

Computer Networks

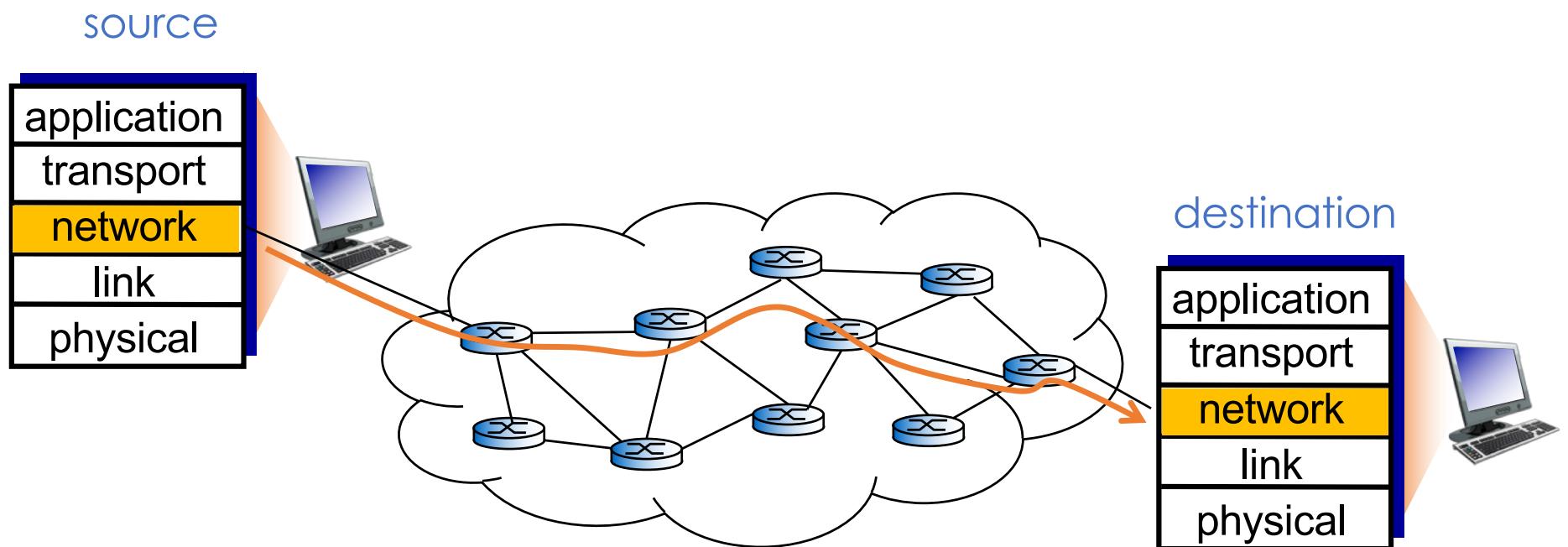
@CS.NCTU

Lecture 4: Network Layer: Data Plane

Instructor: Kate Ching-Ju Lin (林靖茹)

Slides modified from
“Computer Networking: A Top-Down Approach” 7th Edition

from end system to another end system: **routing!**



Outline

- Overview of network layer
- What's inside a router
- IP: Internet Protocol
 - Datagram format
 - IPv4 addressing
 - DHCP
 - Network address translation (NAT)
 - IPv6
- Software defined networking

Two Components

- **Data plane**
 - Per-router function: forwarding
 - For every packet, what should be the output link?
 - Local view
- **Control plane**
 - Network-wise logic: routing
 - traditional routing algorithms: implemented in routers
 - software-defined networking (SDN): implemented in remote servers (controllers)
 - What is the exact path from a source to a destination?
 - Global view

Forwarding vs. Routing

Turn right? Turn left?
(forwarding)

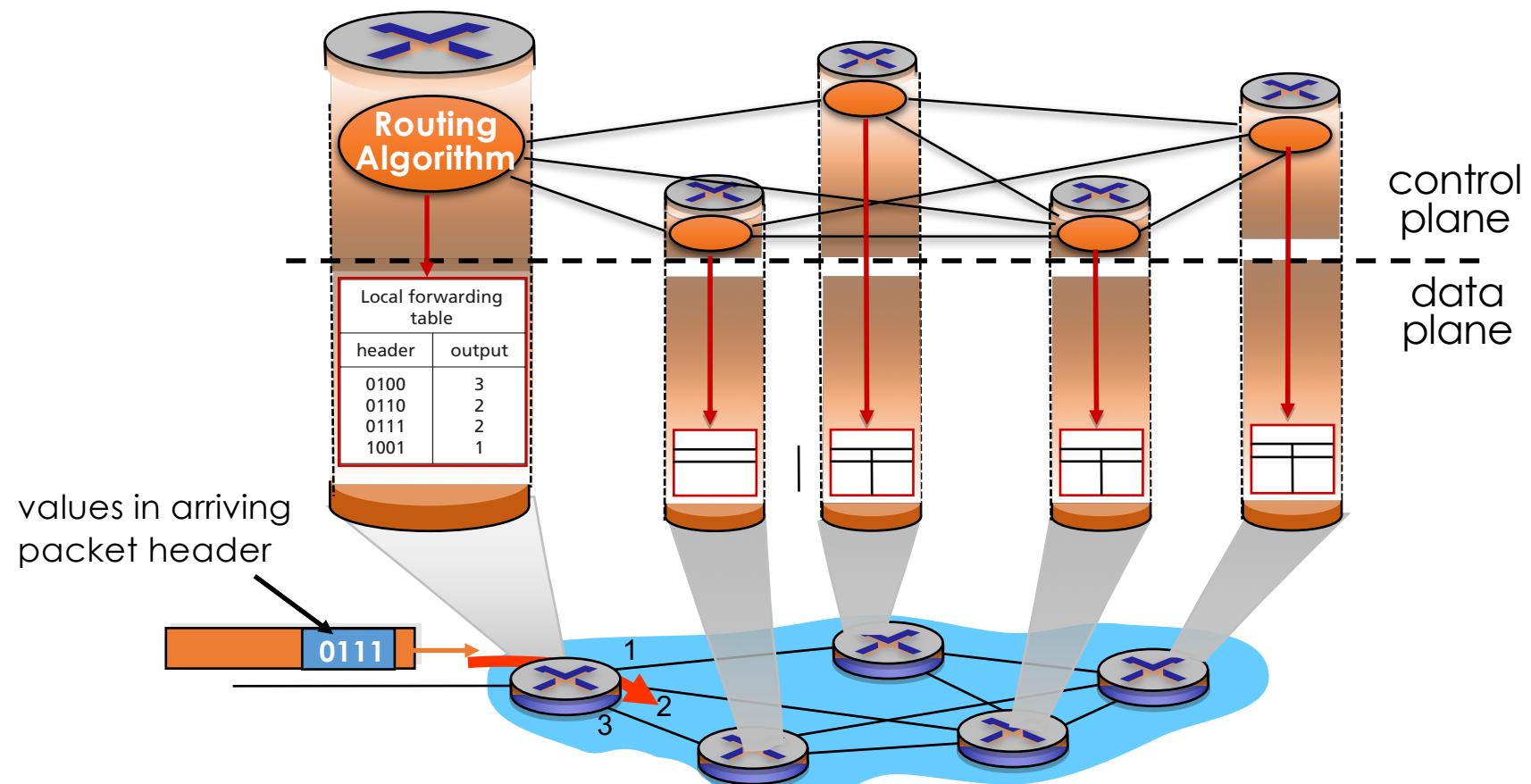


Types of Control Plane

- **Per-router control plane**
 - Controller within each router
- **Centralized control plane**
 - Remote server as a controller, controlling all the routers

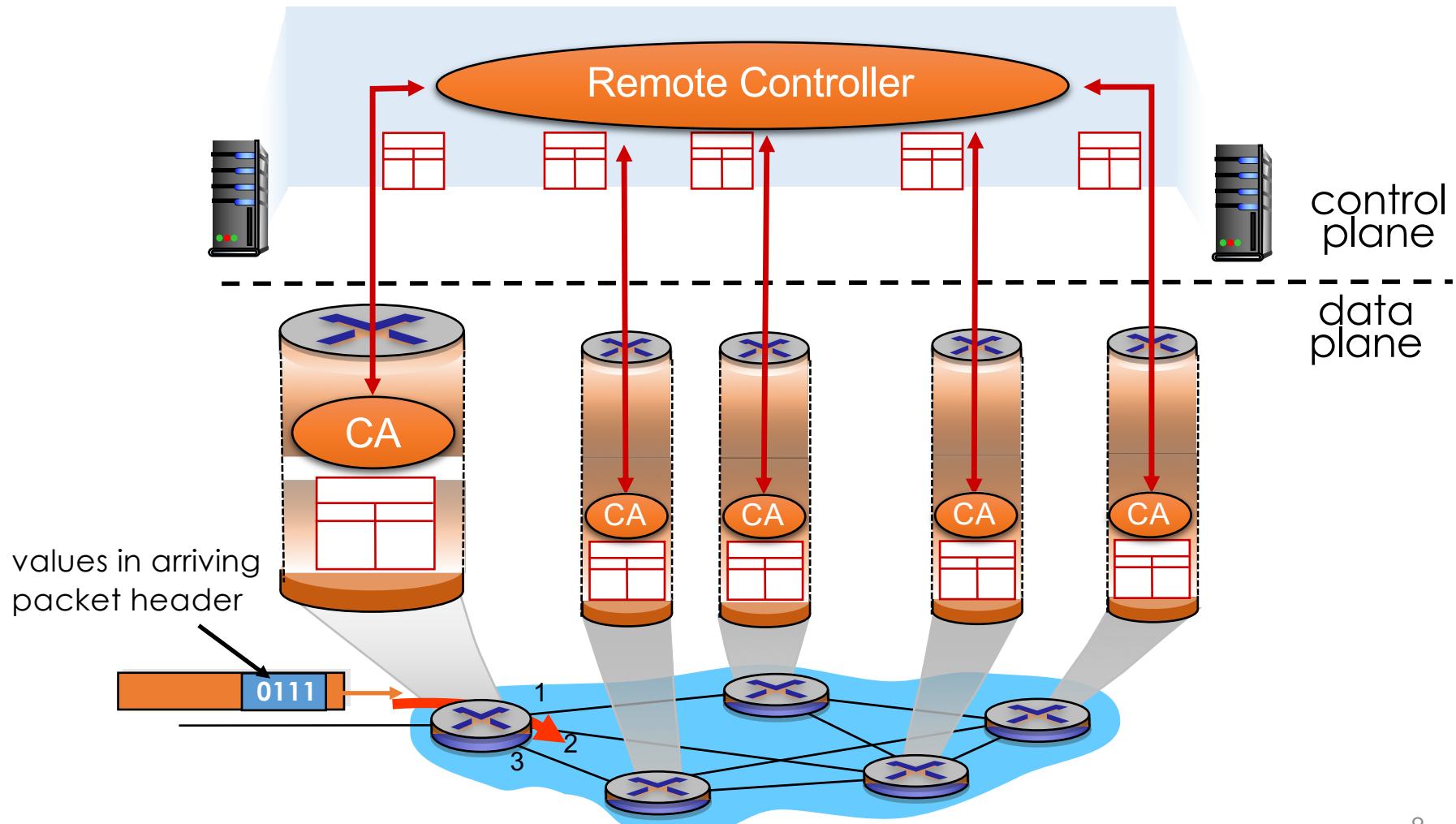
Per-Router Control Plane

- Individual forwarding components in every router interact in the control plane



Centralized Control Plane

- A distinct (typically remote) controller interacts with local control agents (CAs)



Network Service Model

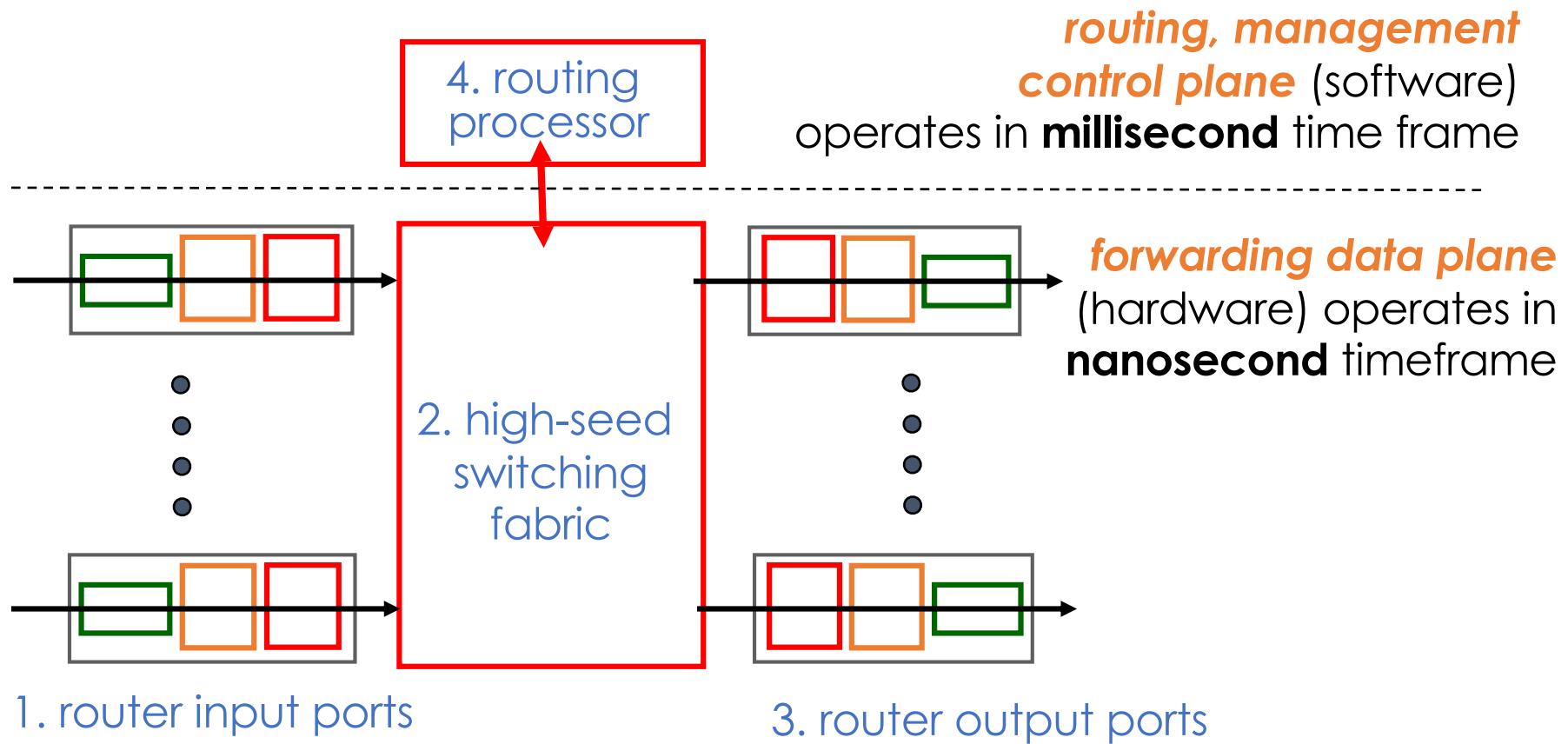
- What are the services required by end-to-end delivery
 - Guaranteed delivery?
 - Bounded delay?
 - In-order packet delivery?
 - Guaranteed minimal bandwidth?
 - Security?
- Types of network protocols
 - **Best-effort**
 - neither delay nor bandwidth is guaranteed
 - **IP (Internet protocol)**
 - **Guaranteed**
 - in-order, bounded delay, min-bandwidth
 - **ATM (Asynchronous Transfer Mode)**

Outline

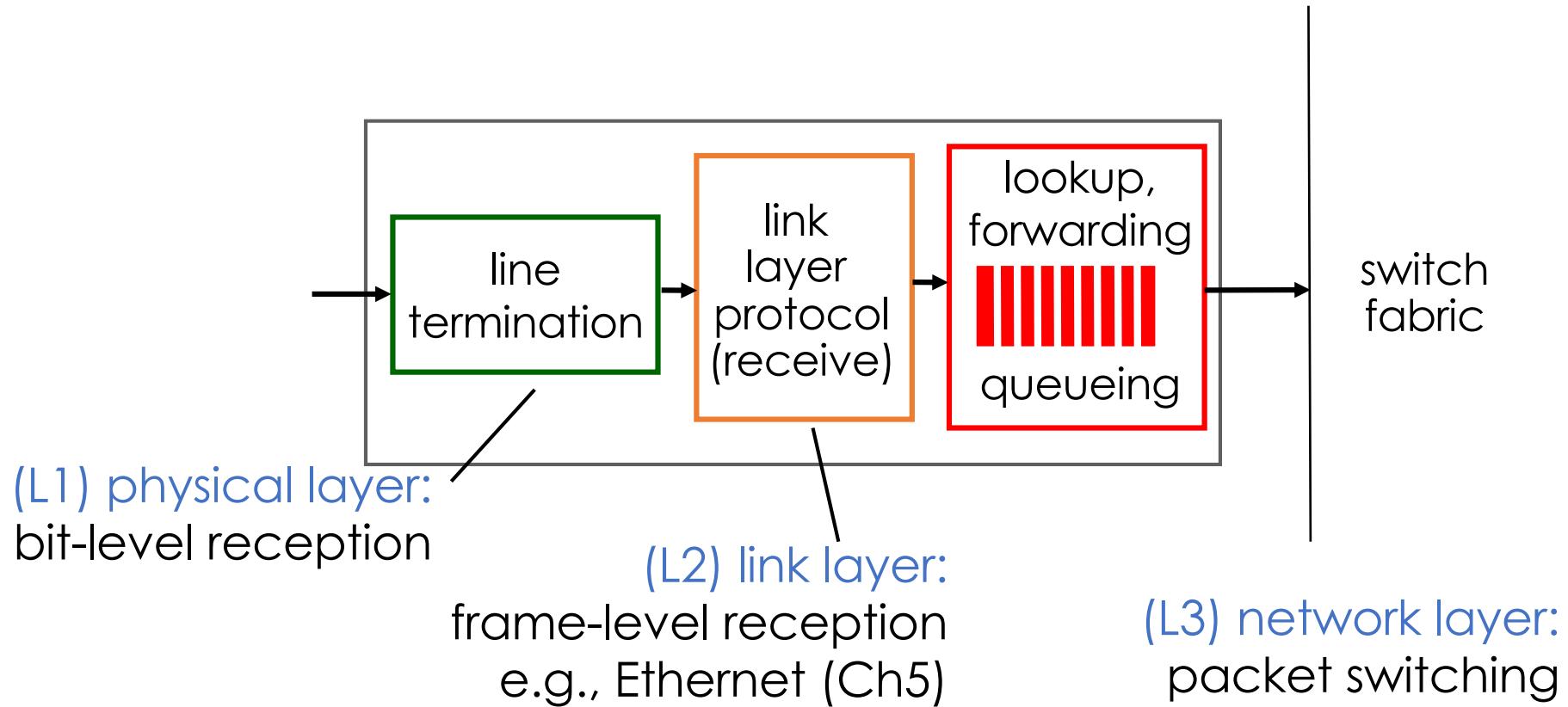
- Overview of network layer
- **What's inside a router**
- IP: Internet Protocol
 - Datagram format
 - IPv4 addressing
 - DHCP
 - Network address translation (NAT)
 - IPv6
- Software defined networking

Generic Router Architecture

- 4 components



1. Input Ports



Input Ports: Decentralized Switching

- Using header field values, lookup output port using forwarding table in input port memory (“match plus action”)
- Why challenging?
 - processing at “line speed”
 - queuing: if datagrams arrive faster than forwarding rate into switch fabric
- Perform **lookup** function
 - Forwarding table used to determine the output port of each arriving packet
 - Control packets forwarded to routing processor
 - Can be updated by routing processor or remote SDN controller

Destination-based Forwarding

forwarding table

Destination Address Range	Link Interface
11001000 00010111 00010000 00000000 through 11001000 00010111 00010111 11111111	0
11001000 00010111 00011000 00000000 Through 11001000 00010111 00011000 11111111	1
11001000 00010111 00011001 00000000 through 11001000 00010111 00011111 11111111	2
otherwise	3

Q: what happens if ranges don't divide up so nicely?

Q: what happens if multiple entries are matched?

Longest Prefix Matching

- When a destination address matches multiple entries, use the one with longest prefix matching
- Example
 - What is the output port of
11001000 00010111 00011000 10101010
 - Port 1 or 2?

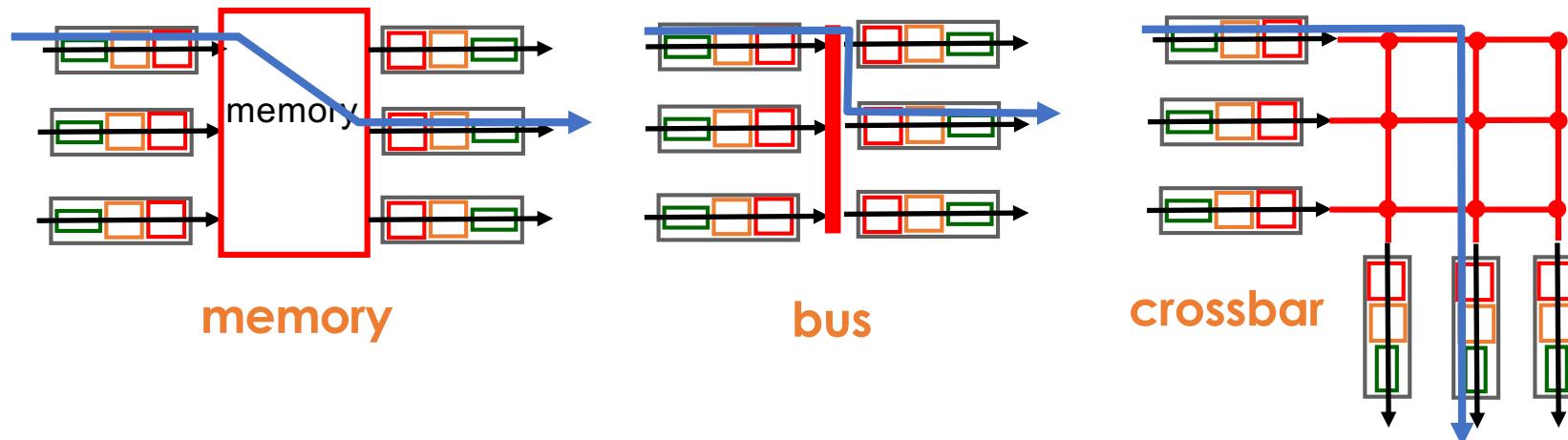
Destination Address Range	Link interface
11001000 00010111 00010*** *****	0
11001000 00010111 00011000 *****	1
11001000 00010111 00011*** *****	2
otherwise	3

Longest Prefix Matching

- The entire table should be searched!
 - Time consuming
 - Packet arrival rate can be Gpbs
- Lookup principles
 - Performed in hardware
 - Simple linear search
 - Short memory access time (e.g., done in **Ternary Content Addressable Memories, TCAM**)
 - Expensive and power consuming
 - Limited capacity: ~1M routing table entries

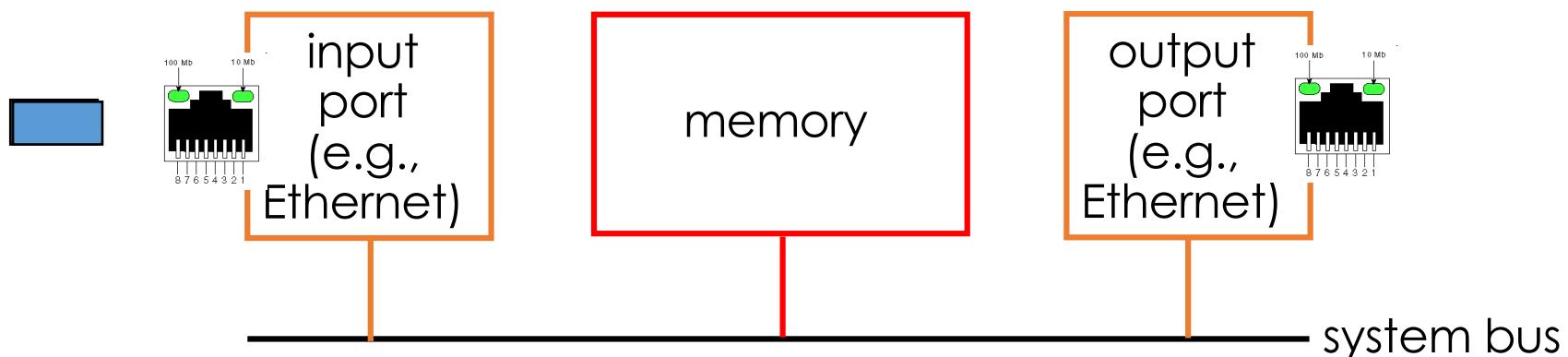
2. Switching Fabrics

- Where packets are actually switched!
 - Transfer a packet from an input port to an output port
- **Switching rate:**
 - Total number of packets transferred per second
 - Switching rate vs. line rate?
 - Some packets might be dropped if switching rate is not high enough to support the line rate
 - Three types:



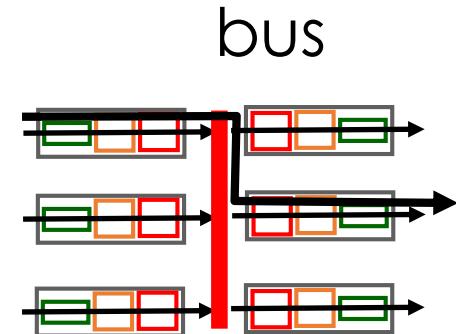
2.1 Switch via Memory

- Simplest, first generation
- Packets copied to processor memory
 - Processor looks up the table and determine the output port
- Memory bandwidth B
 - One memory read/write at a time
 - Forwarding throughput $\leq B/2$



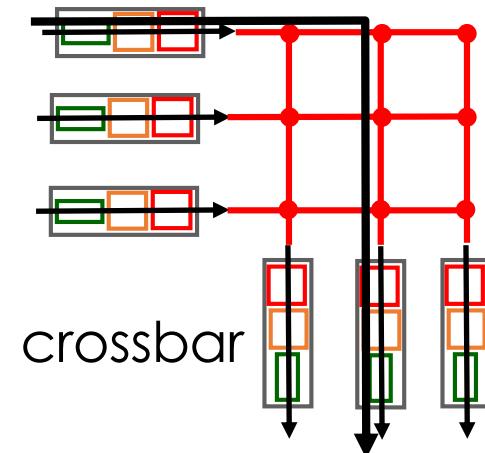
2.2 Switching via Bus

- Packets directly go through a shared bus
 - without intervention by the routing processor
- The input port tags a label to each packet, indicating the output port
 - All output ports receive the packet
 - The matching output port removes the label and forward
 - The others drop the packet
- **Bus contention:** switching speed limited by bus bandwidth
 - Cisco 6500: 32-Gbps bus

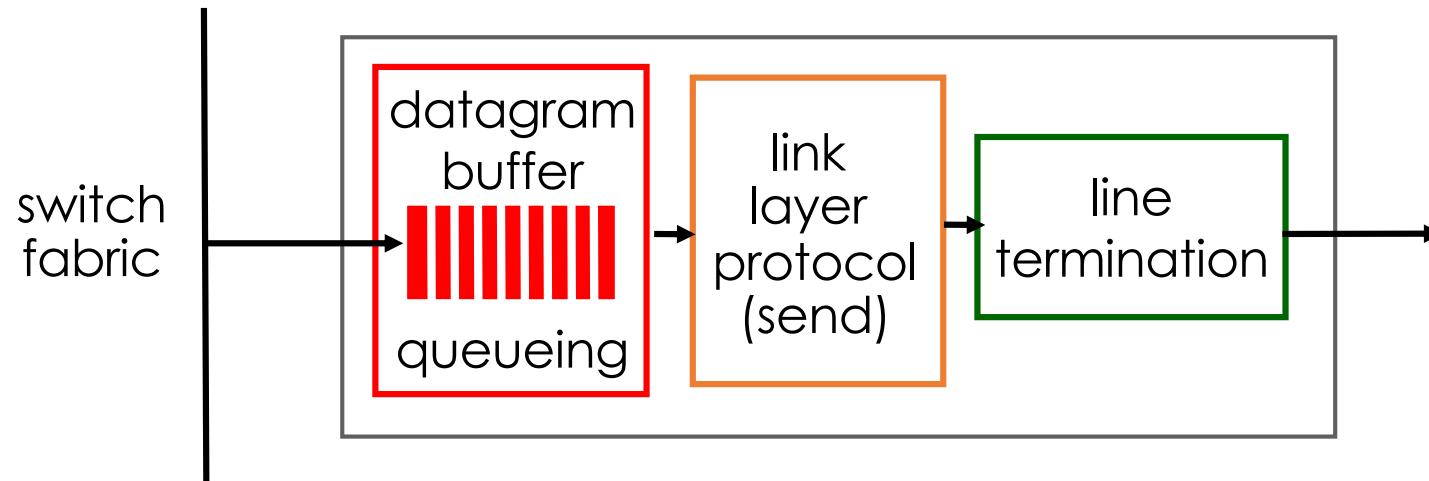


2.3 Switching via Crossbar

- Overcome the bus limitation
 - Cisco 12000: switches 60 Gbps through the interconnection network
- **2N** buses for N input/output ports
- **Non-blocking**
 - Can forward multiple packets **in parallel**
- Advanced version:
multi-staged switching fabric
 1. Packets from different input ports can be sent to the same output port
 2. An input port breaks a packet to K chunks, forwarded by multiple switching fabrics



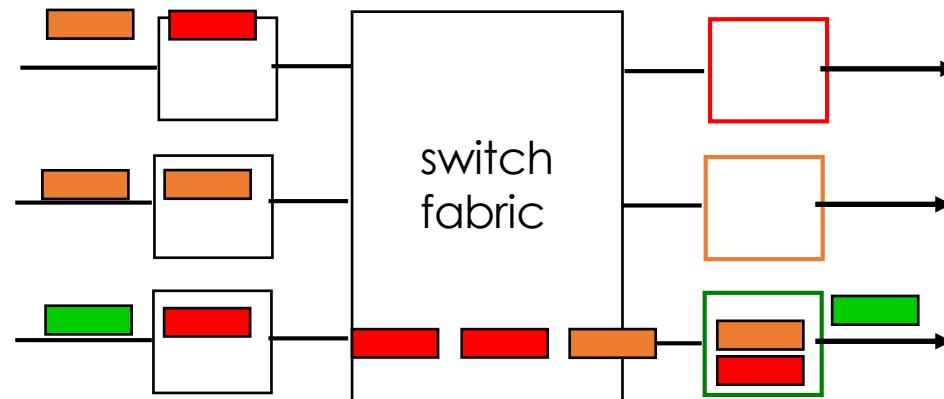
3. Output Port



- **Buffering** required when datagrams arrive from fabric faster than the transmission rate
 - Packet loss due to congestion (overflow)
- **Scheduling discipline** chooses among queued datagrams for transmission
 - Priority or not? (who goes first)

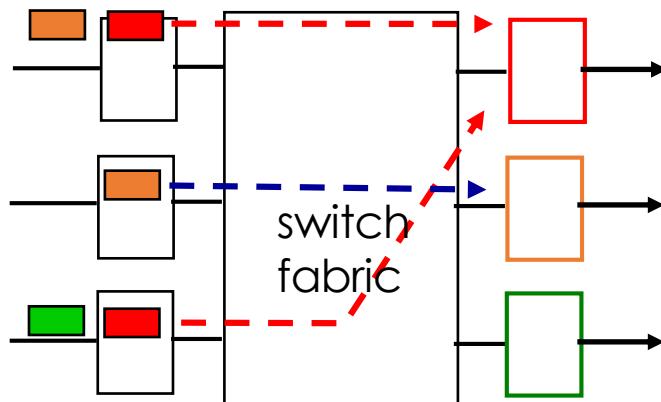
Packet Queueing

- **Input queueing**
 - Arrival rate > switch fabric speed
- **Output queueing**
 - Switch fabric speed > Forwarding rate
- Example:
 - $N * R_{line} = R_{switch}$
all input ports send to the same output port

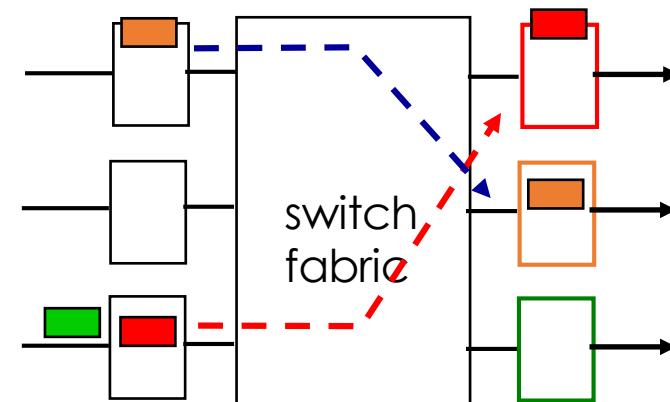


Input Port Queueing

- fabric slower than the aggregated input ports combined → buffer may overflow
- **Head-of-the-Line (HOL) blocking:**
 - queued datagram at front of queue prevents others in queue from moving forward



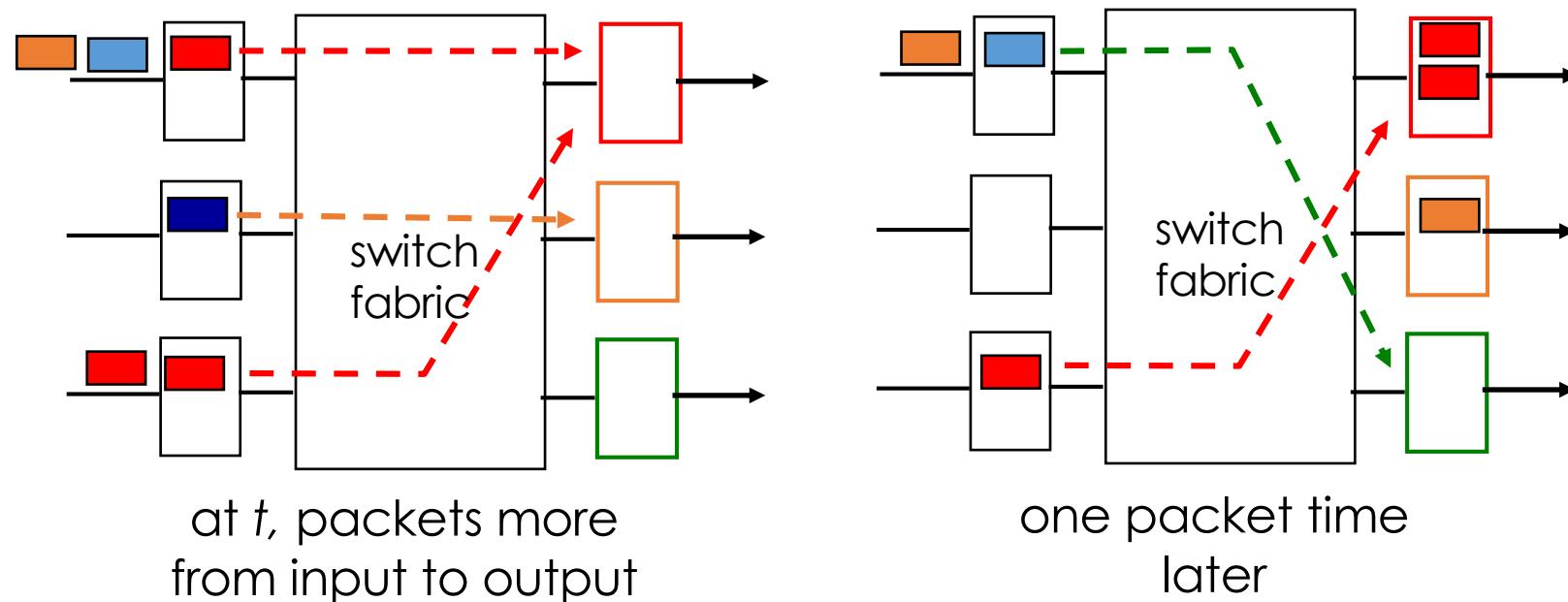
output port contention:
only one red datagram
can be transferred.
lower red packet is blocked



one packet time later:
green packet
experiences HOL blocking

Output Port Queueing

- buffering when arrival rate via switch exceeds output line speed
- queueing (delay) and loss due to output port buffer overflow!



How Much Buffering?

- RFC 3439 rule of thumb
 - average buffering = RTT x link capacity
 - e.g., RTT = 250ms, C = 10 Gbps
→ buffer = $10 * .25 = 2.5 \text{ Gb}$
- [Appenzeller 2004] When N TCP flows pass through a link (as N is large),

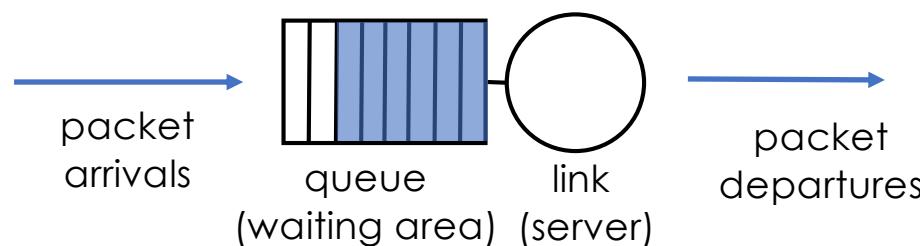
$$\text{buffering} = \frac{\text{RTT} \cdot \text{C}}{\sqrt{N}}$$

4. Packet Scheduling

- Determine the order of packet forwarding
- Four strategies
 - First-in-first-out (FIFO) or first-come-first-serve (FCFS)
 - Priority queue
 - Round robin
 - Weighted fair queueing (WFQ)

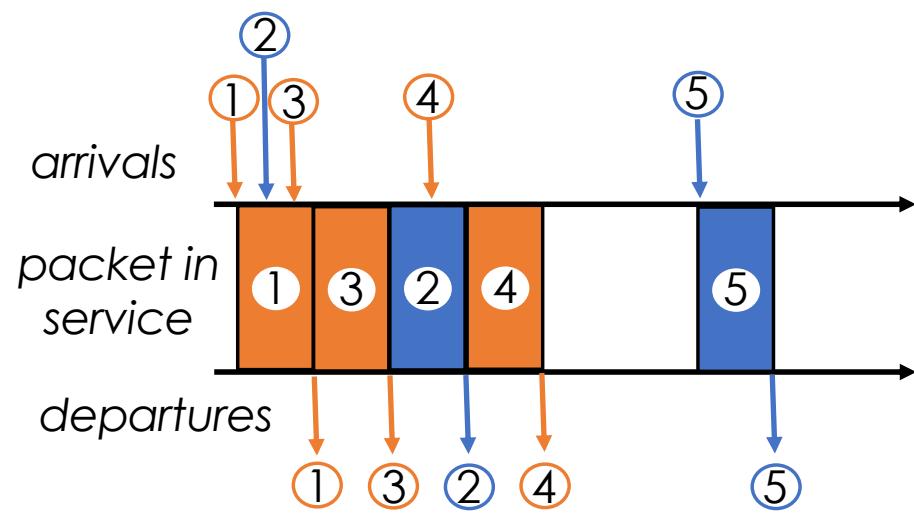
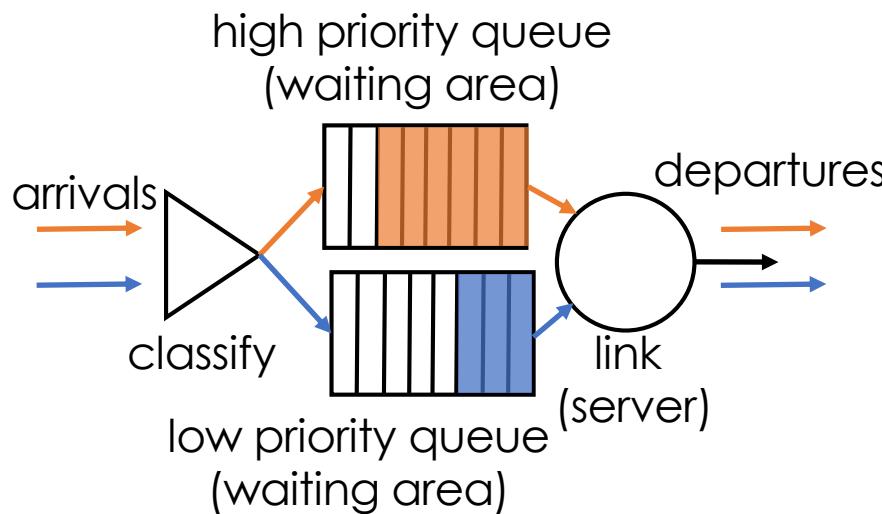
First In First Out

- Packets leave in the same order in which they arrived
- Packet discarding policy:
 - tail drop: drop arriving packet
 - priority: drop/remove on priority basis
 - random: drop/remove randomly



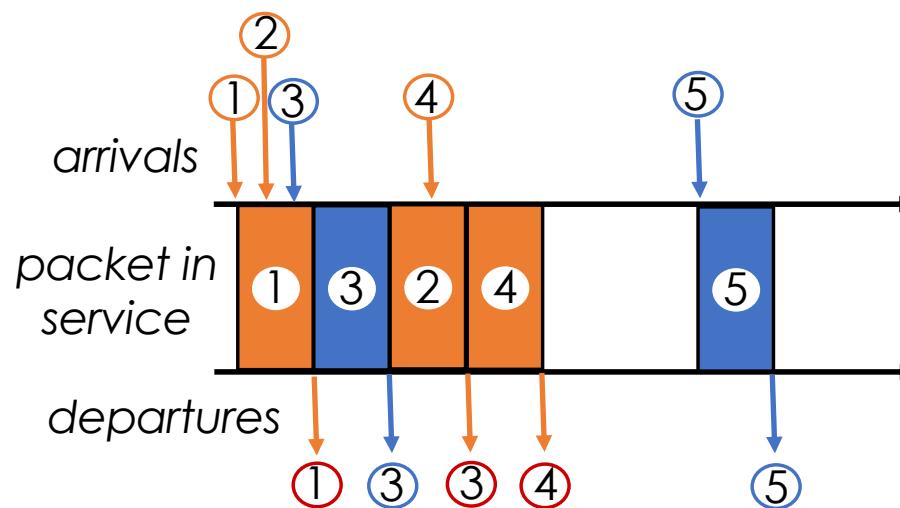
Priority Queueing

- Packets classified into multiple classes
 - Real-time: voice-over-IP
 - Best effort: SMTP, FTP
- Send highest priority queued packet
 - Intra-class: FIFO
 - Non-preemptive: no interruption



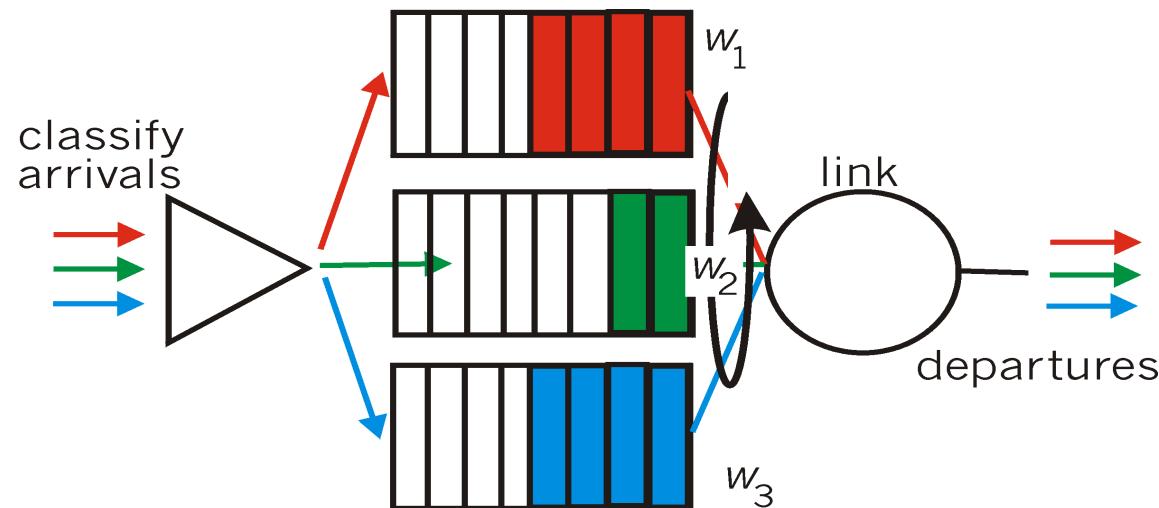
Round Robin

- Packets also classified into multiple classes
- Cyclically scan class queues
 - Send one packet for each class (*if possible*)
- Throughput of each class?
 - Determined by the arrival rate of each class



Weighted Fair Queueing (WFQ)

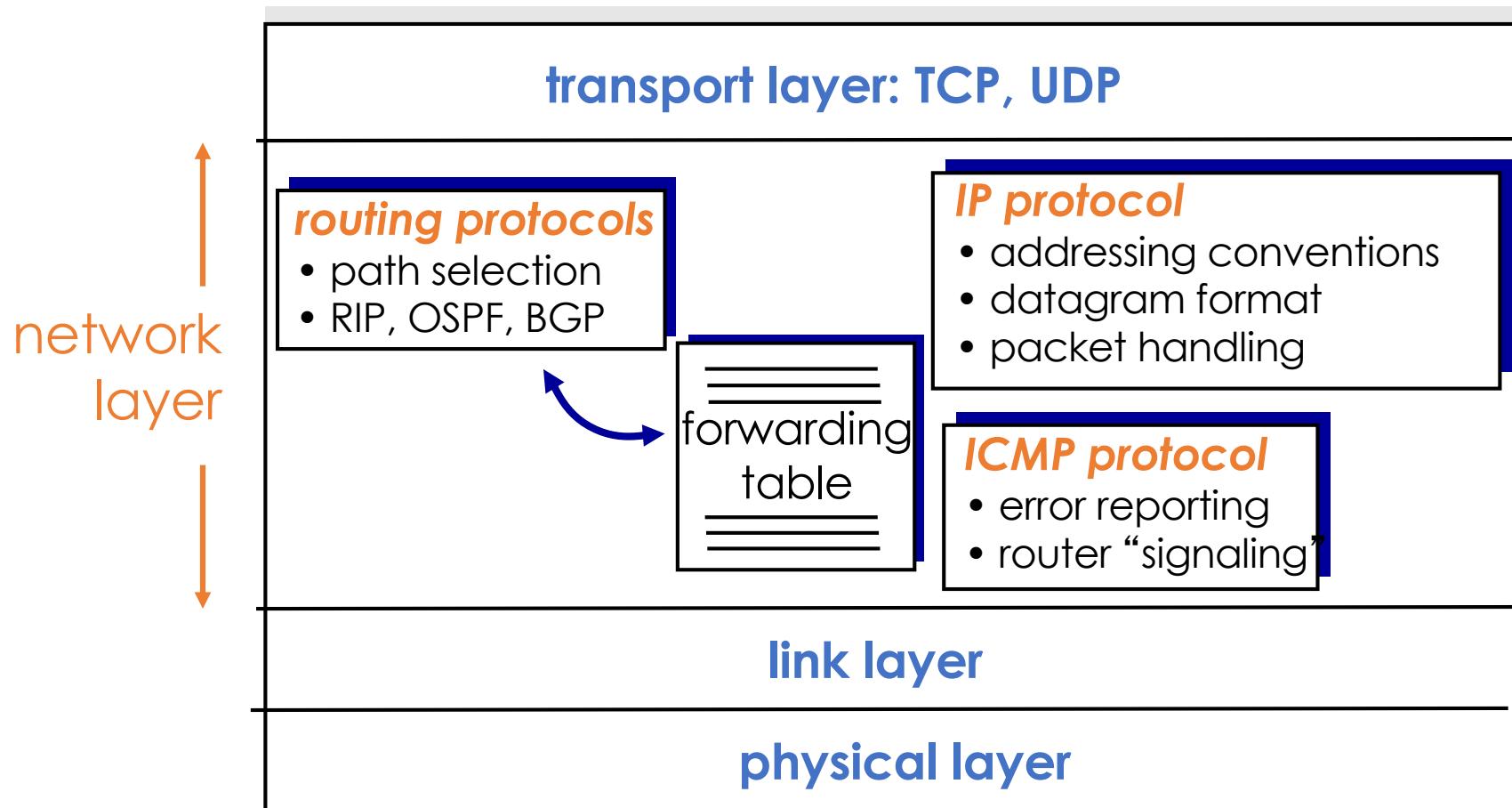
- Achieved by leveraging round robin queueing
- Each class gets weighted amount of service in each cycle
- How?
 - Control the arrival rate of each class
 - Specifically, the interval of pushing a packet into a class queue is controlled



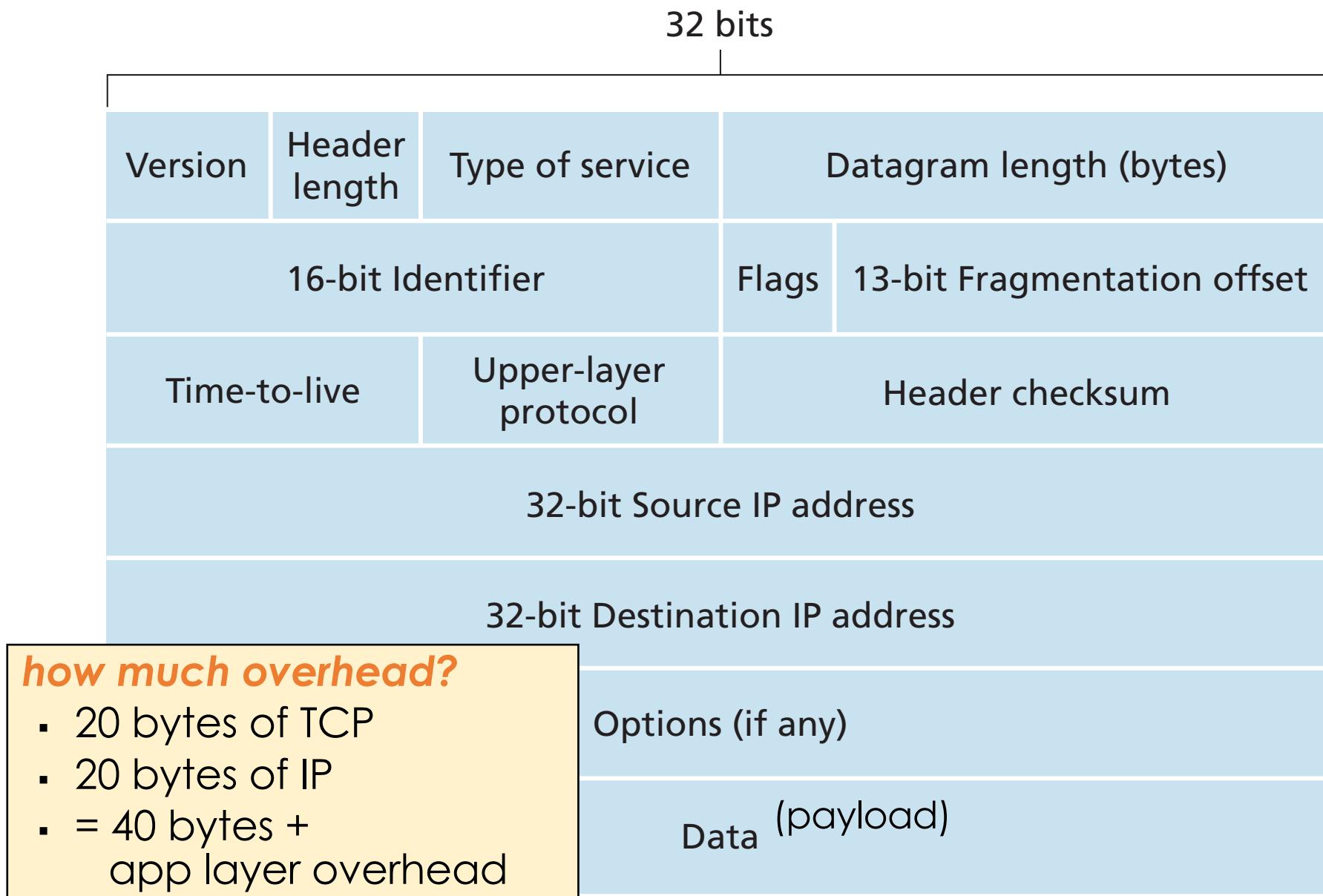
Outline

- Overview of network layer
- What's inside a router
- **IP: Internet Protocol**
 - Datagram format
 - IPv4 addressing
 - DHCP
 - Network address translation (NAT)
 - IPv6
- Software defined networking

Internet Network Layer



IP Datagram Format

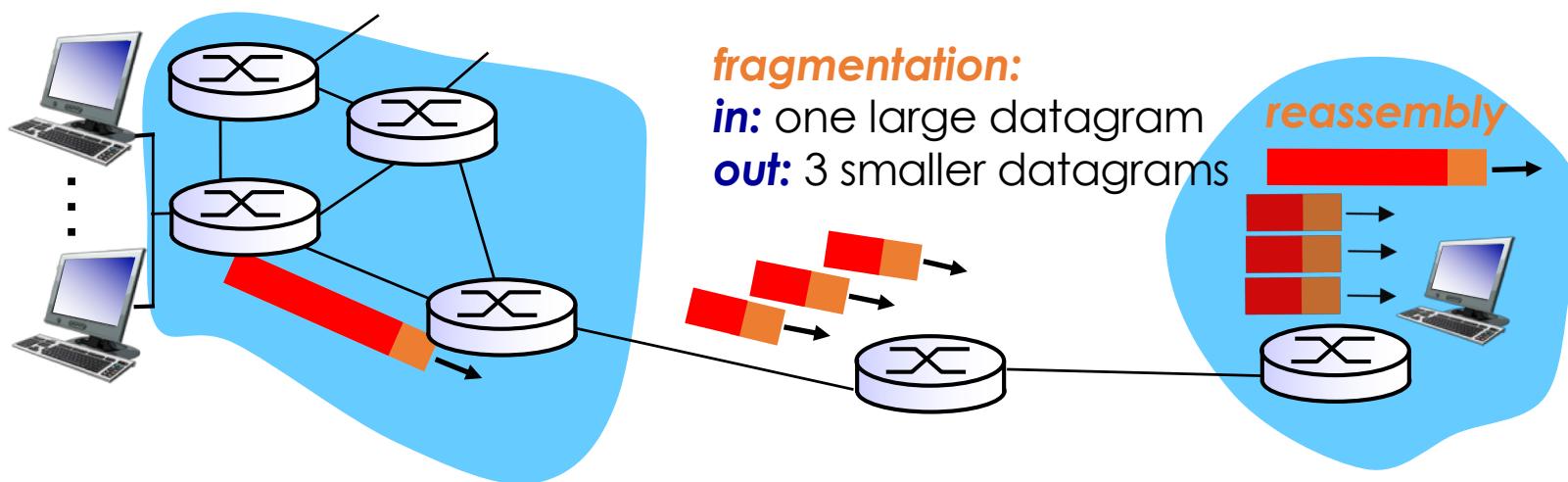


IP Datagram Format

- L3 packet is referred to as a **datagram**
- **Version num**: IPv4 or IPv6
- **Header length**: where the payload begins (typically 20byte)
- **Type of service**: real-time or non-real-time
- **Datagram length**: total length of IP (header + data)
- **Identifier, flag, fragmentation offset**: index for fragments (IPv6 does not support fragmentation)
- **Time-to-live**: TTL, maximum number of hop counts
- **Protocol** : transport-layer protocol, e.g., TCP, UDP
- **Header checksum**: for error detection
 - Recalculated by each router since TTL is updated
 - Why checksum again? TCP/UDP and IP may not used at the same time

IP Fragmentation, Reassembly

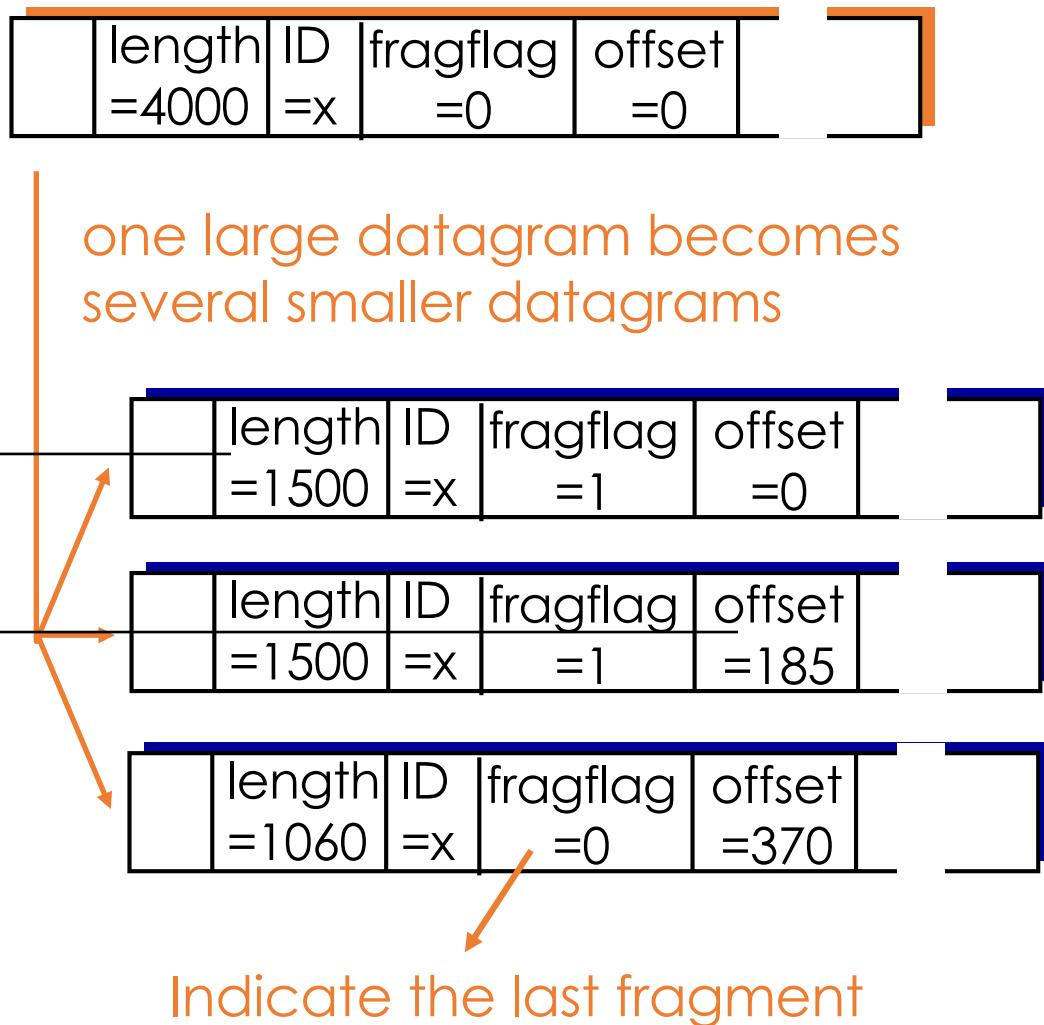
- Each link has (link-layer) **MTU** (maximum transmission unit)
- Large IP datagram divided (“fragmented”)
 - one datagram becomes several datagrams
 - “reassembled” only at final destination
 - IP header bits used to identify, order related fragments



IP Fragmentation, Reassembly

example:

- 4000 byte datagram
- MTU = 1500 bytes



Outline

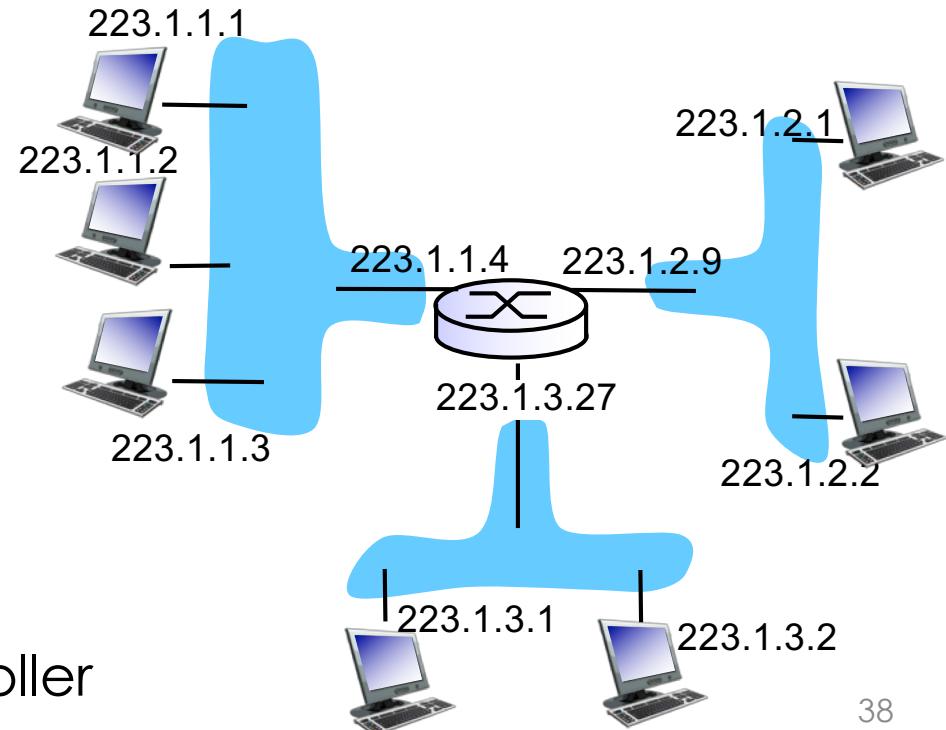
- Overview of network layer
- What's inside a router
- **IP: Internet Protocol**
 - Datagram format
 - **IPv4 addressing**
 - Network address translation (NAT)
 - IPv6
- Software defined networking

IPv4 Addressing

- **32-bit identifier**
 - Dotted-decimal notation: 140.113.94.87
 - Binary: 10001100 01110001 01011110 01010111
- Each “**interface**” needs an IP
 - Interface: connection between a host and a link
 - Try `ifconfig` (Ethernet, WLAN)
 - A router typically has multiple interfaces (IPs)



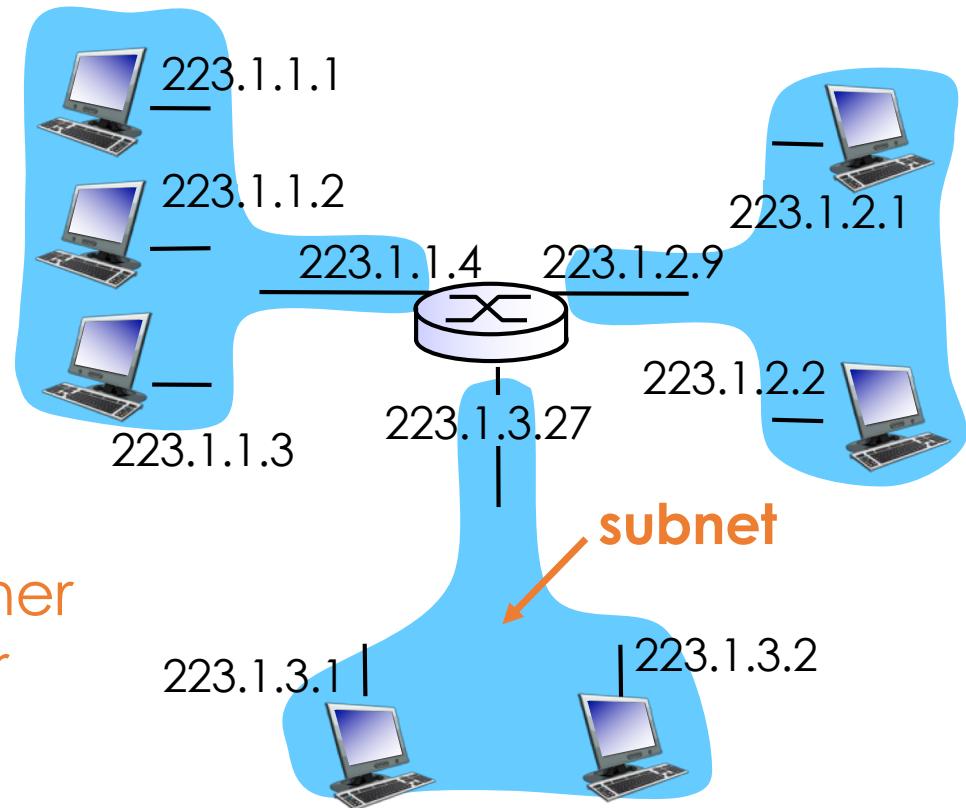
NIC: network interface controller



Subnet

- IP address:
 - Subnet part: prefix
 - Host part: low order bits
- What is a subnet?
 - Interfaces with the same subnet part of IP address
 - Physically reach each other without intervening router
- **Subnet mask**
 - Bitmask used to get the subnet part
 - For 223.1.1.0/24: /24 (255.255.255.0) is the subnet mask

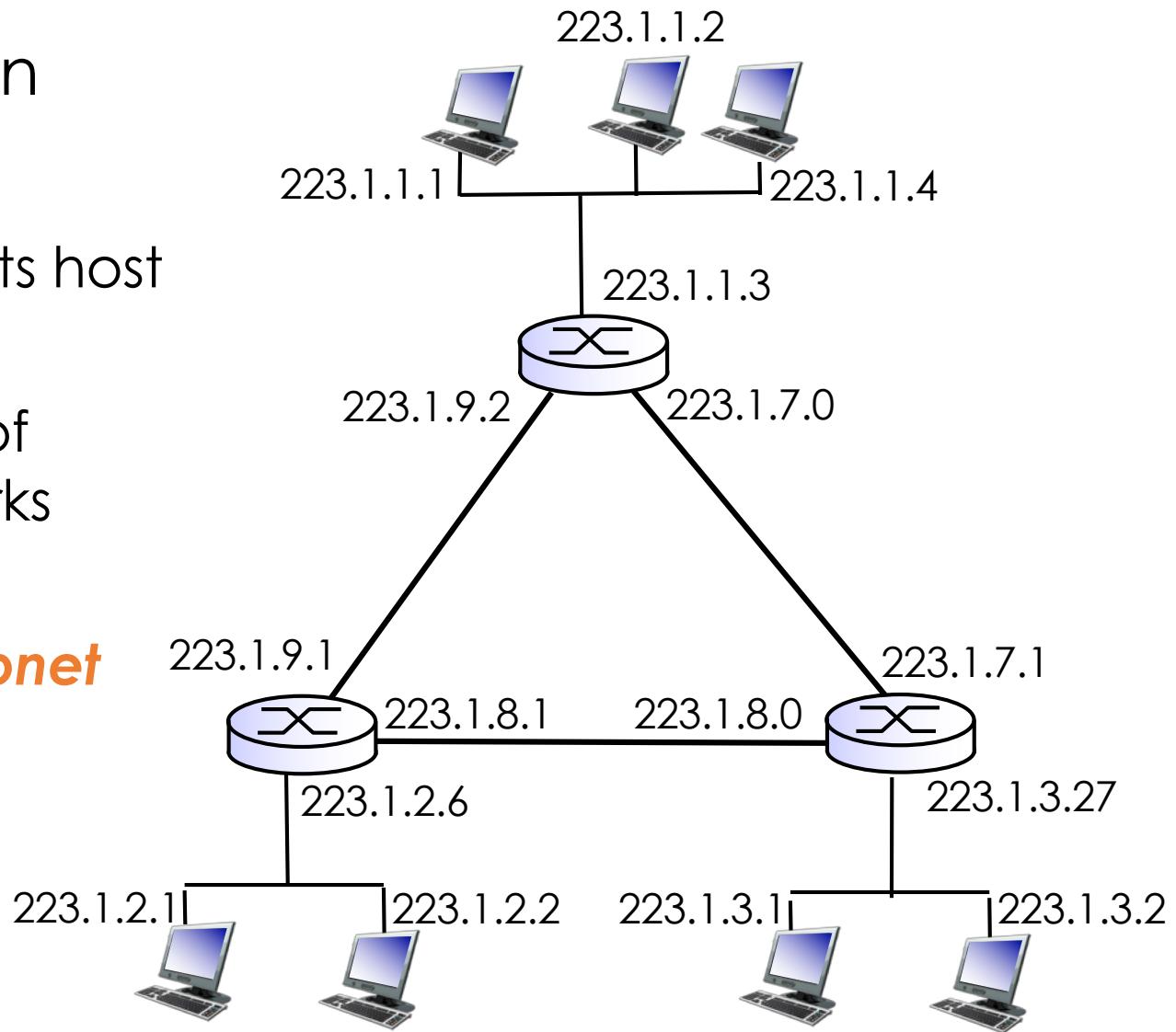
network consisting of 3 subnets



Subnet

- Formal definition

1. detach each interface from its host or router
2. create islands of isolated networks
3. each isolated network is a **subnet**



CIDR

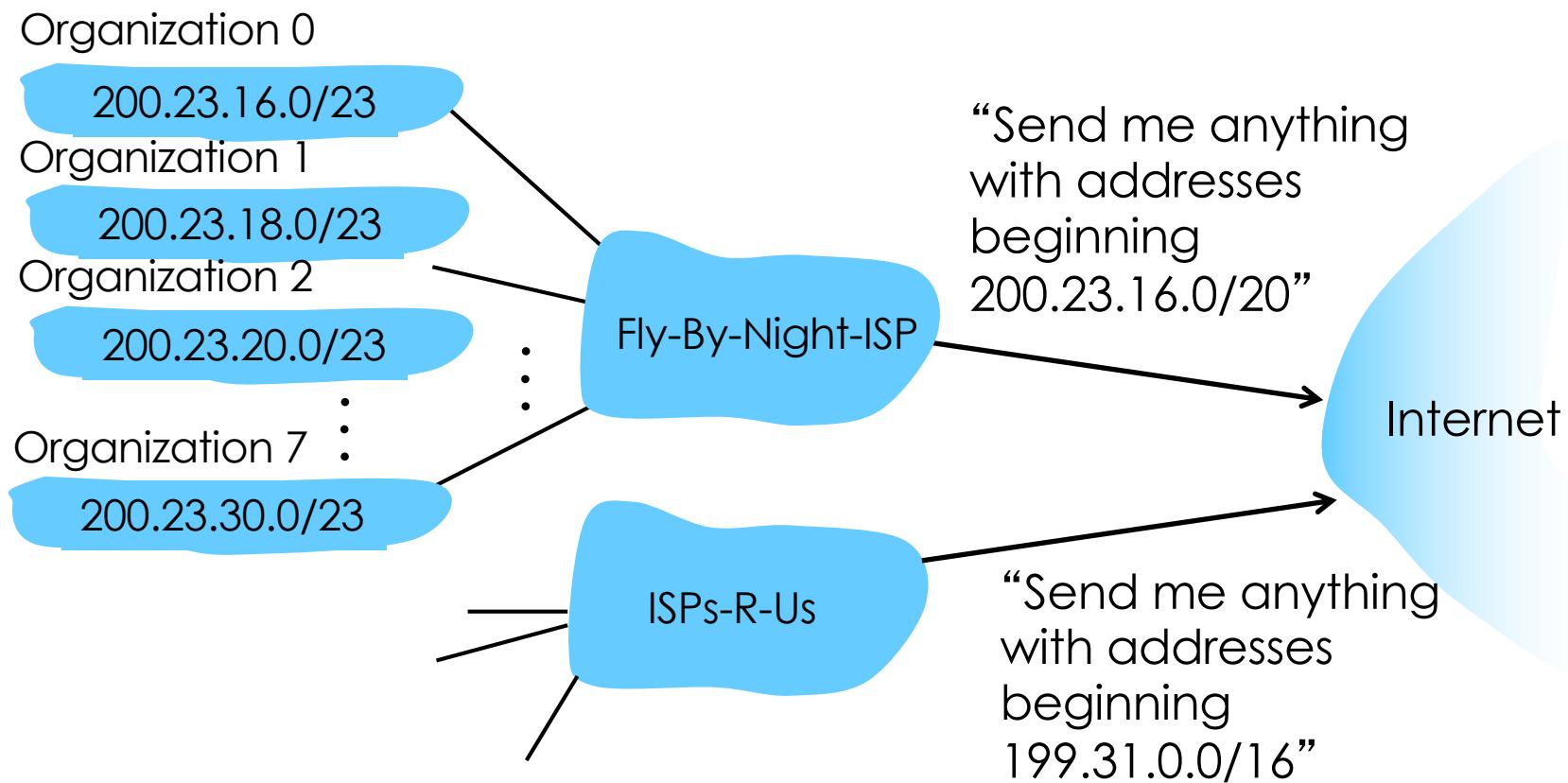
- **Classful addressing:** 8-, 16, 24-bit subnet
 - 24bit subnet: each only includes $2^8 - 2$ hosts
 - 16bit subnet: each includes 65,534 hosts
- **Classless Inter-Domain Routing**
 - Introduced by IETF in 1993
 - Goal: slow the growth of routing tables on routers
- **Variable-length subnet masking!**
 - Subnet part can be of arbitrary length
 - a.b.c.d/**x**, where **x** is # bits in subnet portion of address



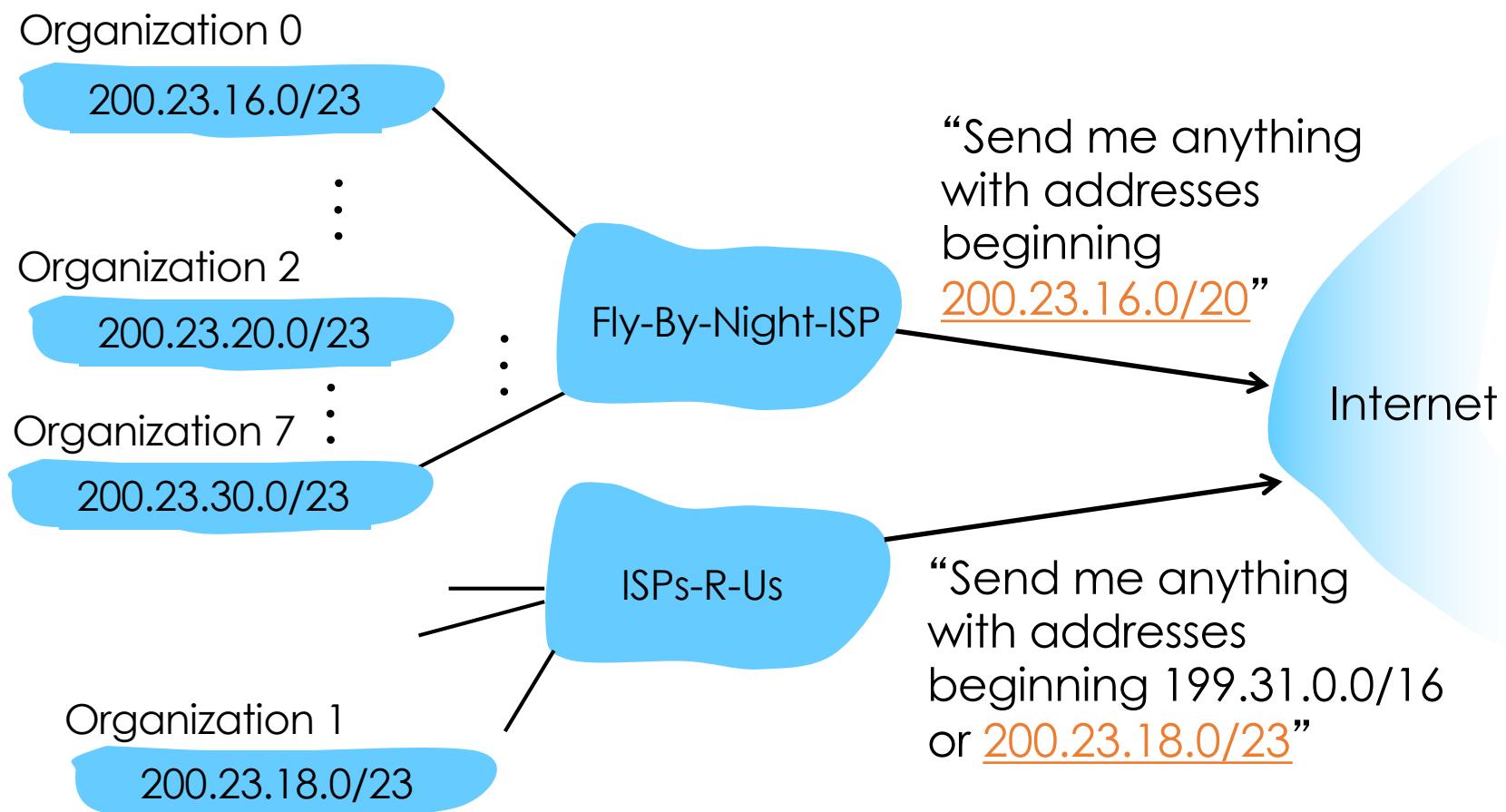
200.23.16.0/23

Hierarchical Addressing

- ISP assigns a block of IP addresses to a subnet



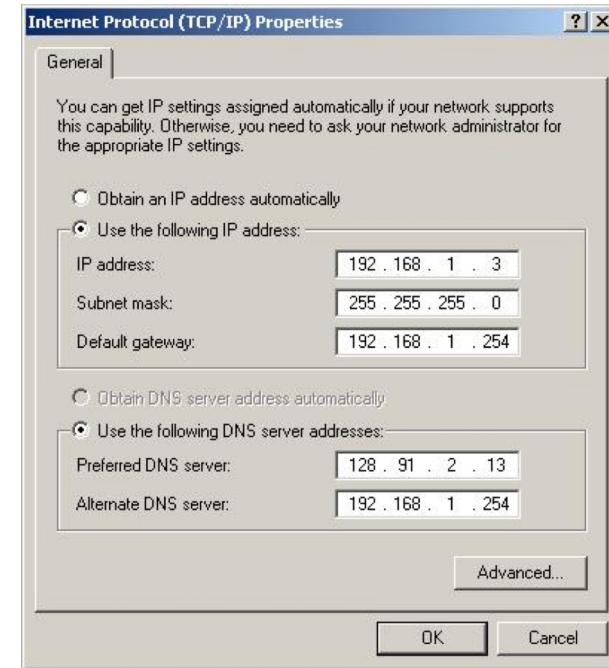
Specific Routes



IP Assignment

- How does a host get a unique IP address?
- **Static IP**
 - Manually configured
 - For Linux: `ifconfig eth0 140.113.94.87`
- **Dynamic IP**
 - DHCP: **Dynamic Host Configuration Protocol**
 - Temporary IP address
 - Plug-and-play
- Also need to configure
 - Subnet mask
 - Default gateway (router)
 - DNS servers

windows



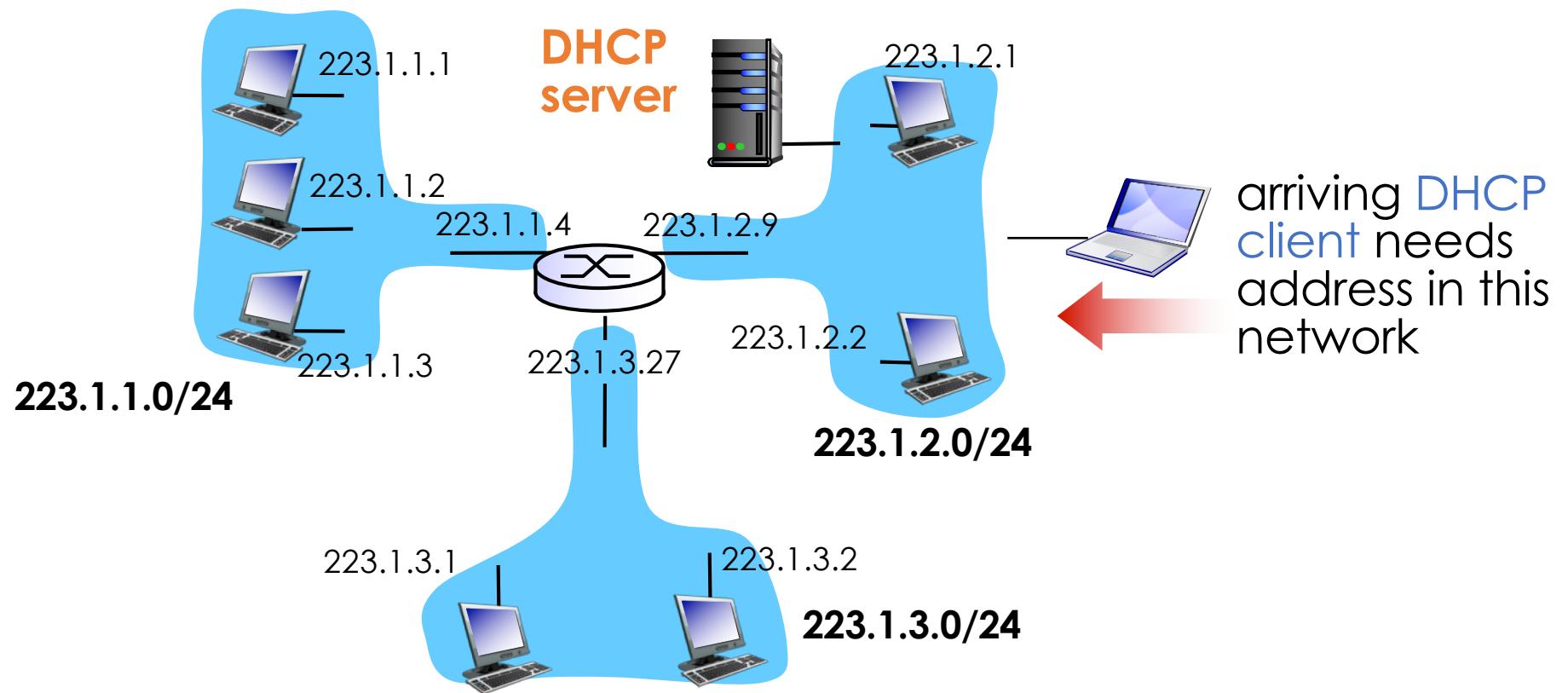
Outline

- Overview of network layer
- What's inside a router
- **IP: Internet Protocol**
 - Datagram format
 - IPv4 addressing
 - **DHCP**
 - Network address translation (NAT)
 - IPv6
- Software defined networking

DHCP

- Allow a host to *dynamically* and *automatically* get an IP address
 - Can **renew** its lease on address in use
 - Allow reuse of addresses (only hold address while connected/“on”)
 - Support for mobile users who want to join network
→ mobile IP (see Ch. 6)
- Client-server architecture
 - Server is found using **link-layer broadcasting**
 - Use UDP, port 67

DHCP Framework



DHCP Flow

DHCP server:
223.1.2.5

1. DHCP discover

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

arriving
client

2. DHCP offer

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

3. DHCP request

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

4. DHCP ACK

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

Try Wireshark

Why steps 1- 2?

May have multiple
DHCP servers in a subnet

Also known as
lease time

Who Assign IP Addresses?

- ICANN: Internet Corporation for Assigned Names and Numbers <http://www.icann.org/>
 - Non-profit organization
 - Allocate addresses
 - Manage DNS
 - Assign domain names, resolves disputes