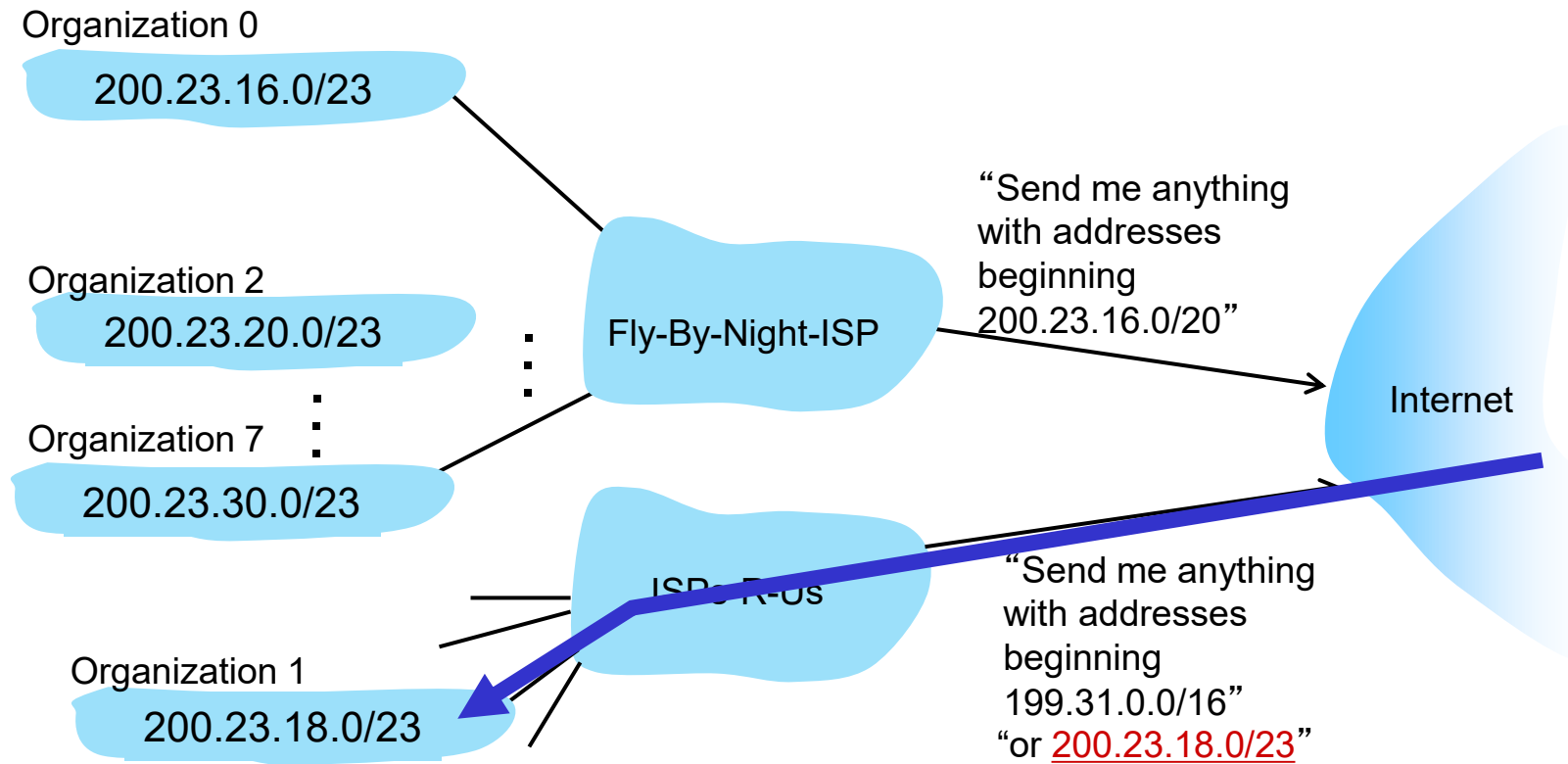
Hierarchical addressing: more specific routes

- Organization 1 moves from Fly-By-Night-ISP to ISPs-R-Us
- ISPs-R-Us now advertises a more specific route to Organization 1



Hierarchical addressing: more specific routes

- Organization 1 moves from Fly-By-Night-ISP to ISPs-R-Us
- ISPs-R-Us now advertises a more specific route to Organization 1



IP addressing: last words ...

Q: how does an ISP get block of addresses?

A: ICANN: Internet Corporation for Assigned Names and Numbers
<http://www.icann.org/>

- allocates IP addresses, through 5 regional registries (RRs) (who may then allocate to local registries)
- manages DNS root zone, including delegation of individual TLD (.com, .edu , ...) management

Q: are there enough 32-bit IP addresses?

- ICANN allocated last chunk of IPv4 addresses to RRs in 2011
- NAT (next) helps IPv4 address space exhaustion
- IPv6 has 128-bit address space

"Who the hell knew how much address space we needed?" Vint Cerf (reflecting on decision to make IPv4 address 32 bits long)

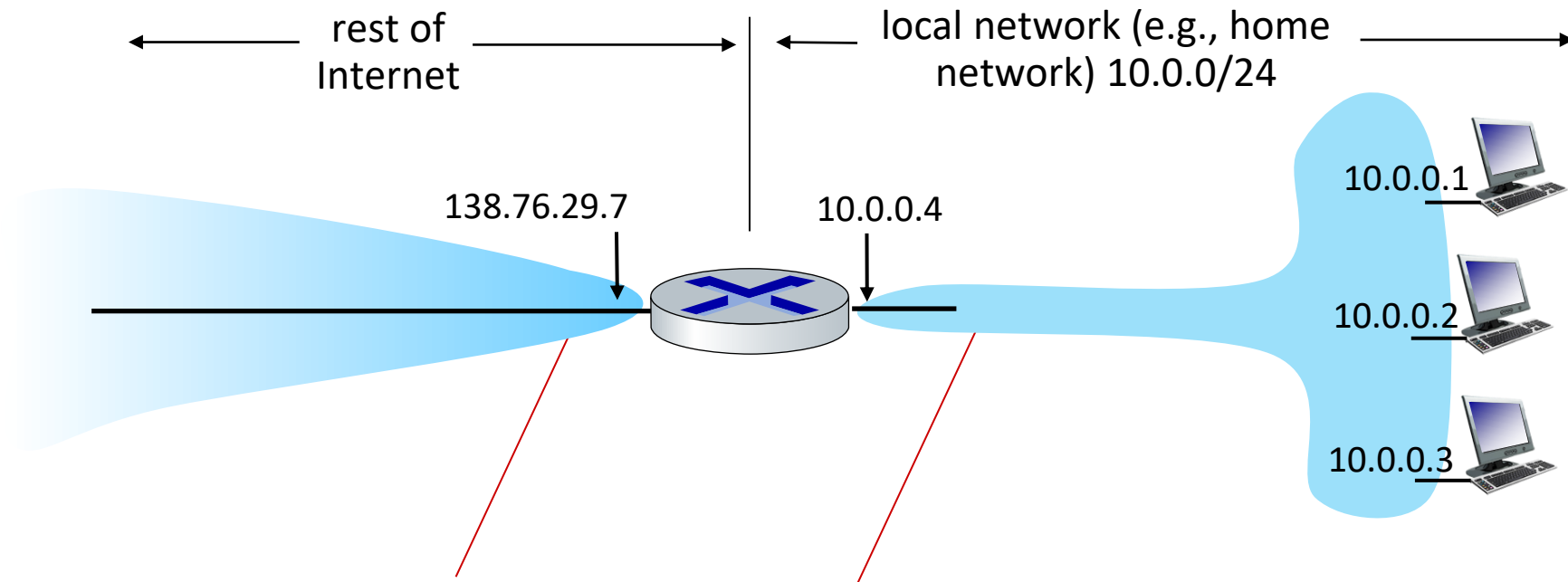
Mặt dữ liệu (Data plane): Lộ trình

- Lớp mạng: Tổng quan
 - Mặt dữ liệu
 - Mặt điều khiển
- Có gì bên trong một router?
 - Cổng vào, kết cấu chuyển mạch, cổng ra
 - Quản lý bộ nhớ đệm
- **IP: Giao thức Internet**
 - Định dạng gói tin datagram
 - Địa chỉ
 - **Dịch địa chỉ mạng**
 - **IPv6**



NAT: Dịch địa chỉ mạng

NAT: Tất cả các thiết bị trong mạng cục bộ chỉ chia sẻ **một** địa chỉ IPv4 khi có liên quan đến thế giới bên ngoài



Tất cả các datagram *rời khỏi* mạng cục bộ có *cùng* địa chỉ IP NAT nguồn: 138.76.29.7, nhưng số cổng dịch vụ nguồn *khác nhau*

Datagram với nguồn hoặc đích trong mạng này có địa chỉ 10.0.0/24 cho nguồn, đích (như thường lệ)

NAT: Dịch địa chỉ mạng

- Tất cả các thiết bị trong mạng cục bộ đều có địa chỉ 32 bits trong không gian địa chỉ IP “**riêng**” (tên tổ 10/8, 172.16/12, 192.168/16) chỉ có thể được sử dụng trong mạng cục bộ
- Ưu điểm:
 - Chỉ cần **một** địa chỉ IP từ nhà cung cấp (ISP) cho tất cả các thiết bị
 - Có thể thay đổi địa chỉ của host trong mạng cục bộ mà không cần thông báo cho thế giới bên ngoài
 - Có thể thay đổi ISP mà không cần thay đổi địa chỉ của các thiết bị trong mạng nội bộ
 - Bảo mật: các thiết bị bên trong mạng cục bộ không thể dịch địa chỉ trực tiếp, thế giới bên ngoài không có thể nhìn thấy mạng cục bộ

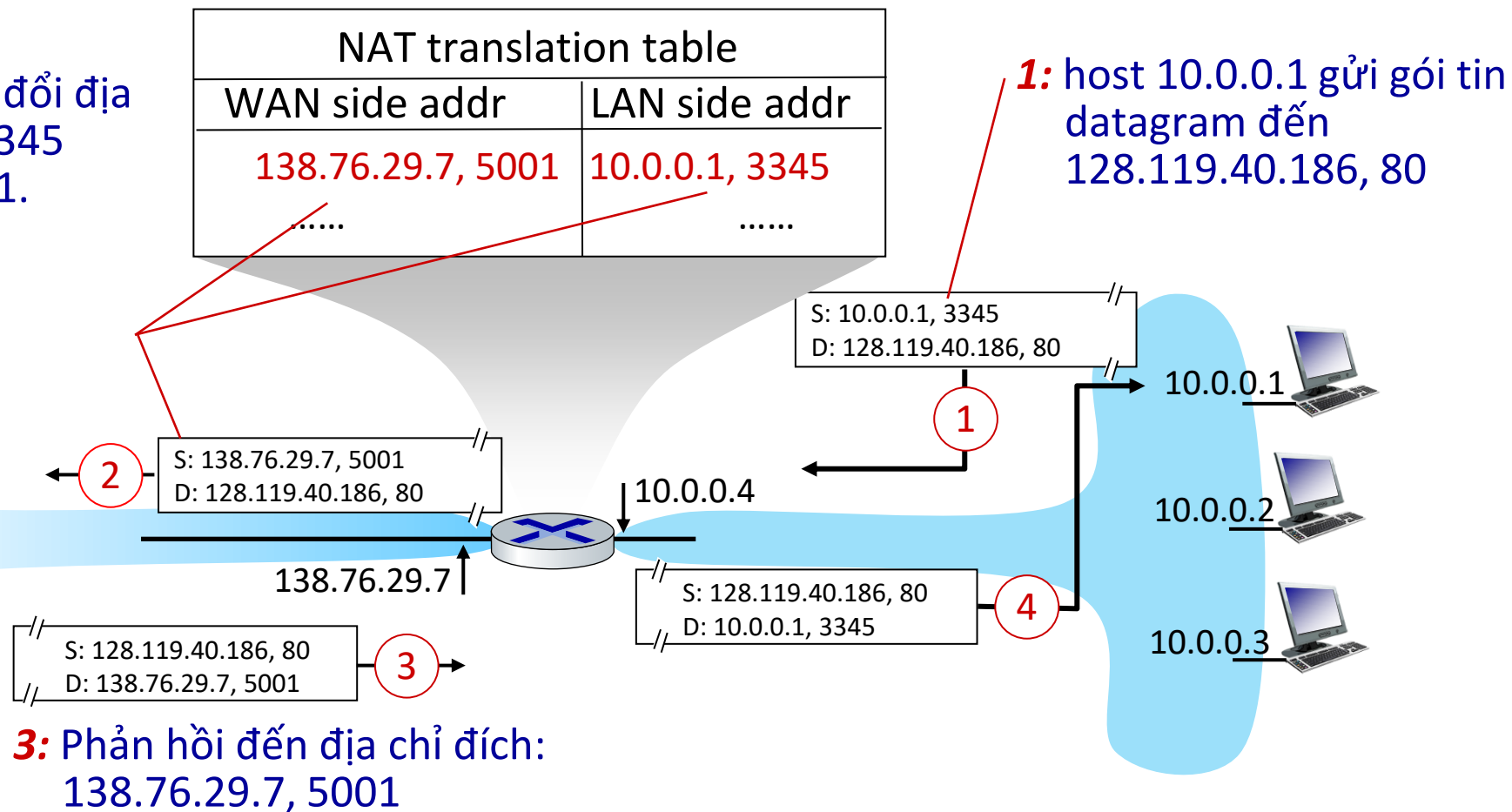
NAT: Dịch địa chỉ mạng

Thực thi: Bộ định tuyến NAT phải (trong suốt):

- **Các gói tin datagram đi: thay thế** (địa chỉ IP nguồn, cổng dịch vụ #) của mọi datagram đi thành (địa chỉ NAT IP, cổng dịch vụ # mới)
 - Clients/servers từ xa sẽ phản hồi bằng cách sử dụng (địa chỉ NAT IP, số cổng dịch vụ mới) làm địa chỉ đích
- **Ghi nhớ (trong bảng dịch NAT)** mọi cặp dịch (địa chỉ IP nguồn, cổng dịch vụ #) sang (địa chỉ NAT IP, cổng dịch vụ # mới)
- **Các gói tin datagram đến: thay thế** (địa chỉ IP NAT, cổng dịch vụ mới) trong các trường thông tin đích của mọi gói tin datagram đến bằng (địa chỉ IP nguồn, cổng dịch vụ #) tương ứng được lưu trữ trong bảng NAT

NAT: Dịch địa chỉ mạng

2: Bộ định tuyến NAT thay đổi địa chỉ nguồn từ 10.0.0.1, 3345 thành 138.76.29.7, 5001.
Cập nhật bảng NAT



NAT: Dịch địa chỉ mạng

- NAT đã gây tranh cãi:
 - Bộ định tuyến chỉ “nên” xử lý tối đa tới lớp mạng
 - Địa chỉ “thiếu” cần được giải quyết bằng IPv6
 - Vi phạm đối số end-to-end (sửa đổi cổng dịch vụ # bởi thiết bị lớp mạng)
 - Truyền tải NAT: nếu máy khách muốn kết nối với máy chủ sau NAT thì sao?
- Nhưng NAT vẫn ở đây:
 - Được sử dụng rộng rãi trong mạng gia đình và tổ chức, mạng di động 4G/5G

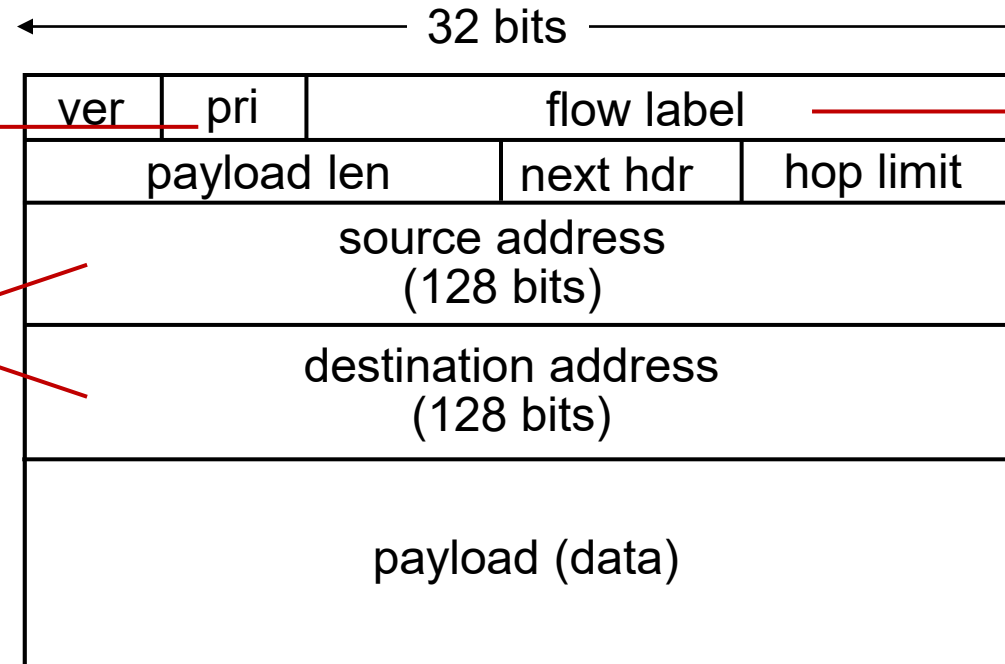
IPv6: Động lực

- **Động lực ban đầu:** không gian địa chỉ IPv4 32-bit sẽ được phân bổ hoàn toàn (phân bổ hết – cạn kiệt)
- Động lực bổ sung:
 - Tốc độ xử lý/chuyển tiếp: tiêu đề có độ dài cố định 40 byte
 - Cho phép xử lý các “luồng” ở lớp mạng khác nhau

IPv6 datagram format

priority: identify priority among datagrams in flow

128-bit IPv6 addresses



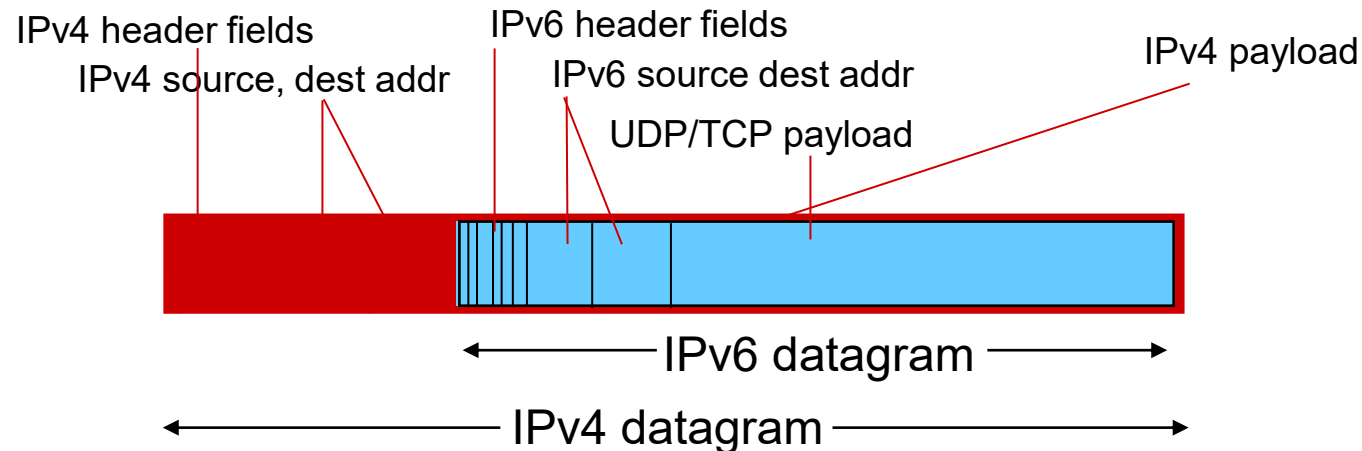
flow label: identify datagrams in same "flow." (concept of "flow" not well defined).

What's missing (compared with IPv4):

- no checksum (to speed processing at routers)
- no fragmentation/reassembly
- no options (available as upper-layer, next-header protocol at router)

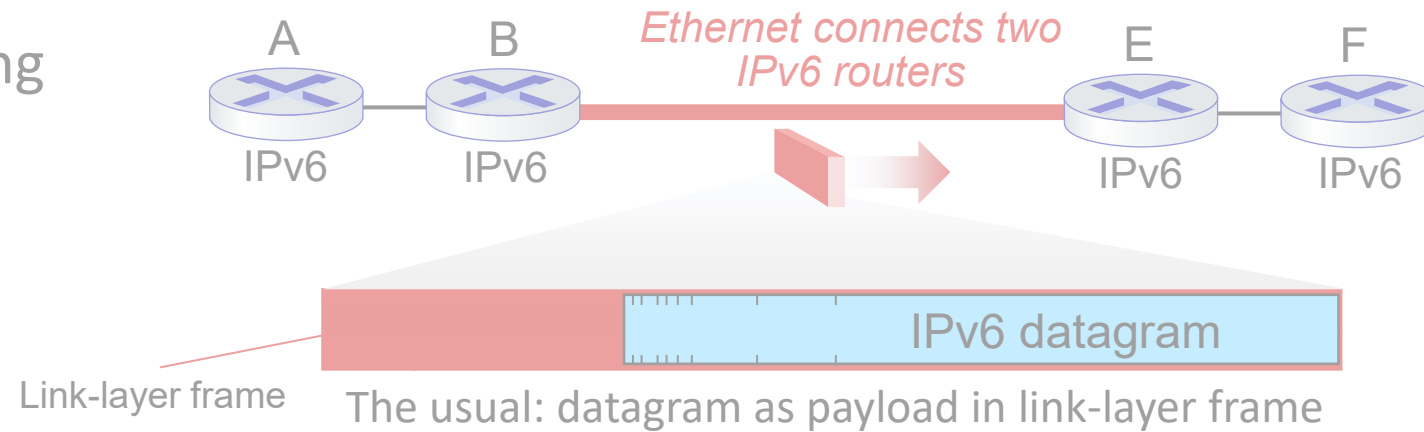
Transition from IPv4 to IPv6

- not all routers can be upgraded simultaneously
 - no “flag days”
 - how will network operate with mixed IPv4 and IPv6 routers?
- **tunneling**: IPv6 datagram carried as *payload* in IPv4 datagram among IPv4 routers (“packet within a packet”)
 - tunneling used extensively in other contexts (4G/5G)

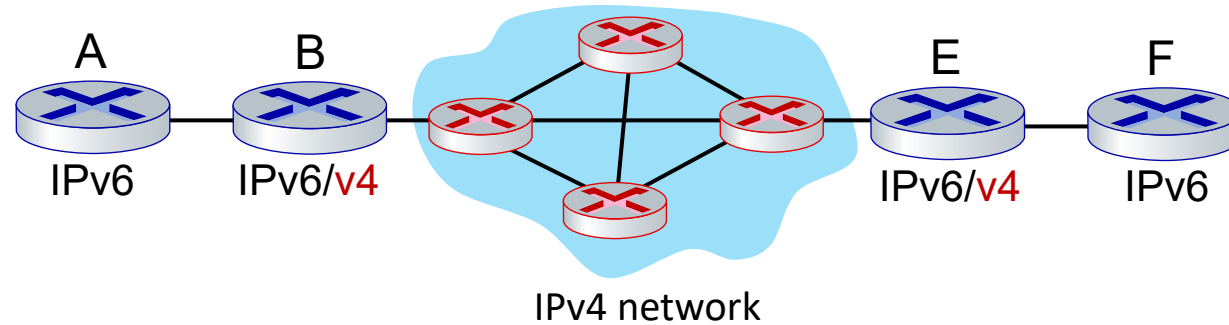


Tunneling and encapsulation

Ethernet connecting two IPv6 routers:

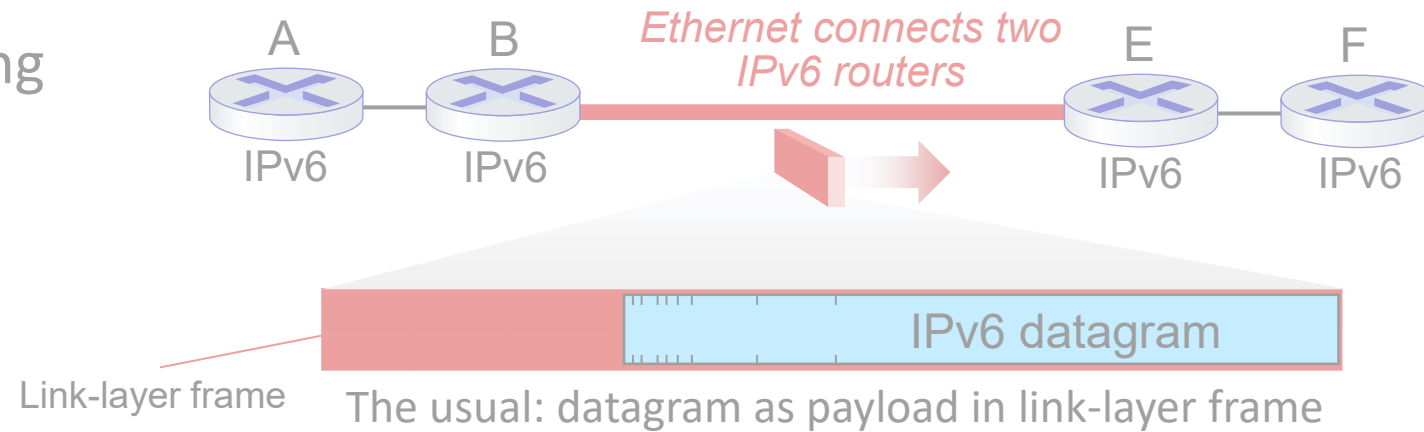


IPv4 network connecting two IPv6 routers

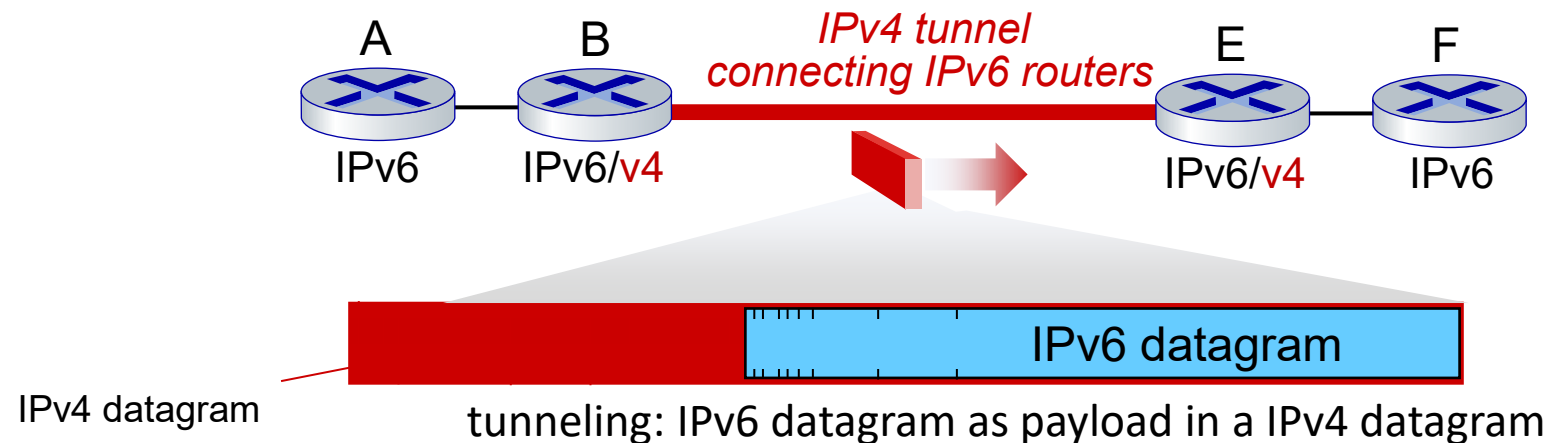


Tunneling and encapsulation

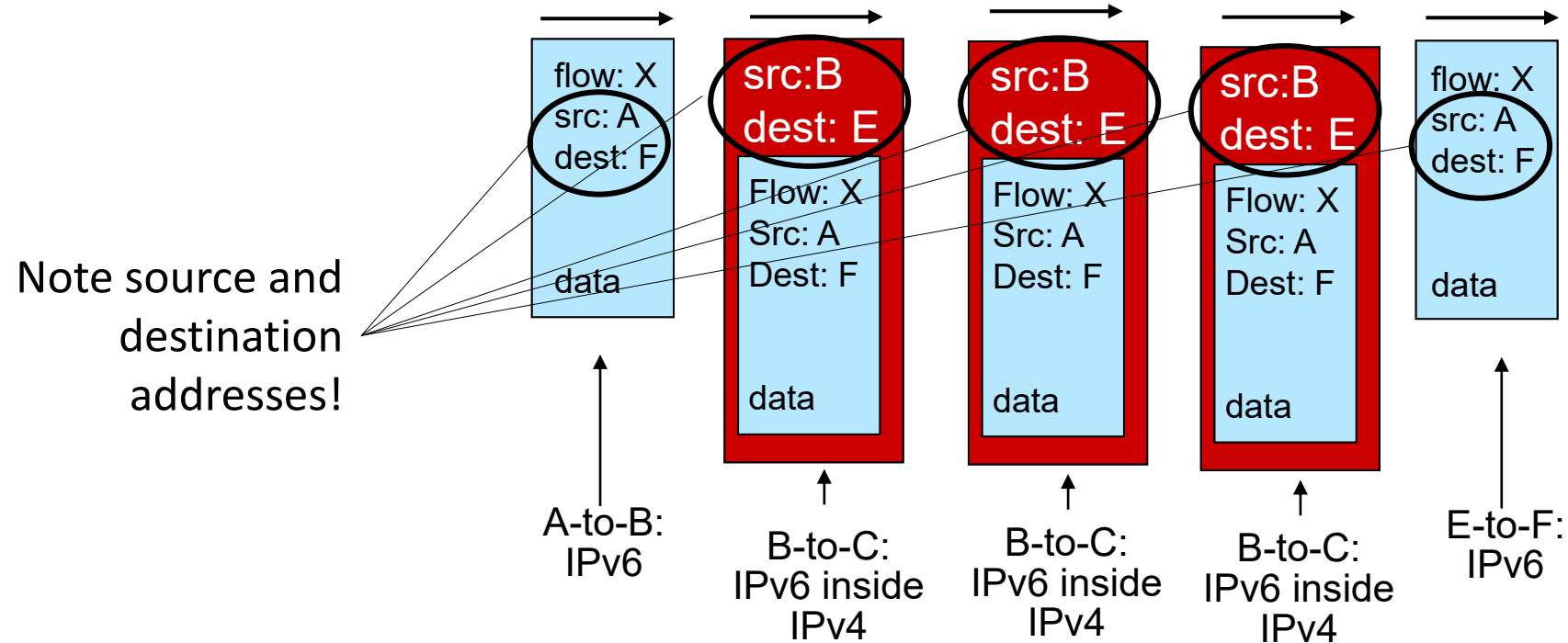
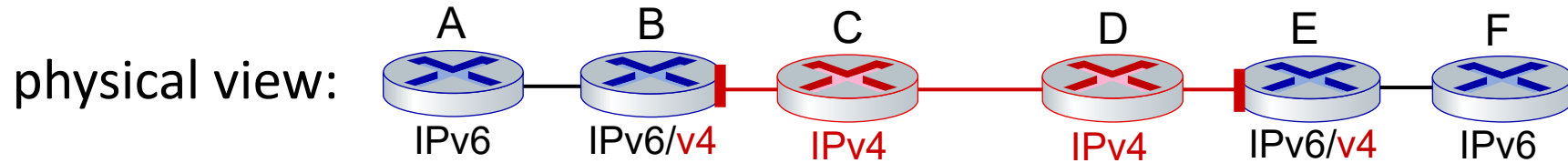
Ethernet connecting two IPv6 routers:



IPv4 tunnel connecting two IPv6 routers



Tunneling

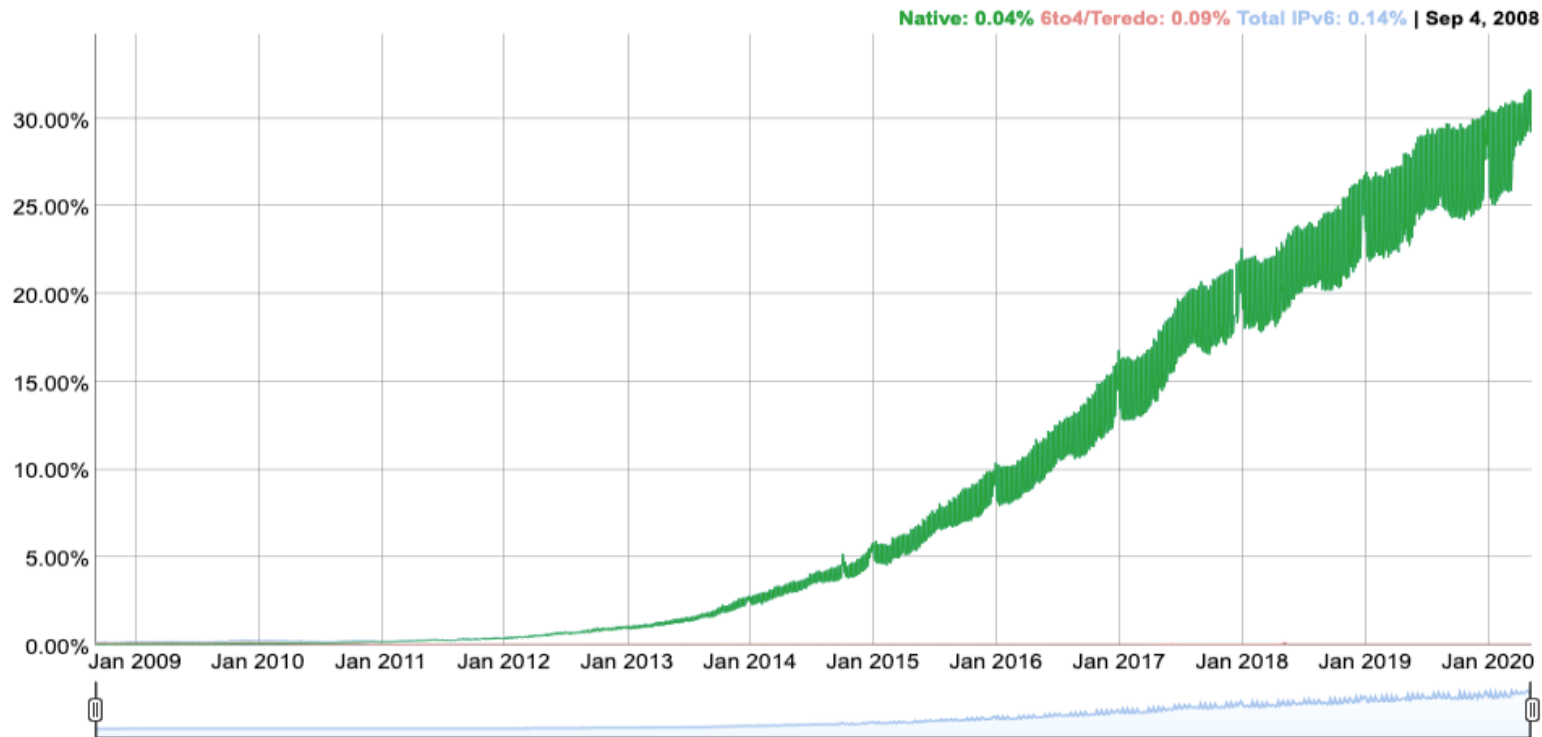


IPv6: Được sử dụng

- Google¹: ~ 30% máy khách truy cập dịch vụ qua IPv6
- NIST: 1/3 miền chính phủ Mỹ là IPv6

IPv6 Adoption

We are continuously measuring the availability of IPv6 connectivity among Google users. The graph shows the percentage of users that access Google over IPv6.



1

<https://www.google.com/intl/en/ipv6/statistics.html>

IPv6: adoption

- Google¹: ~ 30% of clients access services via IPv6
- NIST: 1/3 of all US government domains are IPv6 capable
- Long (long!) time for deployment, use
 - 25 years and counting!
 - think of application-level changes in last 25 years: WWW, social media, streaming media, gaming, telepresence, ...
 - *Why?*

¹ <https://www.google.com/intl/en/ipv6/statistics.html>

Chapter 4: done!

- Lớp mạng: Tổng quan
- Có gì bên trong bộ định tuyến
- IP: Giao thức Internet



Câu hỏi: Bảng chuyển tiếp (chuyển tiếp dựa trên đích đến) được tính như thế nào?

Trả lời: Bảng mặt điều khiển (Chương tiếp theo)