UNIVERSITY OF WATERLOO

CS 456/656
Computer Networks

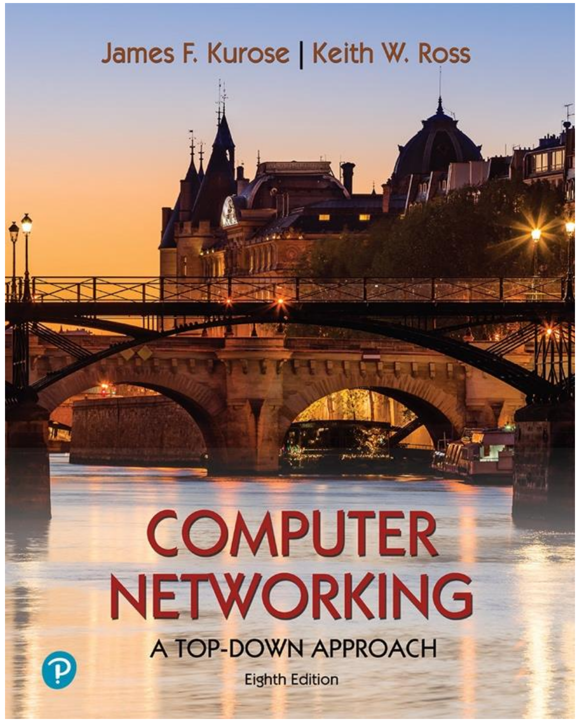
Lecture 11: Network Layer – Part 3

Mina Tahmasbi Arashloo and Uzma Maroof

Fall 2025

A note on the slides

Adapted from the slides that accompany this book.



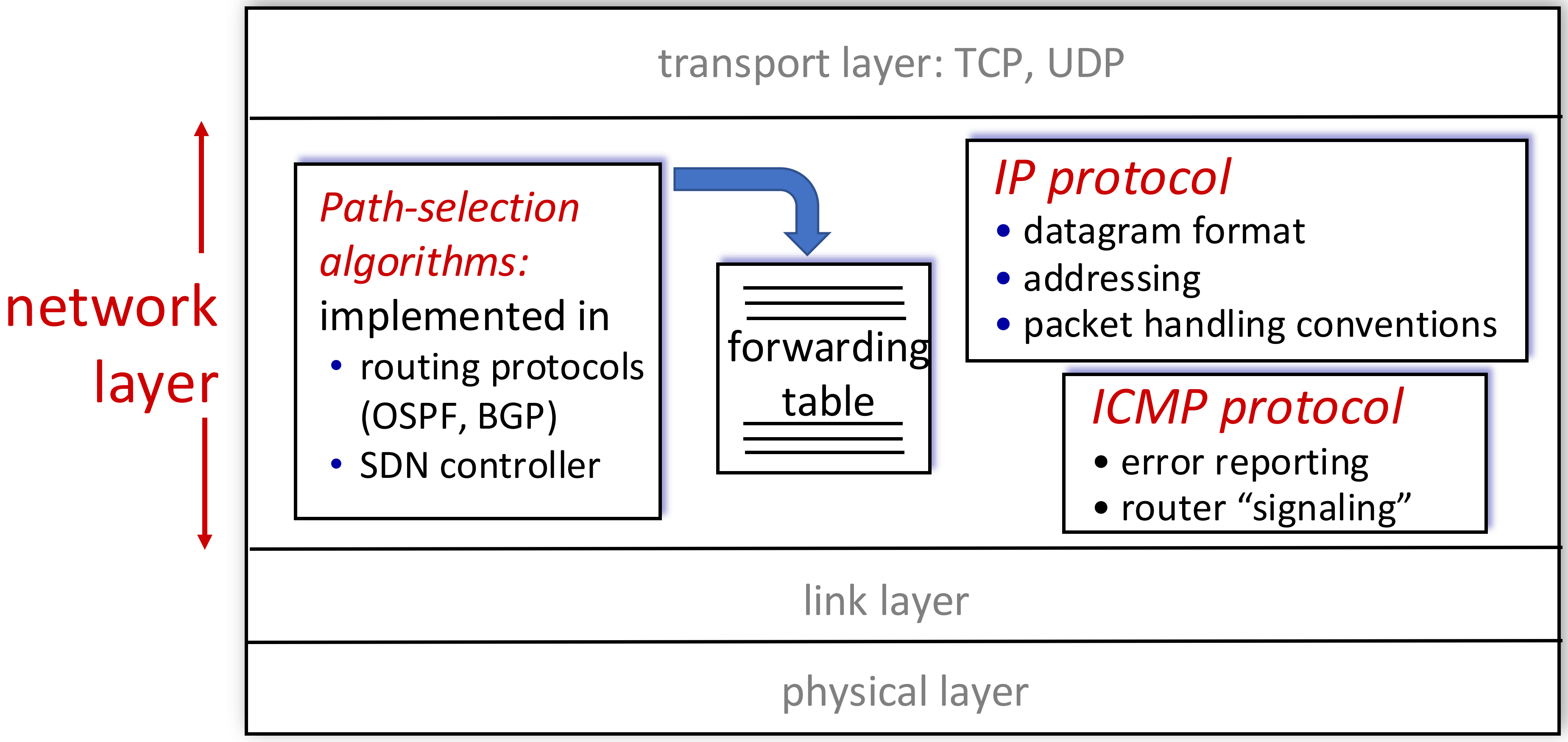
Computer Networking: A Top-Down Approach
8th edition
Jim Kurose, Keith Ross
Pearson, 2020

Network layer: roadmap

- Network layer overview
- Routing algorithms
- <u>Network layer in the Internet</u>

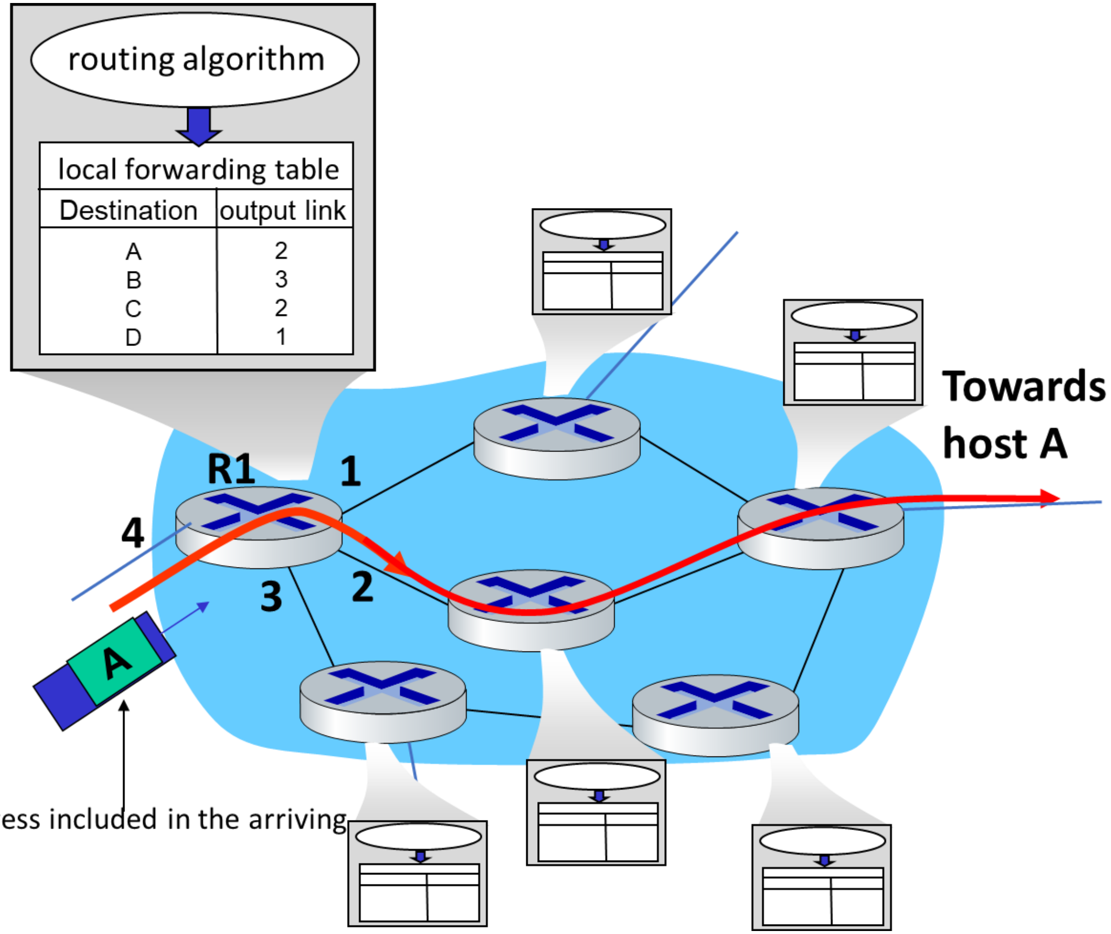
Network Layer in the Internet

host, router network layer functions:



Network Layer in the Internet

- Suppose you want to connect billions of end points together!
- How does this affect addressing, routing, and forwarding in the network layer?



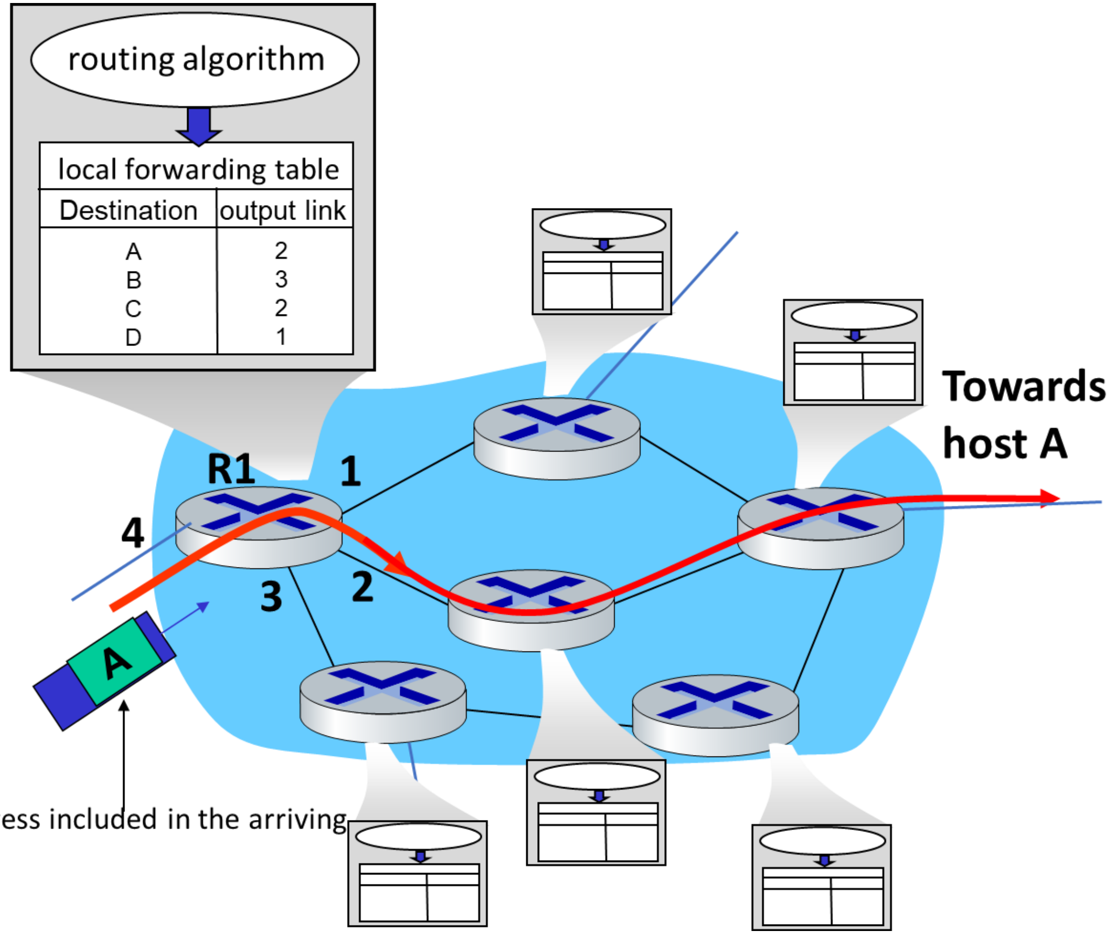
local forwarding table

Destination	output link
A	2
B	3
C	2
D	1

destination address included in the arriving packet.

Network Layer in the Internet

- Suppose you want to connect billions of end points together!
- How does this affect addressing, routing, and forwarding in the network layer?
  - Forwarding tables would have to have billions of entries, one for each address
  - Routing algorithms would have to find a path for each individual address
  - Routing algorithms would have to run across millions of routers
  - …



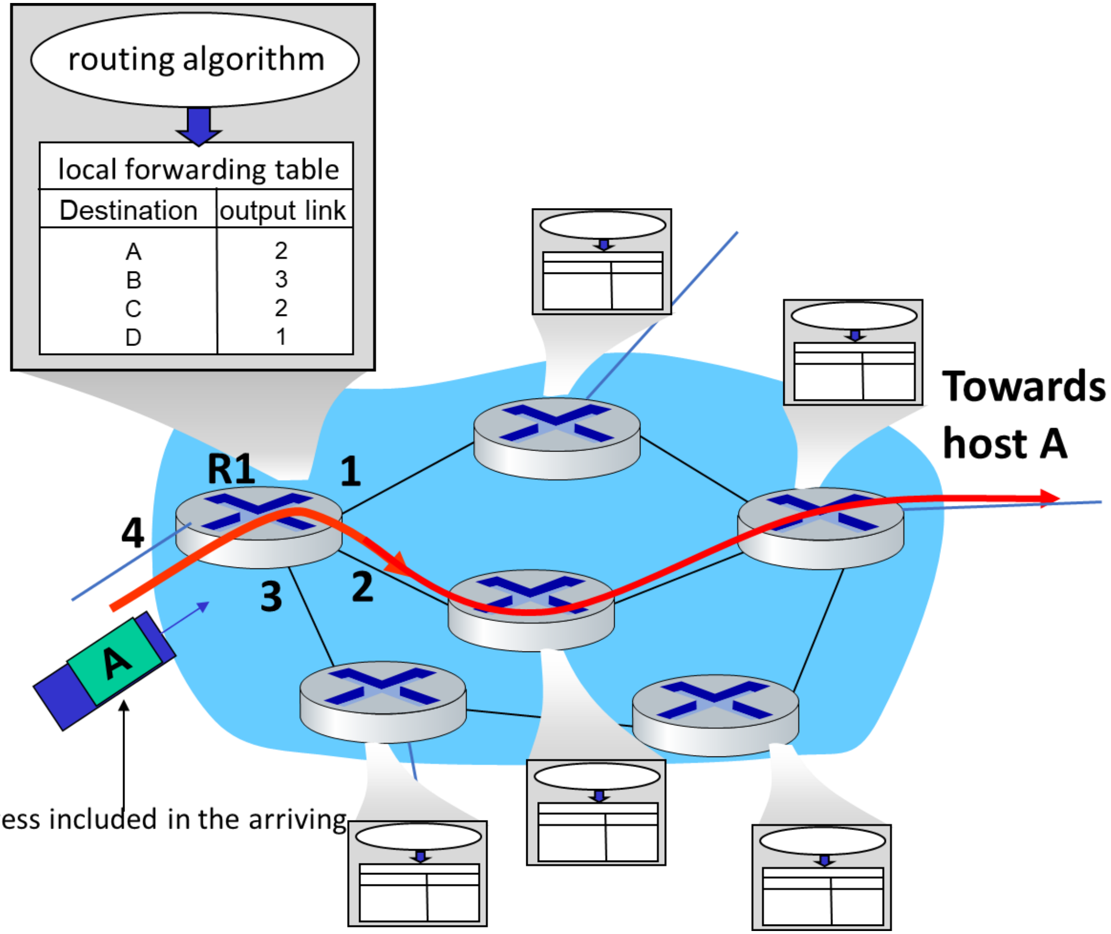
local forwarding table

Destination	output link
A	2
B	3
C	2
D	1

destination address included in the arriving packet.

Network Layer in the Internet

- A "flat" network will not scale!
- Internet and its best-effort network layer are designed with hierarchy in mind
  - Addresses are hierarchical – hosts that are "close" together have addresses that can be combined into one "group identifier"
  - Routing is hierarchical – routers learn how to reach groups of addresses.
  - Forwarding is hierarchical – forwarding tables keep entries about paths to groups of addresses.



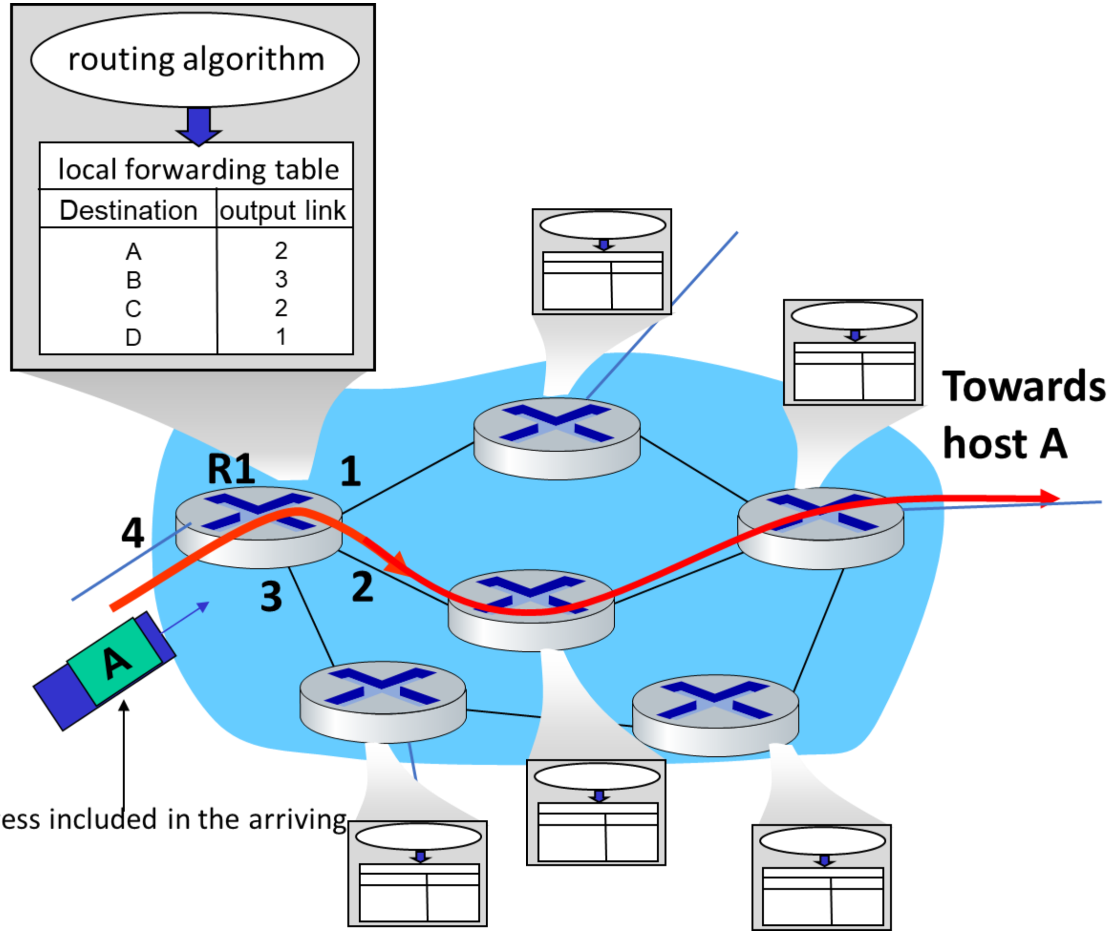
local forwarding table

Destination	output link
A	2
B	3
C	2
D	1

destination address included in the arriving packet.

Network Layer in the Internet

- IP: Internet’s best effort network-layer protocol.
- We’ll discuss:
  - What IP datagrams look like
  - What IP addresses look like
  - How the network finds paths for datagrams going from one IP address to another (i.e., routing algorithms)
  - How routers forward packets
  - …



local forwarding table

Destination	output link
A	2
B	3
C	2
D	1

destination address included in the arriving packet.

Network Layer in the Internet

- The Internet Protocol (IP)
  - Datagram format
  - Addressing
  - Network address translation
  - IPv6
- Internet Routing
- Internet Forwarding

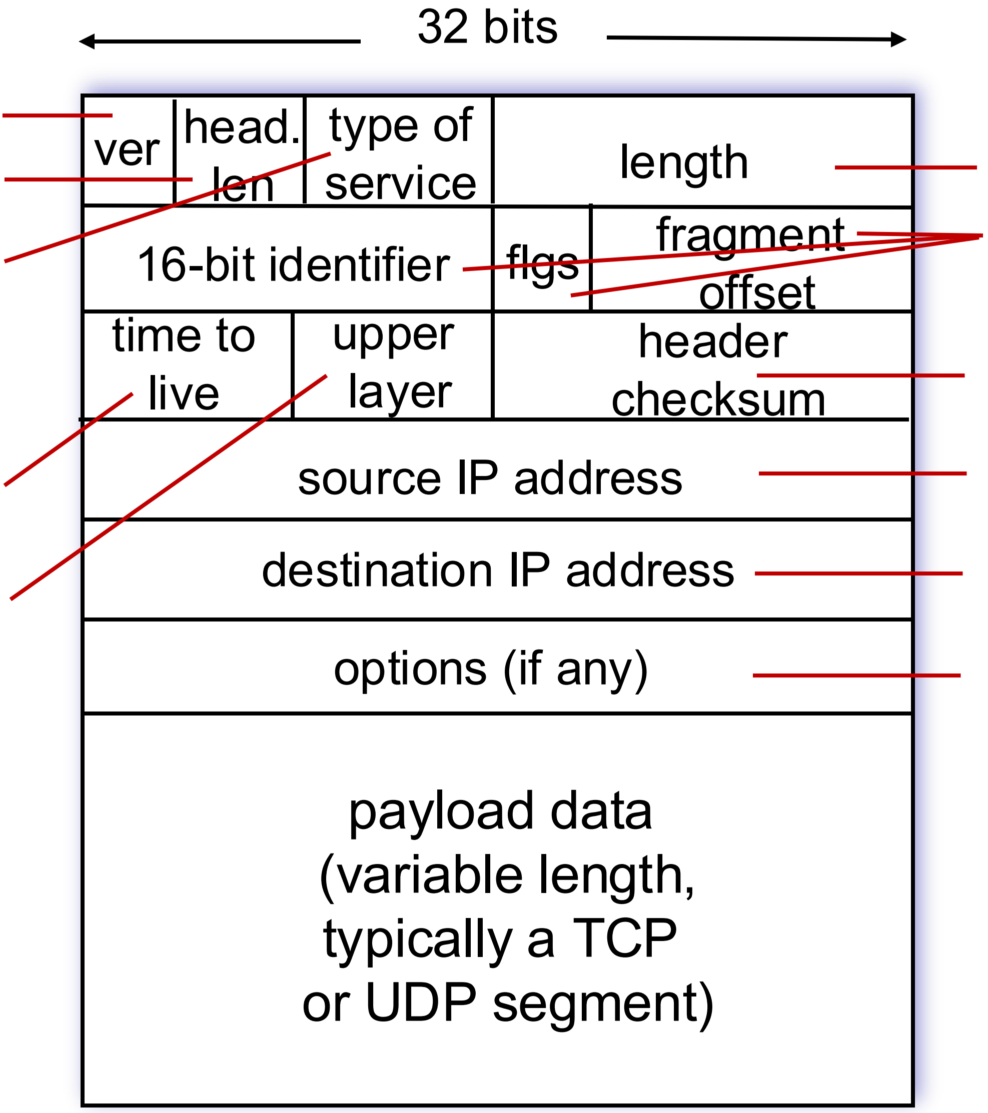
Network Layer in the Internet

- The Internet Protocol (IP)
  - Datagram format
  - Addressing
  - Network address translation
  - IPv6
- Internet Routing
- Internet Forwarding

IP Standard

- IPv4 first and widely deployed Internet protocol
  - many problems in today's Internet
- IPv6 replacement
  - long (long!) time for deployment and use
  - 25 years and counting!
  - think of the much faster application-level changes in last 25 years: WWW, social media, streaming media, gaming, telepresence, …
  - Why? We'll come back to this later in the lecture

IPv4 Datagram format



IP protocol version number

header length(bytes)

"type" of service:

- diffserv (0:5)
- ECN (6:7)

TTL: remaining max hops (decremented at each router)

upper layer protocol (e.g., TCP or UDP)

total datagram length (bytes)

fragmentation/ reassembly

header checksum

32-bit source IP address

Maximum length: 64K bytes
Typically: 1500 bytes or less

e.g., timestamp, record route taken

overhead

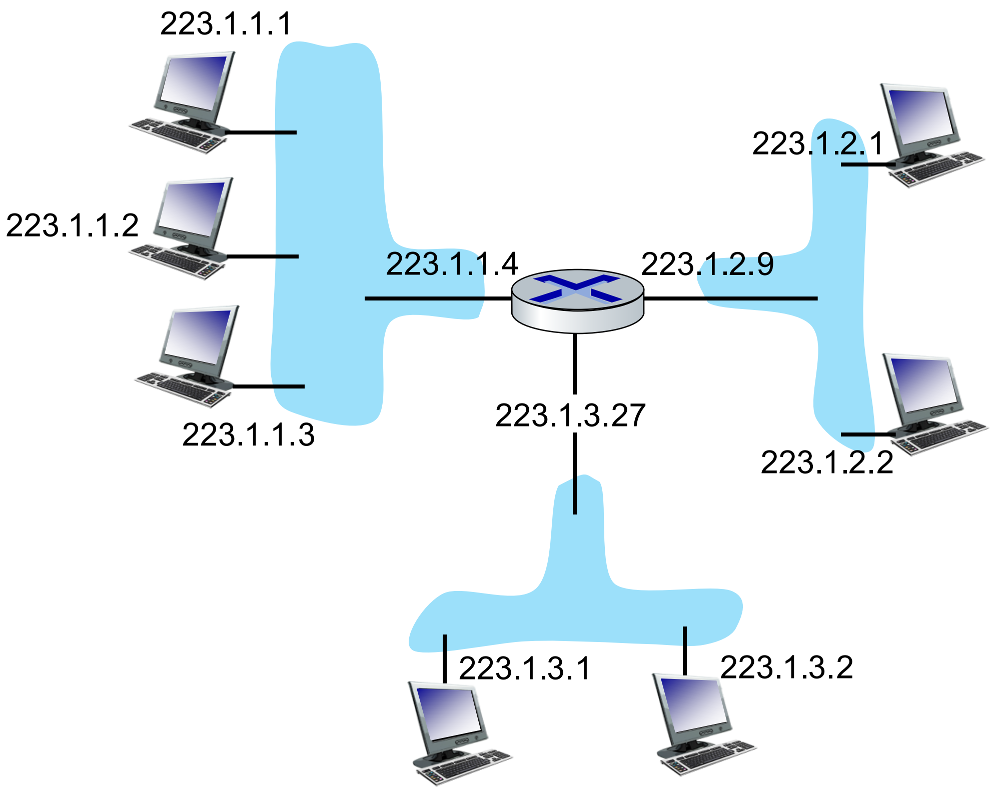
- 20 bytes of TCP
- 20 bytes of IP
- = 40 bytes + app layer overhead for TCP+IP

Network Layer in the Internet

- The Internet Protocol (IP)
  - Datagram format
  - Addressing
  - Network address translation
  - IPv6
- Internet Routing
- Internet Forwarding

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)



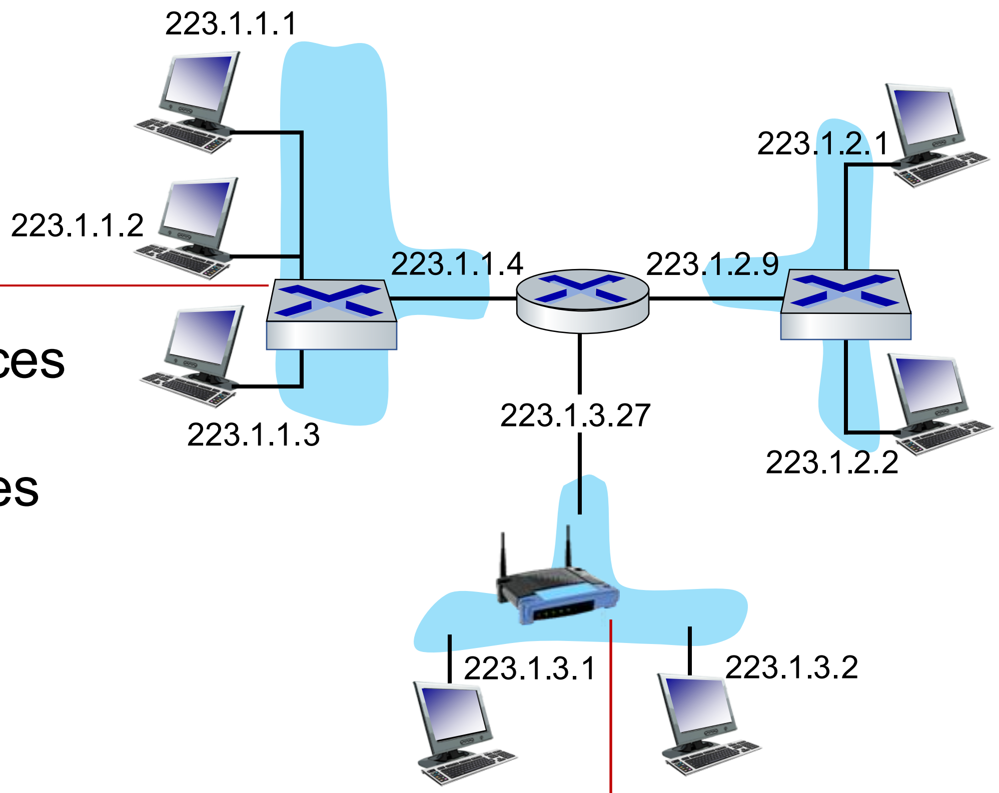
dotted-decimal IP address notation:

$$223.1.1.1 = \underbrace{11011111}_{223}\ \underbrace{00000001}_{1}\ \underbrace{00000001}_{1}\ \underbrace{00000001}_{1}$$

IP addressing: introduction

Q: how are interfaces (without an intervening router) actually connected?

A: we’ll learn about that later



A: wired Ethernet interfaces connected by Ethernet switches

A: wireless WiFi interfaces connected by WiFi base station

For now: don’t need to worry about how two interfaces without an intervening router are connected

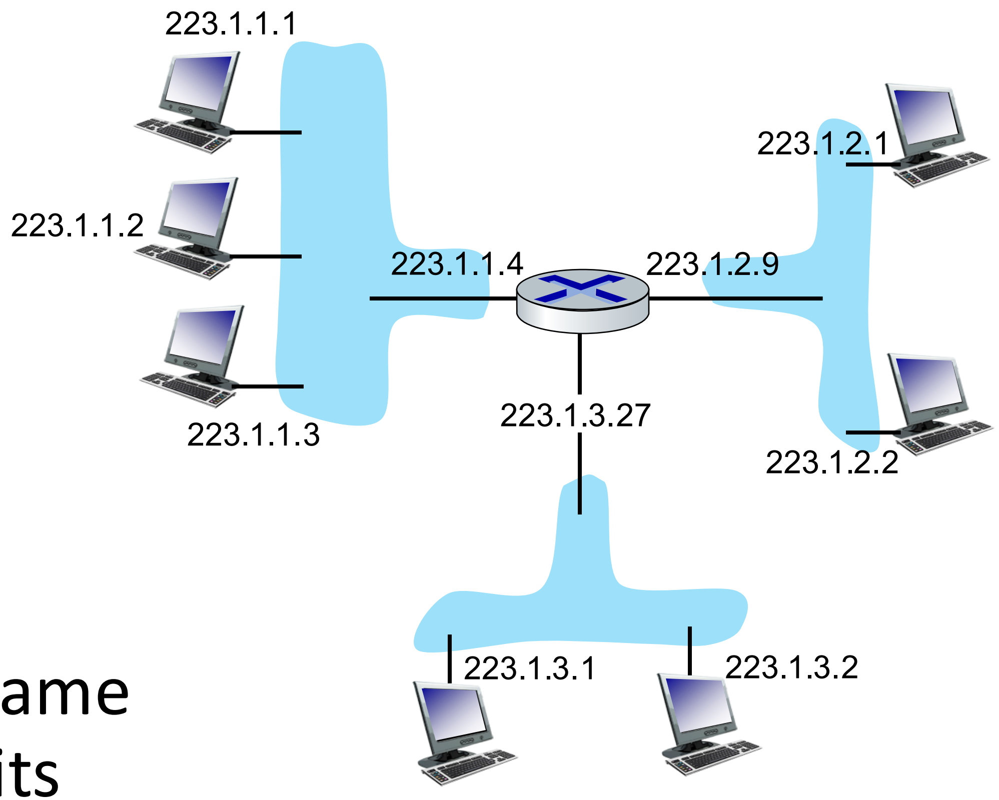
IP addressing: properties

- Global uniqueness:
  - identifies hosts

- Hierarchical structure:
  - consists of Subnet part + Host part
  - is necessary for Internet to scale to large number of hosts
  - aids Internet routing

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 (or *prefix*): devices in same subnet have common high order bits
  - host part: remaining low order bits

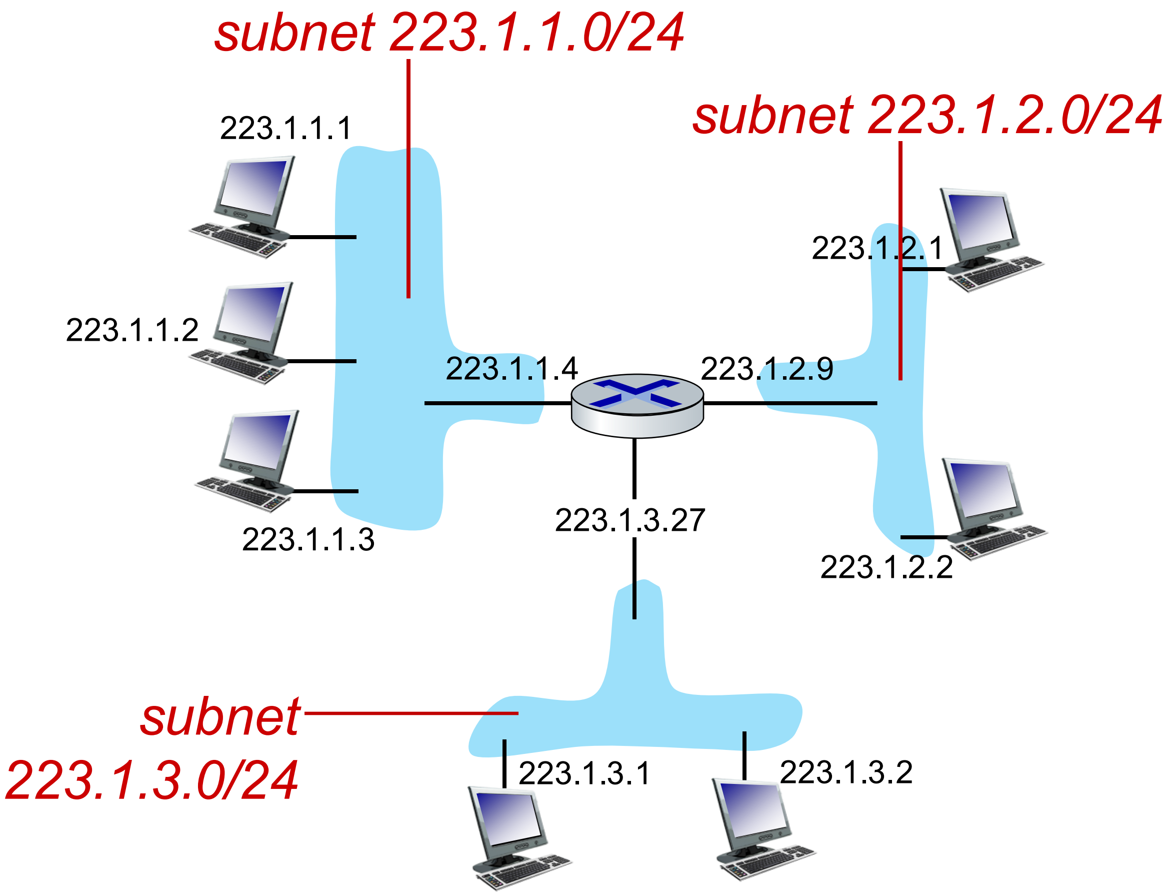


network consisting of 3 subnets

Subnets

Recipe for defining subnets:

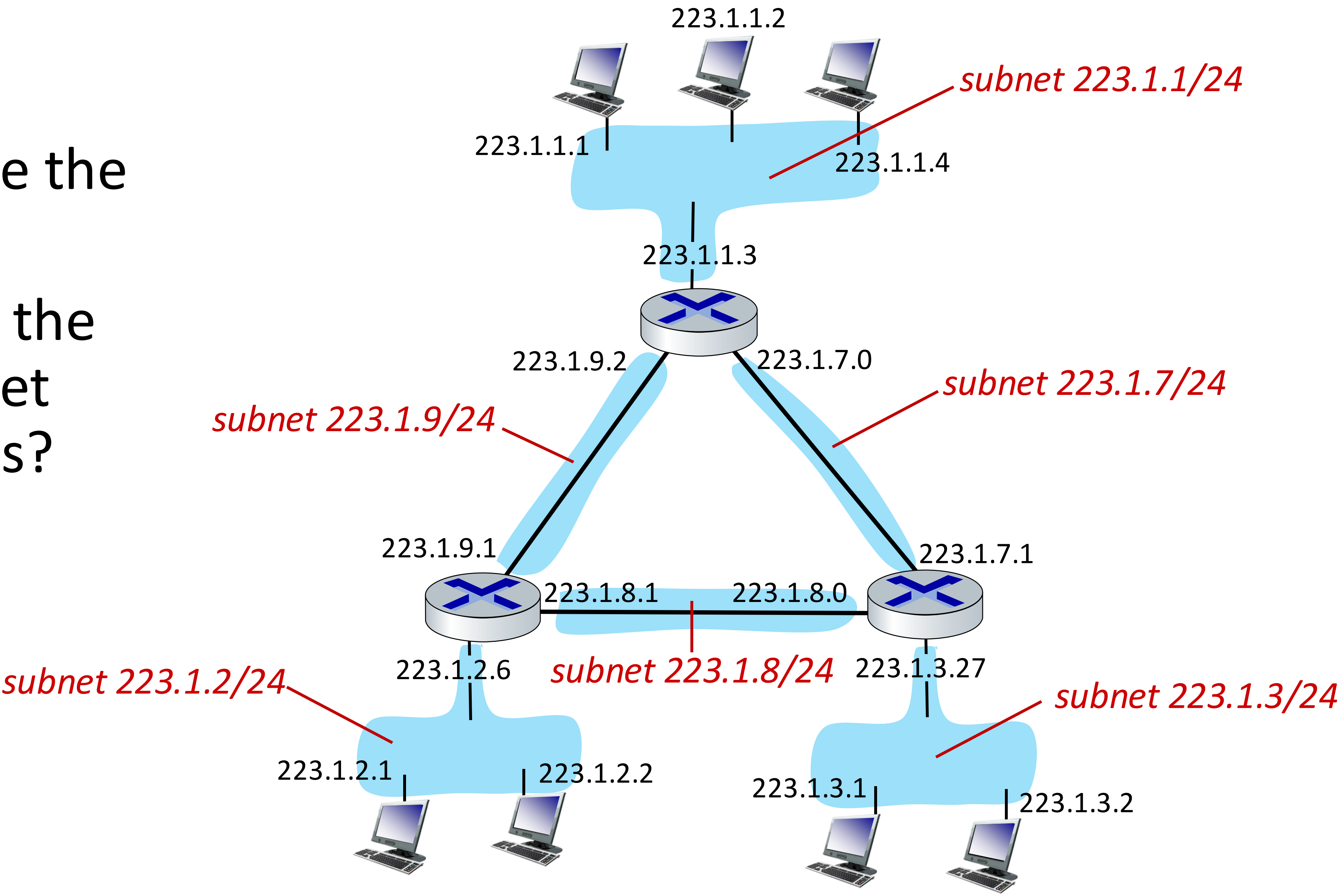
- detach each interface from its host or router, creating "islands" of isolated networks
- each isolated network is called a *subnet*



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

Subnets

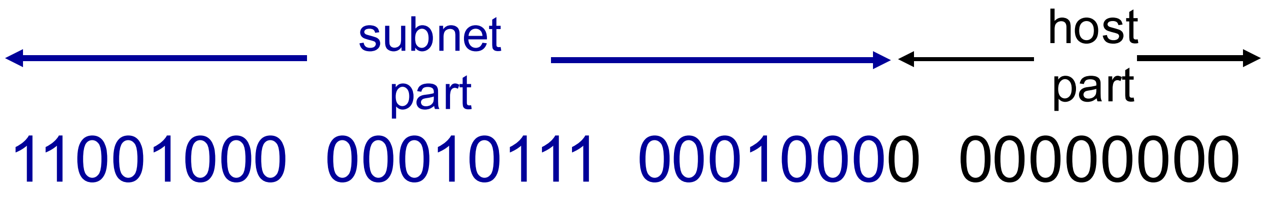
- where are the subnets?
- what are the /24 subnet addresses?



IP addressing: CIDR

CIDR: Classless InterDomain Routing (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



subnet part

host part

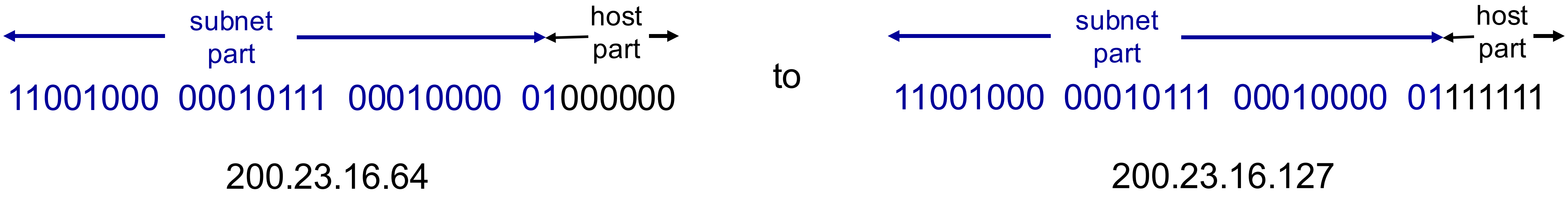
11001000 00010111 00010000 00000000

200.23.16.0/23

IP addressing: CIDR

Q:What is the range of subnet 200.23.16.64/26?

- 6 free bits or 64 addresses
- Last 8 bits are of the format 01XX XXXX
- The range is 200.23.16.64 - 200.23.16.127



subnet part

host part

11001000 00010111 00010000 01000000

200.23.16.64

to

subnet part

host part

11001000 00010111 00010000 01111111

200.23.16.127

Subnet masks

- We’ve seen subnet masks as /24, which is the number of bits in the subnet vs interfaces inside that subnet
- Computers represent subnet masks using binary only, but not in the way you think!
- What is subnet mask of 24?
- 11000?
- No!!!
- It would be:

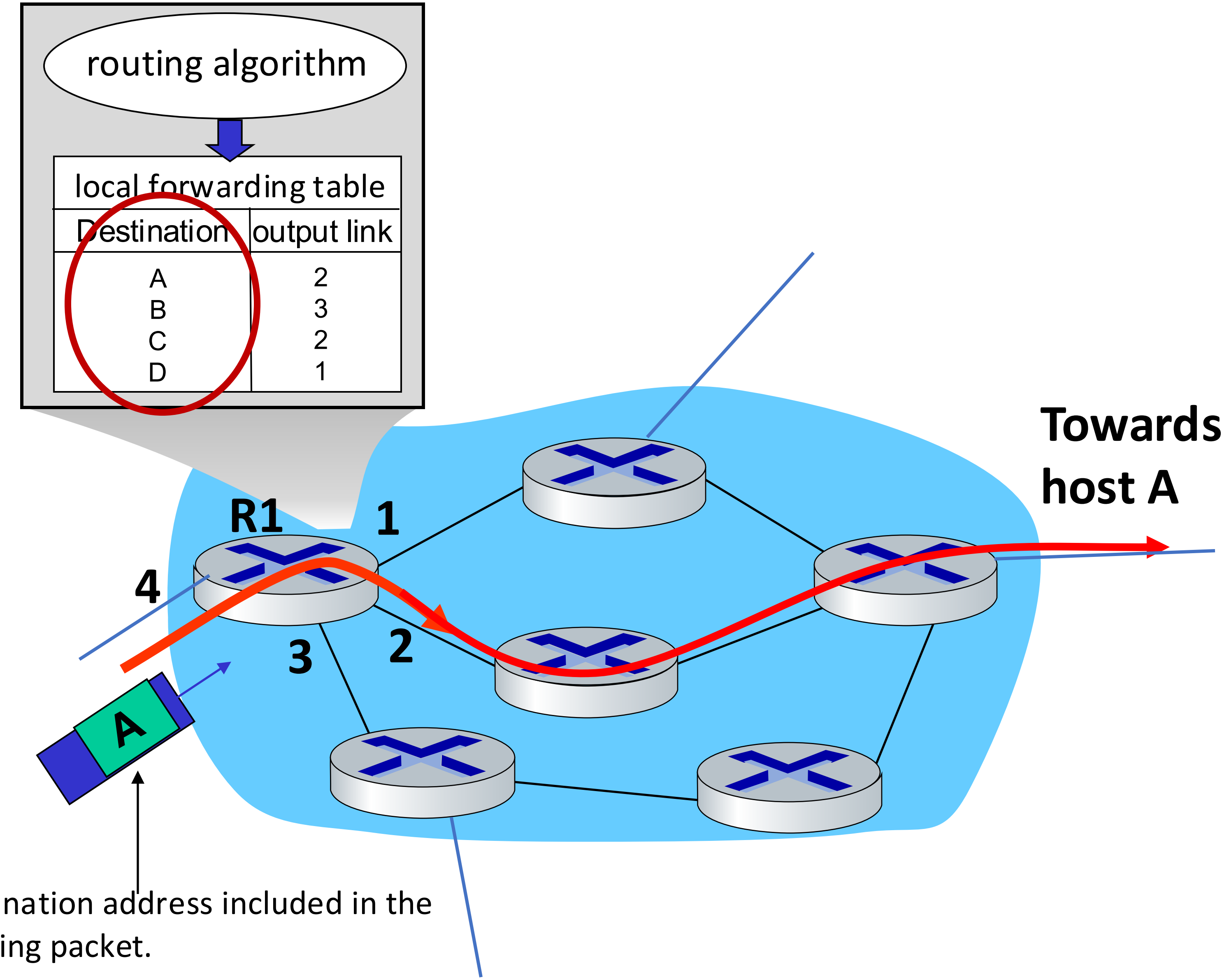
1111 1111 . 1111 1111 . 1111 1111 . 0000 0000

- This represents whether each bit is in the subnet or not!
- OR 255.255.255.0 (in decimal)

Route Aggregation

Recall:

- Forwarding table maps destination addresses to outgoing link
- As the number of possible IP (4 billion!) is huge, rows of forwarding table lists ranges of address, rather than individual hosts



local forwarding table

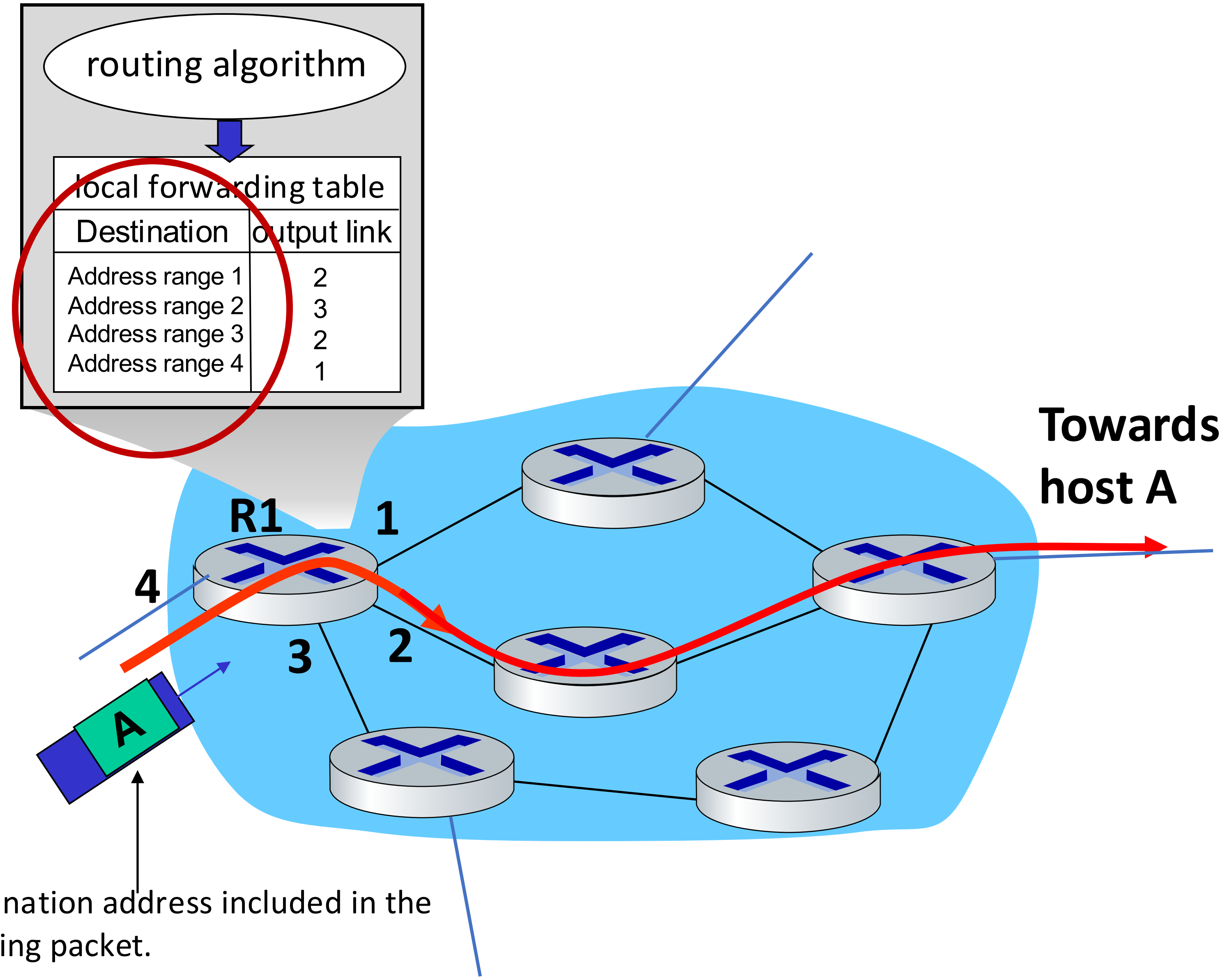
Destination	output link
A	2
B	3
C	2
D	1

destination address included in the arriving packet.

Route Aggregation

Recall:

- Forwarding table maps destination addresses to outgoing link
- As the number of possible IP (4 billion!) is huge, rows of forwarding table lists ranges of address, rather than individual hosts



local forwarding table

Destination	output link
Address range 1	2
Address range 2	3
Address range 3	2
Address range 4	1

destination address included in the arriving packet.

Route Aggregation

Longest prefix matching

- For a given destination address, determine the output link based on the entry with *longest address prefix* that matches the address
- E.g., the destination address 200.23.16.1 falls into ranges of both the first and second rows in the forwarding table; it is forwarded to output link 2 based on longest prefix matching

local forwarding table

Destination	output link
11001000 00010111 00010000 00000000 / 23 200.23.16.0 / 23	2
11001000 00010111 00010000 00000000 / 20 200.23.16.0 / 20	3
11001000 00010111 00011000 00000000 / 23 200.23.24.0 / 23	2
Otherwise	1

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)

Hierarchical addressing

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

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

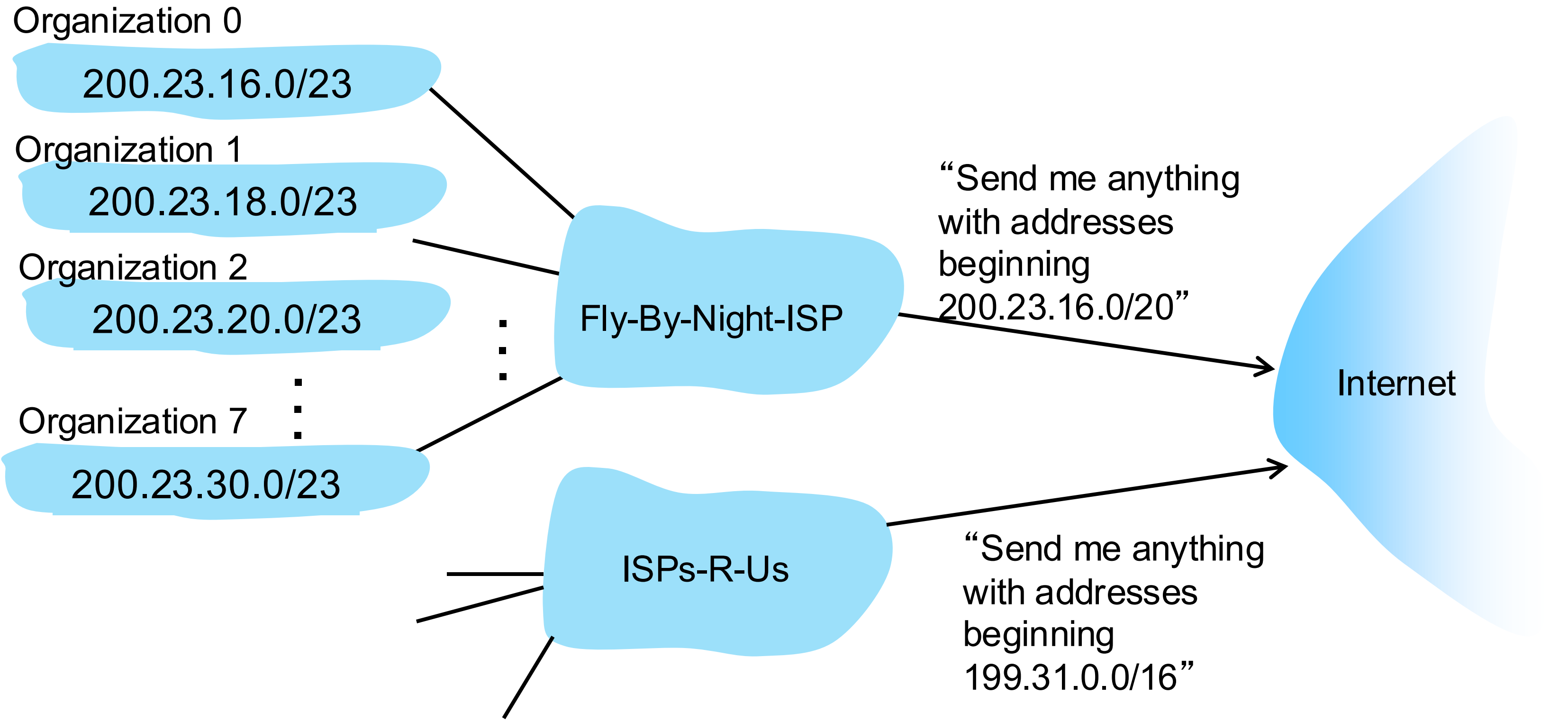
ISP's block	11001000 00010111 00010000 00000000	200.23.16.0/20

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

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

Route aggregation example

hierarchical addressing allows efficient advertisement of routing information:

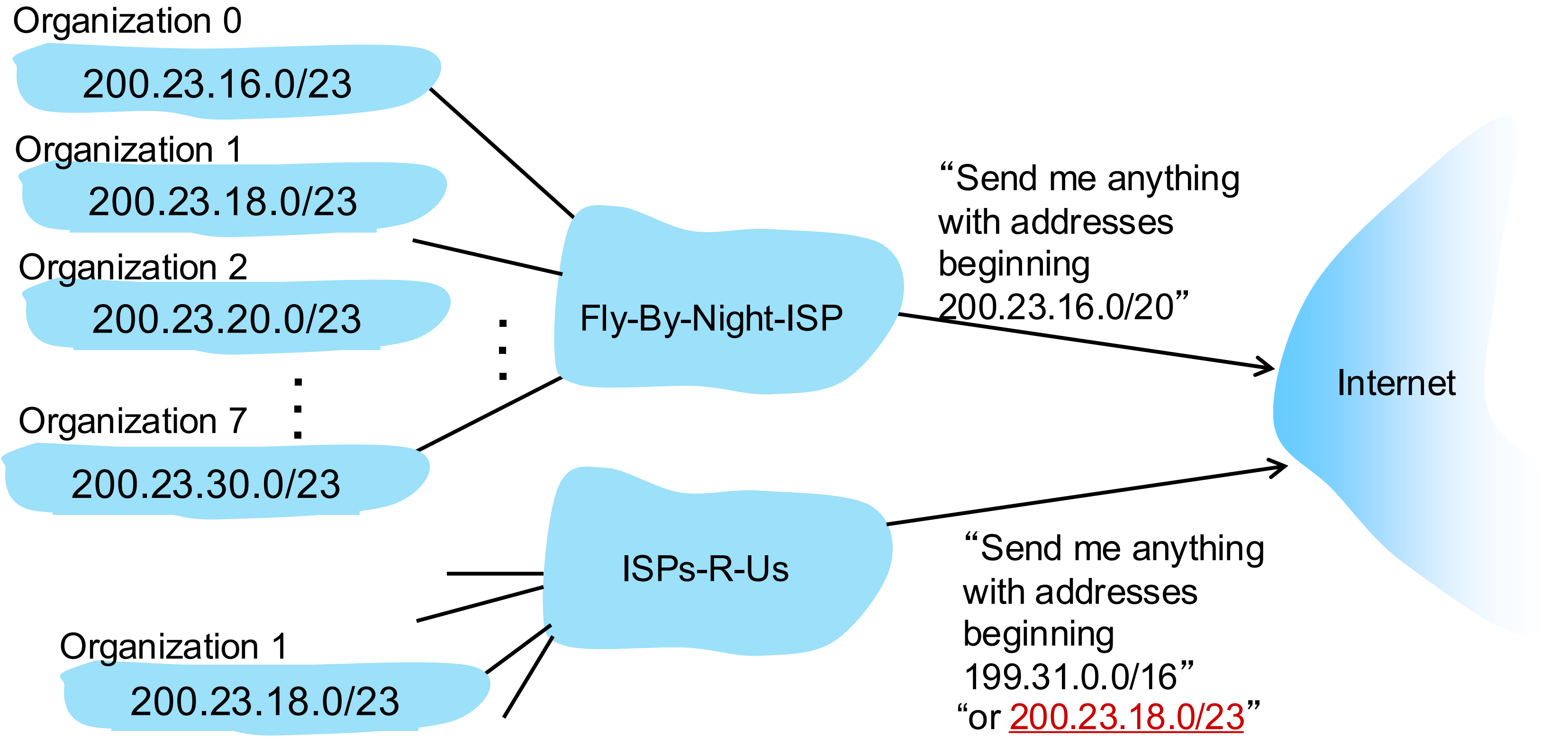


"Send me anything with addresses beginning 200.23.16.0/20"

"Send me anything with addresses beginning 199.31.0.0/16"

Route aggregation example

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

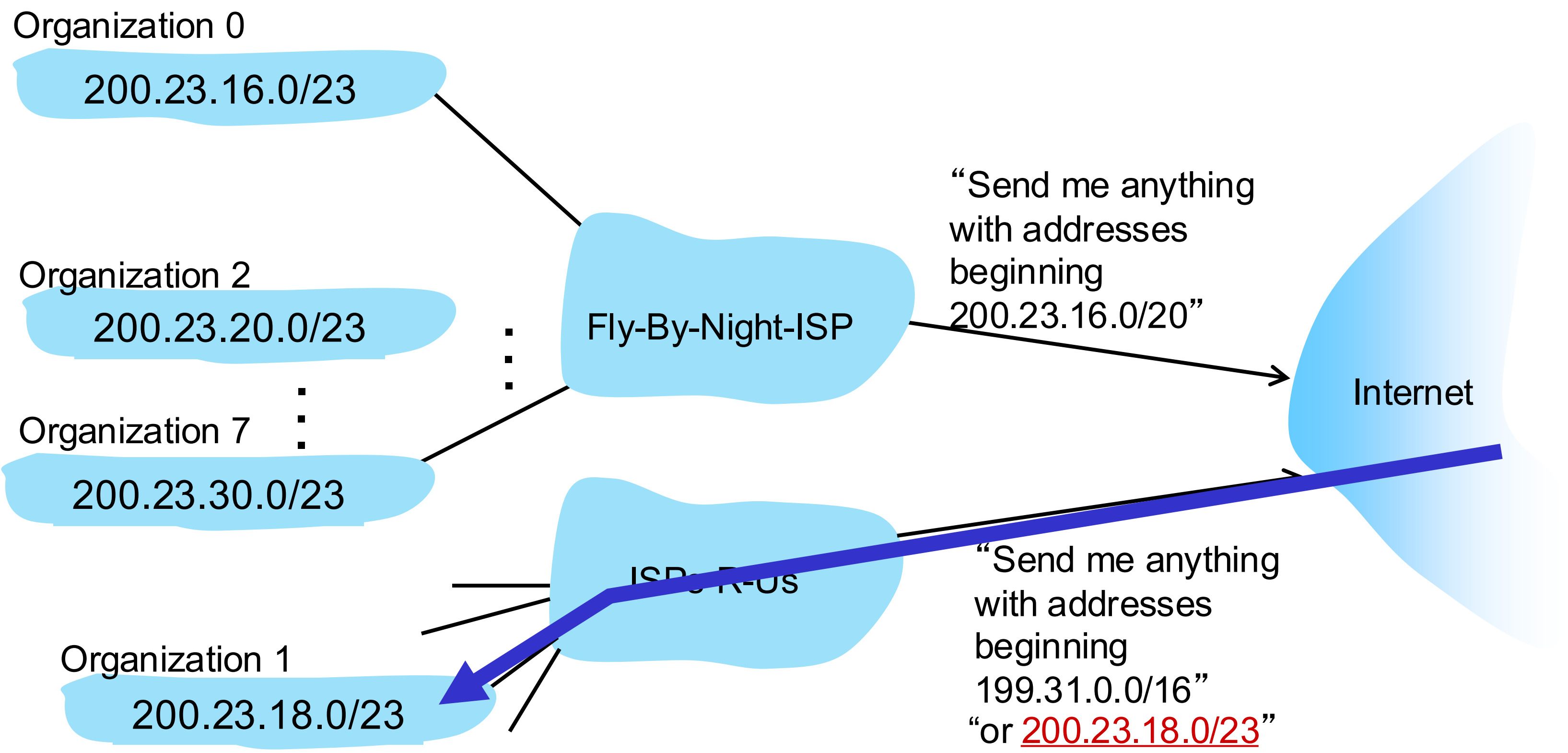


"Send me anything with addresses beginning 200.23.16.0/20"

"Send me anything with addresses beginning 199.31.0.0/16"
"or 200.23.18.0/23"

Route aggregation example

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



In-Class Exercise

Q: A router connects 3 subnets. Each subnet must have prefix 200.23.16.0/24

- Subnet 1 supports up to 123 interfaces
- Subnet 2 and subnet 3 up to 60 interfaces
- What should the 3 subnet addresses be?

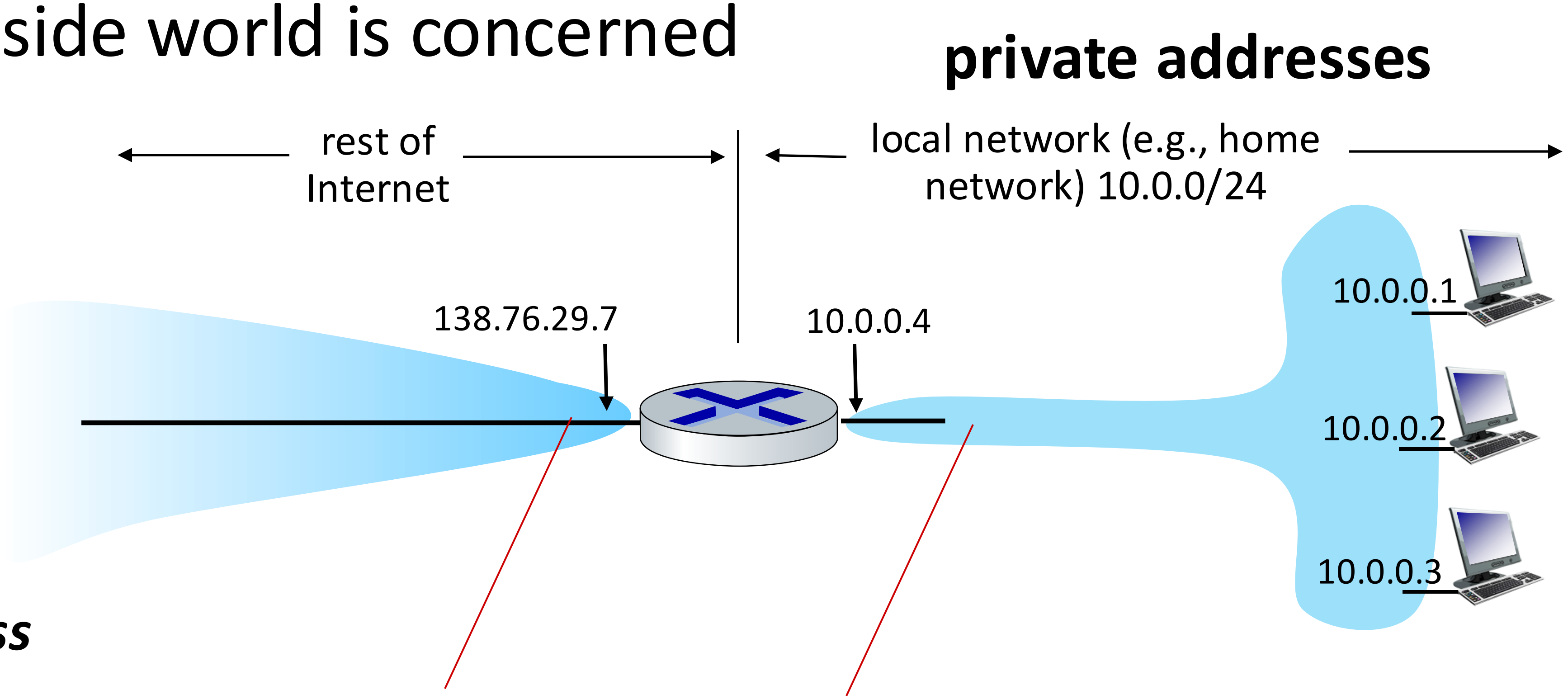
Network Layer in the Internet

- The Internet Protocol (IP)
  - Datagram format
  - Addressing
  - Network address translation
  - IPv6
- Internet Routing
- Internet Forwarding

NAT: network address translation

NAT: all devices in local network share just one IPv4 address as far as outside world is concerned

Router behaves to the outside world as a ***single device*** with a ***single IP address***



all datagrams *leaving* local network have *same* source NAT IP address: 138.76.29.7, but *different* source port numbers

datagrams with source or destination in this network have 10.0.0/24 address for source, destination (as usual)

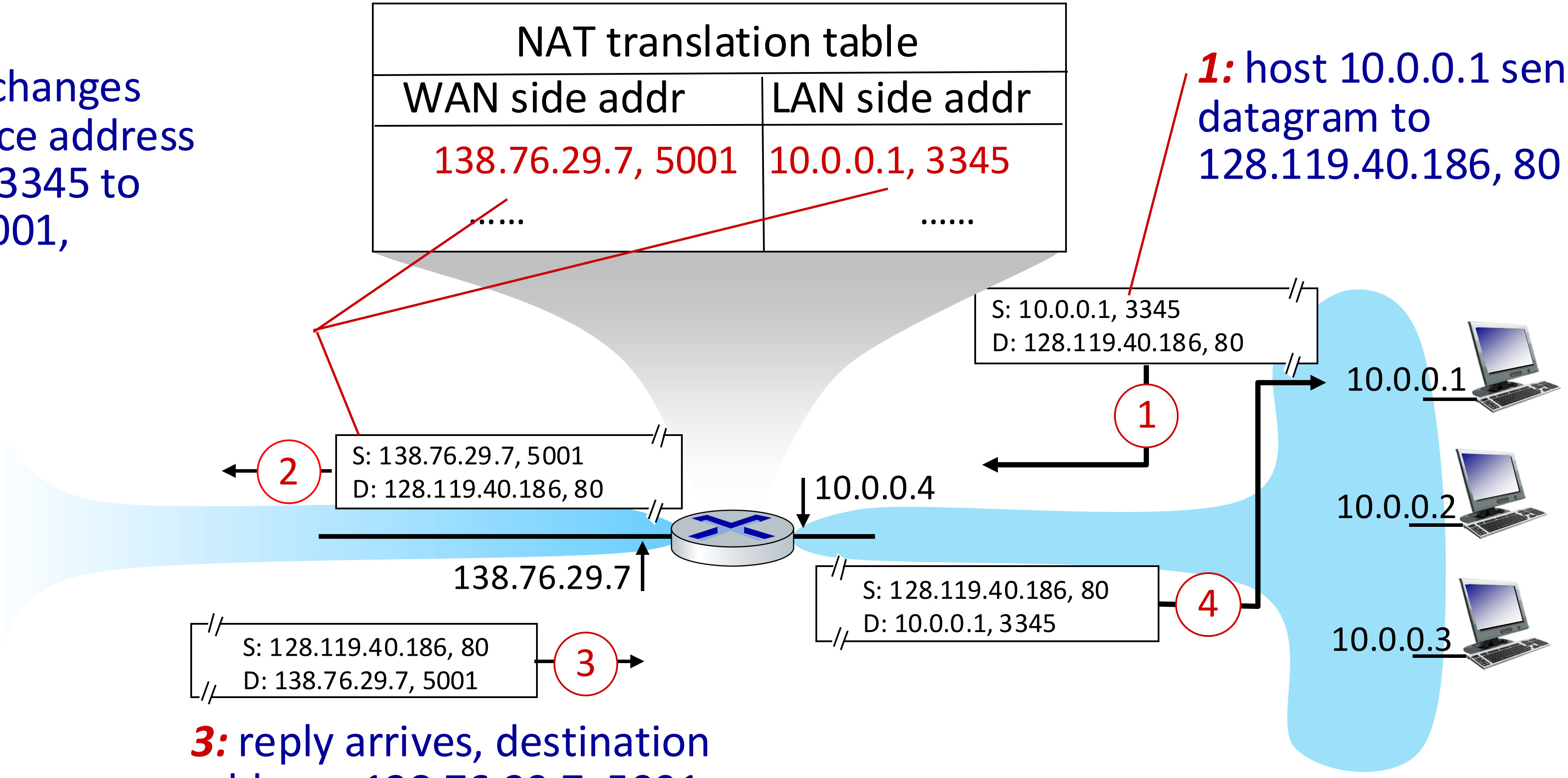
NAT: network address translation

NAT "box" must (transparently):

- outgoing datagrams: replace (source IP address, port #) of every outgoing datagram to (NAT IP address, new port #)
  - remote clients/servers will respond using (NAT IP address, new port #) as destination address
- remember (in NAT translation table) every (source IP address, port #) to (NAT IP address, new port #) translation pair
- incoming datagrams: replace (NAT IP address, new port #) in destination fields of every incoming datagram with corresponding (source IP address, port #) stored in NAT table

NAT: network address translation

2: NAT router changes datagram source address from 10.0.0.1, 3345 to 138.76.29.7, 5001, updates table



NAT translation table

WAN side addr	LAN side addr
138.76.29.7, 5001	10.0.0.1, 3345
……	……

1: host 10.0.0.1 sends datagram to 128.119.40.186, 80

3: reply arrives, destination address: 138.76.29.7, 5001

NAT: network address translation

- all devices in local network have 32-bit addresses in a “private” IP address space (10/8, 172.16/12, 192.168/16 prefixes) that can only be used in local network
- advantages:
  - just one IP address needed from provider ISP for *all* devices
  - can change addresses of host in local network without notifying outside world
  - can change ISP without changing addresses of devices in local network
  - security: devices inside local net not directly addressable, visible by outside world

NAT: network address translation

- 16-bit port-number field:
  - 60,000 simultaneous connections with a single WAN-side address!
- NAT has been controversial:
  - Network devices “should” only process up to layer 3
  - address “shortage” should be solved by IPv6
  - violates end-to-end argument (port # manipulation by network-layer device)
- NAT traversal: what if client wants to connect to server behind NAT?
- but NAT is here to stay:
  - extensively used in home and institutional nets, 4G/5G cellular nets

What you need to know about NATs

- How NATs work
  - E.g., given an example scenario like the ones in the slides, and the port NAT chooses for an incoming connection, you should be able to complete the IP addresses and ports in packets going into and out of the NAT.
- Understand the reasons a NAT can be beneficial
- Understand the problems a NAT can cause.

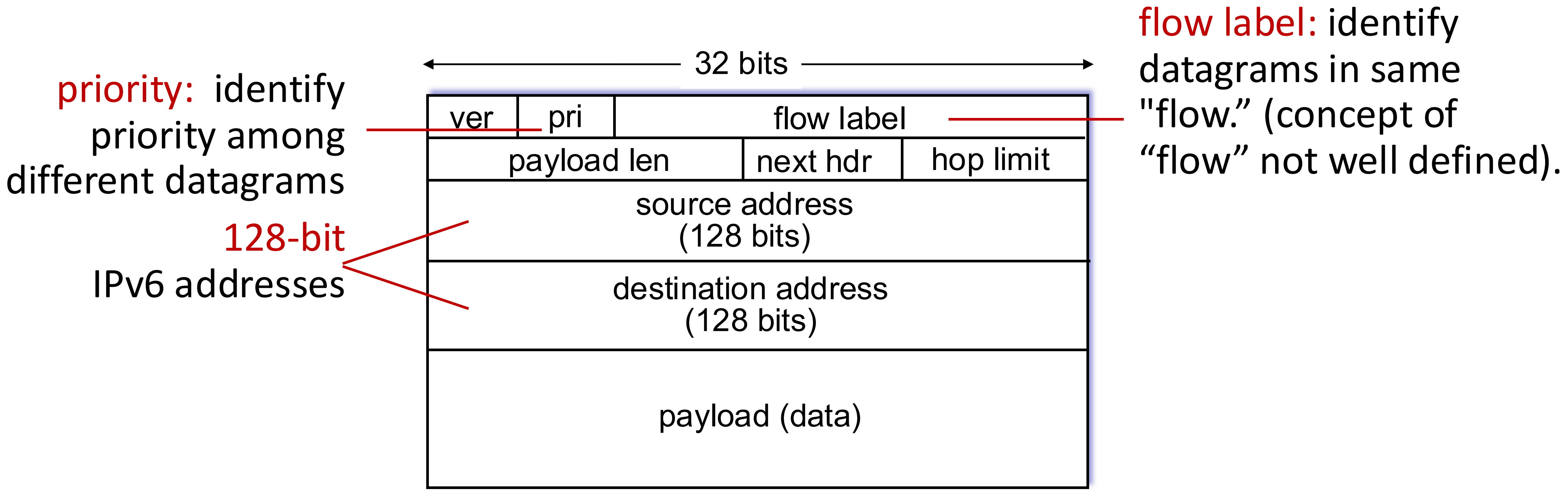
Network Layer in the Internet

- The Internet Protocol (IP)
  - Datagram format
  - Addressing
  - Network address translation
  - IPv6
- Internet Routing
- Internet Forwarding

IPv6: motivation

- initial motivation: 32-bit IPv4 address space would be completely allocated
- additional motivation:
  - speed processing/forwarding
    - E.g., IPv4 has a (long) 20-byte fixed length header
  - enable different network-layer treatment of “flows”
    - E.g., different network-layer service for latency-sensitive traffic such as video conferencing compared to not-so-sensitive traffic such as a file download.

IPv6 datagram format



priority: identify priority among different datagrams

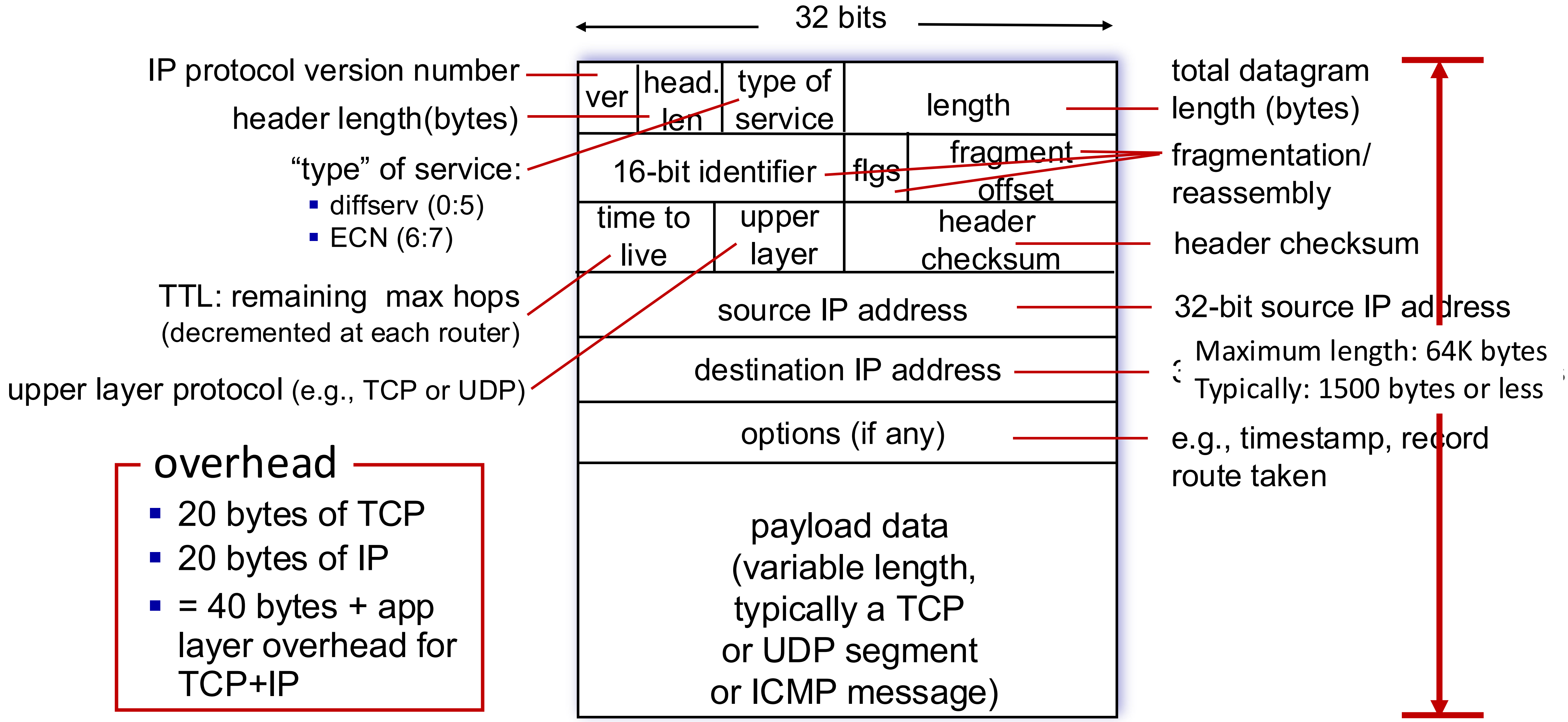
128-bit IPv6 addresses

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)

Compare to IPv4 datagram format…



IP protocol version number

header length(bytes)

“type” of service:

- diffserv (0:5)
- ECN (6:7)

TTL: remaining max hops (decremented at each router)

upper layer protocol (e.g., TCP or UDP)

total datagram length (bytes)

fragmentation/ reassembly

header checksum

32-bit source IP address

Maximum length: 64K bytes
Typically: 1500 bytes or less

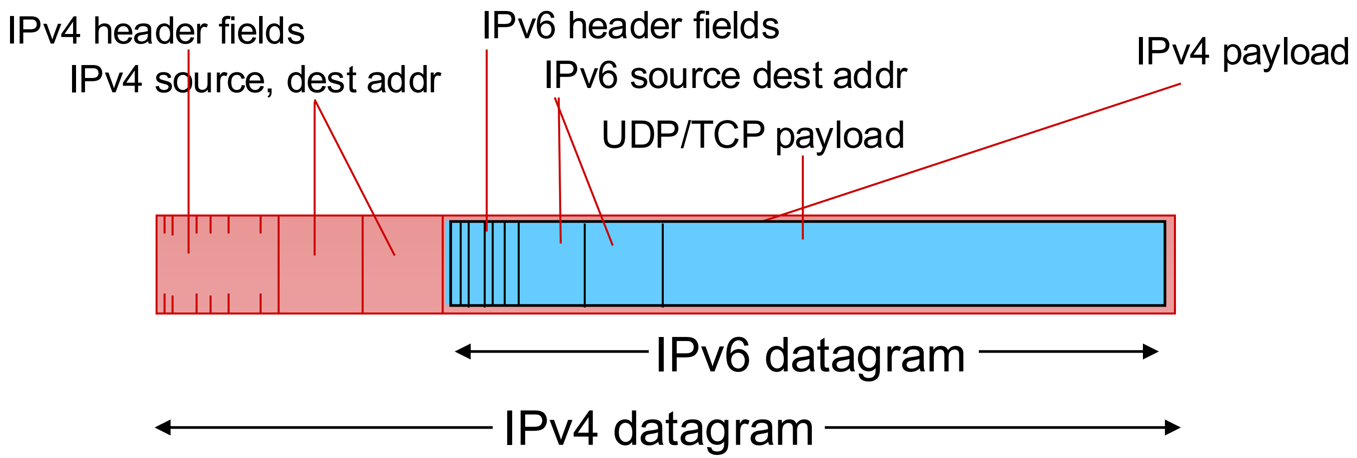
e.g., timestamp, record route taken

overhead

- 20 bytes of TCP
- 20 bytes of IP
- = 40 bytes + app layer overhead for TCP+IP

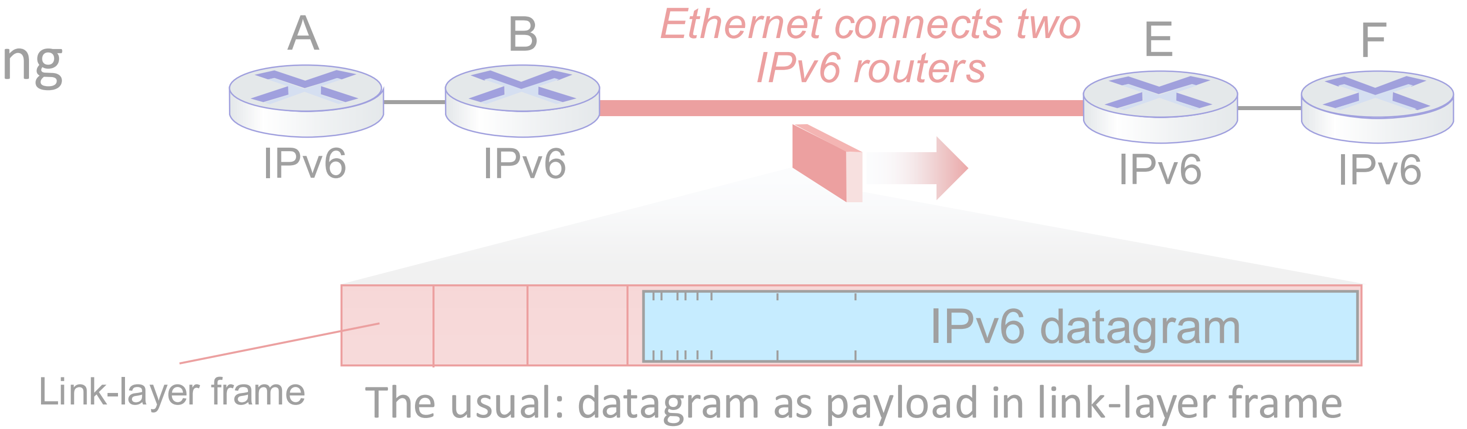
Transition from IPv4 to IPv6

- 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)



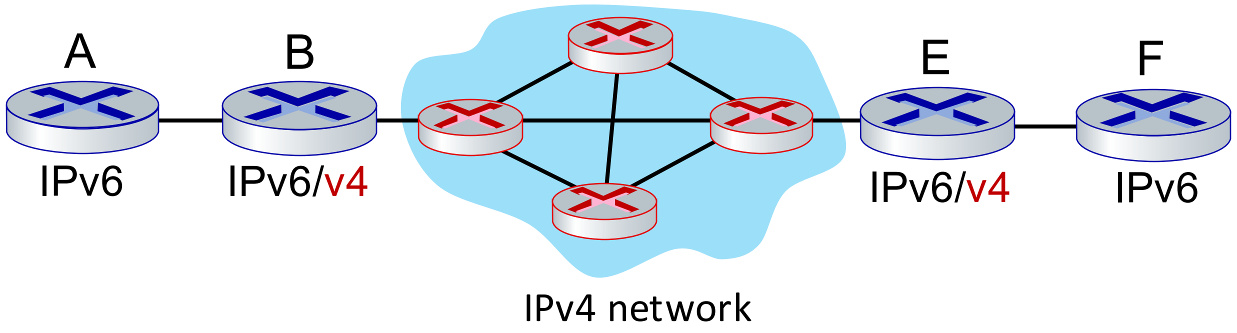
Tunneling and encapsulation

Ethernet connecting two IPv6 routers:



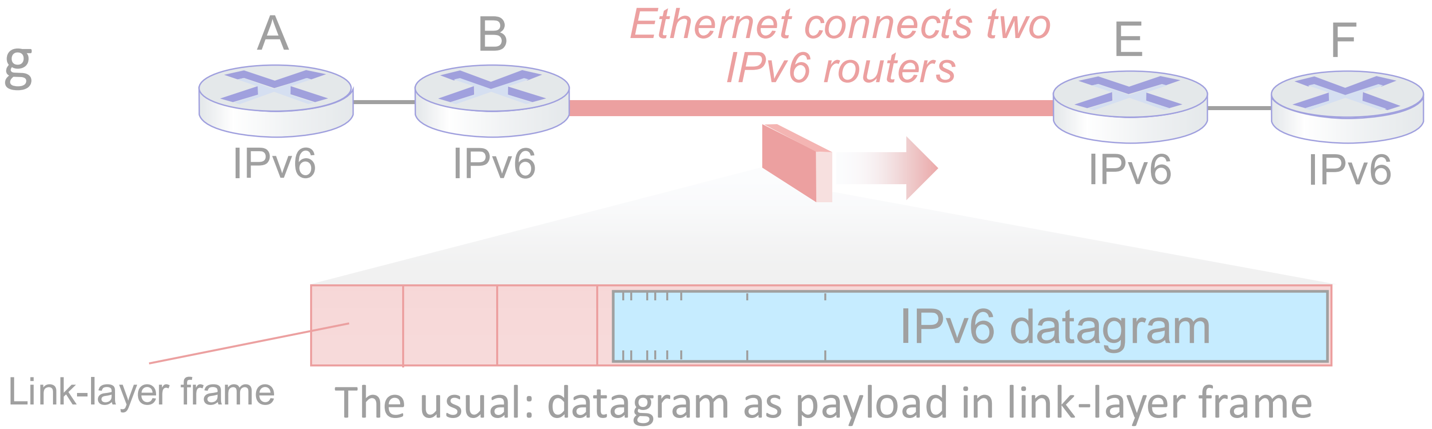
The usual: datagram as payload in link-layer frame

IPv4 network connecting two IPv6 routers



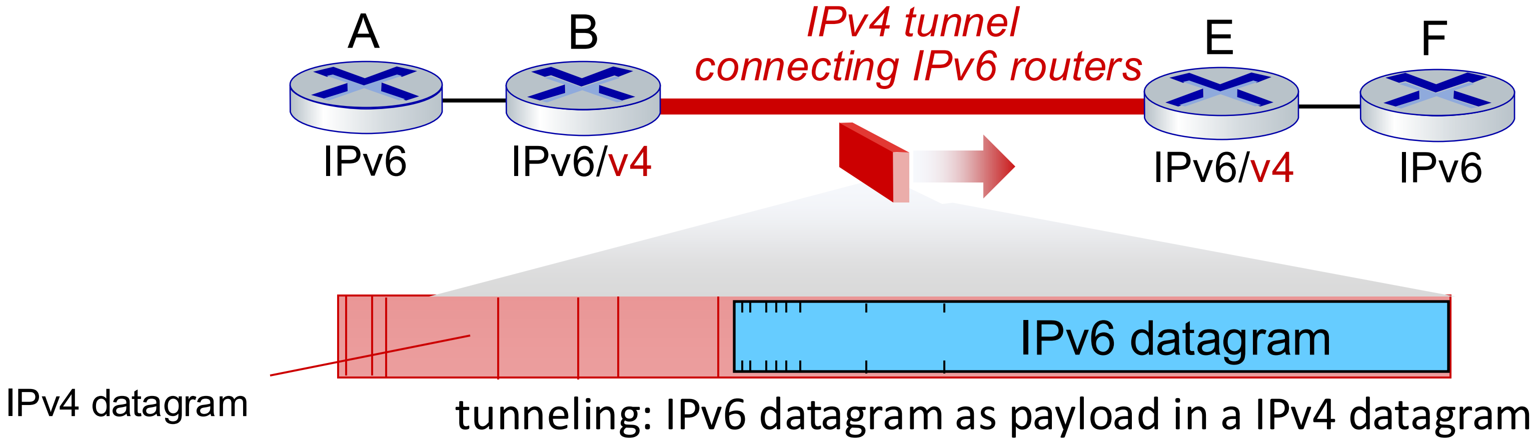
Tunneling and encapsulation

Ethernet connecting two IPv6 routers:



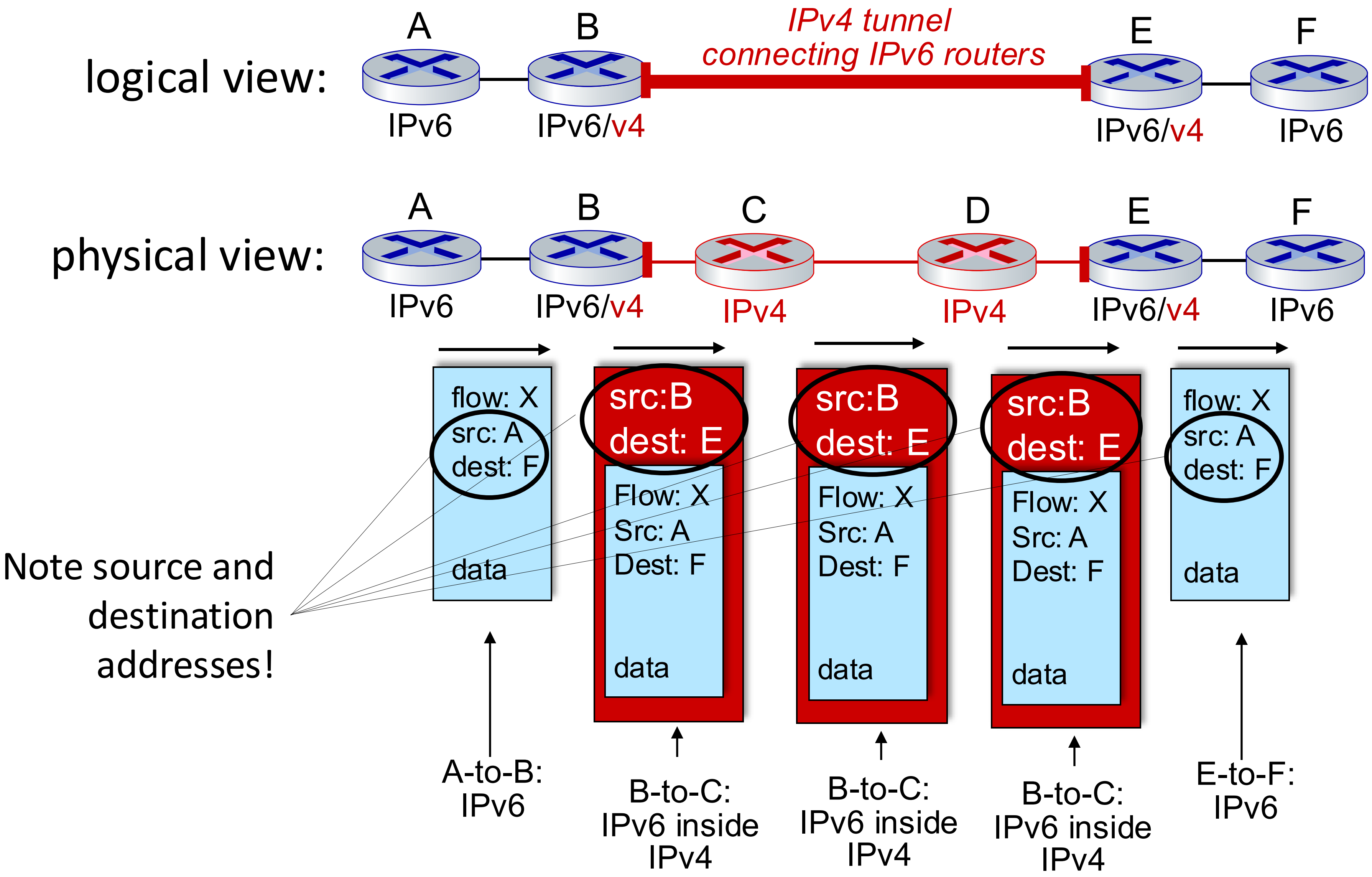
The usual: datagram as payload in link-layer frame

IPv4 tunnel connecting two IPv6 routers



tunneling: IPv6 datagram as payload in a IPv4 datagram

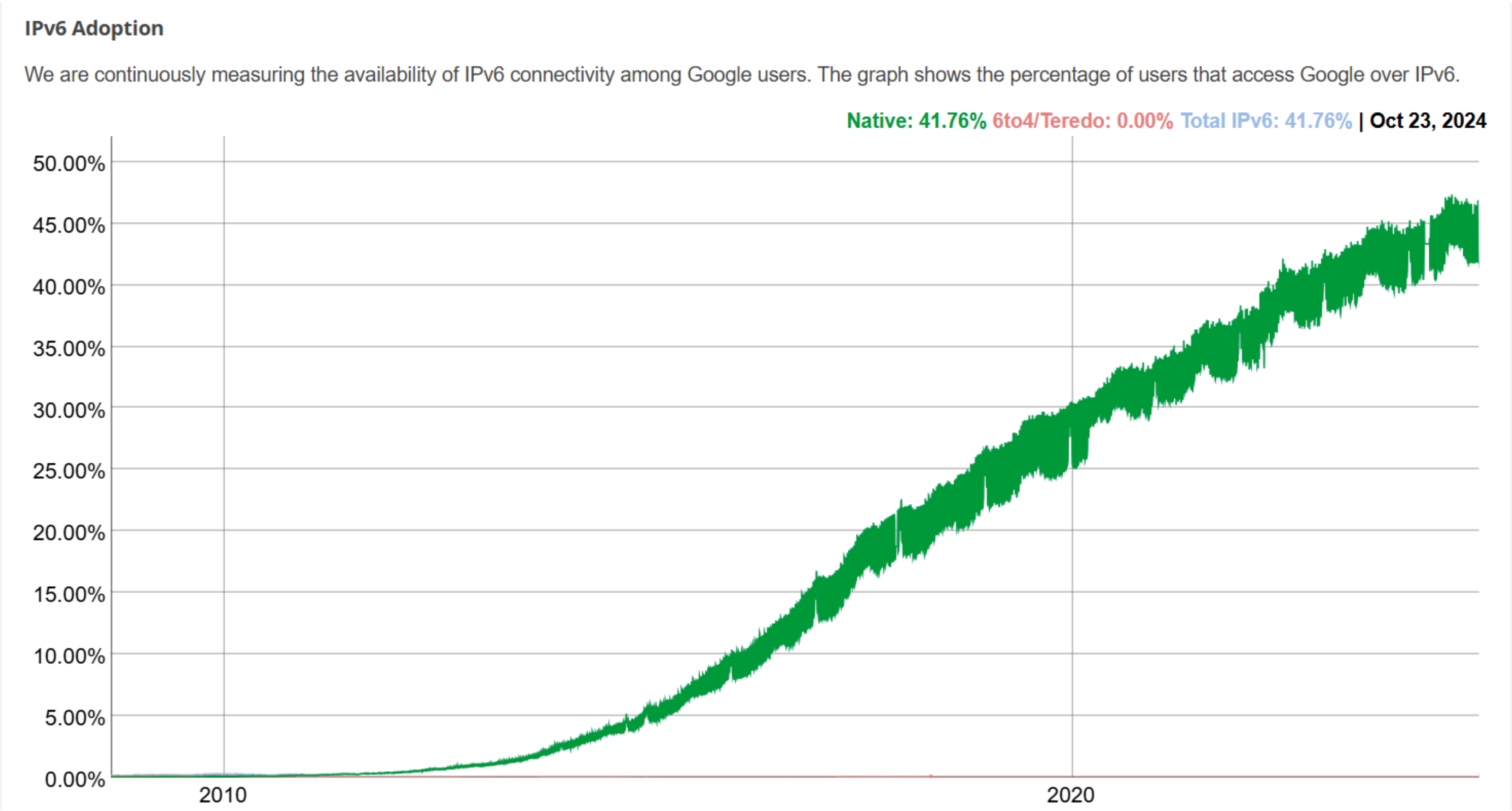
Tunneling



logical view:
A
B
IPv4 tunnel
connecting IPv6 routers
E
F
IPv6
IPv6/v4
IPv6/v4
IPv6
physical view:
A
B
C
D
E
F
IPv6
IPv6/v4
IPv4
IPv4
IPv6/v4
IPv6
flow: X
src: A
dest: F
data
src:B
dest: E
Flow: X
Src: A
Dest: F
data
src:B
dest: E
Flow: X
Src: A
Dest: F
data
src:B
dest: E
Flow: X
Src: A
Dest: F
data
flow: X
src: A
dest: F
data
Note source and
destination
addresses!
A-to-B:
IPv6
B-to-C:
IPv6 inside
IPv4
B-to-C:
IPv6 inside
IPv4
B-to-C:
IPv6 inside
IPv4
E-to-F:
IPv6

IPv6 adoption

- Google[1]: ~ 50% of clients access services via IPv6 (2024)
- NIST: 1/3 of all US government domains are IPv6 capable



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.

Native: 41.76% 6to4/Teredo: 0.00% Total IPv6: 41.76% | Oct 23, 2024

IPv6 adoption

- Google[1]: ~ 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

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)

Q: are there enough 32-bit IP addresses?

- ICANN allocated last chunk of IPv4 addresses to RRs in 2011
- NAT 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)