COMP 3331/9331: Computer Networks and Applications

Week 4

Network Layer: Data Plane (Part 2)

Reading Guide: Chapter 4: Section 4.3

Network Layer, data plane: outline

4.1 Overview of Network layer

- data plane
- control plane

4.2 What's inside a router

4.3 IP: Internet Protocol

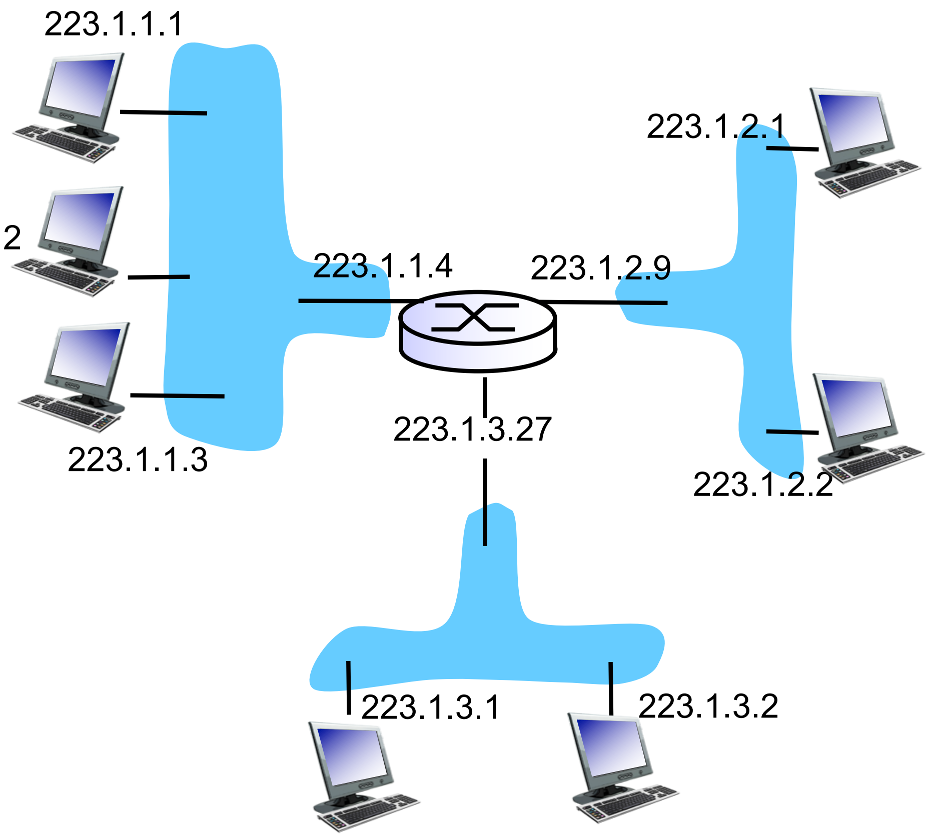
- datagram format
- fragmentation
- IPv4 addressing
- network address translation

4.4 Generalized Forward and SDN

Not Covered

IP addressing: introduction

- *IP address:* 32-bit identifier for host, 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)
- *IP addresses associated with each interface*

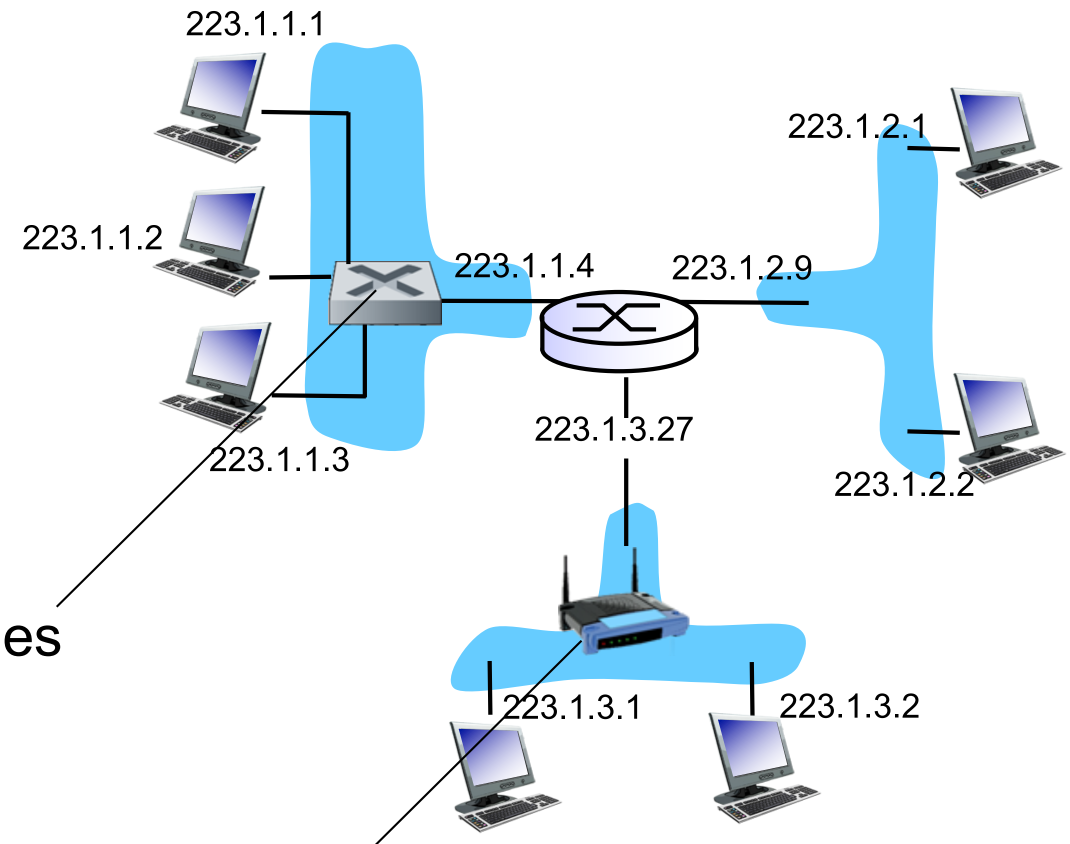


$$223.1.1.1 = \underbrace{11011111}_{223}\ \underbrace{00000001}_{1}\ \underbrace{00000001}_{1}\ \underbrace{00000001}_{1}$$

IP addressing: introduction

Q: how are interfaces actually connected?

A: we'll learn about that in the link layer

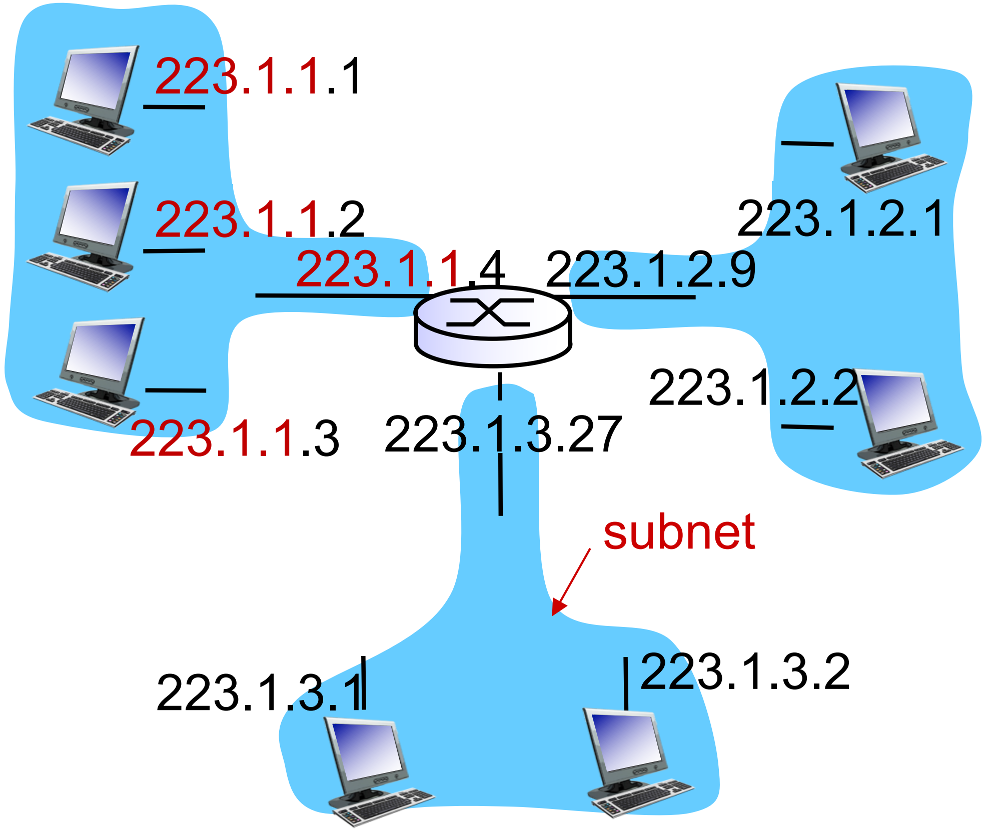


A: wired Ethernet interfaces connected by Ethernet switches

A: wireless WiFi interfaces connected by WiFi base station

Networks

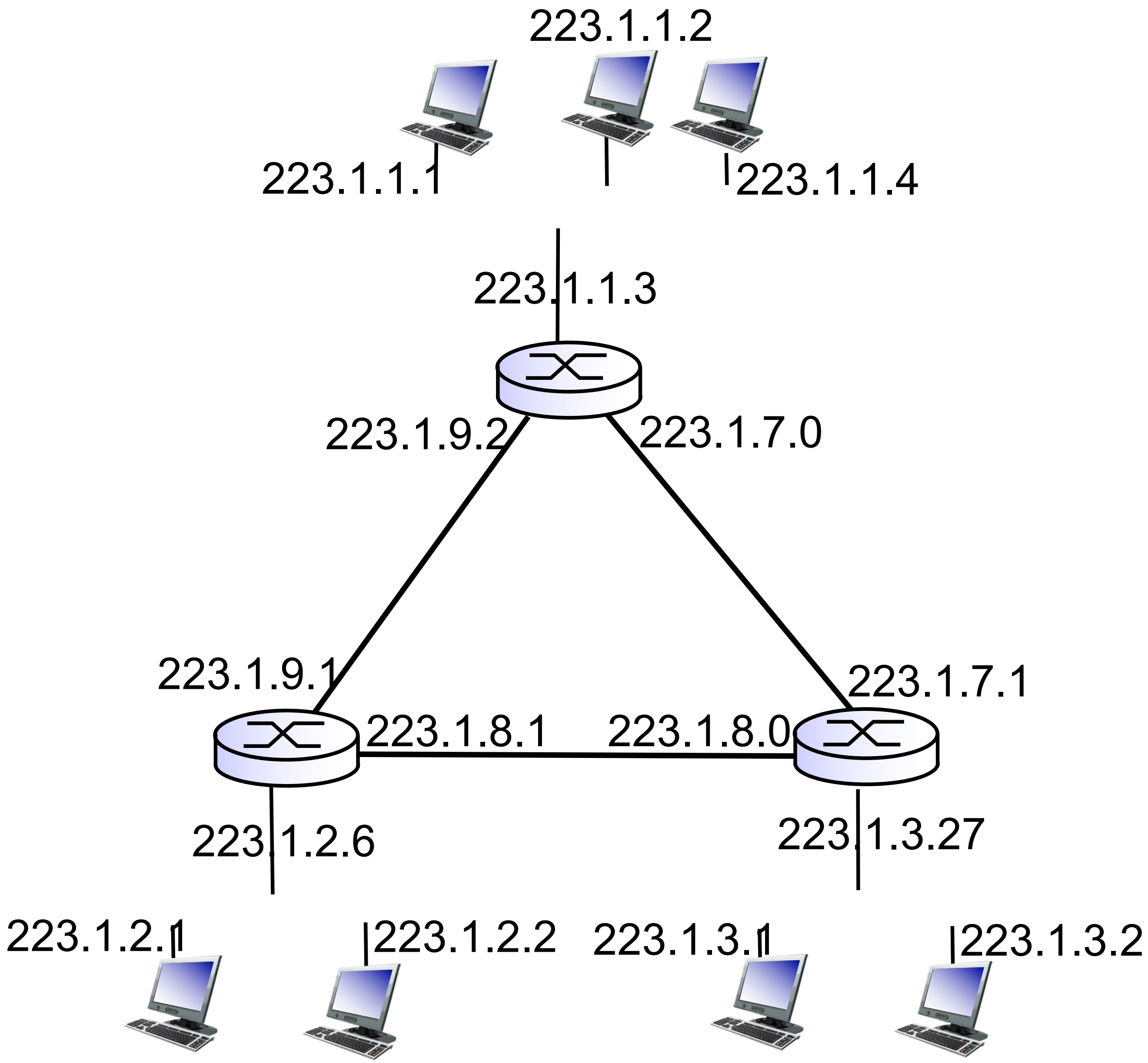
- IP address:
  - network part - high order bits
  - host part - low order bits
- *what's a network ?*
  - device interfaces with same network part of IP address
  - can physically reach each other *without intervening router*



inter-network consisting of 3 nets

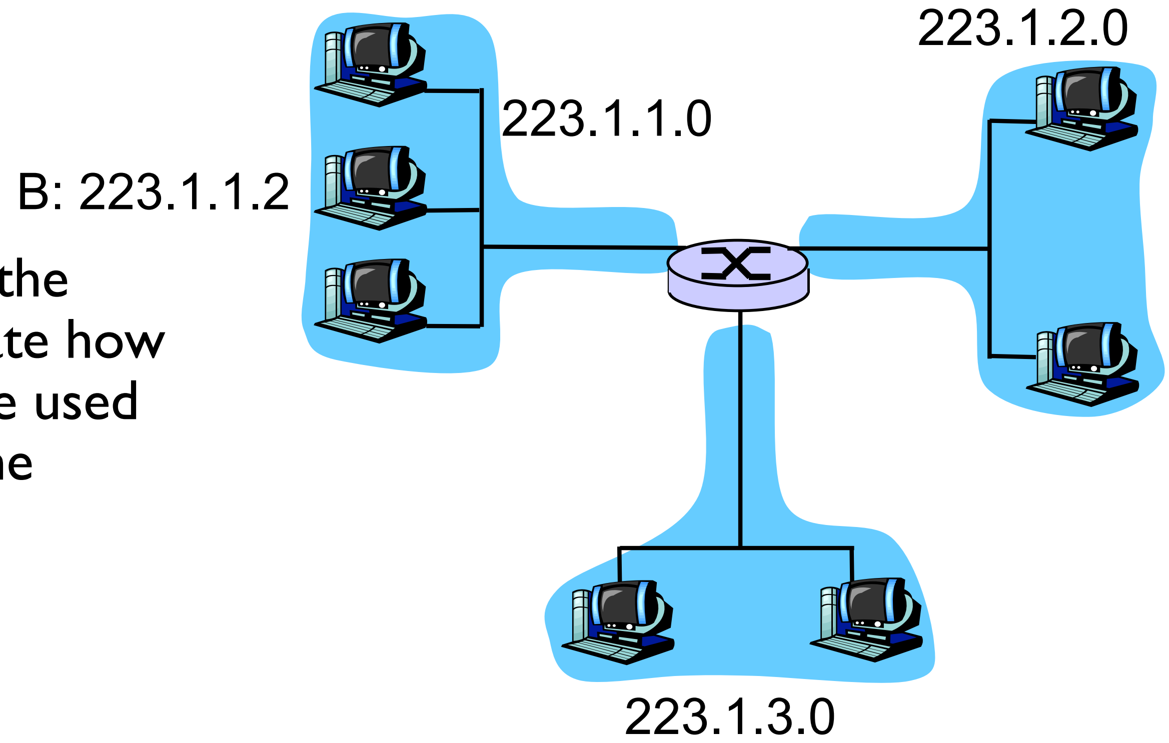
Networks

how many?



Masking

- Mask
  - Used in conjunction with the network address to indicate how many higher order bits are used for the network part of the address
    - Bit-wise AND
  - 223.1.1.0 with mask 255.255.255.0
- Broadcast Address
  - host part is all 111's
  - E.g. 223.1.1.255
- Network Address
  - Host part is all 0000's
  - E.g. 223.1.1.0
- Both of these are not assigned to any host



Host B	Dot-decimal address	Binary
IP address	223.1.1.2	11111101.00000001.00000001.00000010
Mask	255.255.255.0	11111111.11111111.11111111.00000000
Network Part	223.1.1.0	11111101.00000001.00000001.00000000
Host Part	0.0.0.2	00000000.00000000.00000000.00000010

Original Internet Addresses

- First eight bits: network address (/8)
- Last 24 bits: host address, ~16.7 million

Assumed 256 networks were more than enough!

Next design: Class-ful Addresses

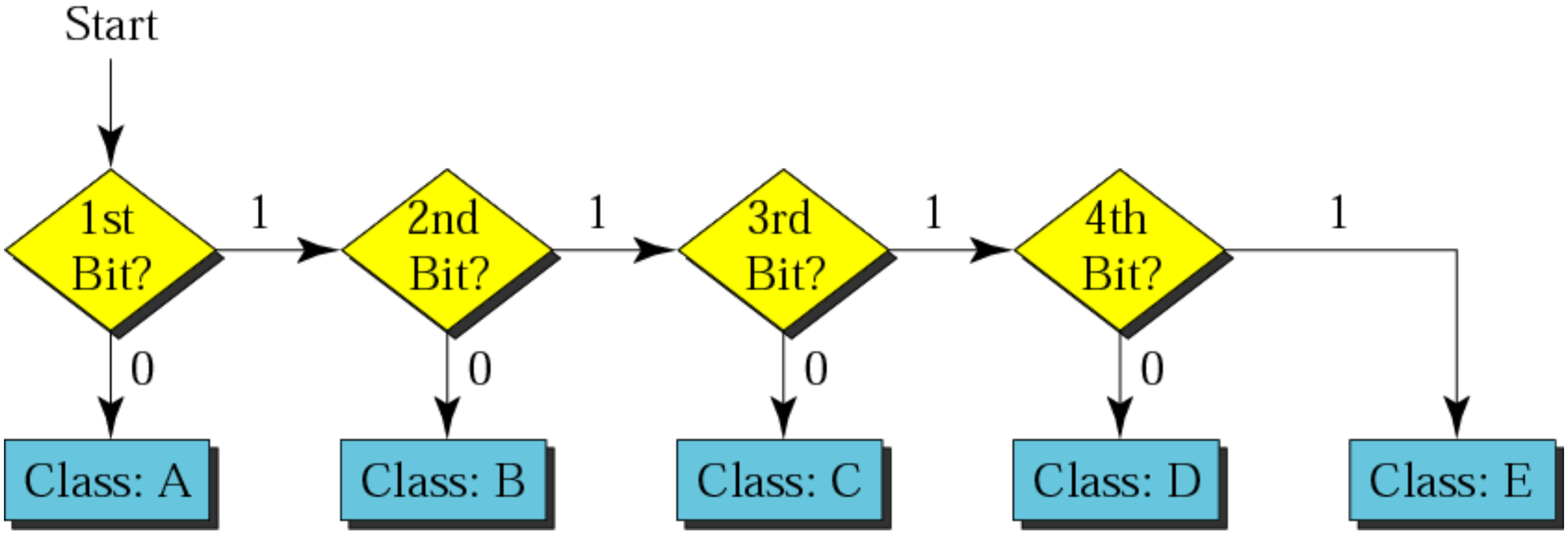
Used till the introduction of CIDR 1993

0 8 16 24 31

Class A	0 *netid*	*hostid*	2^7 nets, 2^{24}hosts	1.0.0.0 to 127.255.255.255
Class B	10 *netid*	*hostid*	2^{14} nets, 2^{16} hosts	128.0.0.0 to 191.255.255.255
Class C	110 *netid*	*hostid*	2^{21} nets, 2^8 hosts	192.0.0.0 to 223.255.255.255
Class D	1110 *multicast address*			224.0.0.0 to 239.255.255.255
Class E	1111 *reserved for future use*			240.0.0.0 to 255.255.255.255

Problem: Networks only come in three sizes!

Finding the address class



What are the issues?

- An organization requires 6 nets each of size 30. Does it have to buy 6 class C address blocks?

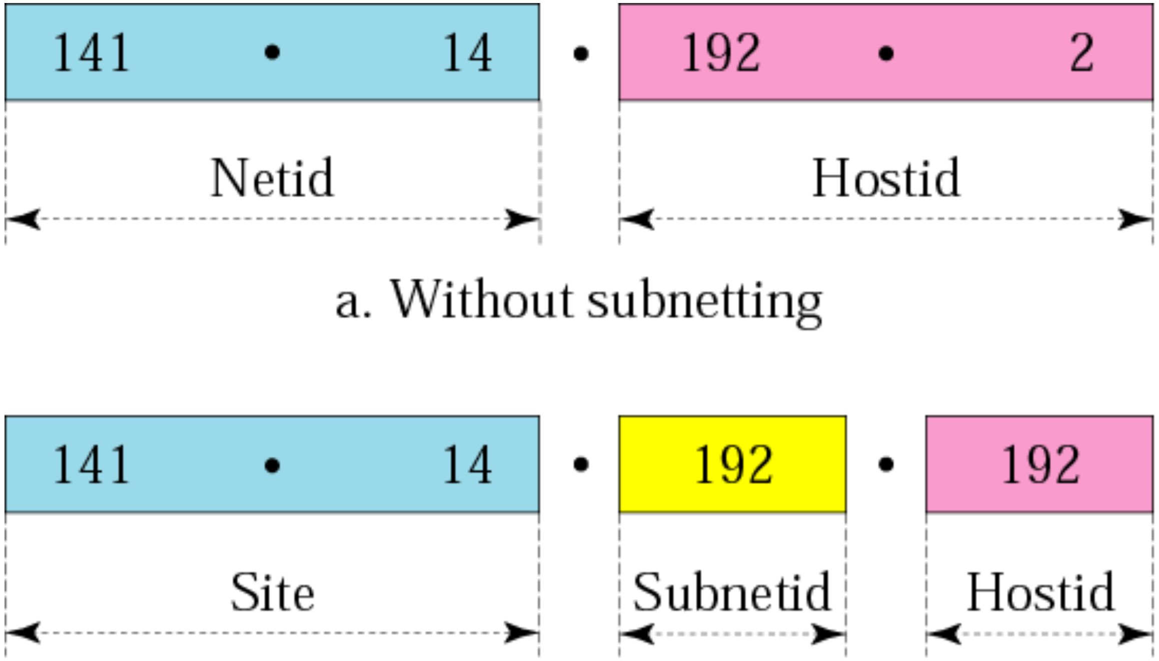
- An organization requires 512 addresses? How many IP addresses should it buy?

Subnetting

- Subnetting is the process of dividing the class A, B or C network into more manageable chunks that are suited to your network' s size and structure.
- Subnetting allows 3 levels of hierarchy
  - netid, subnetid, hostid
- Original netid remains the same and designates the site
- Subnetting remains transparent outside the site

Subnetting

- The process of subnetting simply extends the point where the 1's of Mask stop and 0's start
- You are sacrificing some host ID bits to gain Network ID bits



a. Without subnetting

b. With subnetting

Quiz?

A company is granted the site address 201.70.64.0 (class C). The company needs six subnets. Design the subnets.

The company needs six subnets. 6 is not a power of 2. The next number that is a power of 2 is 8 (2^3). We need 3 more 1s in the subnet mask. The total number of 1s in the subnet mask is 27 (24 + 3). The mask is

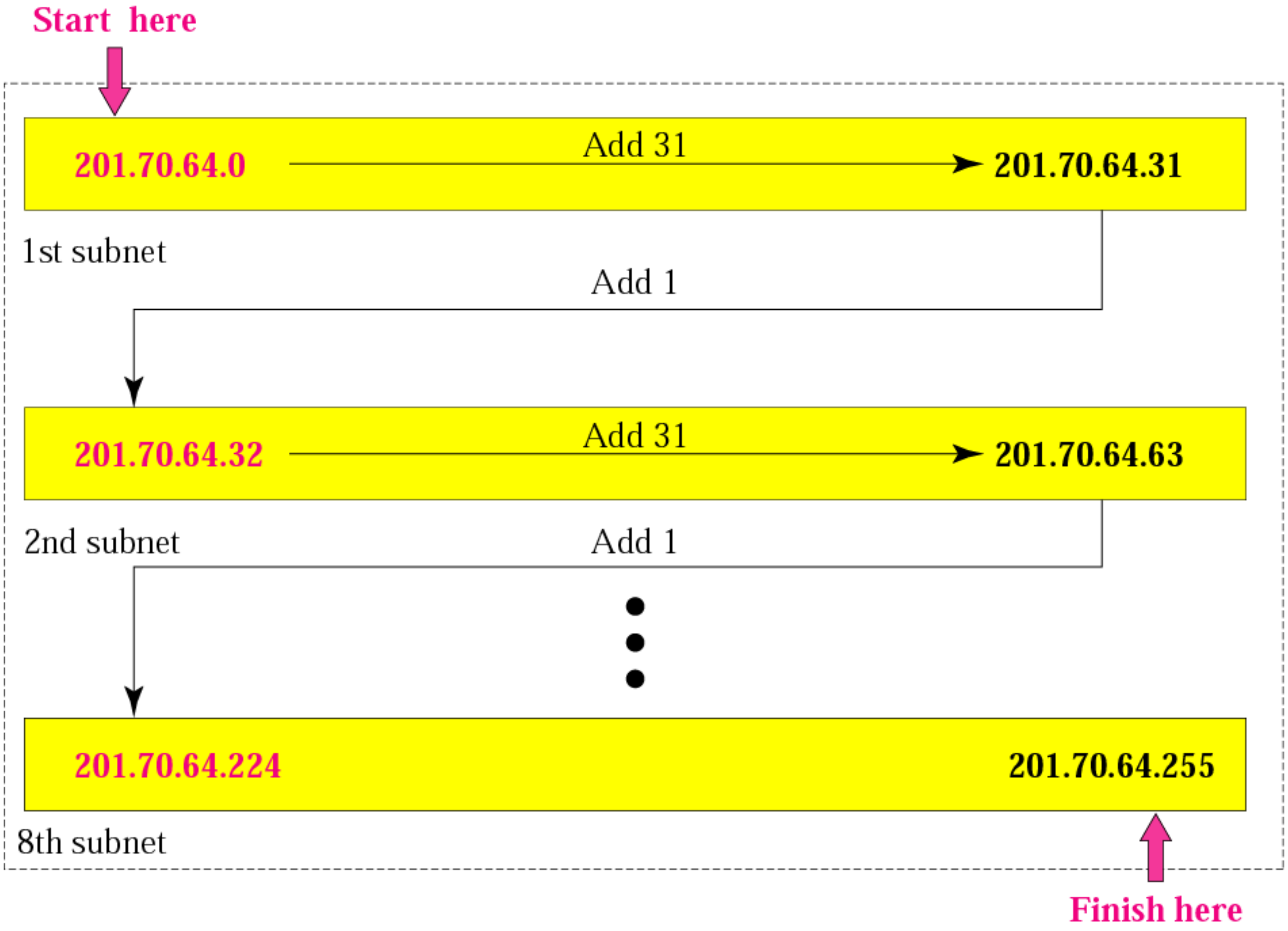
11111111 11111111 11111111 11100000

or 255.255.255.224

Number of addresses in each subnet = 2^5

= 32

The number of addresses in each subnet is 2^5 or 32.

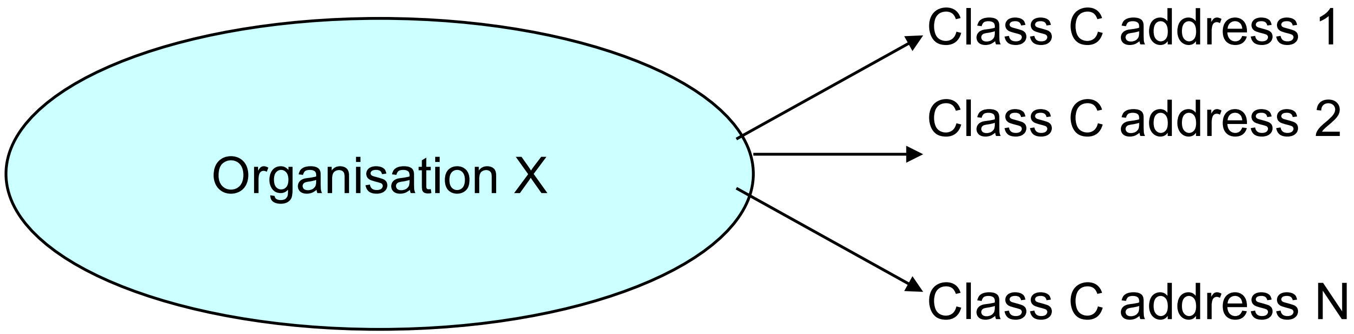


Supernetting

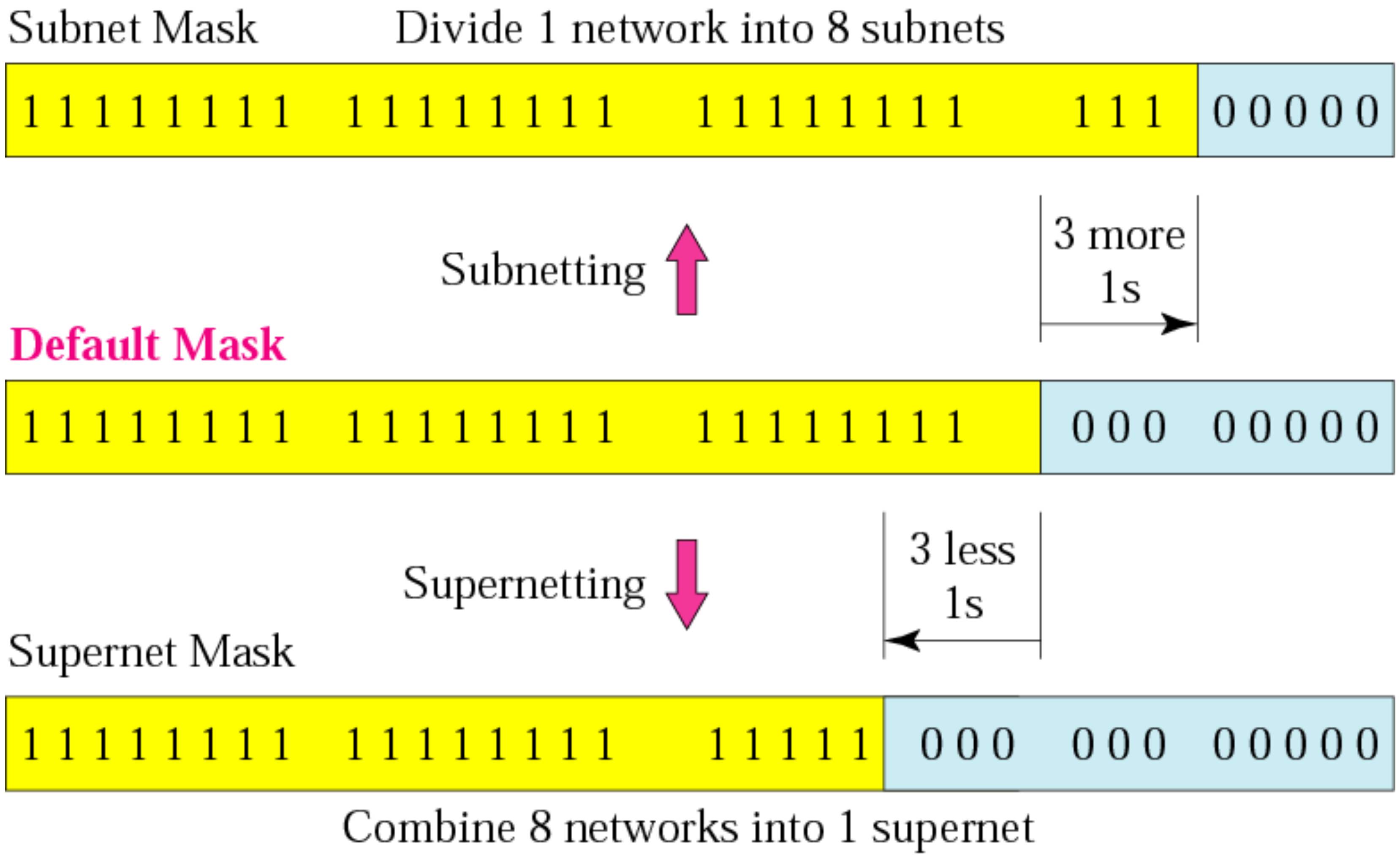
- Traditionally, one organisation one global netid
- Supernetting allows the addresses assigned to *a single organization to span multiple classed prefixes*
- *Supernetting*, one organisation, many netids (a block of netids)
- With supernetting, a single organisation is known to the rest of the world by multiple (a block) netids

Supernetting Example

- An organisation needing 1000 addresses no longer needs a class B (and waste 95% of its address space) address, it can be allocated 4 *contiguous* unused class C addresses.



Comparison of subnet, default, and supernet



Supernet Mask Example

- 11111111.11111111.11111100.00000000
  - allocate 4 Class C addresses
- 11111111.11111111.11111000.00000000
  - allocate 8 Class C addresses
- 11111111.11111111.11110000.00000000
  - allocate 16 Class C addresses

1s
0s

Supernetting

- Follow Three rules (for class C networks)

1. The number of blocks must be a power of 2 (2, 4, 8, 16, . . .).
2. The blocks must be contiguous in the address space (no gaps between the blocks).
3. The third byte of the first address in the superblock must be evenly divisible by the number of blocks. In other words, if the number of blocks is *N*, the third byte must be divisible by *N*.

Supernetting

- 205.109.16.0 to 205.109.23.0
  - Address blocks contiguous
  - Lowest address in block is 205.109.16.0
  - 8 class C addresses. 16/8 is OK
  - Need 3 bits from netid (24-3=255.255.248.0)

16 = 00010000
23 = 00010111

24 = 00011000

- Two items to specify a given block of addresses
  - 32-bit lowest address in the block :205.109.16.0
  - 32-bit mask : 255.255.248.0

- When a packet is received, the router
  - applies the mask to the destination address
  - if result matches the lowest address, packet belongs to the supernet

Example

- Lowest address: 214.128.32.0
  - Bit Mask : 255.255.252.0 (2 bit supernetting)

- IP Packets with following destination addresses will belong to this supernet: 214.128.32.0, 214.128.33.0,214.128.34.0,214.128.35.0

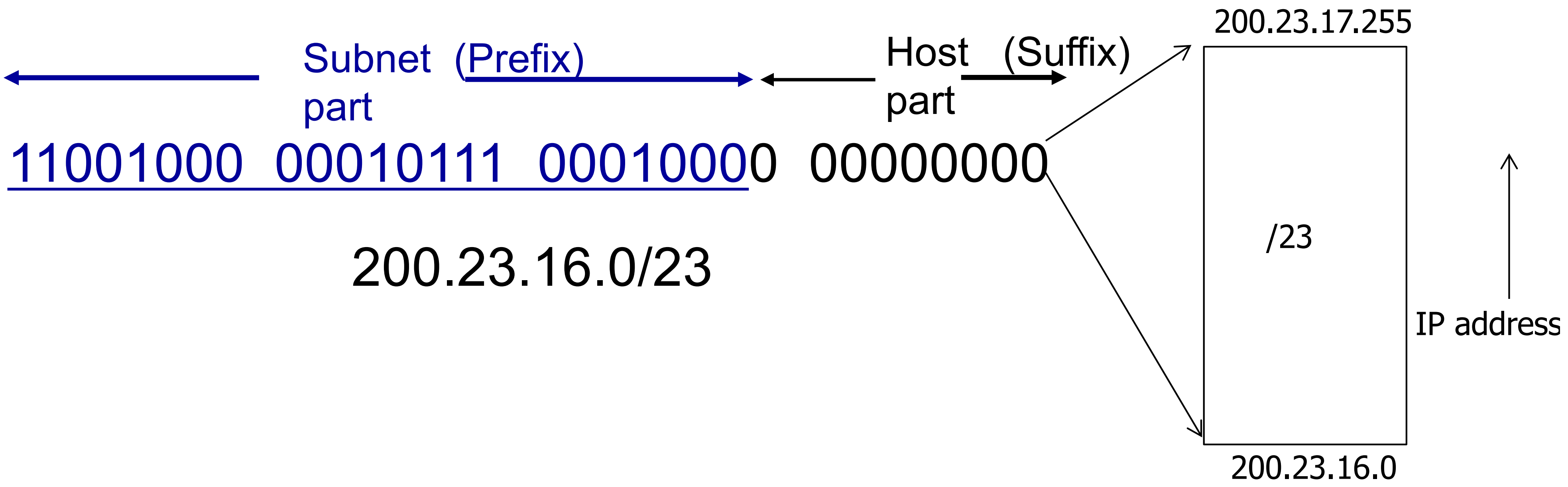
Packet arrives with destination address = 214.128.35.6

Destination Address	11010110.10000000.00100011.00000110	AND OP
Bit Mask	11111111 .11111111 .11111100.00000000	
Lowest Address	11010110.10000000.00100000.00000000	

Today's addressing: CIDR

CIDR: Classless InterDomain Routing

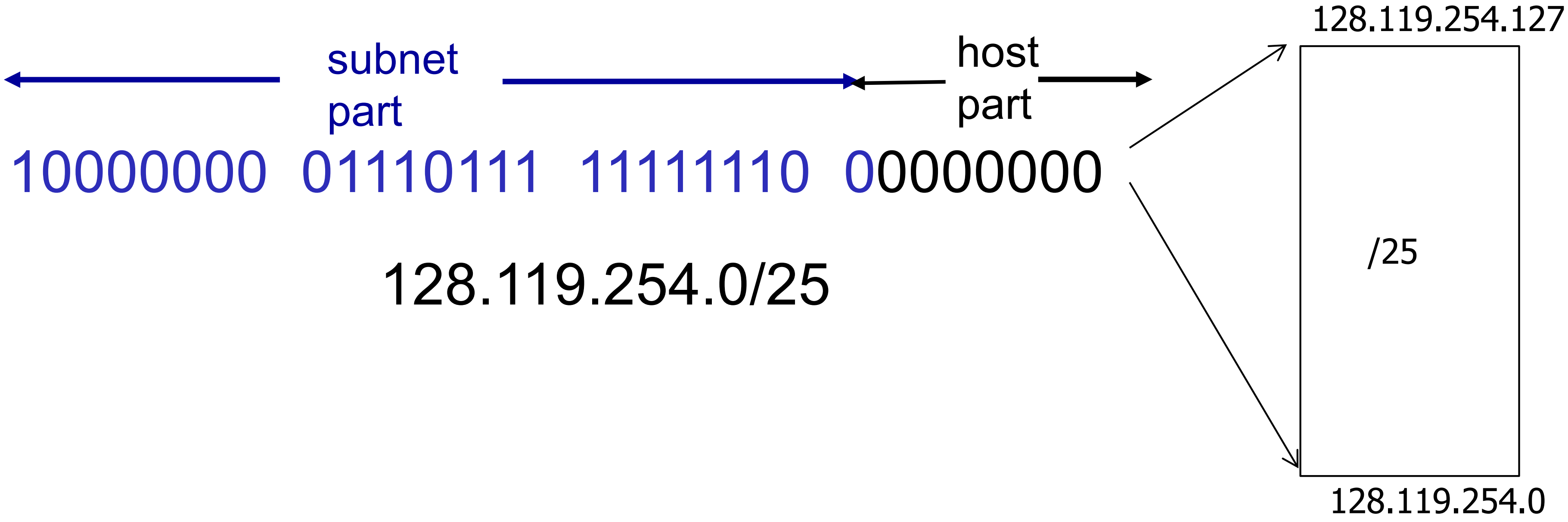
- subnet portion of address of arbitrary length
- address format: a.b.c.d/x, where x is # bits in subnet portion of address



Quiz: IP Addressing

- How many IP addresses belong to the subnet 128.119.254.0/25 ? What are the IP addresses at the two end-points of this range ?

Answer: 2^7 = 128 addresses (126 are usable)



128.119.254.0/25

Quiz: IP Addressing

- A small organization is given a block with the beginning address and the prefix length 205.16.37.24/29 (in slash notation). What is the range of the block?

The beginning address is 205.16.37.24. To find the last address we keep the first 29 bits and change the last 3 bits to 1s.

Beginning:11001111 00010000 00100101 00011000

Ending : 11001111 00010000 00100101 00011111

There are only 8 addresses in this block.

205.16.37.24 to 205.16.37.31

Quiz: IP Addressing

- An ISP is granted a block of addresses starting with 190.100.0.0/16. The ISP needs to distribute these addresses to three groups of customers as follows:

1. The first group has 64 customers; each needs 256 addresses.
2. The second group has 128 customers; each needs 128 addresses.
3. The third group has 128 customers; each needs 64 addresses.

Design the subblocks and give the slash notation for each subblock. Find out how many addresses are still available after these allocations.

Group 1

For this group, each customer needs 256 addresses. This means the suffix length is 8 ($2^8 = 256$). The prefix length is then $32 - 8 = 24$.

01: 190.100.0.0/24 ➔190.100.0.255/24

02: 190.100.1.0/24 ➔190.100.1.255/24

…………………………………..

64: 190.100.63.0/24➔190.100.63.255/24

Total = 64 × 256 = 16,384

Group 2

For this group, each customer needs 128 addresses. This means the suffix length is 7 ($2^7 = 128$). The prefix length is then 32 – 7 = 25. The addresses are:

001: 190.100.64.0/25 → 190.100.64.127/25

002: 190.100.64.128/25 → 190.100.64.255/25

……………………………………………

128: 190.100.127.128/25 → 190.100.127.255/25

Total = 128 × 128 = 16,384

Group 3

For this group, each customer needs 64 addresses. This means the suffix length is 6 ($2^6 = 64$). The prefix length is then 32 – 6 = 26.

001:190.100.128.0/26 → 190.100.128.63/26

002:190.100.128.64/26 → 190.100.128.127/26

…………………………

128:190.100.159.192/26 → 190.100.159.255/26

Total = 128 × 64 = 8,192

Number of granted addresses: 65,536

Number of allocated addresses: 40,960

Number of available addresses: 24,576

IP addresses: how to get one?

Q: How does a *host* get IP address?

- hard-coded by system admin in a file
  - Windows: control-panel->network->configuration->tcp/ip->properties
  - UNIX: /etc/rc.config
- DHCP: Dynamic Host Configuration Protocol: dynamically get address from as server
  - "plug-and-play"

DHCP

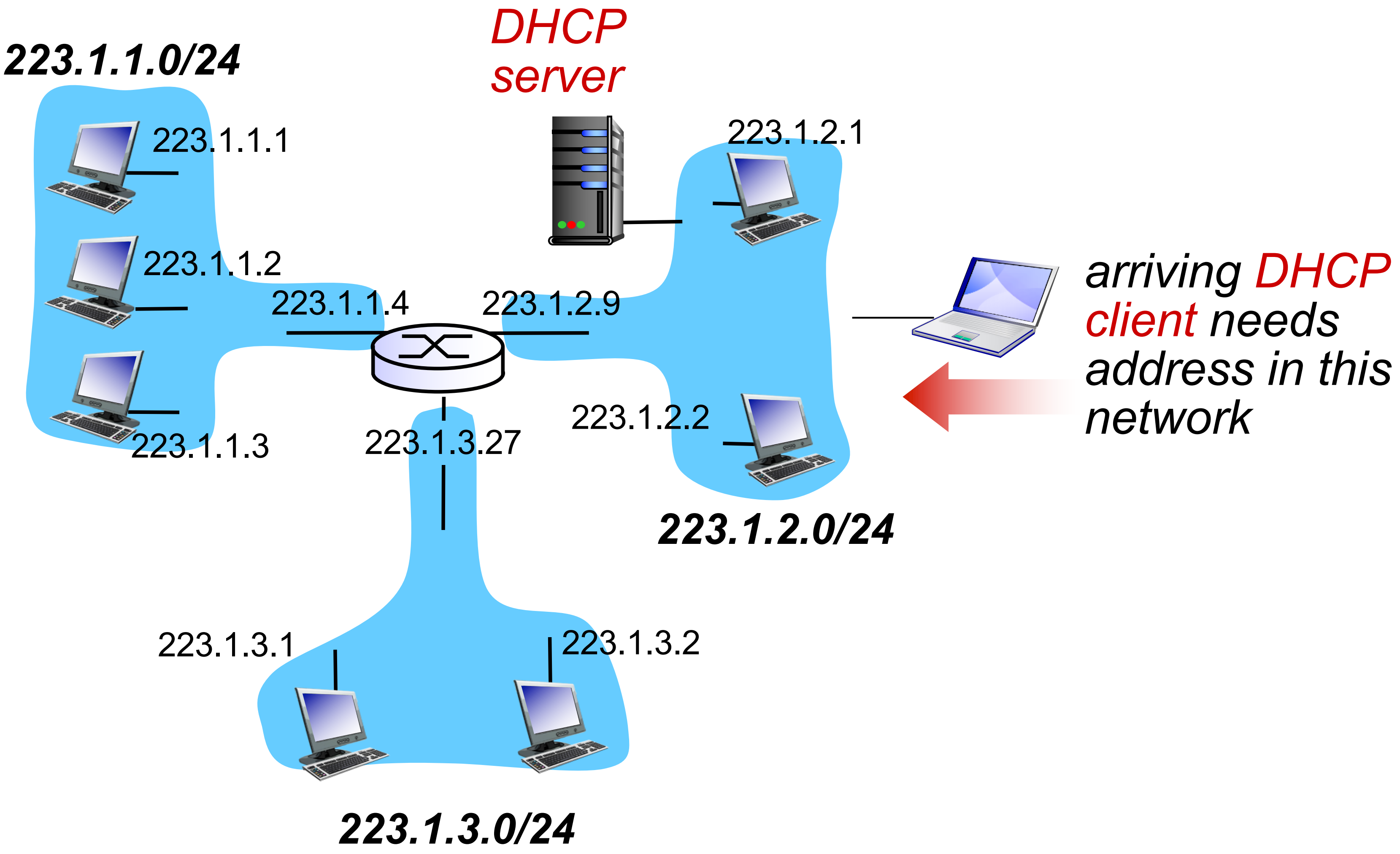
goal: allow host to *dynamically* obtain its IP address from network server when it joins network

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

DHCP overview:

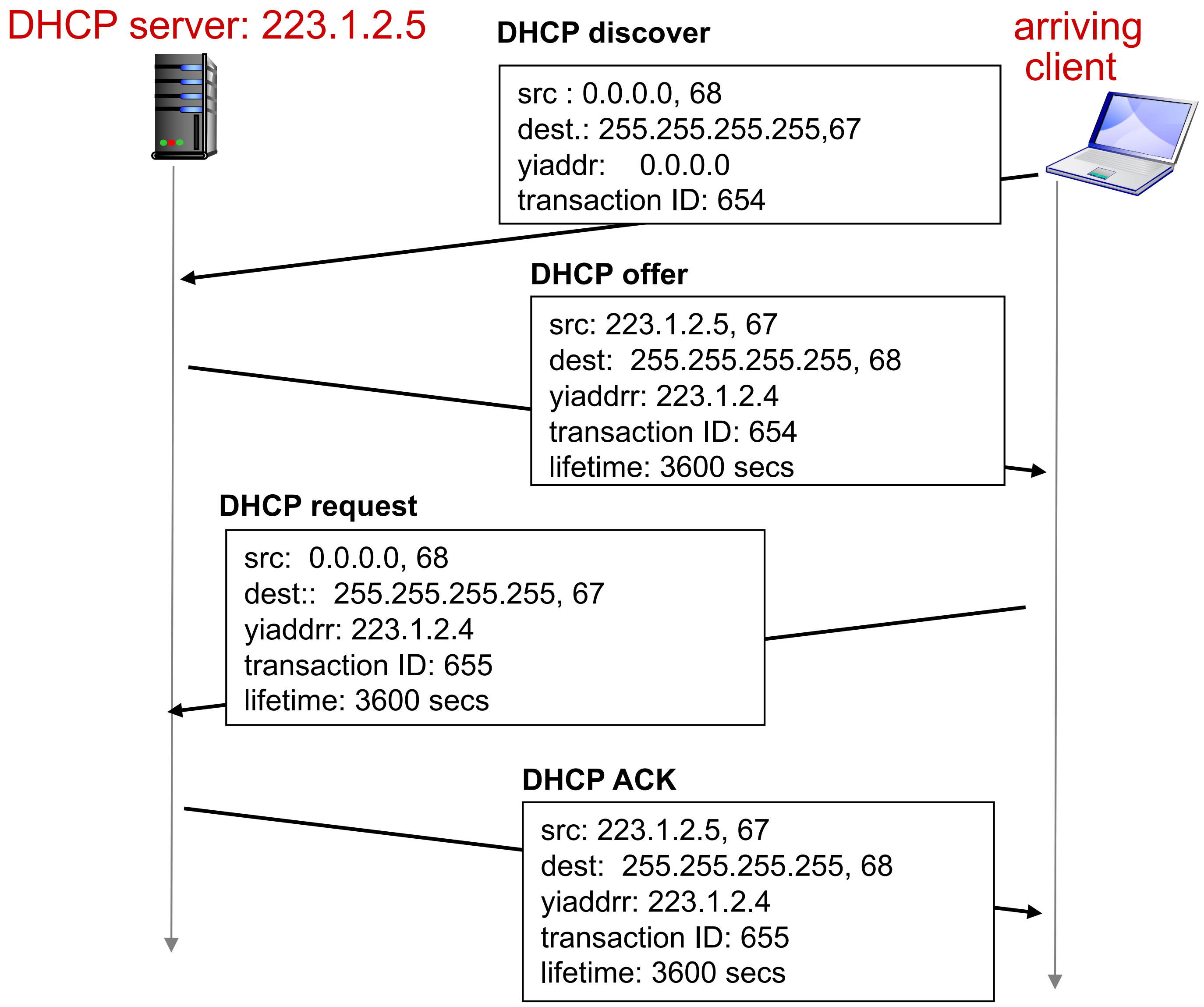
- host broadcasts "DHCP discover" msg
- DHCP server responds with "DHCP offer" msg
- host requests IP address: "DHCP request" msg
- DHCP server sends address: "DHCP ack" msg

DHCP client-server scenario



arriving DHCP client needs address in this network

DHCP client-server scenario



DHCP server: 223.1.2.5

arriving client

DHCP discover

src : 0.0.0.0, 68
dest.: 255.255.255.255,67
yiaddr: 0.0.0.0
transaction ID: 654

DHCP offer

src: 223.1.2.5, 67
dest: 255.255.255.255, 68
yiaddrr: 223.1.2.4
transaction ID: 654
lifetime: 3600 secs

DHCP request

src: 0.0.0.0, 68
dest:: 255.255.255.255, 67
yiaddrr: 223.1.2.4
transaction ID: 655
lifetime: 3600 secs

DHCP ACK

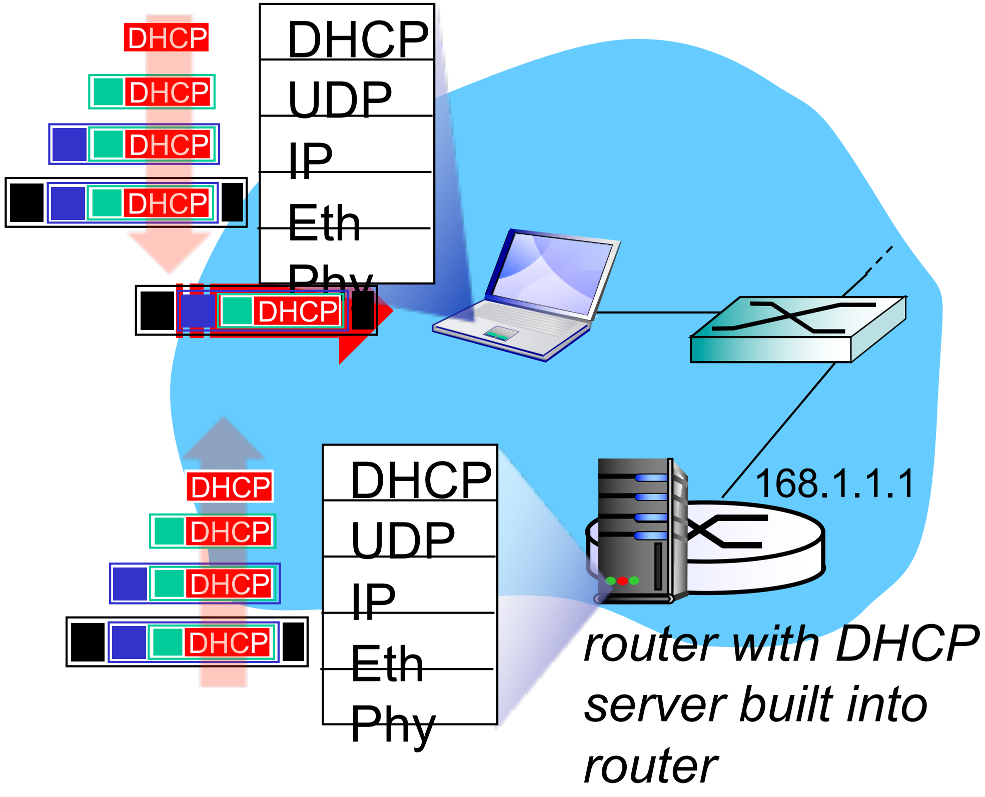
src: 223.1.2.5, 67
dest: 255.255.255.255, 68
yiaddrr: 223.1.2.4
transaction ID: 655
lifetime: 3600 secs

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)

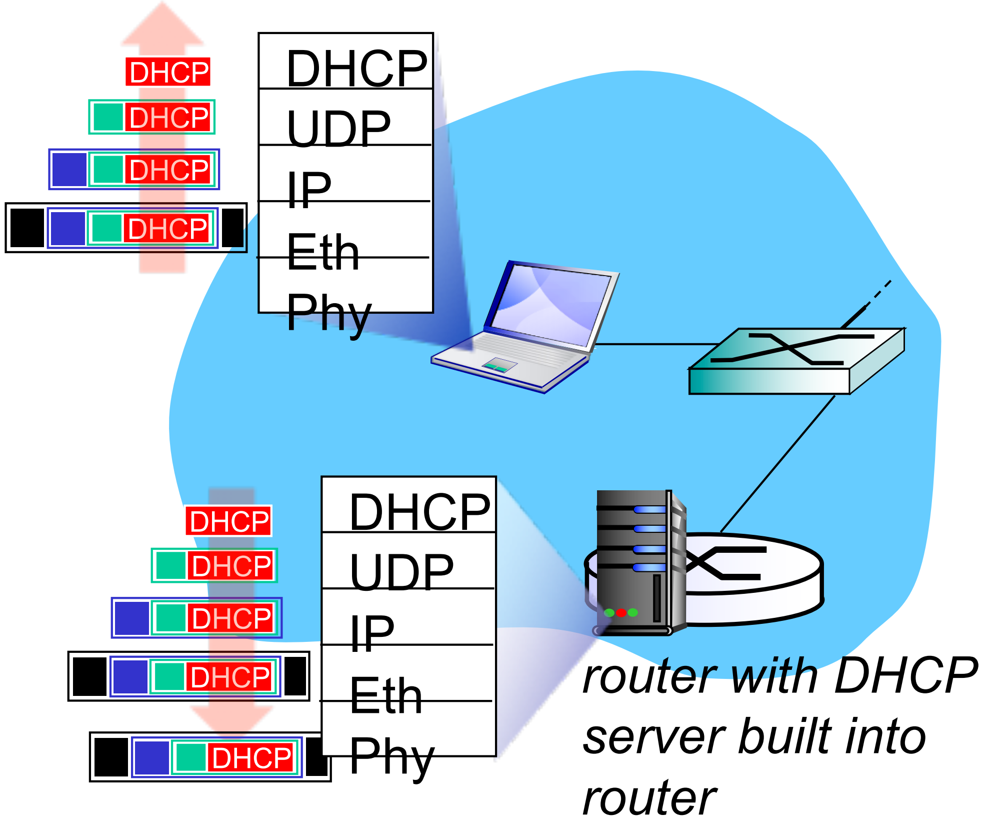
DHCP: example



router with DHCP server built into router

- connecting laptop needs its IP address, addr of first-hop router, addr of DNS server: use DHCP
- DHCP request encapsulated in UDP, encapsulated in IP, encapsulated in 802.1 Ethernet
- Ethernet frame broadcast (dest: FFFFFFFFFFFF) on LAN, received at router running DHCP server
- Ethernet demuxed to IP demuxed, UDP demuxed to DHCP

DHCP: example



router with DHCP server built into router

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

DHCP: Wireshark output (home LAN)

request

```
Message type: Boot Request (1)
Hardware type: Ethernet
Hardware address length: 6
Hops: 0
Transaction ID: 0x6b3a11b7
Seconds elapsed: 0
Bootp flags: 0x0000 (Unicast)
Client IP address: 0.0.0.0 (0.0.0.0)
Your (client) IP address: 0.0.0.0 (0.0.0.0)
Next server IP address: 0.0.0.0 (0.0.0.0)
Relay agent IP address: 0.0.0.0 (0.0.0.0)
Client MAC address: Wistron_23:68:8a (00:16:d3:23:68:8a)
Server host name not given
Boot file name not given
Magic cookie: (OK)
Option: (t=53,l=1) DHCP Message Type = DHCP Request
Option: (61) Client identifier
    Length: 7; Value: 010016D323688A;
    Hardware type: Ethernet
    Client MAC address: Wistron_23:68:8a (00:16:d3:23:68:8a)
Option: (t=50,l=4) Requested IP Address = 192.168.1.101
Option: (t=12,l=5) Host Name = "nomad"
Option: (55) Parameter Request List
    Length: 11; Value: 010F03062C2E2F1F21F92B
    1 = Subnet Mask; 15 = Domain Name
    3 = Router; 6 = Domain Name Server
    44 = NetBIOS over TCP/IP Name Server
    ……
```

reply

```
Message type: Boot Reply (2)
Hardware type: Ethernet
Hardware address length: 6
Hops: 0
Transaction ID: 0x6b3a11b7
Seconds elapsed: 0
Bootp flags: 0x0000 (Unicast)
Client IP address: 192.168.1.101 (192.168.1.101)
Your (client) IP address: 0.0.0.0 (0.0.0.0)
Next server IP address: 192.168.1.1 (192.168.1.1)
Relay agent IP address: 0.0.0.0 (0.0.0.0)
Client MAC address: Wistron_23:68:8a (00:16:d3:23:68:8a)
Server host name not given
Boot file name not given
Magic cookie: (OK)
Option: (t=53,l=1) DHCP Message Type = DHCP ACK
Option: (t=54,l=4) Server Identifier = 192.168.1.1
Option: (t=1,l=4) Subnet Mask = 255.255.255.0
Option: (t=3,l=4) Router = 192.168.1.1
Option: (6) Domain Name Server
    Length: 12; Value: 445747E2445749F244574092;
     IP Address: 68.87.71.226;
     IP Address: 68.87.73.242;
     IP Address: 68.87.64.146
Option: (t=15,l=20) Domain Name = "hsd1.ma.comcast.net."
```

DHCP: further details

- DHCP uses UDP and port numbers 67 (server side) and 68 (client side)
- Usually the MAC address is used to identify clients
  - DHCP server can be configured with a "registered list" of acceptable MAC addresses
- DHCP offer message includes ip address, length of lease, subnet mask, DNS servers, default gateway
- DHCP security holes
  - DoS attack by exhausting pool of IP addresses
  - Masquerading as a DHCP server
  - Authentication for DHCP - RFC 3118

IP addresses: how to get one?

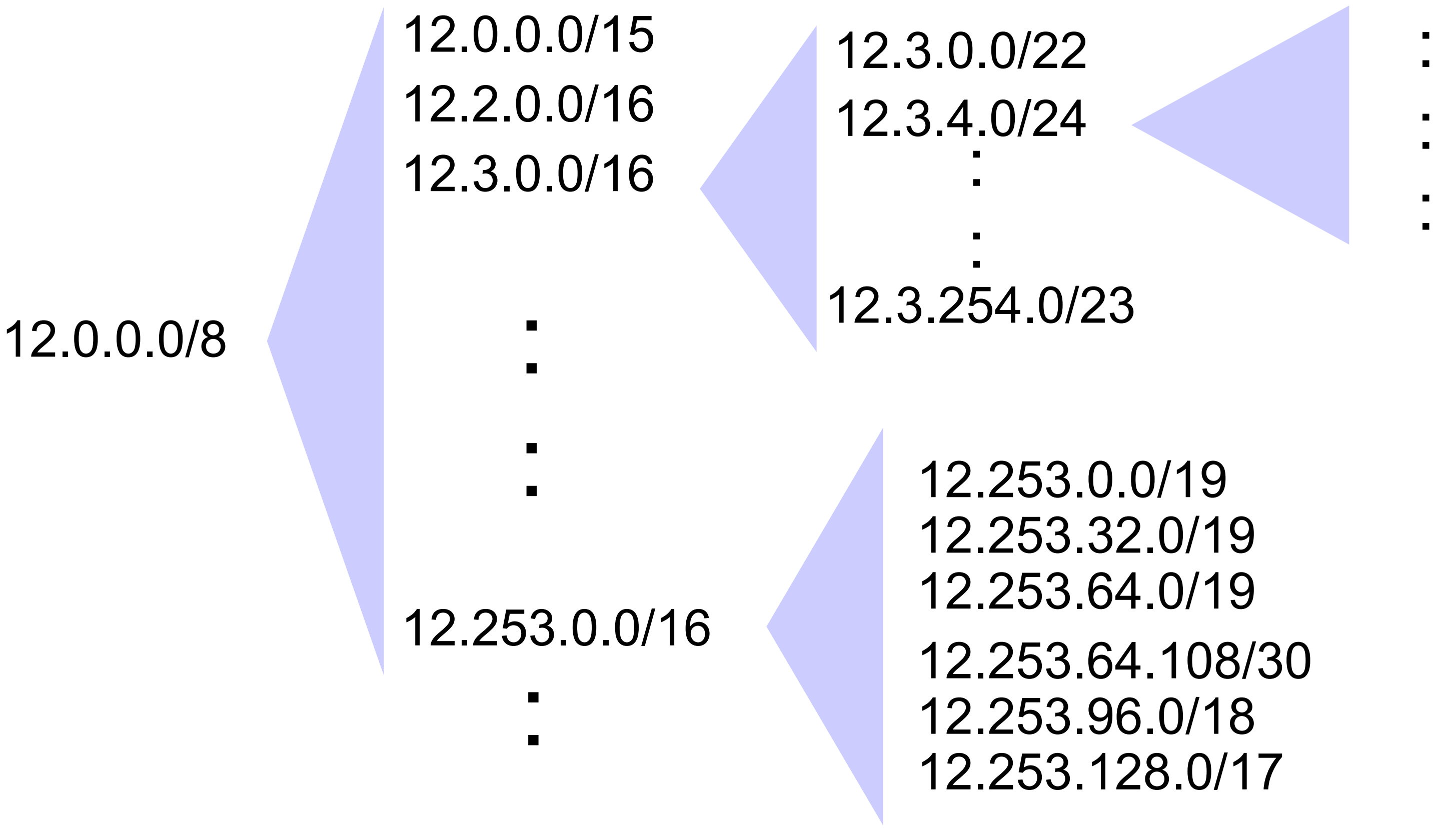
Q: how does *network* get subnet part of IP addr?

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

ISP's block	11001000 00010111 00010000 00000000	200.23.16.0/20
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

CIDR: Addresses allocated in contiguous prefix chunks

Recursively break down chunks as get closer to host



12.0.0.0/8

12.0.0.0/15
12.2.0.0/16
12.3.0.0/16
:
:
12.253.0.0/16
:

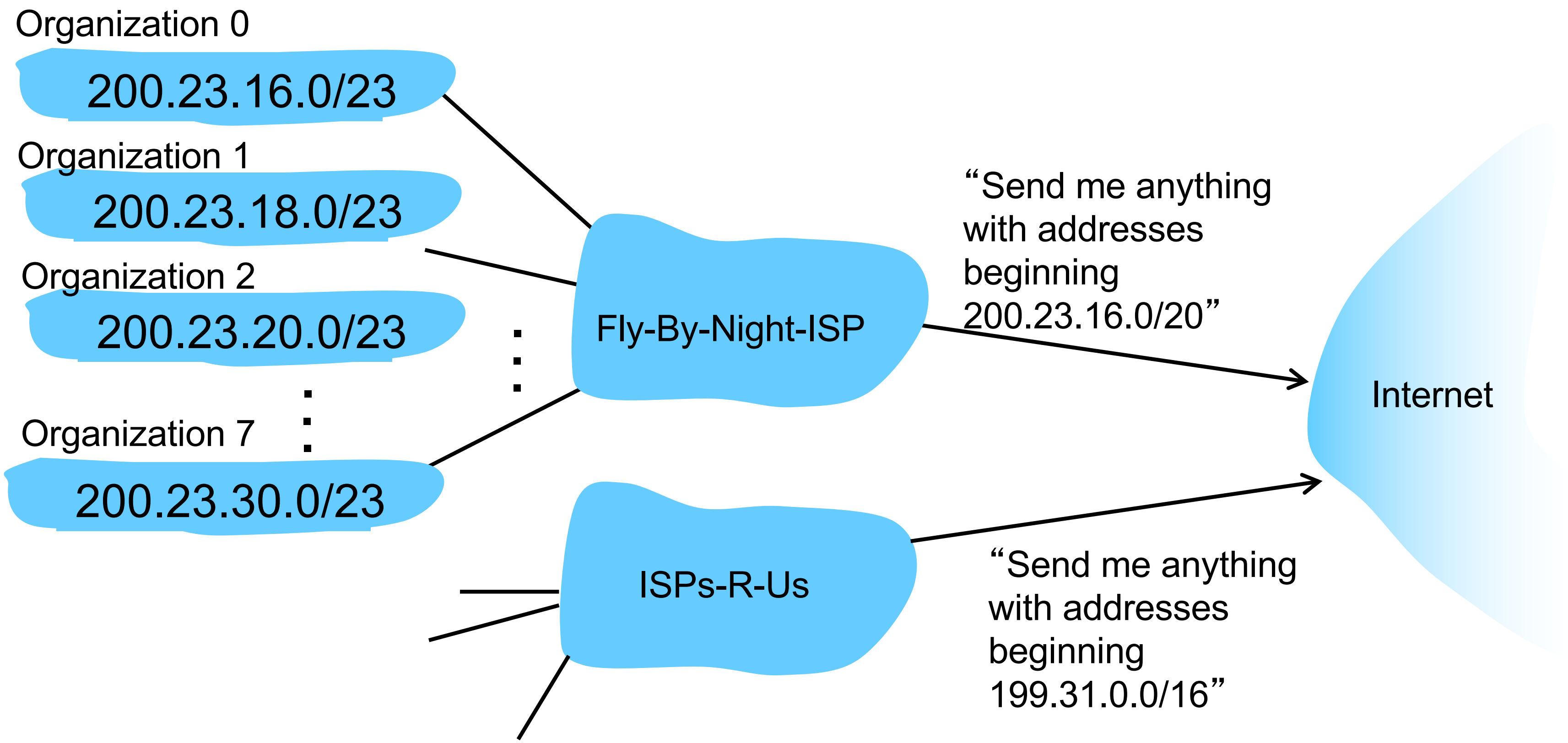
12.3.0.0/22
12.3.4.0/24
:
:
12.3.254.0/23

:
:
:

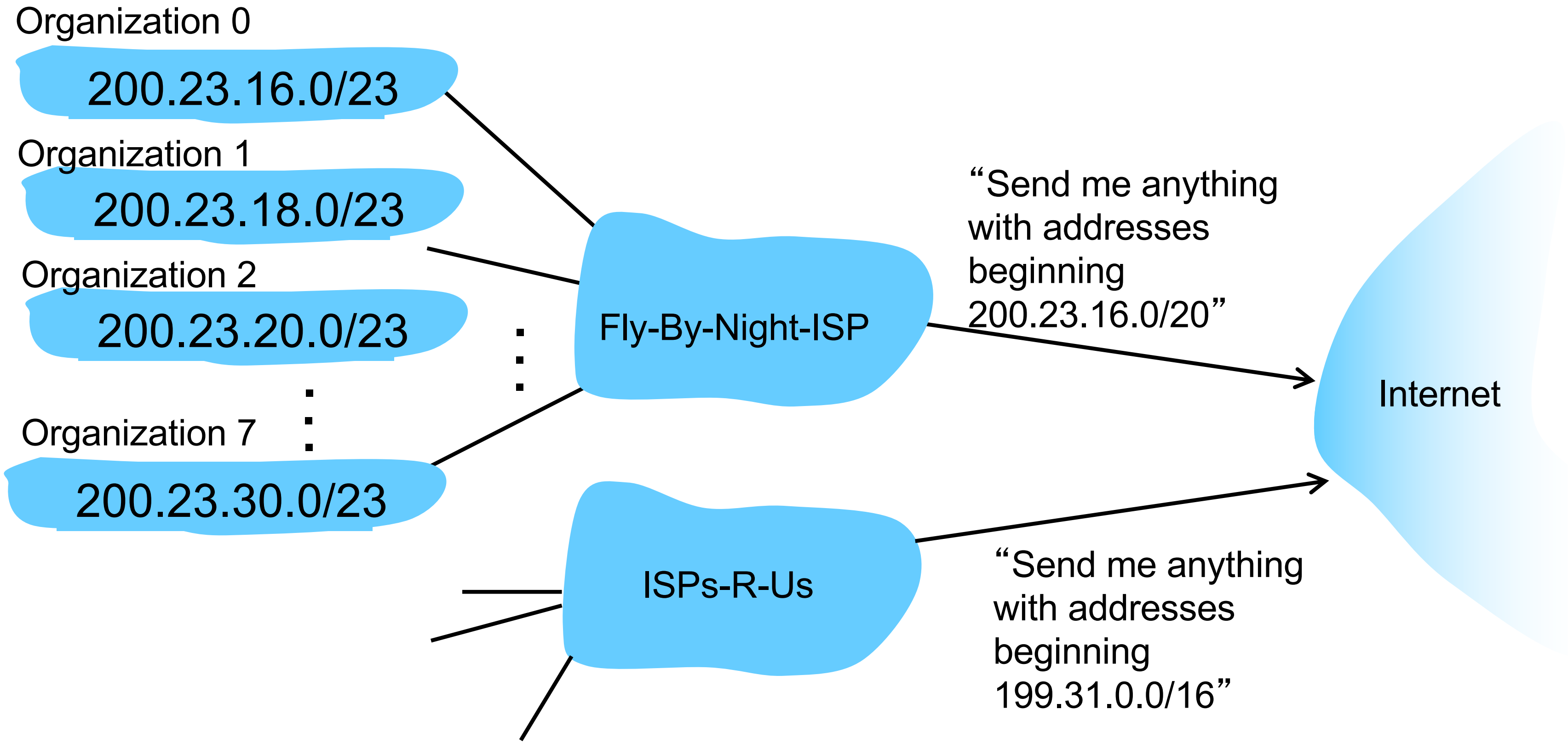
12.253.0.0/19
12.253.32.0/19
12.253.64.0/19
12.253.64.108/30
12.253.96.0/18
12.253.128.0/17

Hierarchical addressing: route aggregation

hierarchical addressing allows efficient advertisement of routing information:



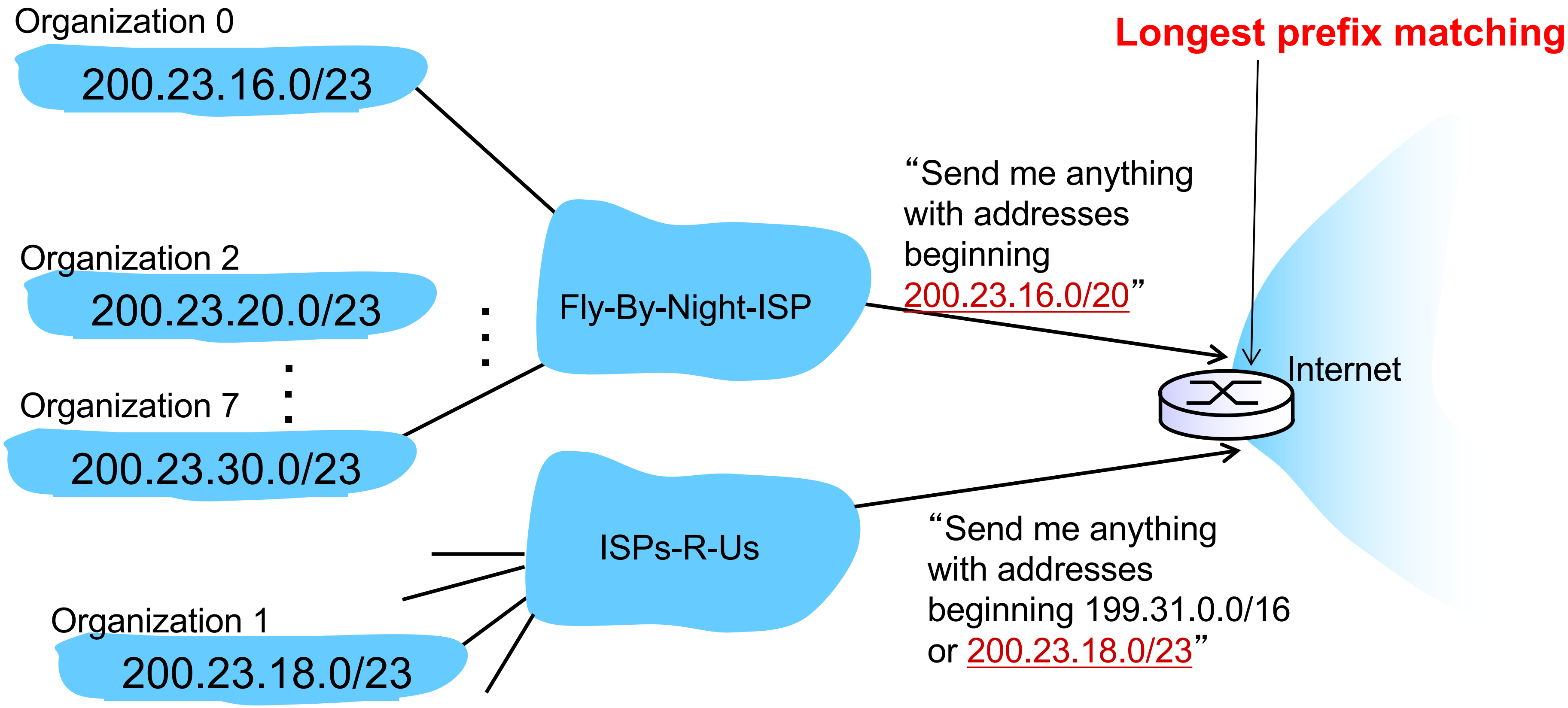
Quiz: What should we do if organization 1 decides to switch to ISPs-R-Us



A: Move 200.23.18.0/23 to ISPs-R-Us (and break up Fly-By-Night's/20 block).
B: Give new addresses to Organization 1 (and force them to change all their addresses)
C: Some other solution

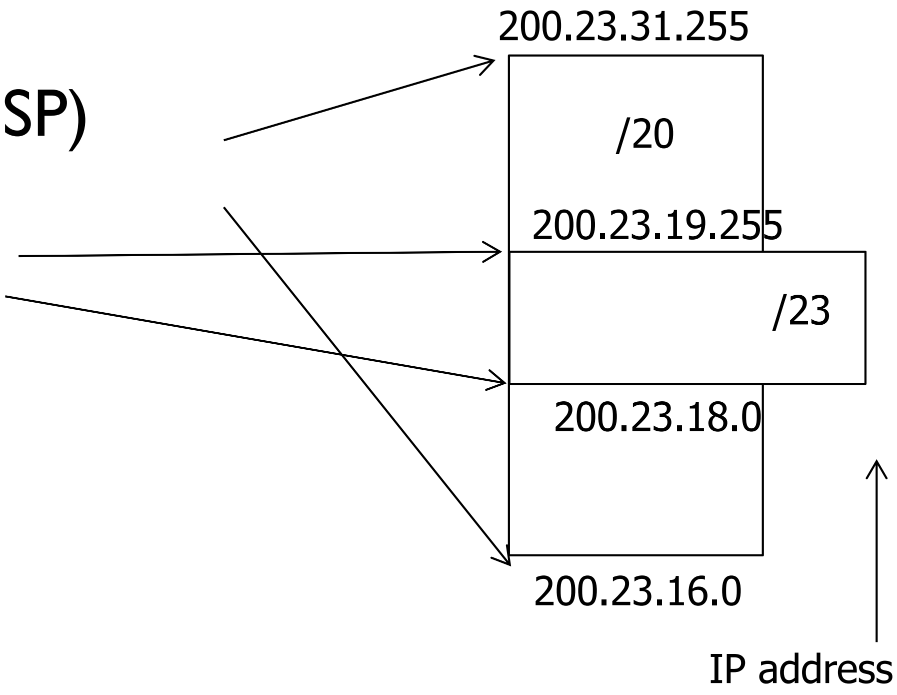
Hierarchical addressing: more specific routes

ISPs-R-Us has a more specific route to Organization 1



Example: continued

- But how will this work?
- Routers in the Internet will have two entries in their tables
  - 200.23.16.0/20 (Fly-by-Night-ISP)
  - 200.23.18.0/23 (ISPs-R-Us)
- Longest prefix match



Longest prefix matching

longest prefix matching

when looking for forwarding table entry for given destination address, use *longest* address prefix that matches destination address.

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

examples:

DA: 11001000 00010111 00010110 10100001 which interface?

DA: 11001000 00010111 00011000 10101010 which interface?

Quiz: Longest prefix matching

- On which outgoing interface will a packet destined to 11011001 be forwarded?

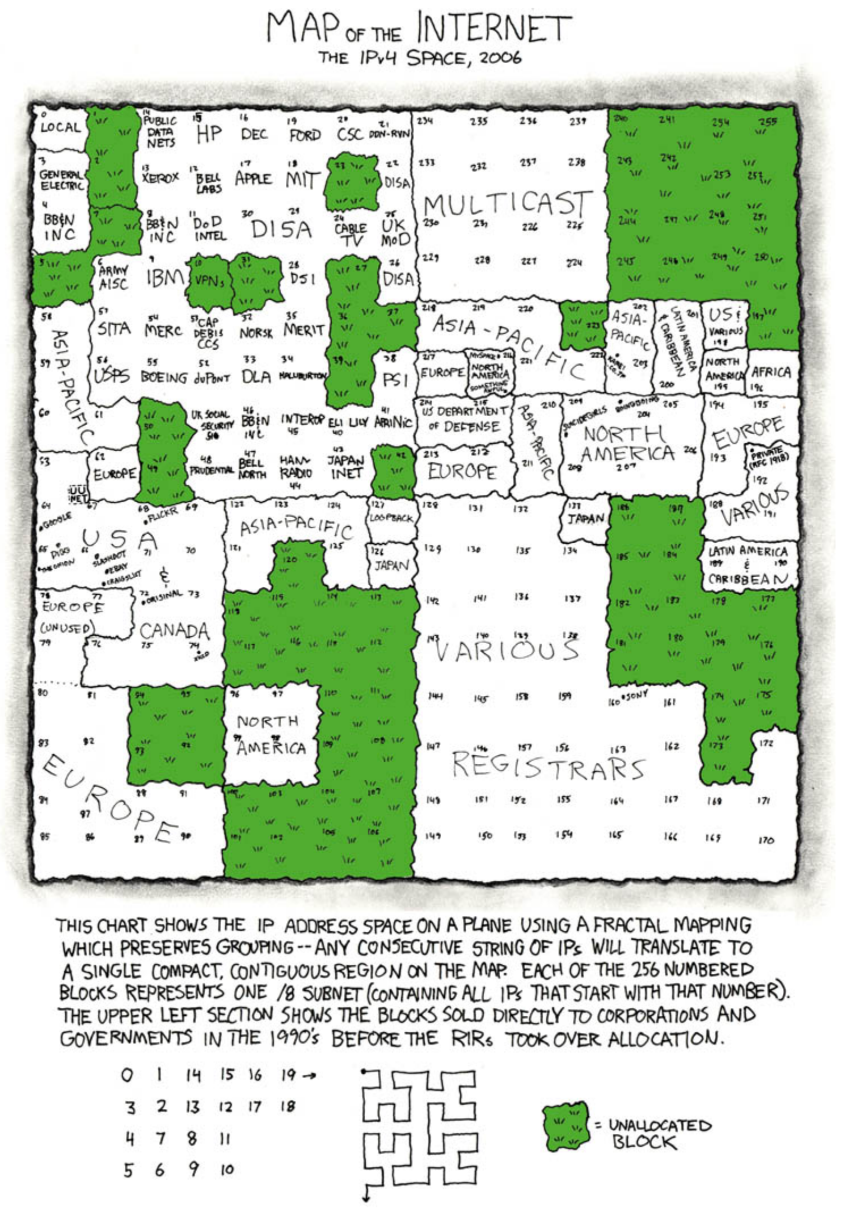
Prefix	Interface
1*	A
11*	B
111*	C
Default	D

More on IP addresses

Source: www.xkcd.com

- IP addresses are allocated as blocks and have geographical significance
- It is possible to determine the geographical location of an IP address

http://www.geobytes.com/IpLocator.htm



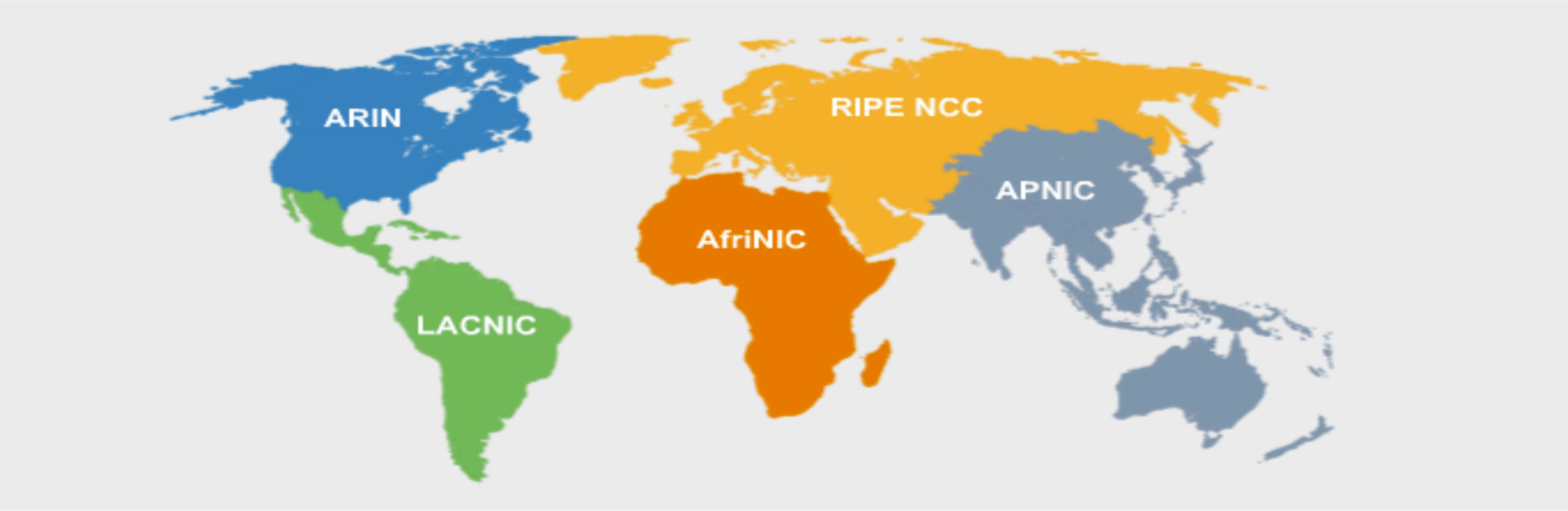
THIS CHART SHOWS THE IP ADDRESS SPACE ON A PLANE USING A FRACTAL MAPPING WHICH PRESERVES GROUPING -- ANY CONSECUTIVE STRING OF IPs WILL TRANSLATE TO A SINGLE COMPACT, CONTIGUOUS REGION ON THE MAP. EACH OF THE 256 NUMBERED BLOCKS REPRESENTS ONE /8 SUBNET (CONTAINING ALL IPs THAT START WITH THAT NUMBER). THE UPPER LEFT SECTION SHOWS THE BLOCKS SOLD DIRECTLY TO CORPORATIONS AND GOVERNMENTS IN THE 1990's BEFORE THE RIRs TOOK OVER ALLOCATION.

IP Addressing: the last word...

Q: How does an ISP get block of addresses?

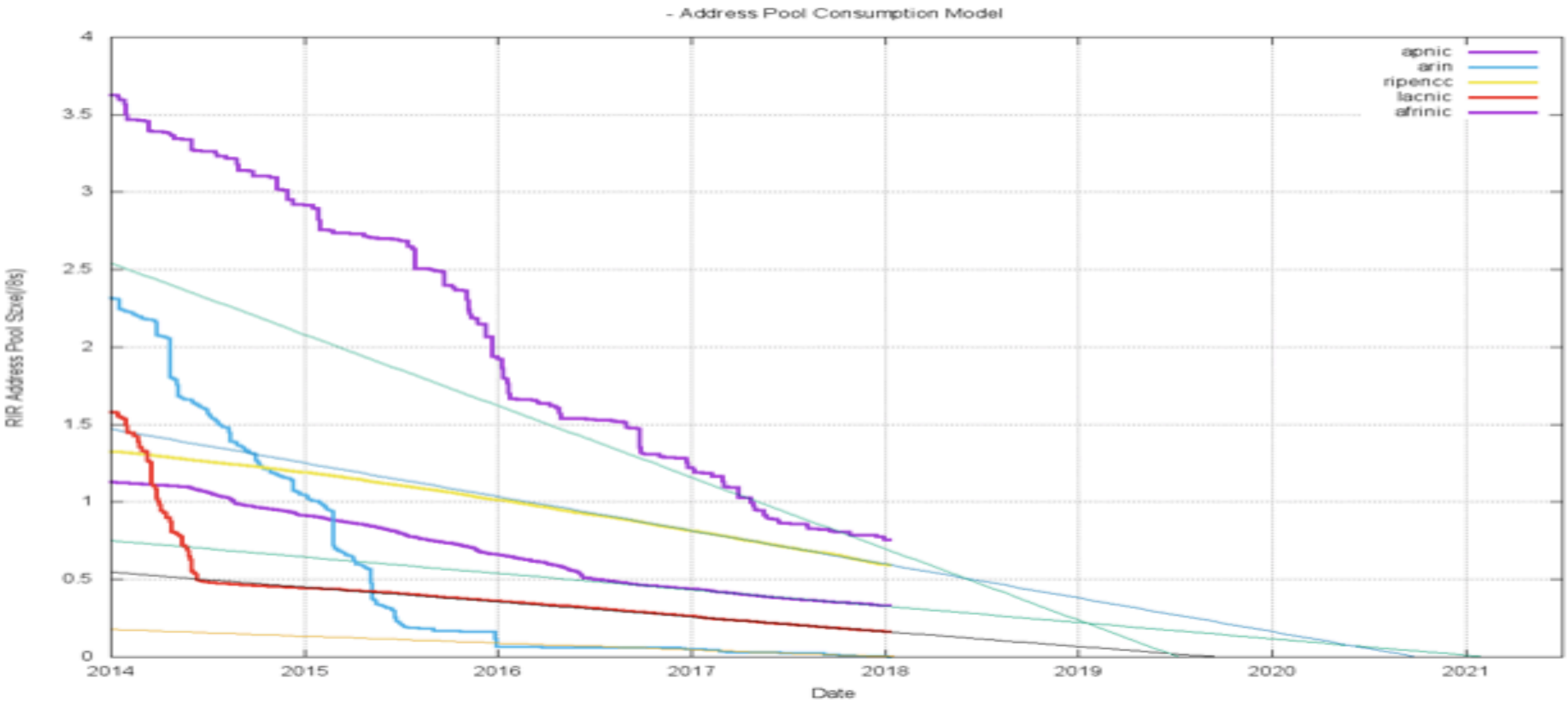
A: ICANN: Internet Corporation for Assigned Names and Numbers http://www.icann.org/

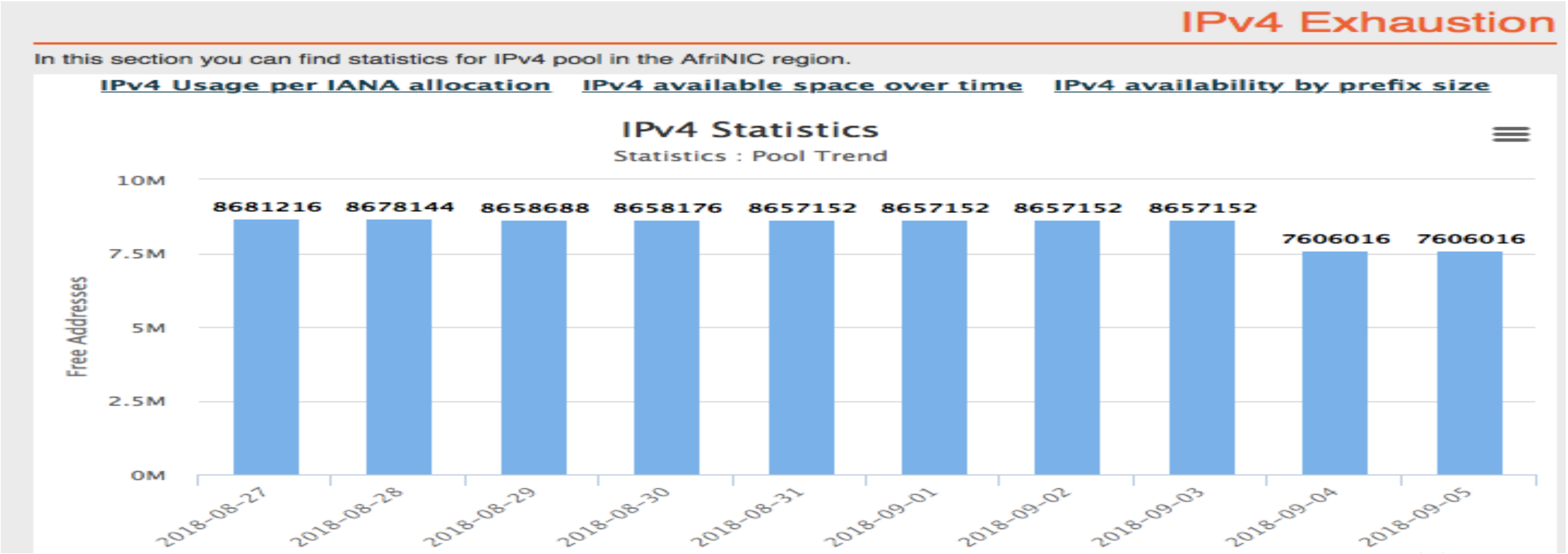
iana Internet Assigned Numbers Authority	IANA works through Regional Internet Registries (RIRs):	RIPE NCC	IRéseaux IP Européens Network Coordination Centre
ARIN American Registry for Internet Numbers	American Registry for Internet Numbers	APNIC	Asia-Pacific Network Information Center
LACNIC	Latin America and Caribbean Network Information Centre	AfriNIC	African Network Information Centre



Made-up Example in More Detail

- ICANN gives APNIC several /8s
- APNIC gives Telstra one /8, **129.0/8**
  - Network Prefix: **10000001**
- Telstra gives UNSW a /16, **129.94/16**
  - Network Prefix: **1000000101011110**
- UNSW gives CSE a /24, **129.94.242/24**
  - Network Prefix: **100000010101111011110010**
- CSE gives me a specific address **129.94.242.51**
  - Address: **10000001010111101111001000110011**





IPv4 Exhaustion

In this section you can find statistics for IPv4 pool in the AfriNIC region.

IPv4 Usage per IANA allocation IPv4 available space over time IPv4 availability by prefix size

Network Layer, data plane: outline

4.1 Overview of Network layer

- data plane
- control plane

4.2 What's inside a router

4.3 IP: Internet Protocol

- datagram format
- fragmentation
- IPv4 addressing
- network address translation

4.4 Generalized Forward and SDN

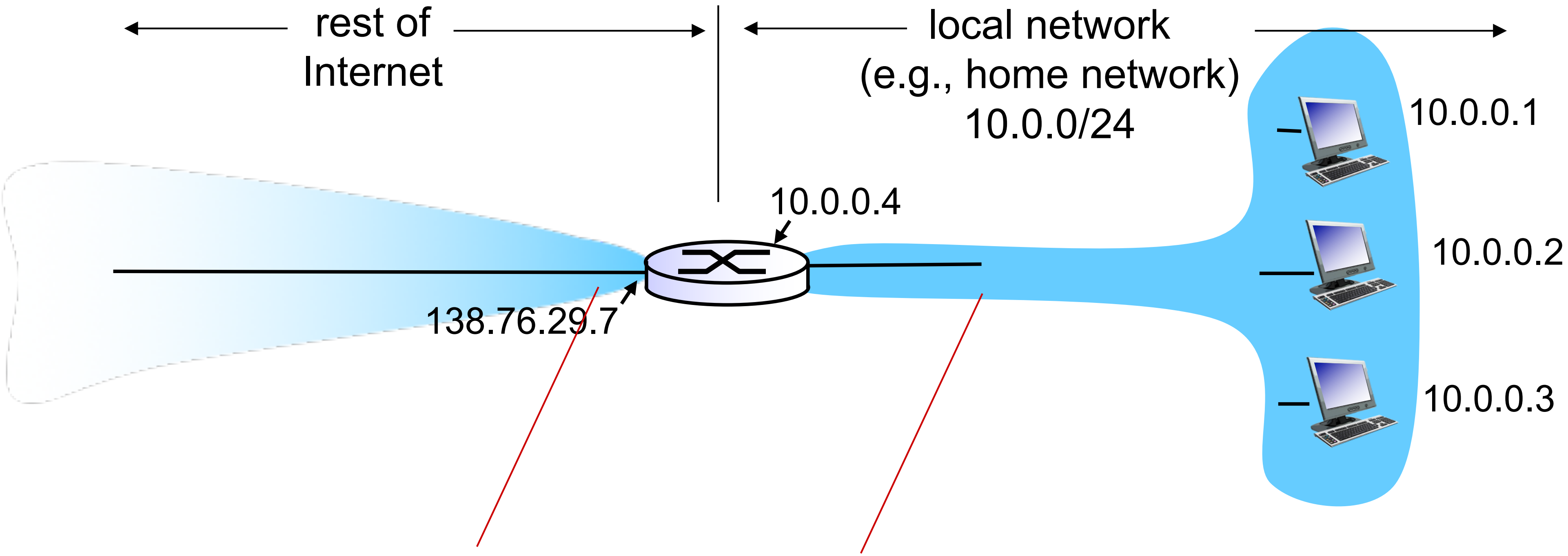
- match
- action
- OpenFlow examples of match-plus-action in action

Private Addresses

- Defined in RFC 1918:
  - 10.0.0.0/8 (16,777,216 hosts)
  - 172.16.0.0/12 (1,048,576 hosts)
  - 192.168.0.0/16 (65536 hosts)

- These addresses cannot be routed
  - Anyone can use them
  - Typically used for NAT

NAT: network address translation



all datagrams *leaving* local network have *same* single source NAT IP address: 138.76.29.7,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

implementation: NAT router must:

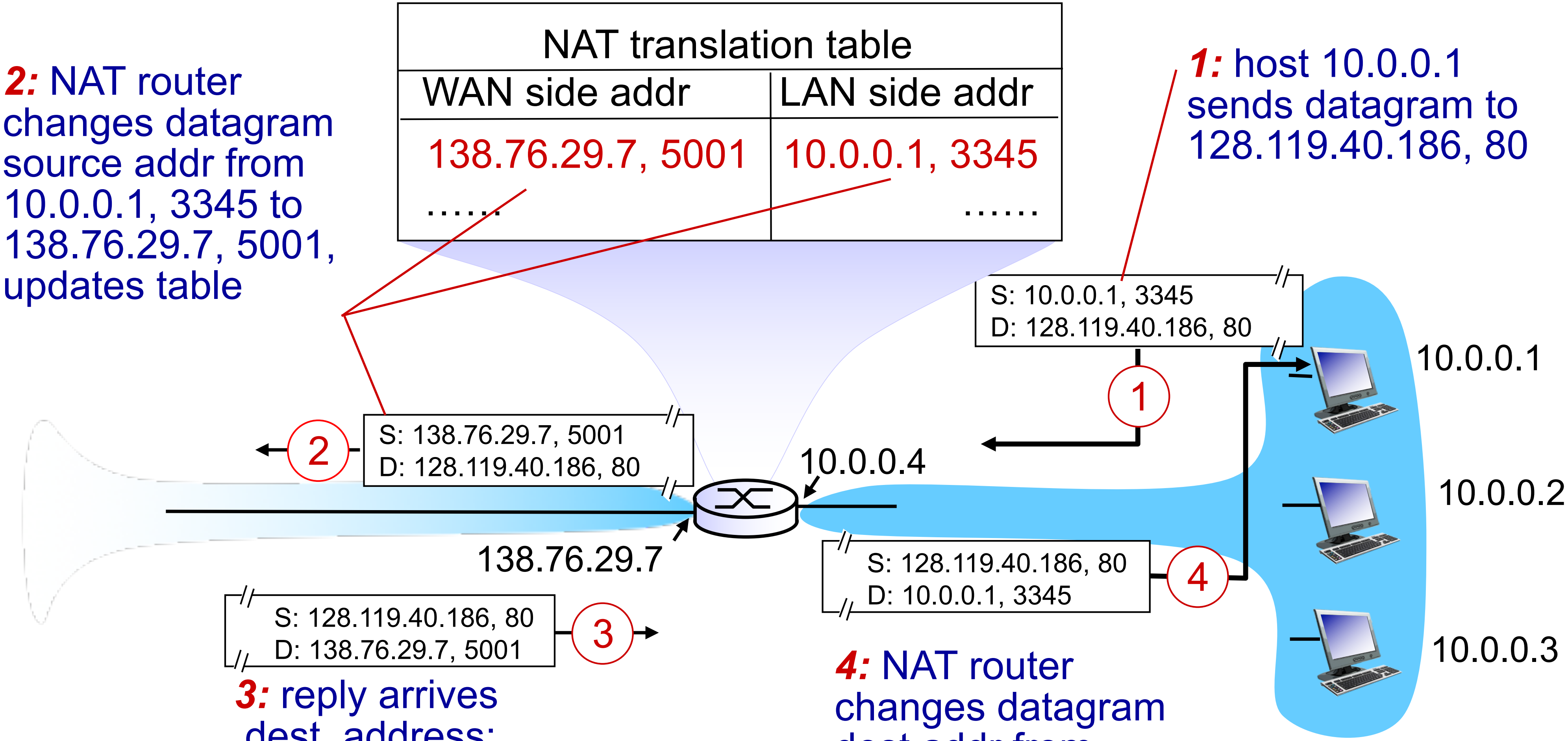
- *outgoing datagrams: replace* (source IP address, port #) of every outgoing datagram to (NAT IP address, new port #)

  . . . remote clients/servers will respond using (NAT IP address, new port #) as destination addr

- *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 dest fields of every incoming datagram with corresponding (source IP address, port #) stored in NAT table

NAT: network address translation



NAT translation table

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

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

1: host 10.0.0.1 sends datagram to 128.119.40.186, 80

S: 10.0.0.1, 3345
D: 128.119.40.186, 80

S: 138.76.29.7, 5001
D: 128.119.40.186, 80

S: 128.119.40.186, 80
D: 138.76.29.7, 5001

S: 128.119.40.186, 80
D: 10.0.0.1, 3345

3: reply arrives dest. address: 138.76.29.7, 5001

4: NAT router changes datagram dest addr from 138.76.29.7, 5001 to 10.0.0.1, 3345

NAT Advantages

Local network uses just one IP address as far as outside world is concerned:

- range of addresses not needed from ISP: just one IP address for all devices
- can change addresses of devices in local network without notifying outside world
- can change ISP without changing addresses of devices in local network

NAT Disdvantages

- NAT violates the architectural model of IP
  - Every IP address uniquely identifies a single node on Internet
  - routers should only process up to layer 3
- NAT changes the Internet from connection less to a kind of connection oriented network

Discussion: NAT

- Devices inside the local network are not explicitly addressable or visible by outside world.

A: This is an advantage

B: This is a disadvantage

NAT: network address translation

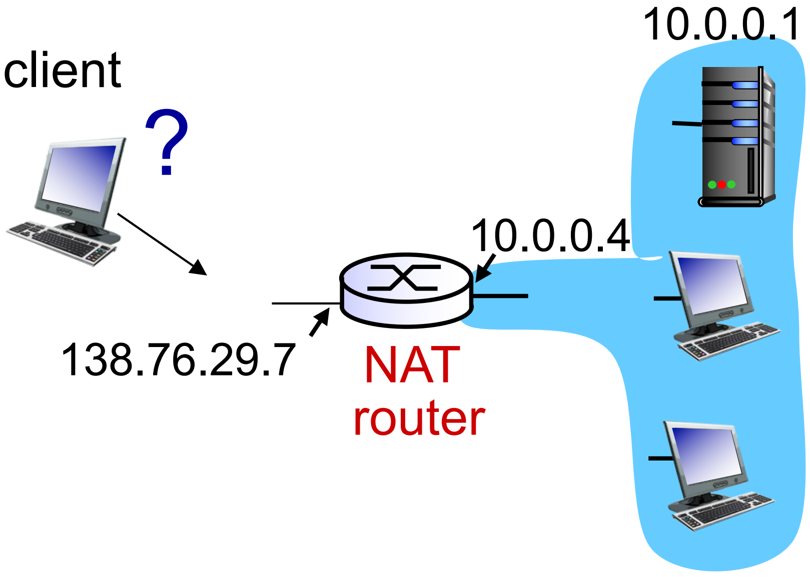
- 16-bit port-number field:
  - ~65,000 simultaneous connections with a single WAN-side address!
- NAT is controversial:
  - routers should only process up to layer 3
  - violates end-to-end argument
    - NAT possibility must be taken into account by app designers, e.g., P2P applications
  - address shortage should instead be solved by IPv6

NAT: Practical Issues

- NAT modifies port # and IP address
  - *Requires recalculation of TCP and IP checksum*
- Some applications embed IP address or port numbers in their message payloads
  - DNS, FTP (PORT command), SIP, H.323
  - For legacy protocols, NAT must look into these packets and translate the embedded IP addresses/port numbers
  - Duh, What if these fields are encrypted ?? (SSL/TLS, IPSEC, etc)
  - **Q: In some cases why may NAT need to change TCP sequence number??**
- If applications change port numbers periodically, the NAT must be aware of this
- NAT Traversal Problems
  - E.g: How to setup a server behind a NAT router?
  - How to talk to a Skype user behind a NAT router?

NAT traversal problem

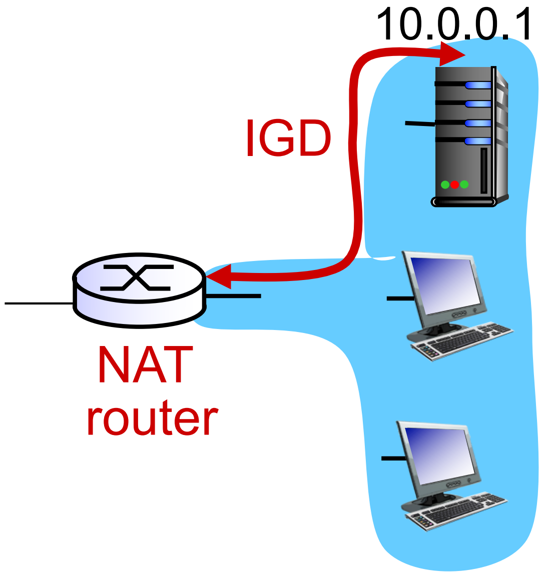
- client wants to connect to server with address 10.0.0.1
  - server address 10.0.0.1 local to LAN (client can't use it as destination addr)
  - only one externally visible NATed address: 138.76.29.7
- *solution1:* Inbound-NAT. Statically configure NAT to forward incoming connection requests at given port to server
  - e.g., (138.76.29.7, port 2500) always forwarded to 10.0.0.1 port 25000



NAT traversal problem

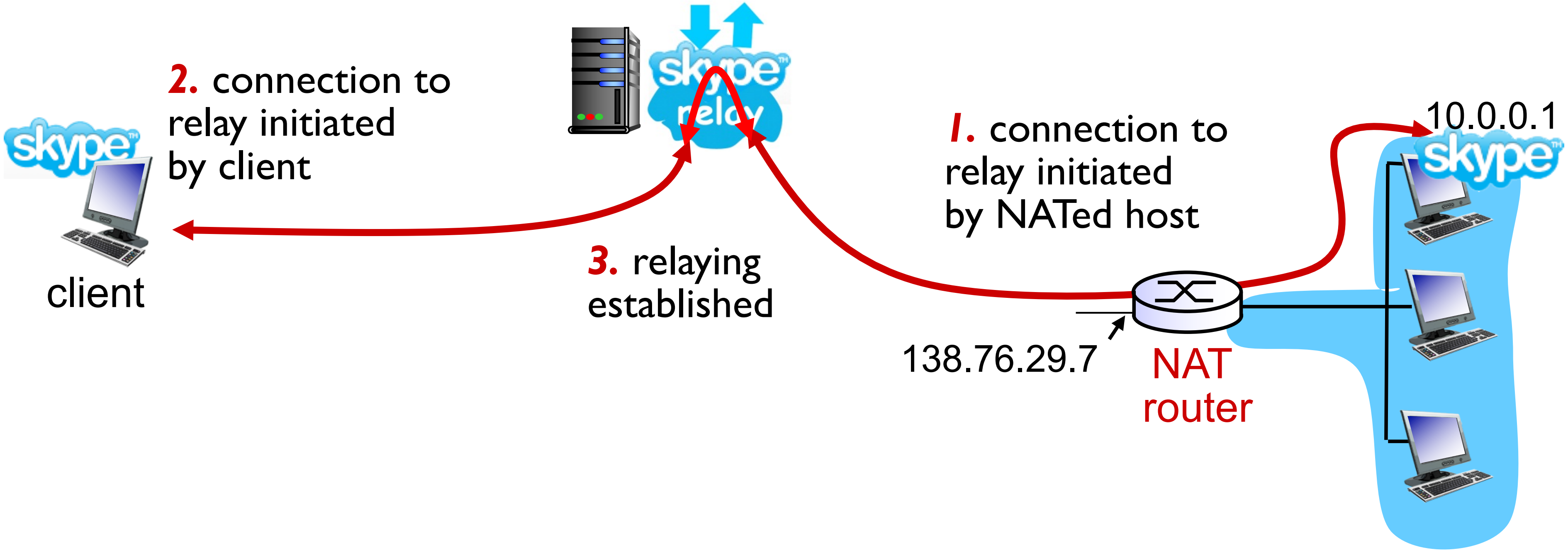
- *solution 2:* Universal Plug and Play (UPnP) Internet Gateway Device (IGD) Protocol. Allows NATed host to:
  - learn public IP address (138.76.29.7)
  - add/remove port mappings (with lease times)

  i.e., automate static NAT port map configuration



NAT traversal problem

- *solution 3:* relaying (used in Skype)
  - NATed client establishes connection to relay
  - external client connects to relay
  - relay bridges packets between to connections



2. connection to relay initiated by client

1. connection to relay initiated by NATed host

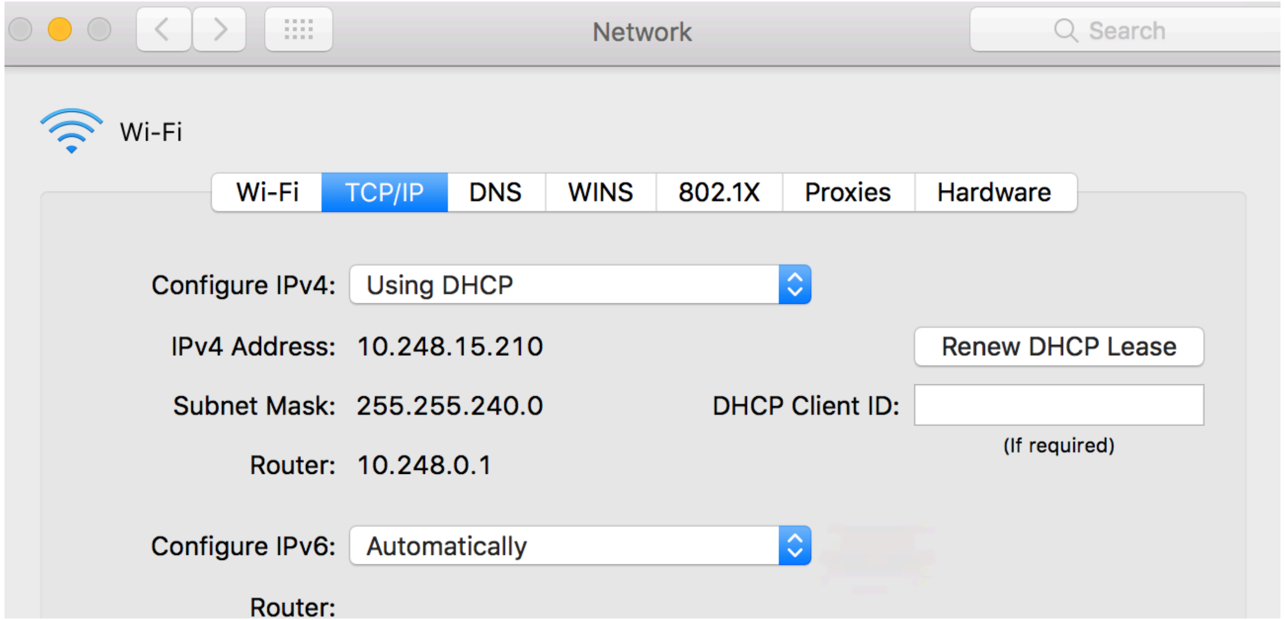
3. relaying established

NAT: Devil in the details

- Despite the problems, NAT has been widely deployed
- Most protocols can be successfully passed through a NAT, including VPN
- Modern hardware can easily perform NAT functions at > 100 Mbps
- IPv6 is still not widely deployed commercially, so the need for NAT is real
- After years of refusing to work on NAT, the IETF has been developing "NAT control protocols" for hosts
- Lot of practical variations
  - Full-cone NAT, Restricted Cone NAT, Port Restricted Cone NAT, Symmetric NAT, …..
    - The devil is in the detail

Discussion

- The picture below shows you the IP address of my machine connected to the uniwide wireless network.



Configure IPv4: Using DHCP

IPv4 Address: 10.248.15.210

Subnet Mask: 255.255.240.0

DHCP Client ID:

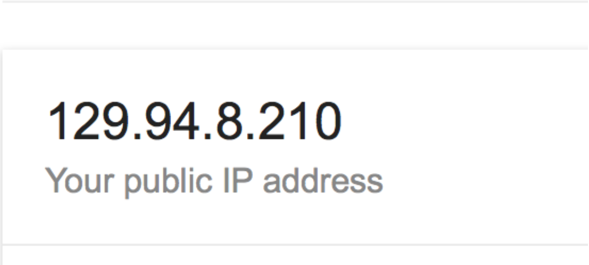
(If required)

Router: 10.248.0.1

Configure IPv6: Automatically

Router:

- However when I ask Google it says my IP address is as noted below. Can you explain the discrepancy?



129.94.8.210

Your public IP address

Network Layer, data plane: outline

4.1 Overview of Network layer

- data plane
- control plane

4.2 What's inside a router

4.3 IP: Internet Protocol

- datagram format
- fragmentation
- IPv4 addressing
- network address translation

4.4 Generalized Forward and SDN

(Not Covered)