Unit No:4

Network Layer

Marwadi University

Department of Computer Engineering

Computer Networks (3150710)

Dr. Sushil Kumar Singh
Associate Professor

Chapter 4: Network Layer

Syllabus & Goals

- 4. 1 Introduction
- 4.2 Virtual circuit and datagram networks
- 4.3 What's inside a router
- 4.4 IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - NAT
  - ICMP
  - IPv6
- 4.5 Routing algorithms
  - Link state
  - Distance Vector
  - Hierarchical routing
- 4.6 Routing in the Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Broadcast and multicast routing

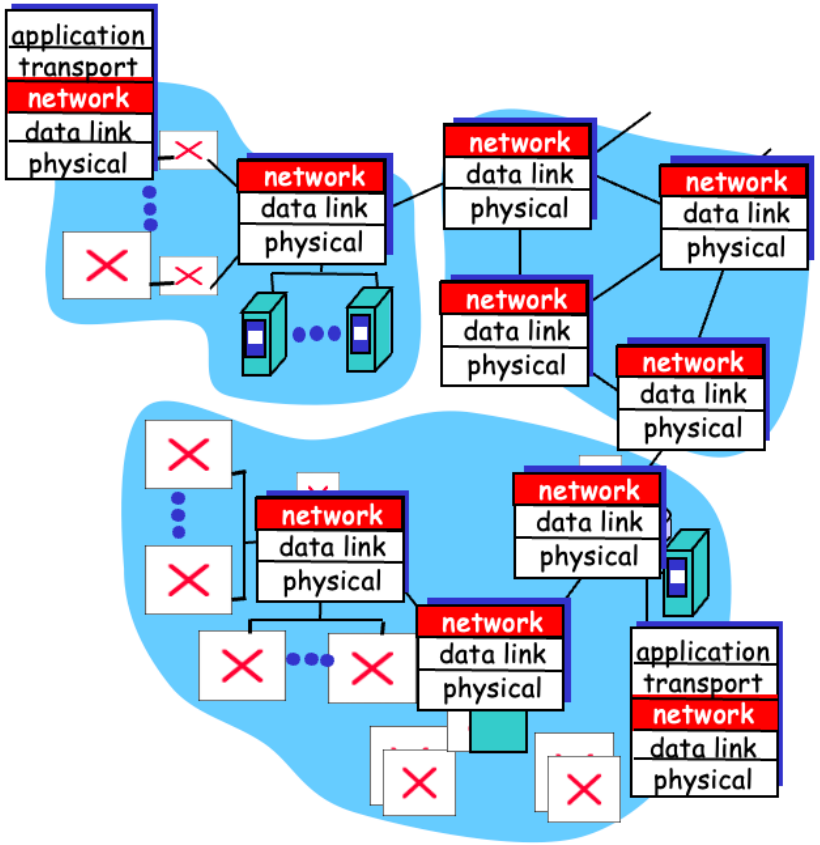
Chapter goals:

- understand principles behind network layer services:
  - network layer service models
  - forwarding versus routing
  - how a router works
  - routing (path selection)
  - dealing with scale
  - advanced topics: IPv6, mobility
- instantiation, implementation in the Internet

Network Layer

Network layer

- on sending side encapsulates segments into datagrams
- on rcving side, delivers segments to transport layer
- network layer protocols in *every* host, router
- Router examines header fields in all IP datagrams passing through it



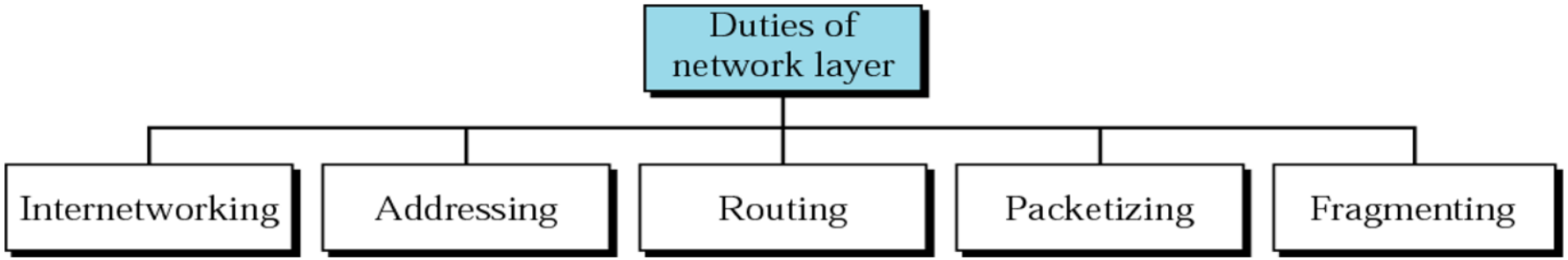
Key Network-Layer Functions

Network Layer (Functions)

- *forwarding:* move packets from router's input to appropriate router output
- *routing:* determine route taken by packets from source to dest.
  - *Routing algorithms*

analogy:

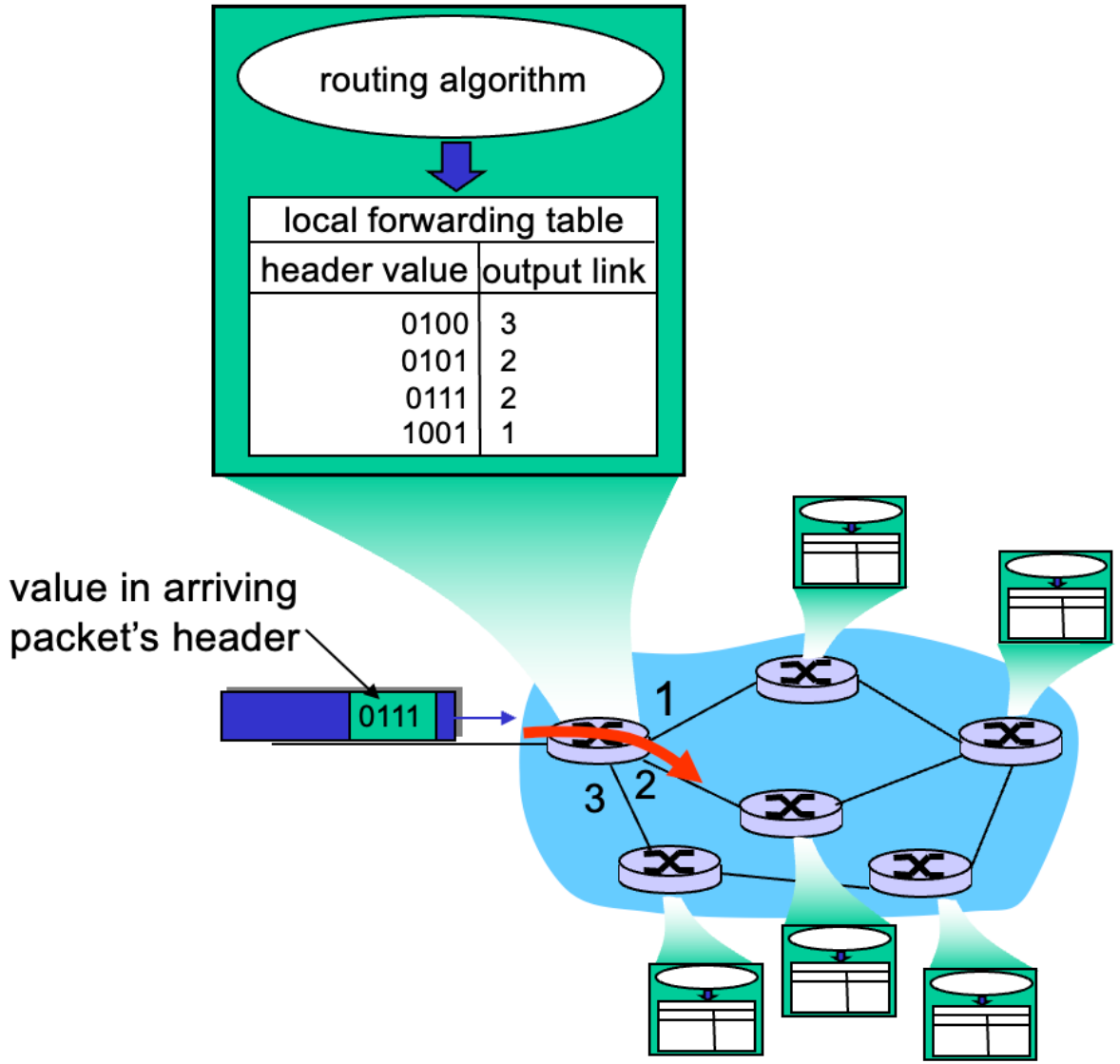
- routing: process of planning trip from source to dest
- forwarding: process of correct left turns, right turns, exits, etc.



Network Layer

Interplay between routing and forwarding

Interplay between routing and forwarding



Network Layer
Connection Setup

Connection setup

- important function in *some* network architectures:
  - ATM, frame relay, X.25
- Before datagrams flow, two hosts and intervening routers establish virtual connection
  - Routers get involved
- Network and transport layer cnctn service:
  - Network: between two hosts
  - Transport: between two processes

Network Layer

Chapter 4: Network Layer

- 4. 1 Introduction
- 4.2 Virtual circuit and datagram networks
- 4.3 What's inside a router
- 4.4 IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- 4.5 Routing algorithms
  - Link state
  - Distance Vector
  - Hierarchical routing
- 4.6 Routing in the Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Broadcast and multicast routing

Network Layer
Connection, connection-less service

Network layer connection and connection-less service

- Datagram network provides network-layer connectionless service
- VC network provides network-layer connection service
- Analogous to the transport-layer services, but:
  - Service: host-to-host
  - No choice: network provides one or the other
  - Implementation: in the core

Network Layer
Virtual Circuits

Virtual circuits

"source-to-dest path behaves much like telephone circuit"

- performance-wise
- network actions along source-to-dest path

- call setup, teardown for each call *before* data can flow
- each packet carries VC identifier (not destination host address)
- *every* router on source-dest path maintains "state" for each passing connection
- link, router resources (bandwidth, buffers) may be *allocated* to VC

Network Layer
VC
Implementation

VC implementation

A VC consists of:

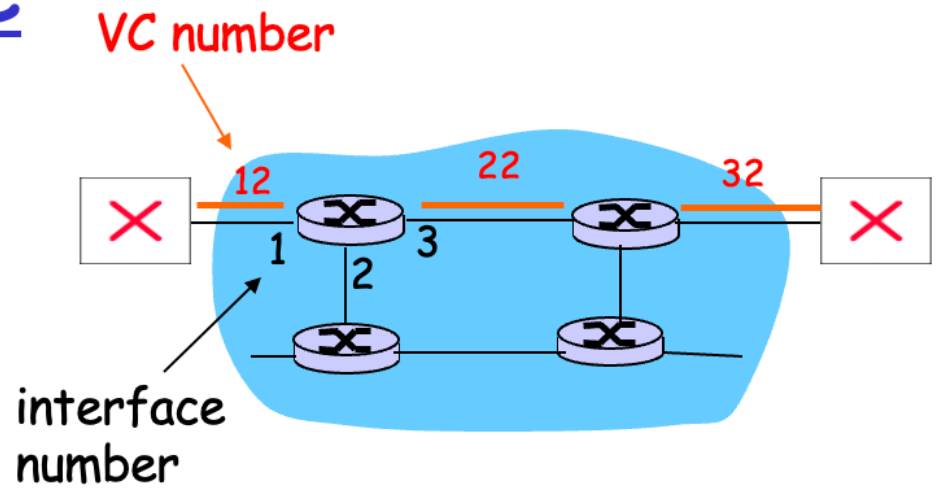
1. Path from source to destination
2. VC numbers, one number for each link along path
3. Entries in forwarding tables in routers along path

Example next slide

- Packet belonging to VC carries a VC number.
- VC number must be changed on each link.
  - New VC number comes from forwarding table

Network Layer
VC Forwarding Table

Forwarding table



Forwarding table in northwest router:

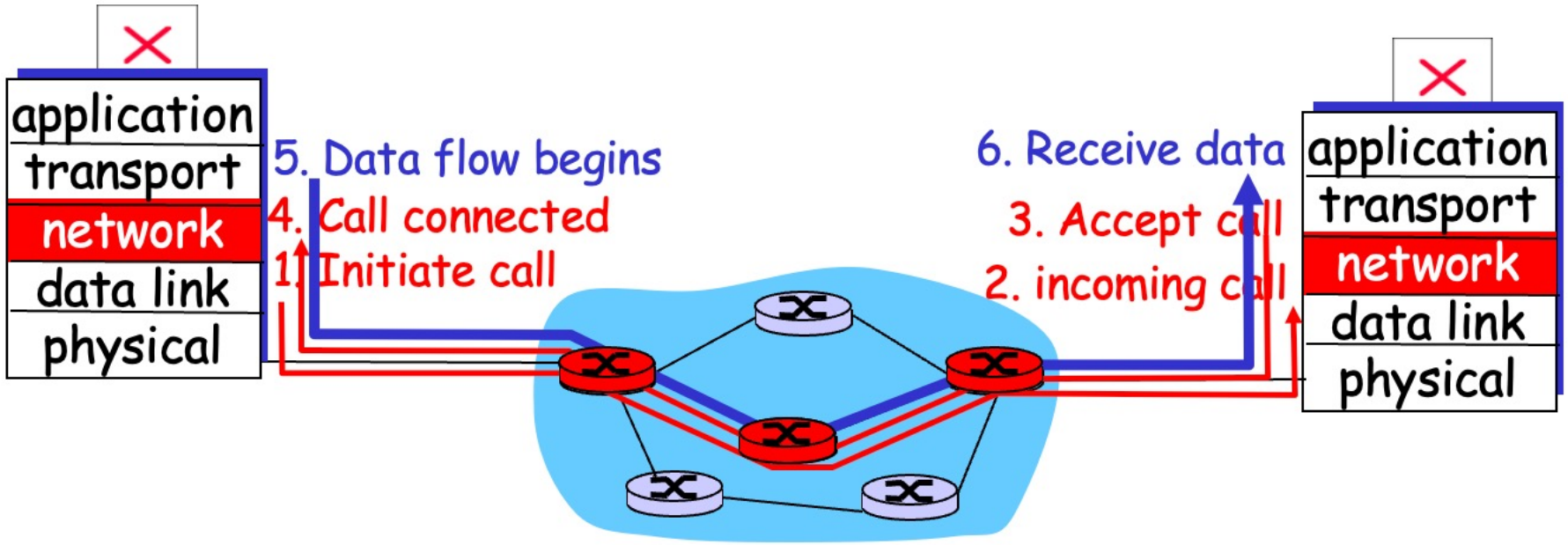
Incoming interface	Incoming VC #	Outgoing interface	Outgoing VC #
1	12	3	22
2	63	1	18
3	7	2	17
1	97	3	87
…	…	…	…

Routers maintain connection state information!

Network Layer
Virtual circuits: signaling protocols

Virtual circuits: signaling protocols

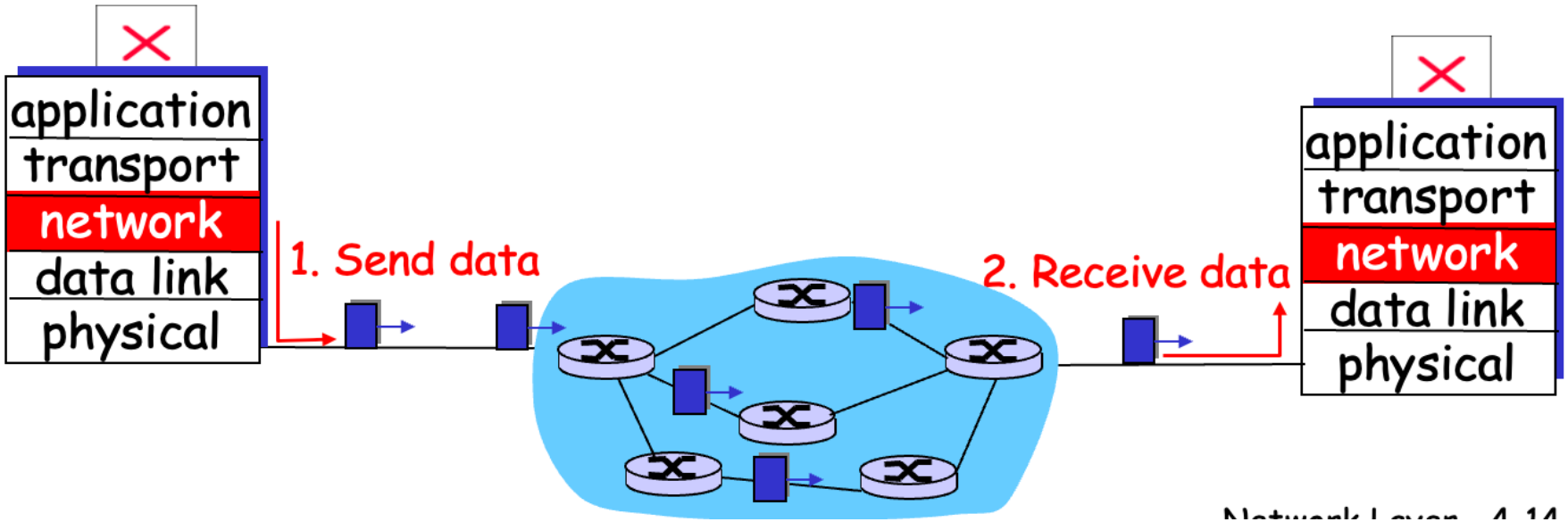
- used to setup, maintain teardown VC
- used in ATM, frame-relay, X.25
- not used in today's Internet



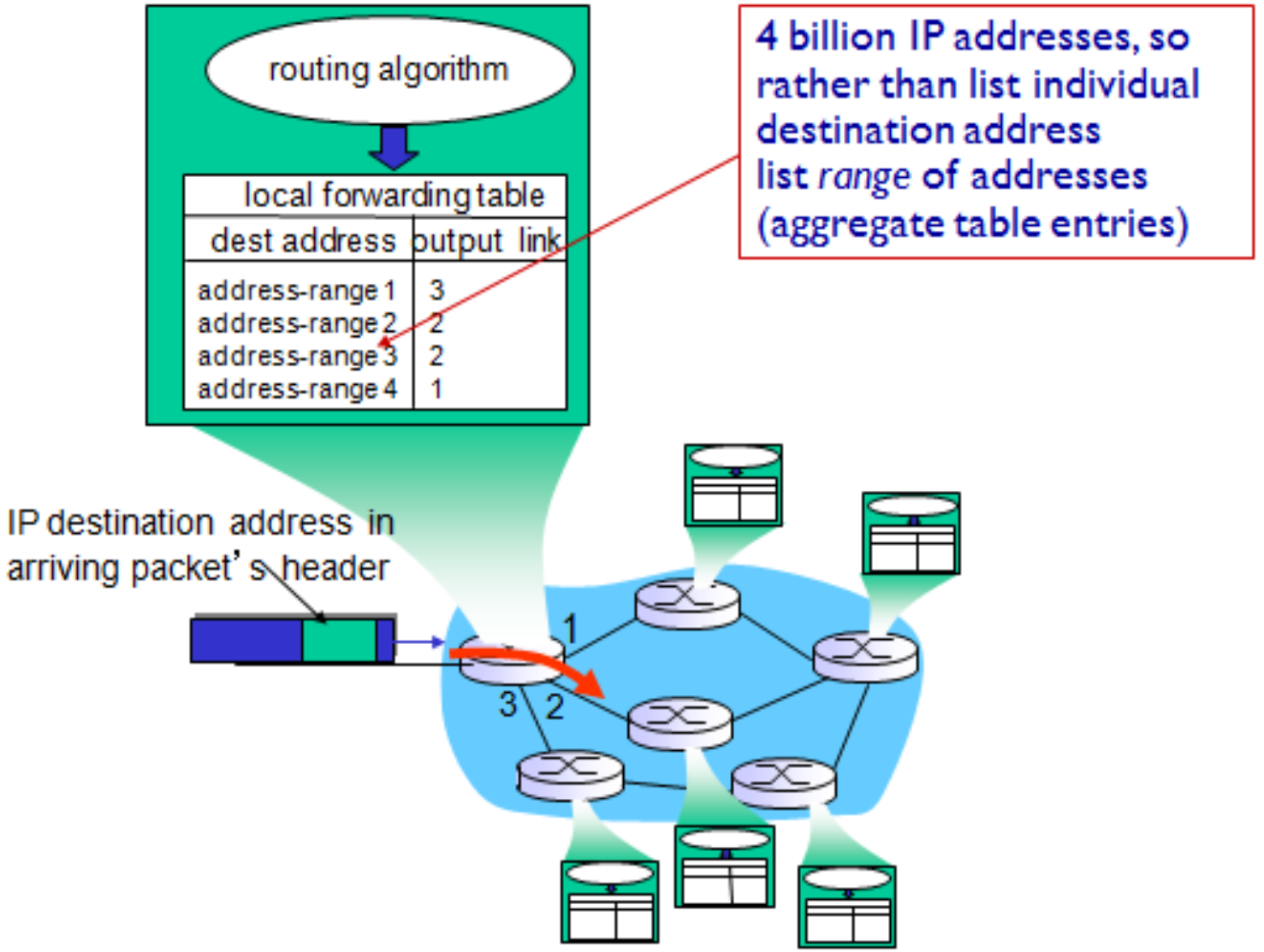
Network Layer
Datagram networks

Datagram networks

- no call setup at network layer
- routers: no state about end-to-end connections
  - no network-level concept of "connection"
- packets forwarded using destination host address
  - packets between same source-dest pair may take different paths



Network Layer
Datagram forwarding table



Network Layer

Datagram vs. VC network

Datagram vs. VC network

Issue	Datagram	Virtual Circuit
Connection Setup	None	Required
Addressing	Packet contains full source and destination address	Packet contains short virtual circuit number identifier.
State Information	None other than router table containing destination network	Each virtual circuit number entered to table on setup, used for routing.
Routing	Packets routed independently	Route established at setup, all packets follow same route.
Effect of Router Failure	Only on packets lost during crash	All virtual circuits passing through failed router terminated.
Congestion Control	Difficult since all packets routed independently router resource requirements can vary.	Simple by pre-allocating enough buffers to each virtual circuit at setup, since maximum number of circuits fixed.

Network Layer
Datagram vs. VC network

Datagram or VC network: why?

Internet

- data exchange among computers
  - "elastic" service, no strict timing req.
- "smart" end systems (computers)
  - can adapt, perform control, error recovery
  - simple inside network, complexity at "edge"
- many link types
  - different characteristics
  - uniform service difficult

ATM

- evolved from telephony
- human conversation:
  - strict timing, reliability requirements
  - need for guaranteed service
- "dumb" end systems
  - telephones
  - complexity inside network

Network Layer

Chapter 4: Network Layer

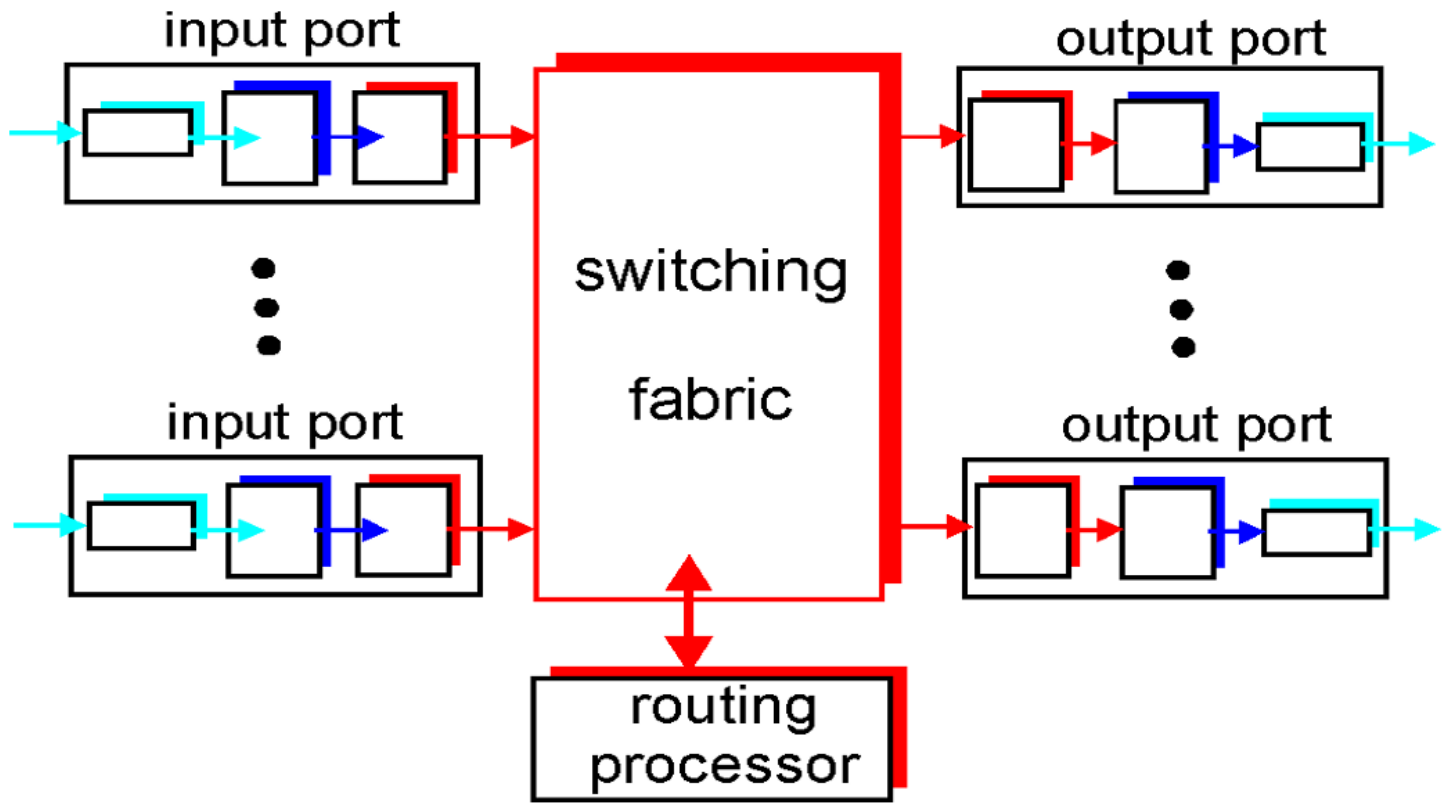
- 4. 1 Introduction
- 4.2 Virtual circuit and datagram networks
- 4.3 What's inside a router
- 4.4 IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- 4.5 Routing algorithms
  - Link state
  - Distance Vector
  - Hierarchical routing
- 4.6 Routing in the Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Broadcast and multicast routing

Network Layer
(Router)

Router Architecture Overview

Two key router functions:

- run routing algorithms/protocol (RIP, OSPF, BGP)
- *forwarding* datagrams from incoming to outgoing link



Network Layer

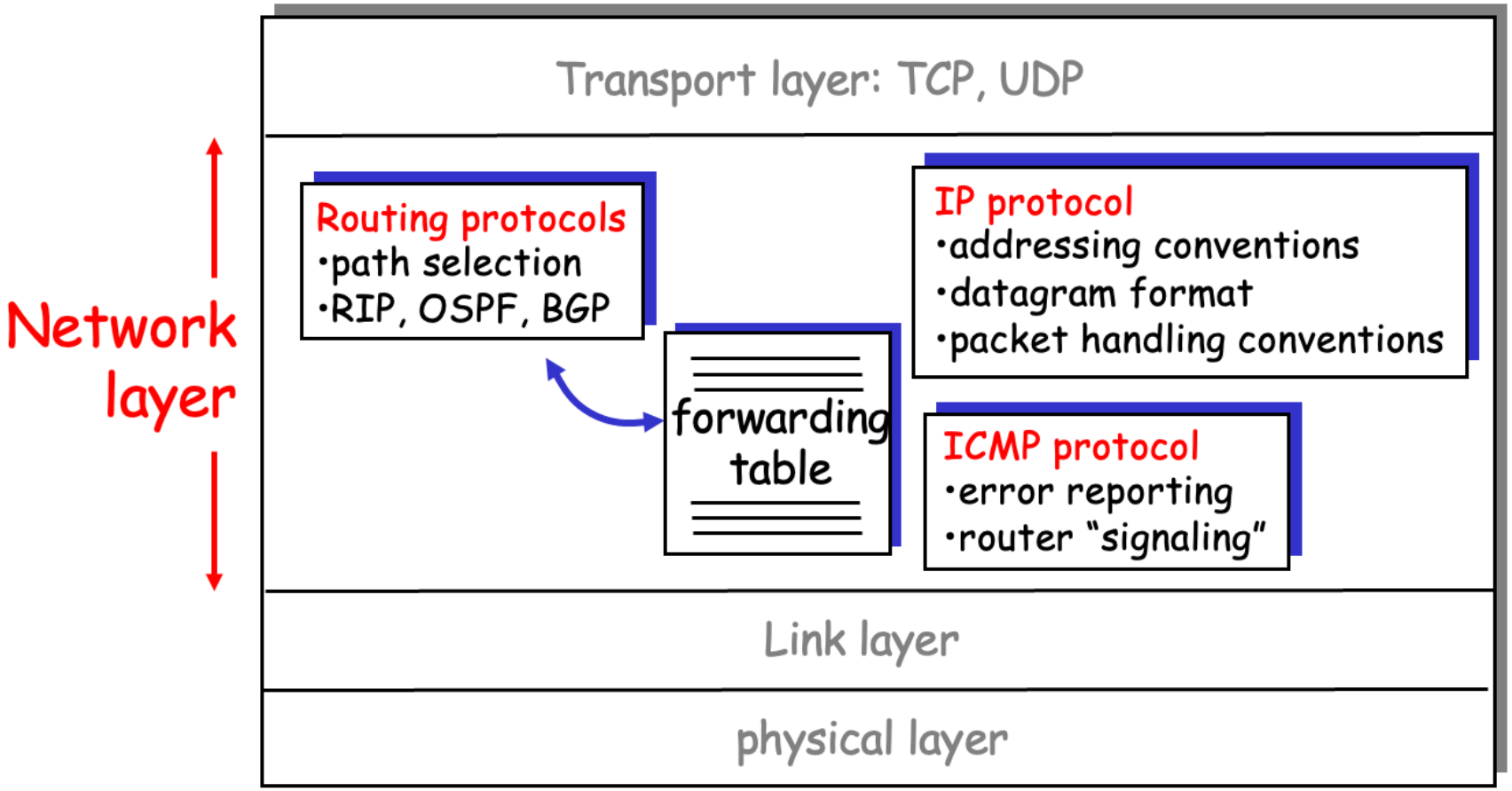
Chapter 4: Network Layer

- 4. 1 Introduction
- 4.2 Virtual circuit and datagram networks
- 4.3 What's inside a router
- 4.4 IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- 4.5 Routing algorithms
  - Link state
  - Distance Vector
  - Hierarchical routing
- 4.6 Routing in the Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Broadcast and multicast routing

Network Layer (Host, Router Network Layer Functions)

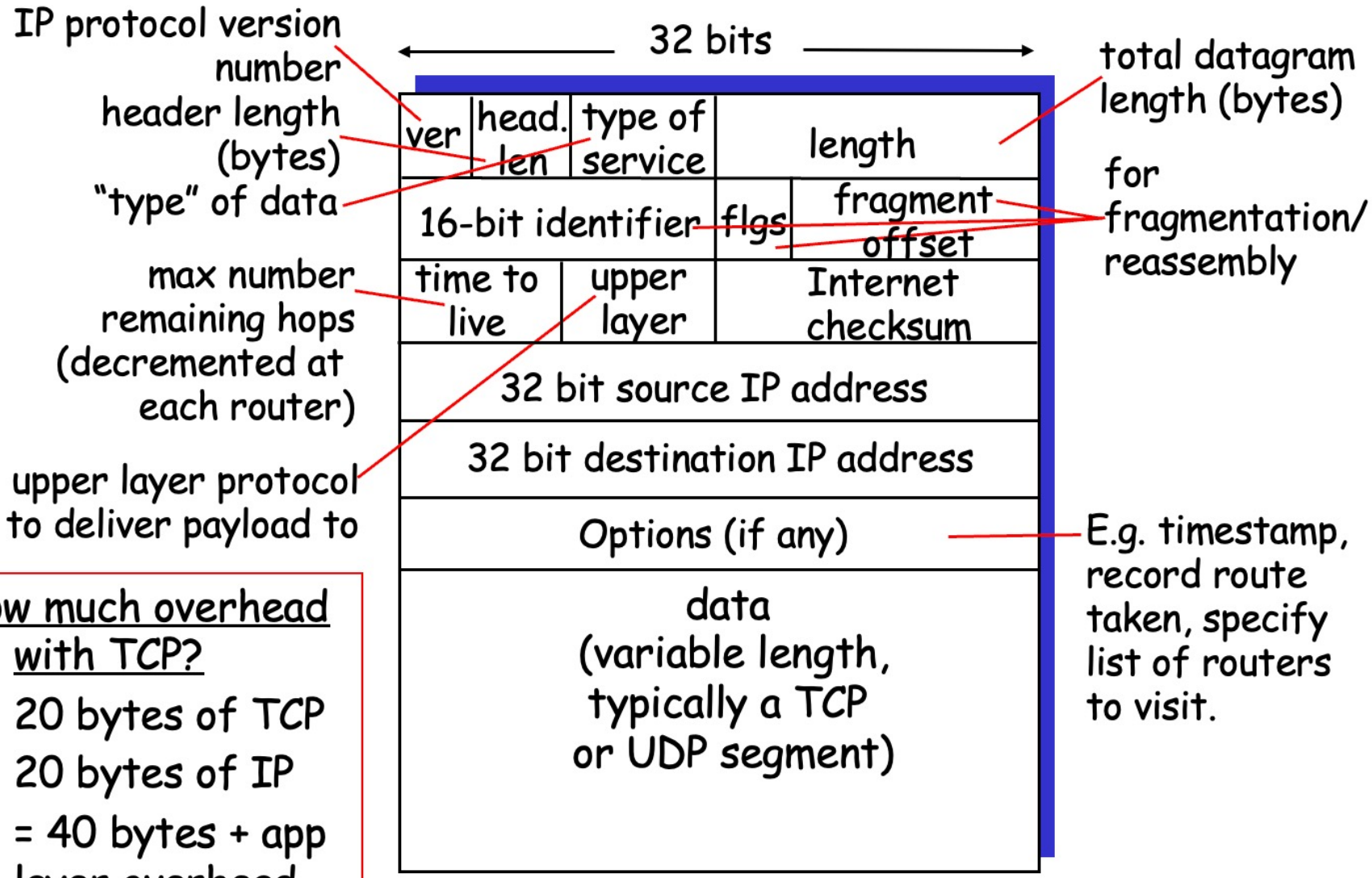
The Internet Network layer

Host, router network layer functions:



Network Layer
IP datagram format

IP datagram format



how much overhead with TCP?

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

Network Layer
IP datagram fields

- **Version number**: These 4 bits specify the IP protocol version of the datagram. It determines how to interpret the header. Currently the only permitted values are 4 (0100) or 6 (0110).
- **Header length:** Specifies the length of the IP header, in 32-bit words.
- **Type of service:** The type of service (TOS) bits were included in the IPv4 header to allow different types of IP datagrams (for example, datagrams particularly requiring low delay, high throughput, or reliability) to be distinguished from each other.
- **Datagram length:** This is the total length of the IP datagram (header plus data), measured in bytes.
- **Identifier:** Uniquely identifies the datagram. It is incremented by 1 each time a datagram is sent. All fragments of a datagram contain the same identification value. This allows the destination host to determine which fragment belongs to which datagram.
- **Flags:** In order for the destination host to be absolutely sure it has received the last fragment of the original datagram, the last fragment has a flag bit set to 0, whereas all the other fragments have this flag bit set to 1.
- **Fragmentation offset:** When fragmentation of a message occurs, this field specifies the offset, or position, in the overall message where the data in this fragment goes. It is specified in units of 8 bytes (64 bits).

Network Layer Continue..

- **Time-to-live:** Specifies how long the datagram is allowed to "live" on the network. Each router decrements the value of the TTL field (reduces it by one) prior to transmitting it. If the TTL field drops to zero, the datagram is assumed to have taken too long a route and is discarded.
- **Protocol**: This field is used only when an IP datagram reaches its final destination. The value of this field indicates the specific transport-layer protocol to which the data portion of this IP datagram should be passed. For example, a value of 6 indicates that the data portion is passed to TCP, while a value of 17 indicates that the data is passed to UDP.
- **Header checksum:** The header checksum aids a router in detecting bit errors in a received IP datagram.
- **Source and destination IP addresses:** When a source creates a datagram, it inserts its IP address into the source IP address field and inserts the address of the ultimate destination into the destination IP address field.
- **Options:** The options fields allow an IP header to be extended.
- **Data (payload):** The data to be transmitted in the datagram, either an entire higher-layer message or a fragment of one.

Network Layer
IP Addressing

Addressing

Internet Address

Classful Addressing

Subnetting

Supernetting

Classless Addressing

Dynamic Address Configuration

Network Address Translation

Network Layer
IP Addressing

Dotted Decimal Notation

Note:

An IP address is a 32-bit address.

The IP addresses are unique and universal.

10000000 00001011 00000011 00011111

128.11.3.31

Network Layer
IP Addressing

Dotted Decimal Notation

Example 1

Change the following IP addresses from binary notation to dotted-decimal notation.

a. 10000001 00001011 00001011 11101111

b. 11111001 10011011 11111011 00001111

Solution

We replace each group of 8 bits with its equivalent decimal number (see Appendix B) and add dots for separation:

a. 129.11.11.239

b. 249.155.251.15

Network Layer
IP Addressing

Dotted Decimal Notation

Example 2

Change the following IP addresses from dotted-decimal notation to binary notation.

a. 111.56.45.78

b. 75.45.34.78

Solution

We replace each decimal number with its binary equivalent (see Appendix B):

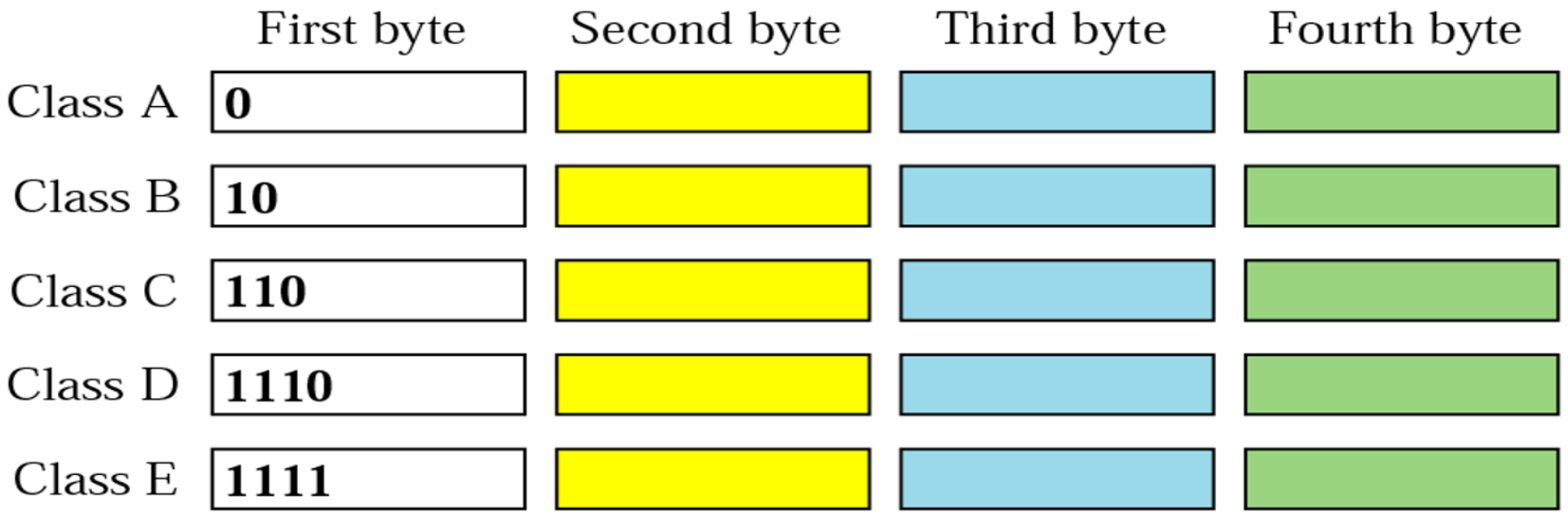
a. 01101111 00111000 00101101 01001110
b. 01001011 00101101 00100010 01001110

Note:

In classful addressing, the address space is divided into five classes: A, B, C, D, and E.

Network Layer
IP Addressing

Finding the class in binary notation



Example 3

Find the class of each address:

a. **0**0000001 00001011 00001011 11101111

b. **1111**0011 10011011 11111011 00001111

Solution

See the procedure in Figure 19.11.

a. **The first bit is 0; this is a class A address.**

b. **The first 4 bits are 1s; this is a class E address.**

Network Layer
IP Addressing

Finding the class in decimal notation

	First byte	Second byte	Third byte	Fourth byte
Class A	0 to 127			
Class B	128 to 191			
Class C	192 to 223			
Class D	224 to 239			
Class E	240 to 255			

Example 4

Find the class of each address:

a. **227**.12.14.87

b. **252**.5.15.111

c. **134**.11.78.56

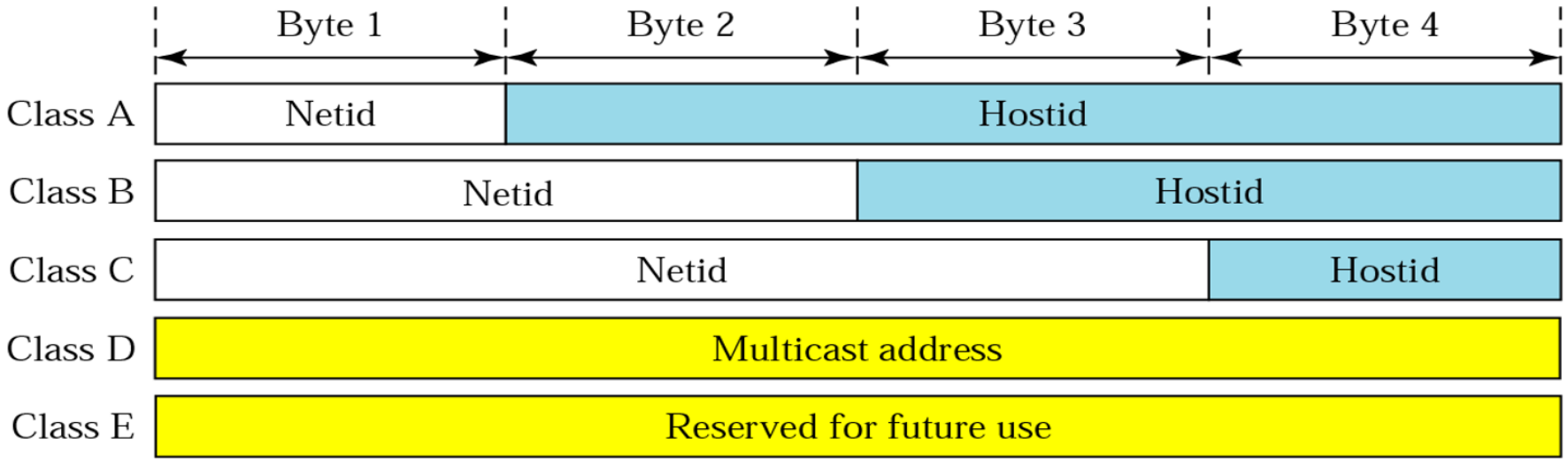
Solution

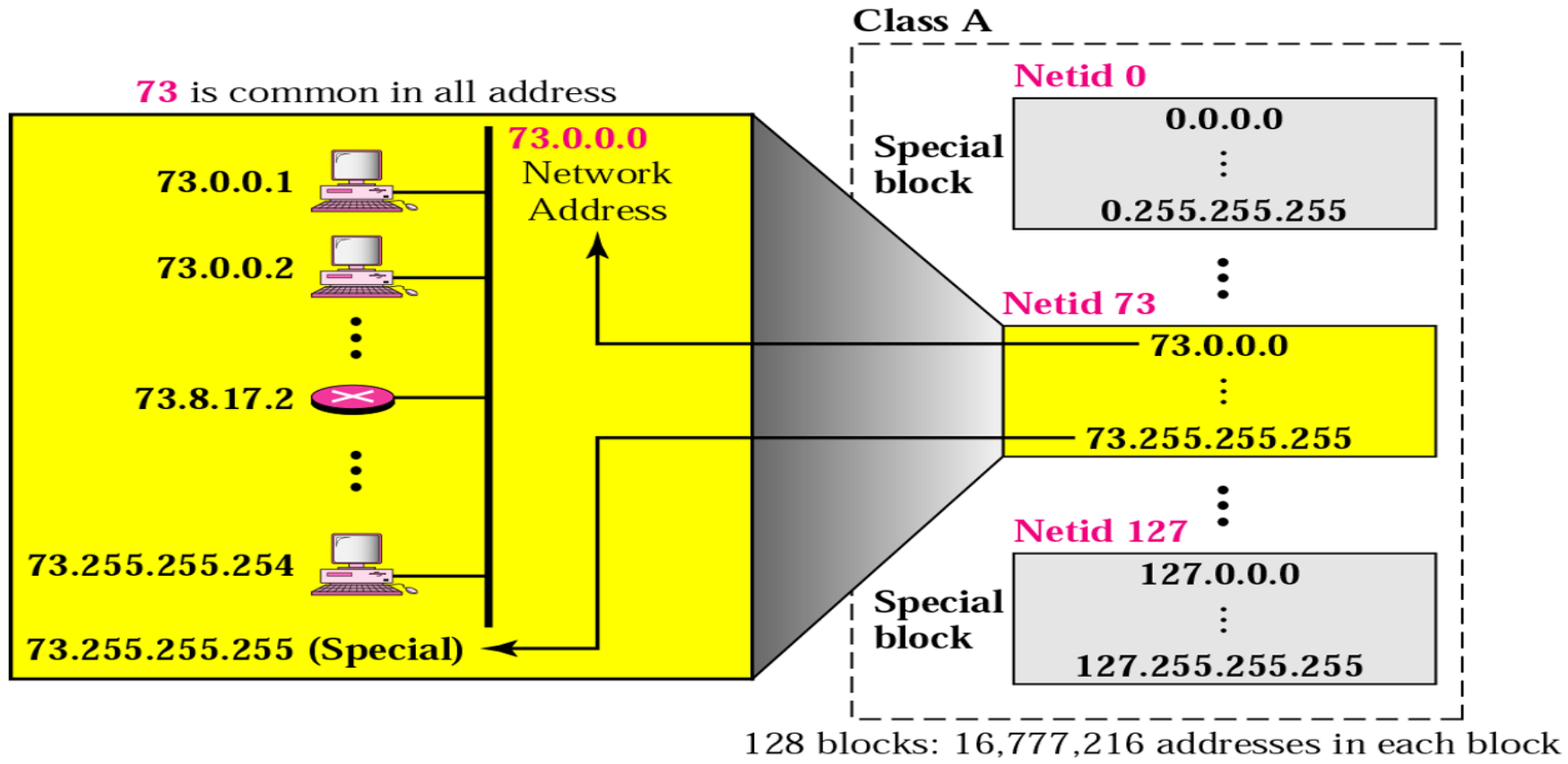
a. The first byte is 227 (between 224 and 239); the class is D.
b. The first byte is 252 (between 240 and 255); the class is E.
c. The first byte is 134 (between 128 and 191); the class is B.

Network Layer
IP Addressing

NetID, HostID

Blocks in class A





Network Layer

IP Addressing

NetID, HostID

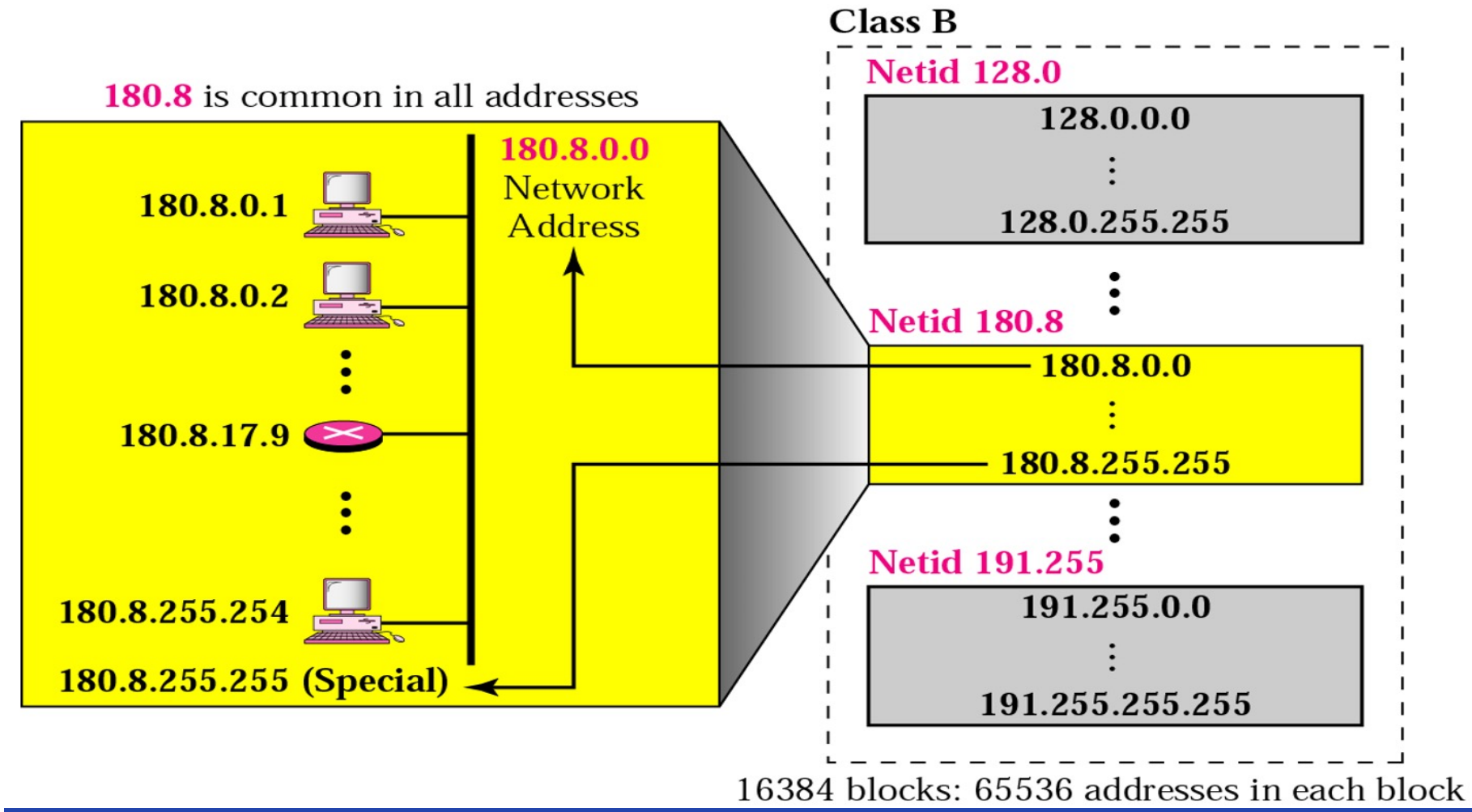
Blocks in class A

Note:

Millions of class A addresses are wasted.

Network Layer

Blocks in class B



Note:

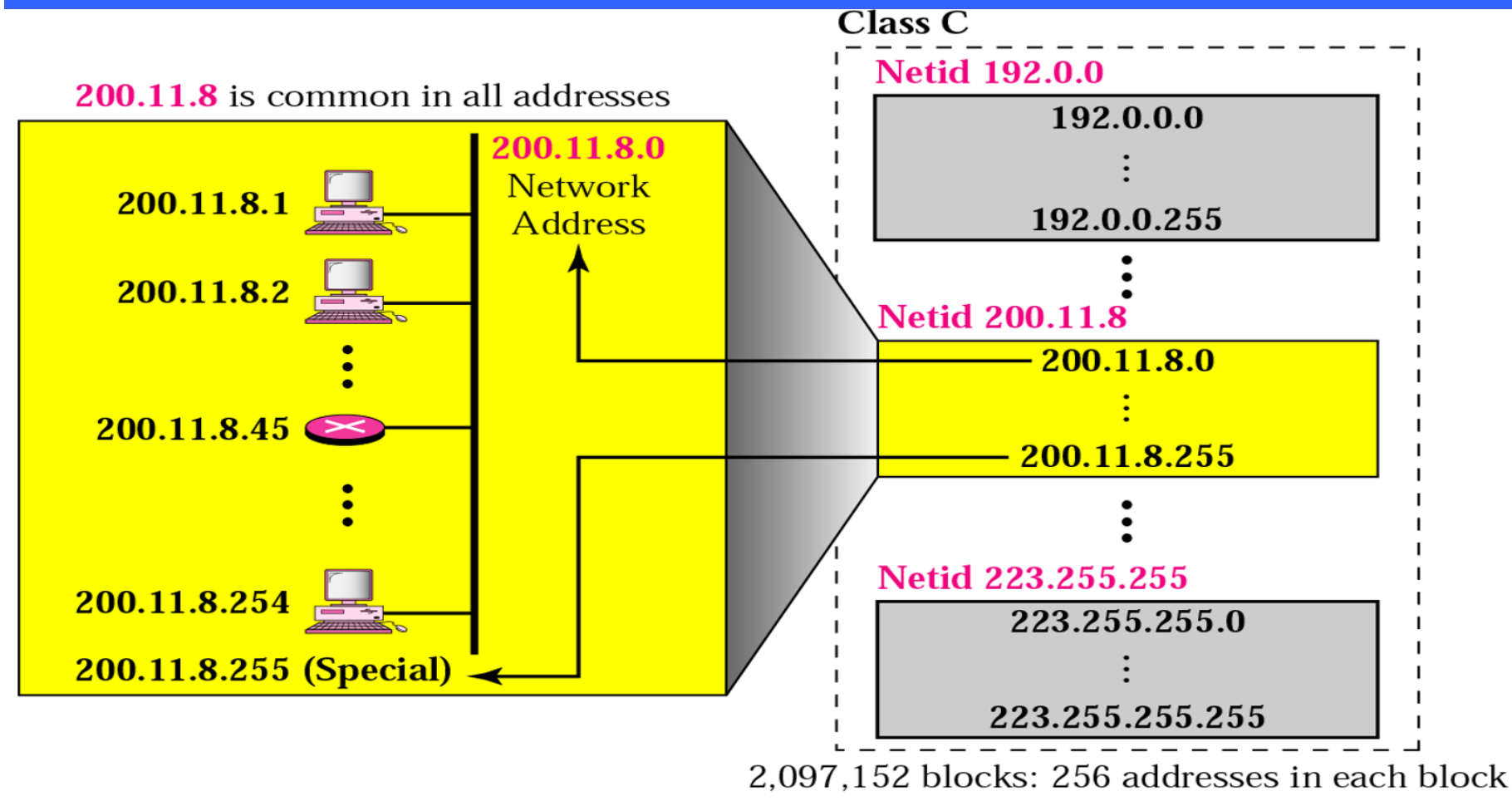
Many class B addresses are wasted.

Network Layer

Blocks in class C

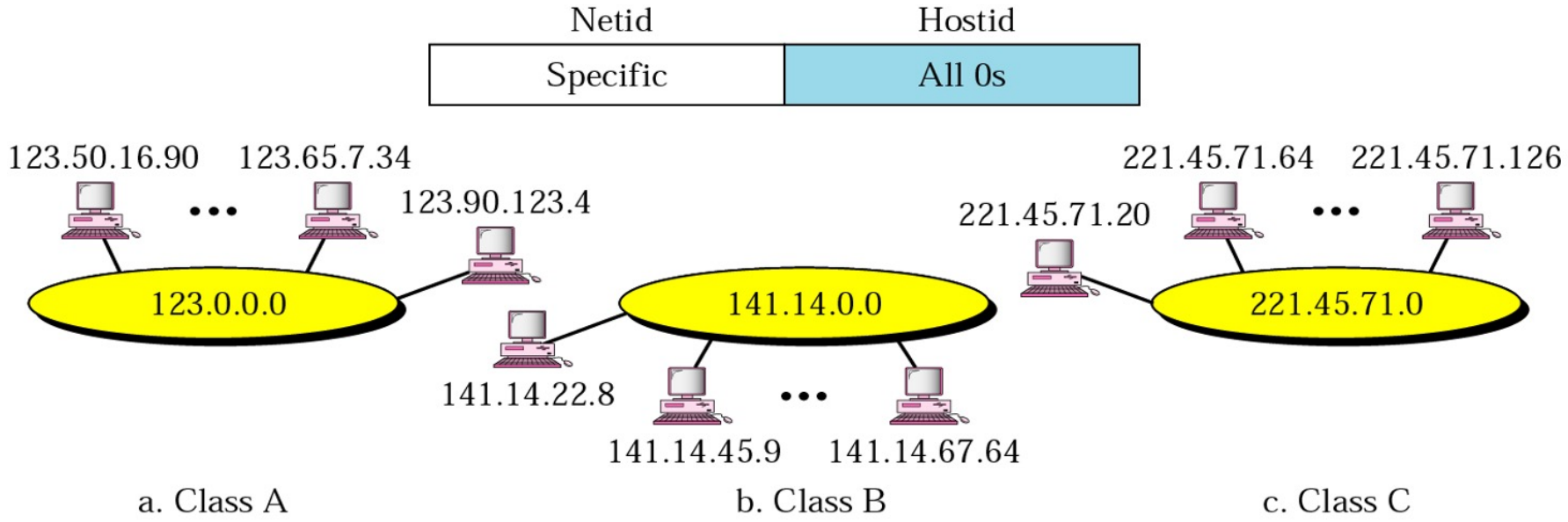
Note:

The number of addresses in class C is smaller than the needs of most organizations.



Network Layer

Network Address



Note:

In classful addressing, the network address is the one that is assigned to the organization.

Network Layer

Network Address

Example 5

Given the address 23.56.7.91, find the network address.

Solution

The class is A. Only the first byte defines the netid. We can find the network address by replacing the hostid bytes (56.7.91) with 0s. Therefore, the network address is 23.0.0.0.

Example 6

Given the address 132.6.17.85, find the network address.

Solution

The class is B. The first 2 bytes defines the netid. We can find the network address by replacing the hostid bytes (17.85) with 0s. Therefore, the network address is 132.6.0.0.

Example 7

Given the network address 17.0.0.0, find the class.

Solution

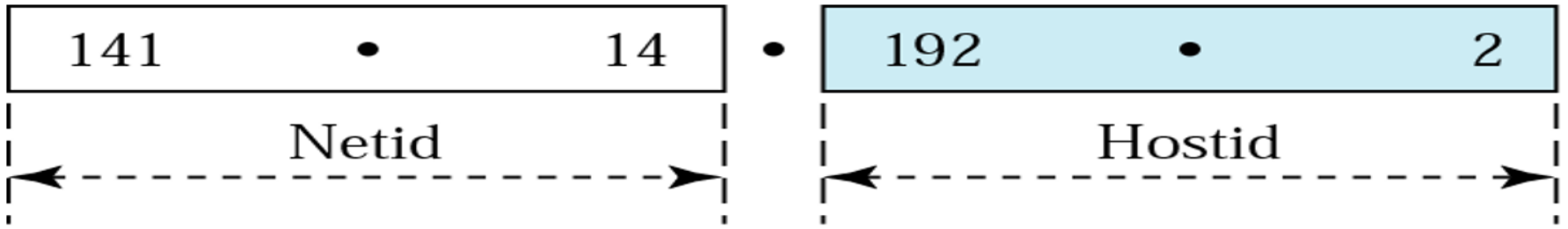
The class is A because the netid is only 1 byte.

Note:

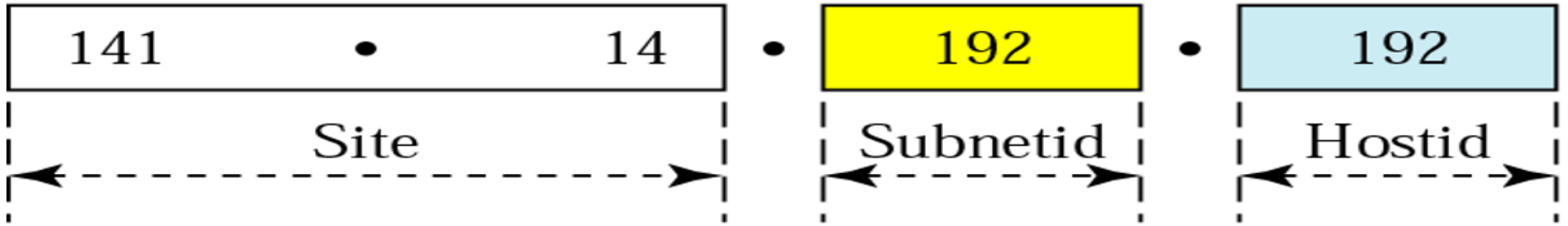
A network address is different from a netid. A network address has both netid and hostid, with 0s for the hostid.

Network Layer

Addresses in a network with and without subnetting



a. Without subnetting



b. With subnetting

Table 19.1 Default masks

Class	*In Binary*	*In Dotted-Decimal*	*Using Slash*
A	11111111 00000000 00000000 00000000	255.0.0.0	/8
B	11111111 11111111 00000000 00000000	255.255.0.0	/16
C	11111111 111111111 11111111 00000000	255.255.255.0	/24

Network Layer

Numerical

Note:

The network address can be found by applying the default mask to any address in the block (including itself). It retains the netid of the block and sets the hostid to 0s.

Example 8

A router outside the organization receives a packet with destination address 190.240.7.91. Show how it finds the network address to route the packet.

Solution

The router follows three steps:

1. The router looks at the first byte of the address to find the class. It is class B.
2. The default mask for class B is 255.255.0.0. The router ANDs this mask with the address to get 190.240.0.0.
3. The router looks in its routing table to find out how to route the packet to this destination. Later, we will see what happens if this destination does not exist.

Network Layer

CIDR

IP addressing: CIDR

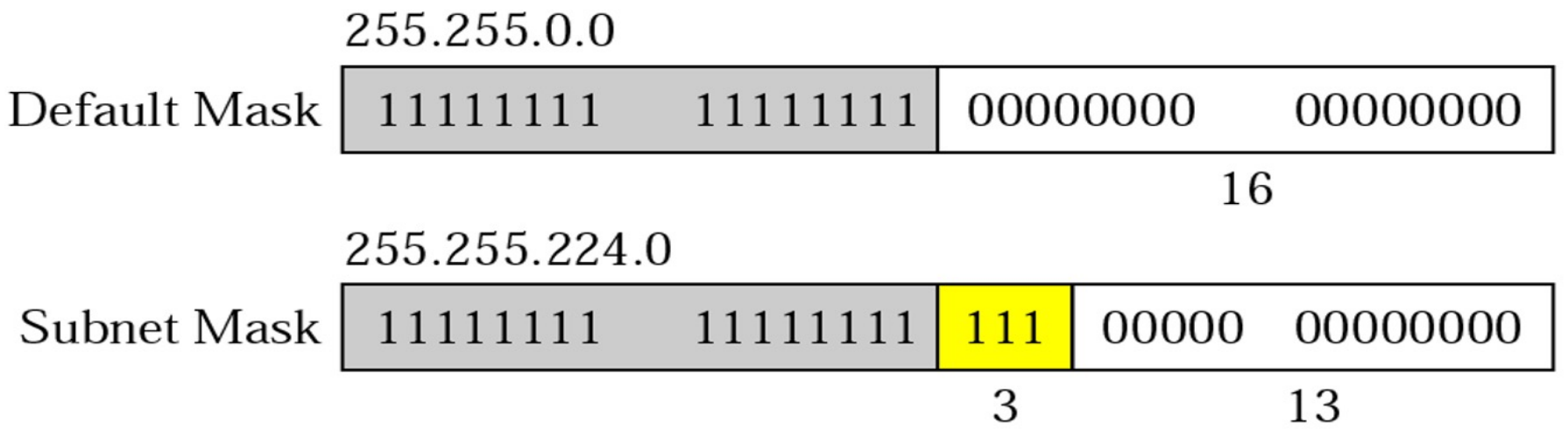
CIDR: Classless InterDomain Routing

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

200.23.16.0/23

Network Layer

Subnet Mask



Example 9

A router inside the organization receives the same packet with destination address 190.240.33.91. Show how it finds the subnetwork address to route the packet.

Solution

The router follows three steps:

1. **The router must know the mask. We assume it is /19, as shown in Figure 19.23.**
2. **The router applies the mask to the address, 190.240.33.91. The subnet address is 190.240.32.0.**
3. **The router looks in its routing table to find how to route the packet to this destination. Later, we will see what happens if this destination does not exist.**

Network Layer

Default Mask

Table 19.2 Default masks

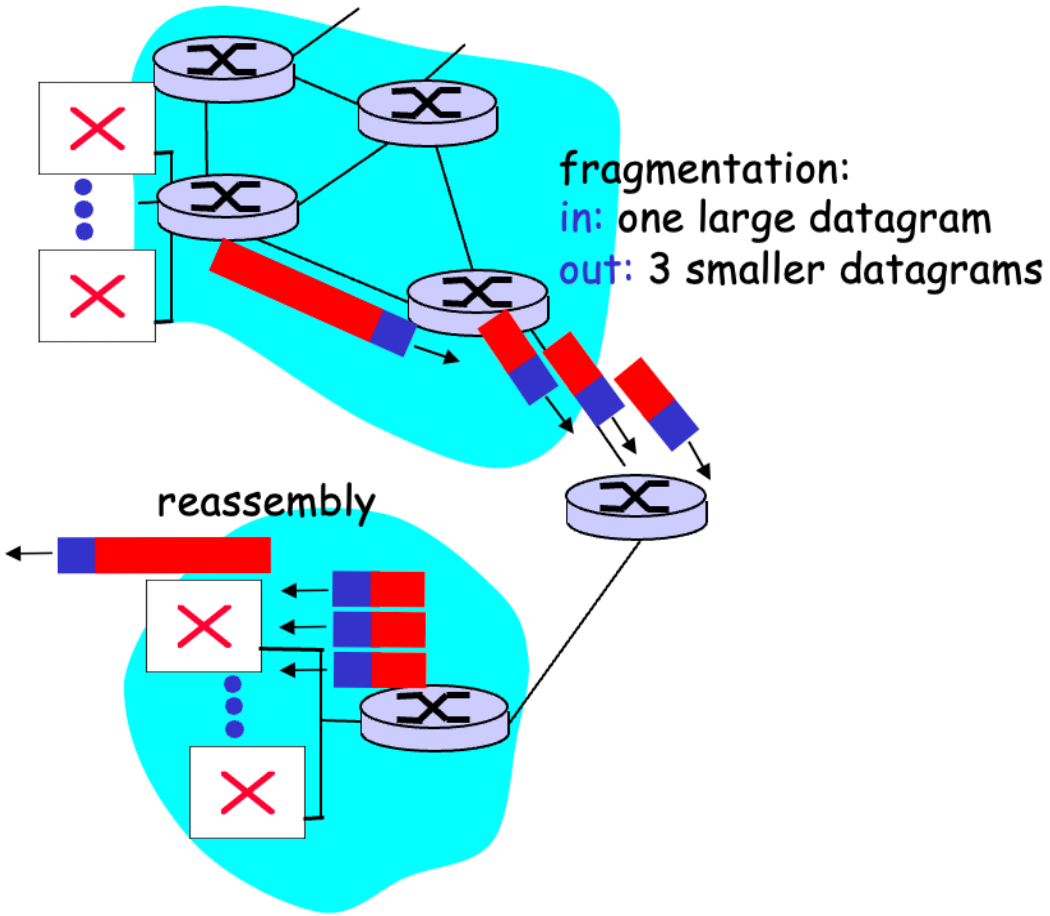
Range			Total
10.0.0.0	to	10.255.255.255	2^{24}
172.16.0.0	to	172.31.255.255	2^{20}
192.168.0.0	to	192.168.255.255	2^{16}

Network Layer

IP Fragmentation & Reassembly

IP Fragmentation & Reassembly

- network links have MTU (max.transfer size) - largest possible link-level frame.
  - different link types, different MTUs
- large IP datagram divided ("fragmented") within net
  - one datagram becomes several datagrams
  - "reassembled" only at final destination
  - IP header bits used to identify, order related fragments



Network Layer

IPv6

IPv6

IPv6 Addresses

Categories of Addresses

IPv6 Packet Format

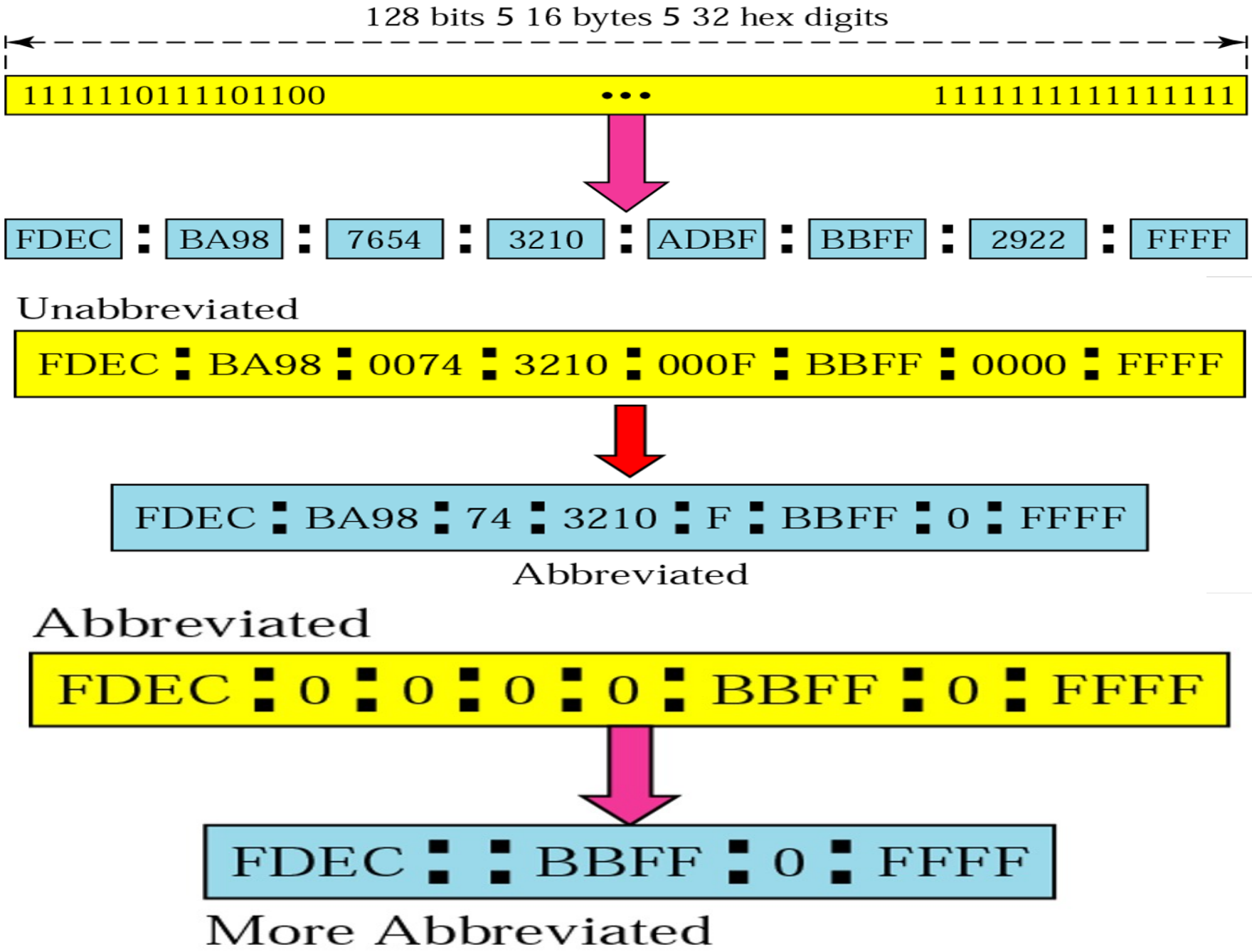
Fragmentation

ICMPv6

Transition

Network Layer

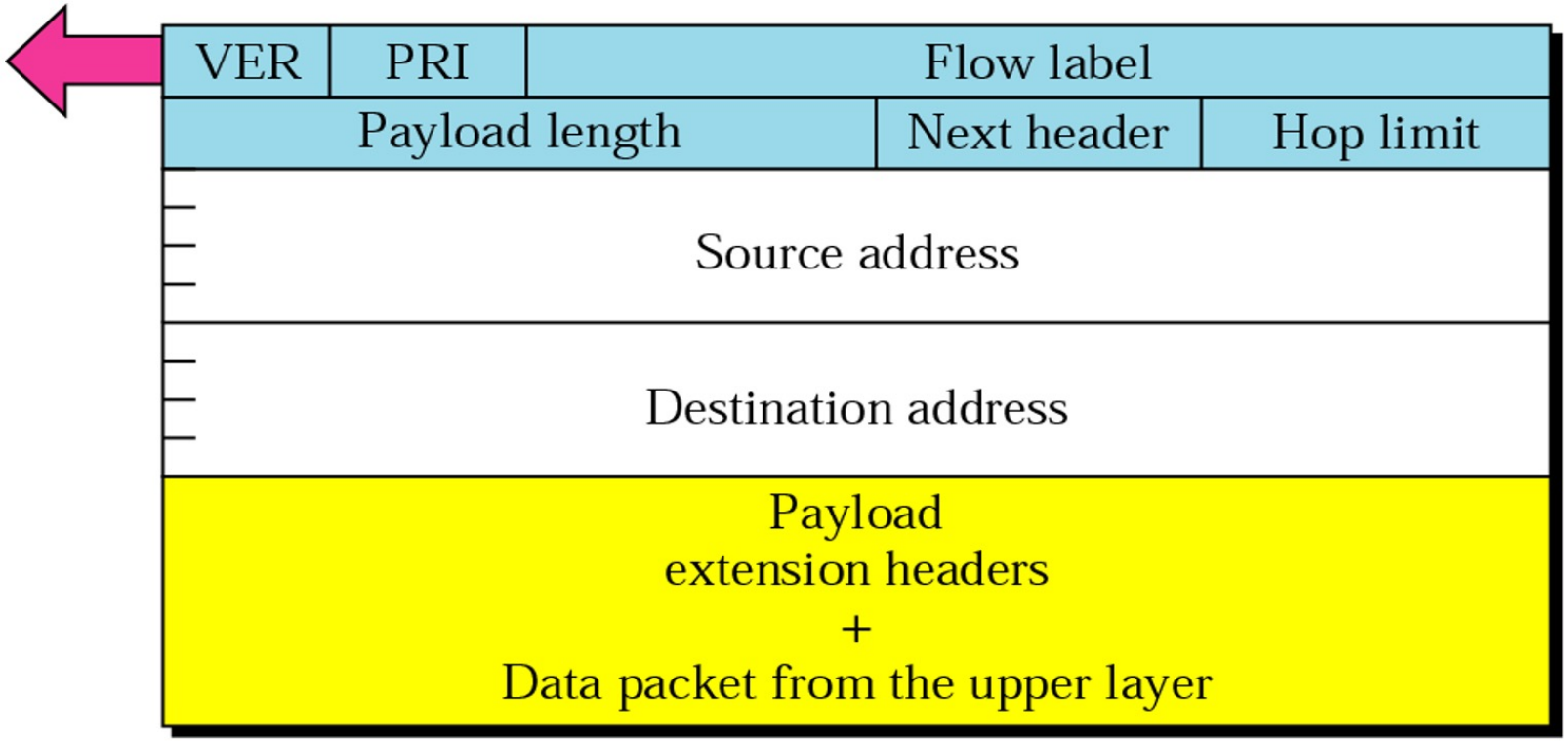
IPv6 Address Abbreviated Address



Network Layer

Format of an IPv6 datagram

Format of an IPv6 datagram



CIDR address

FDEC : 0 : 0 : 0 : 0 : BBFF : 0 : FFFF/60

Difference between IPv4 and IPv6

Network Layer

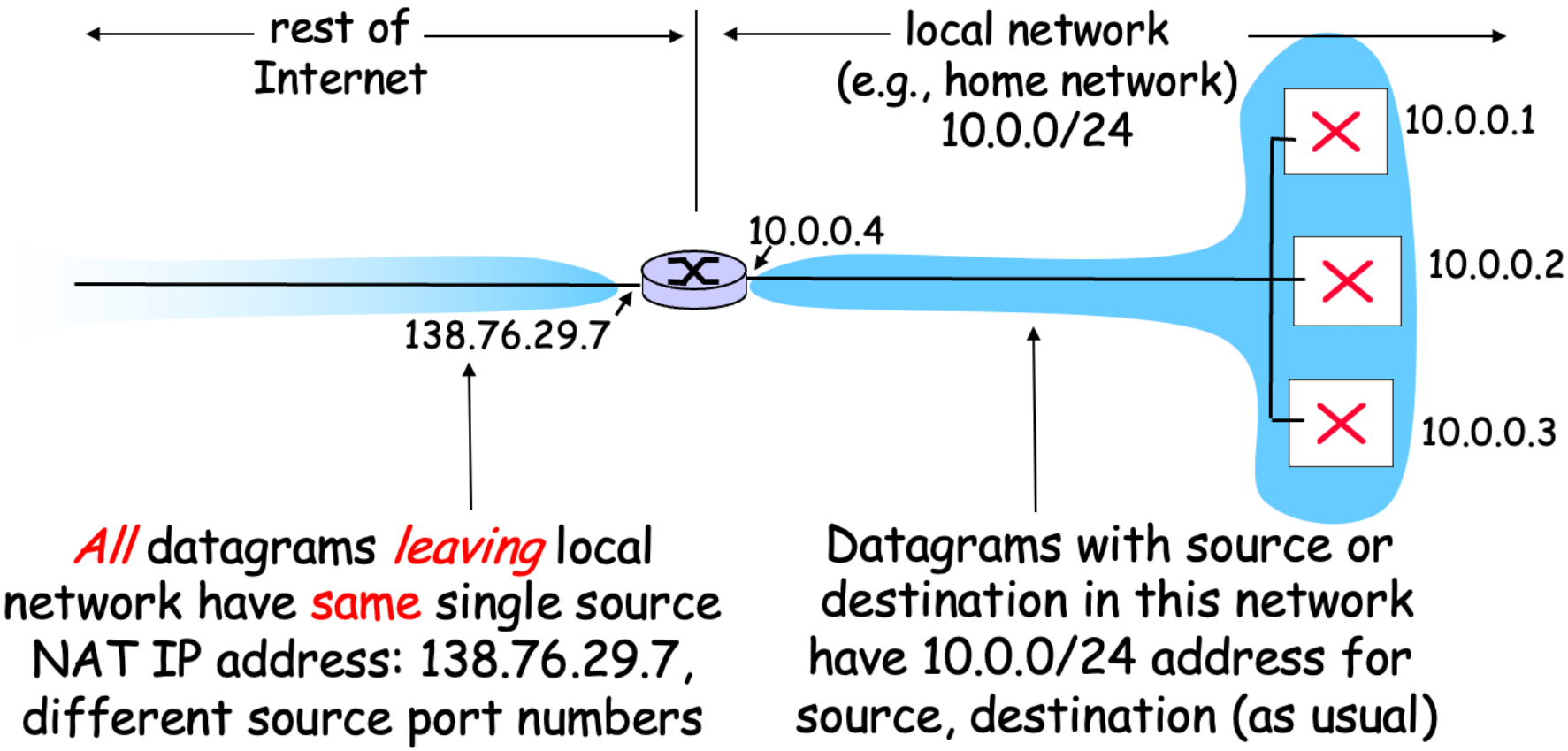
Difference between IPv4 and IPv6

IPv4	IPv6
• IPv4 addresses are 32 bit length.	• IPv6 addresses are 128 bit length.
• Fragmentation is done by sender and forwarding routers.	• Fragmentation is done only by sender.
• No packet flow identification.	• Packet flow identification is available within the IPv6 header using the Flow Label field.
• Checksum field is available in header	• No checksum field in header.
• Options fields are available in header.	• No option fields, but Extension headers are available.
• Address Resolution Protocol (ARP)is available to map IPv4 addresses to MAC addresses.	• Address Resolution Protocol (ARP) is replaced with Neighbour Discovery Protocol.
• Broadcast messages are available.	• Broadcast messages are not available.
• Manual configuration (Static) of IP addresses or DHCP (Dynamic configuration) is required to configure IP addresses.	• Auto-configuration of addresses is available.

Network Layer

NAT- Network Address Translation

NAT: Network Address Translation



Network Layer

NAT- Network Address Translation

NAT: Network Address Translation

- Motivation: 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
  - devices inside local net NOT explicitly addressable, visible by outside world (a security plus).

Network Layer

NAT- Network Address Translation

NAT: Network Address Translation

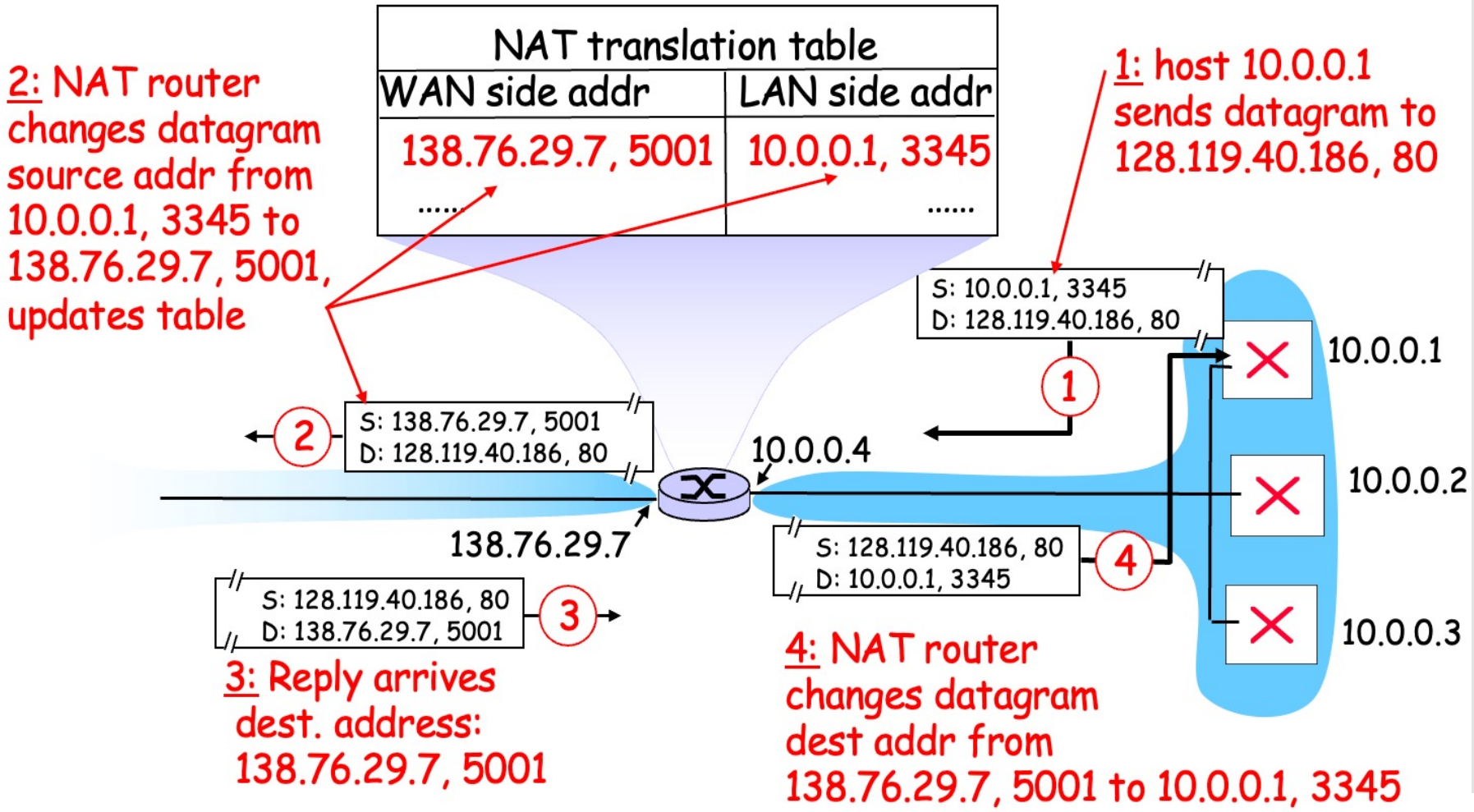
Implementation: NAT router must:

- *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

Network Layer

NAT- Network Address Translation

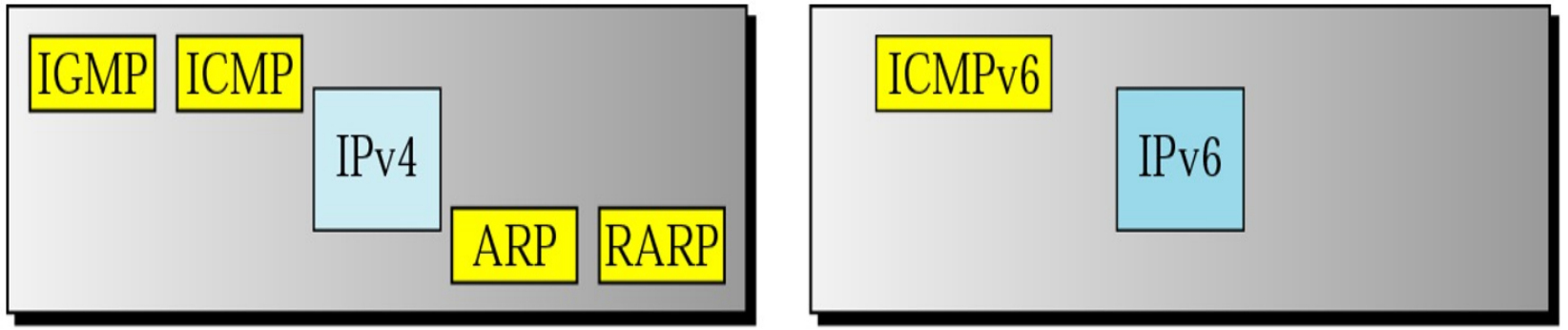
NAT: Network Address Translation



Network Layer

Network Layer Protocol

Comparison of network layers in version 4 and version 6



Network layer in version 4

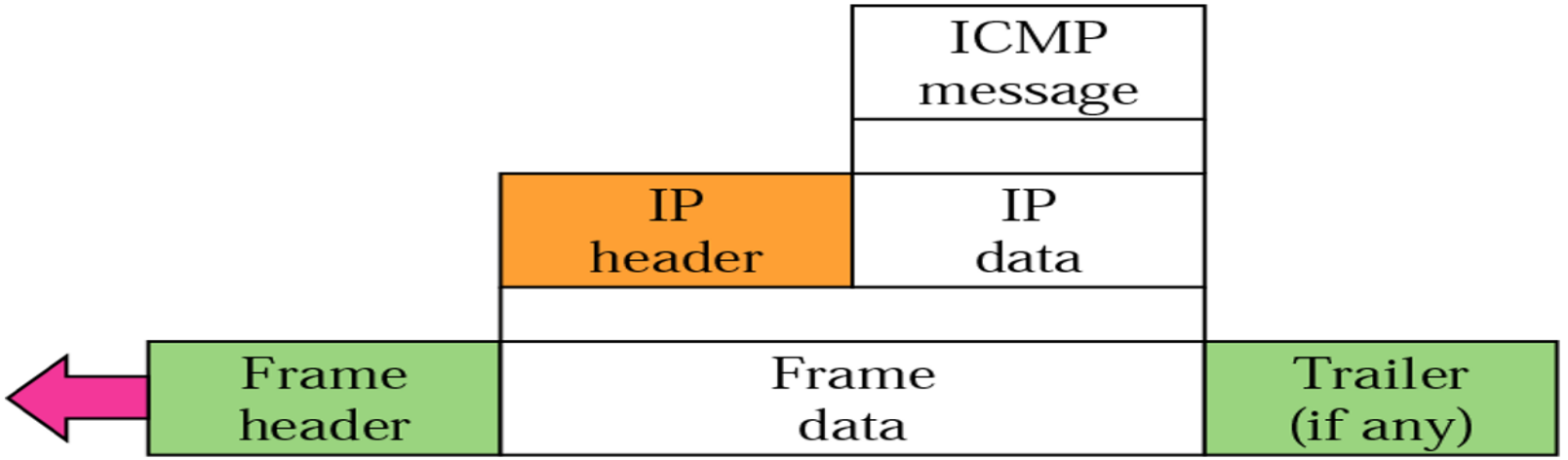
Network layer in version 6

Network Layer

ICMP (Internet Control Message Protocol)

Note:

ICMP always reports error messages to the original source.



Note:

There is no flow control or congestion control mechanism in IP.

Network Layer

ICMP (Internet Control Message Protocol)

21-2 ICMP

The IP protocol has no error-reporting or error-correcting mechanism. The IP protocol also lacks a mechanism for host and management queries. The Internet Control Message Protocol (ICMP) has been designed to compensate for the above two deficiencies. It is a companion to the IP protocol.

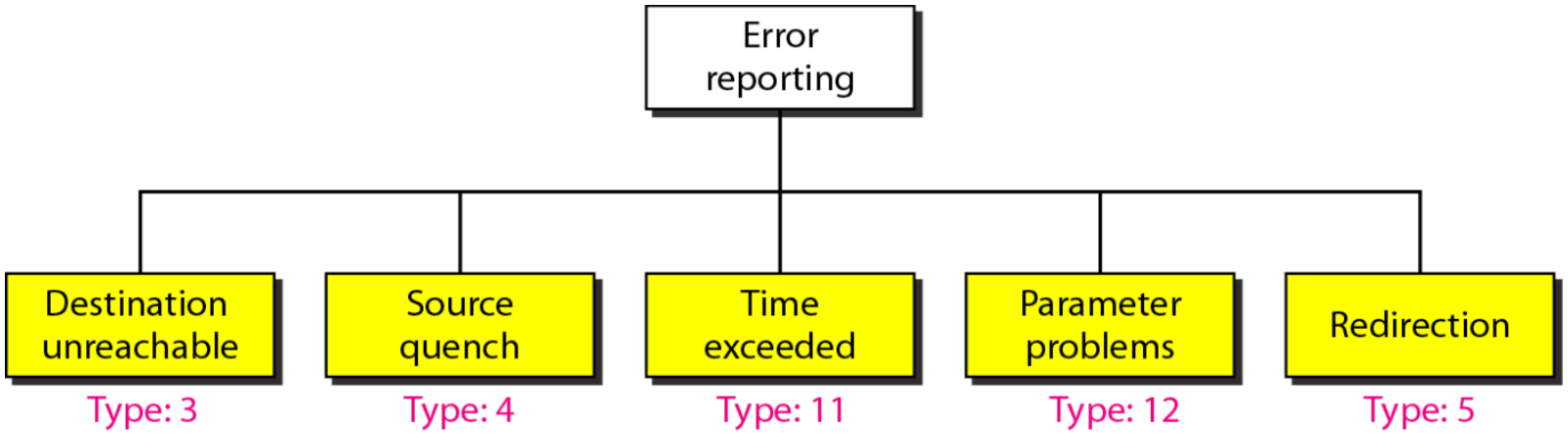
Topics discussed in this section:

Types of Messages
Message Format
Error Reporting and Query
Debugging Tools

Network Layer

(Error Reporting Message)

Error-reporting messages



Note

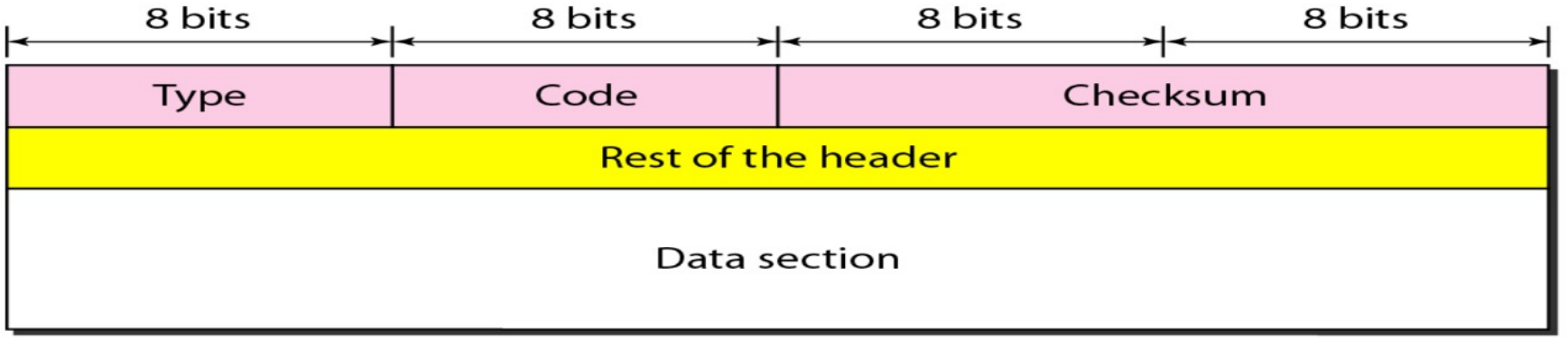
Important points about ICMP error messages:

- **No ICMP error message will be generated in response to a datagram carrying an ICMP error message.**
- **No ICMP error message will be generated for a fragmented datagram that is not the first fragment.**
- **No ICMP error message will be generated for a datagram having a multicast address.**
- **No ICMP error message will be generated for a datagram having a special address such as 127.0.0.0 or 0.0.0.0.**

Network Layer

ICMP (General Format)

General format of ICMP messages



Types of Messages

ICMP messages are divided into two broad categories:

Error-reporting messages and
Query messages

Code Field

The code field specifies the reason for the particular message type

Data Section

The data section in error messages carries information for finding the original packet that had the error.
In query messages, the data section carries extra information based on the type of the query.

Network Layer

IGMP (Internet Group Message Protocol)

21-3 IGMP

The IP protocol can be involved in two types of communication: unicasting and multicasting. The Internet Group Management Protocol (IGMP) is one of the necessary, but not sufficient, protocols that is involved in multicasting. IGMP is a companion to the IP protocol.

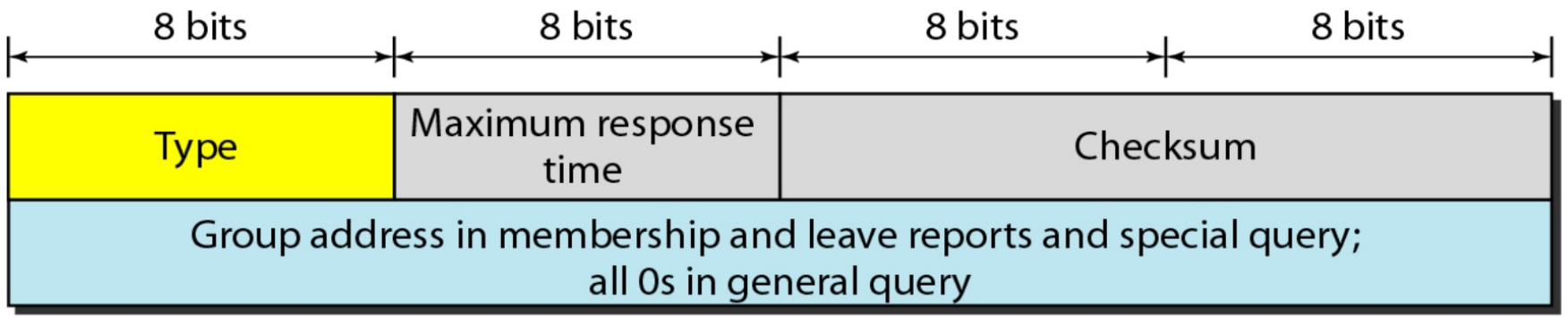
Topics discussed in this section:

Group Management
IGMP Messages and IGMP Operation
Encapsulation

Network Layer

IGMP (Message Format)

IGMP message format



Type

This 8-bit field defines the type of message, as shown in Table. The value of the type is shown in both hexadecimal and binary notation.

Type	*Value*
General or special query	0x11 or 00010001
Membership report	0x16 or 00010110
Leave report	0x17 or 00010111

Maximum Response Time

This 8-bit field defines the amount of time in which a query must be answered

Checksum This is a 16-bit field carrying the checksum. The checksum is calculated over the 8-byte message.

Group address

The value of this field is 0 for a general query message. The value defines the groupid (multicast address of the group) in the special query, the membership report, and the leave report messages.

Network Layer

IGMP (Operations)

ICMP: Internet Control Message Protocol

- used by hosts & routers to communicate network-level information
  - error reporting: unreachable host, network, port, protocol
  - echo request/reply (used by ping)
- network-layer "above" IP:
  - ICMP msgs carried in IP datagrams
- ICMP message: type, code plus first 8 bytes of IP datagram causing error

Type	Code	description
0	0	echo reply (ping)
3	0	dest. network unreachable
3	1	dest host unreachable
3	2	dest protocol unreachable
3	3	dest port unreachable
3	6	dest network unknown
3	7	dest host unknown
4	0	source quench (congestion control - not used)
8	0	echo request (ping)
9	0	route advertisement
10	0	router discovery
11	0	TTL expired
12	0	bad IP header

Network Layer

(Traceroute and ICMP)

Traceroute and ICMP

- Source sends series of UDP segments to dest
  - First has TTL =1
  - Second has TTL=2, etc.
- When nth datagram arrives to nth router:
  - Router discards datagram
  - And sends to source an ICMP message (type 11, code 0)
  - Message includes name of router& IP address
- When ICMP message arrives, source calculates RTT
- Traceroute does this 3 times

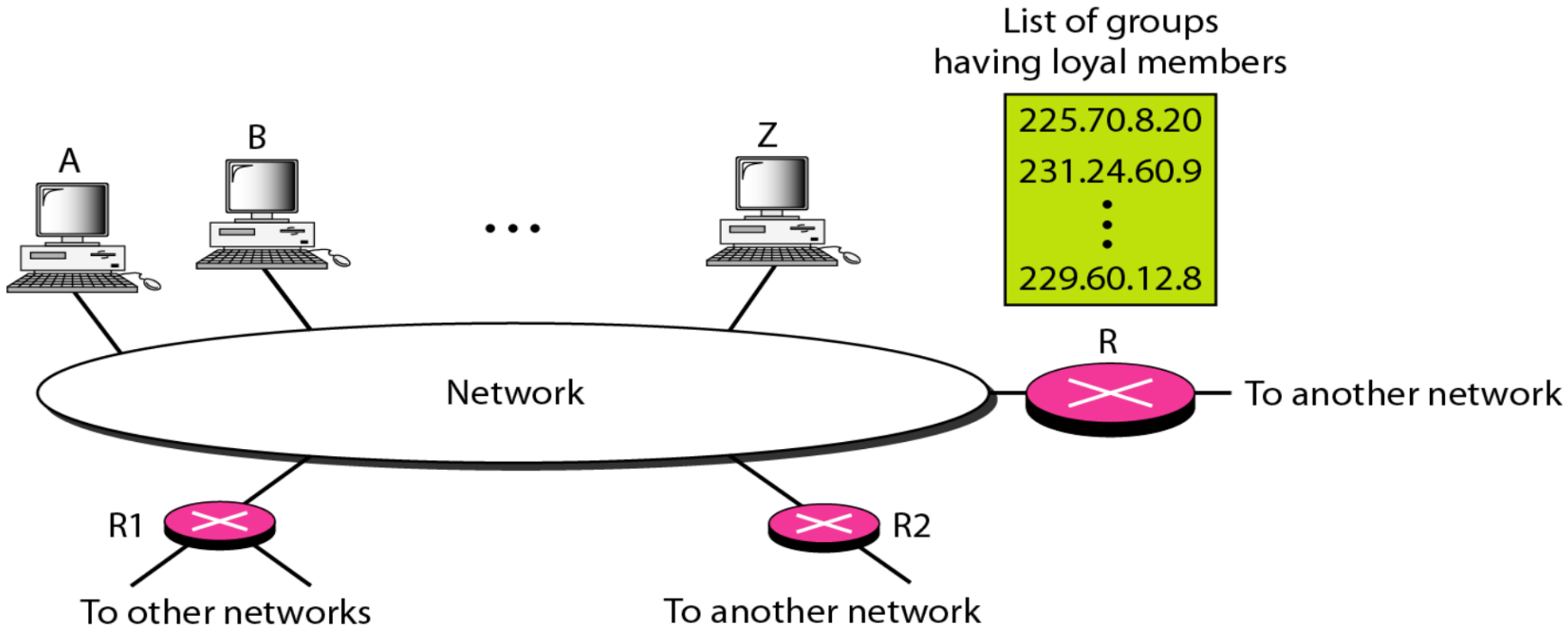
Stopping criterion

- UDP segment eventually arrives at destination host
- Destination returns ICMP "host unreachable" packet (type 3, code 3)
- When source gets this ICMP, stops.

Network Layer

IGMP (Operations)

IGMP operation



IGMP operation

Joining a Group

Leaving a Group

Monitoring Membership

Delayed Response

Network Layer

IGMP (Operations)

Note

In IGMP, a membership report is sent twice, one after the other.

Note

The general query message does not define a particular group.

Network Layer

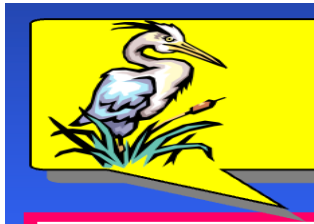
Chapter 4: Network Layer

- 4. 1 Introduction
- 4.2 Virtual circuit and datagram networks
- 4.3 What's inside a router
- 4.4 IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- 4.5 Routing algorithms
  - Link state
  - Distance Vector
  - Hierarchical routing
- 4.6 Routing in the Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Broadcast and multicast routing

Network Layer
(Unicast Routing)

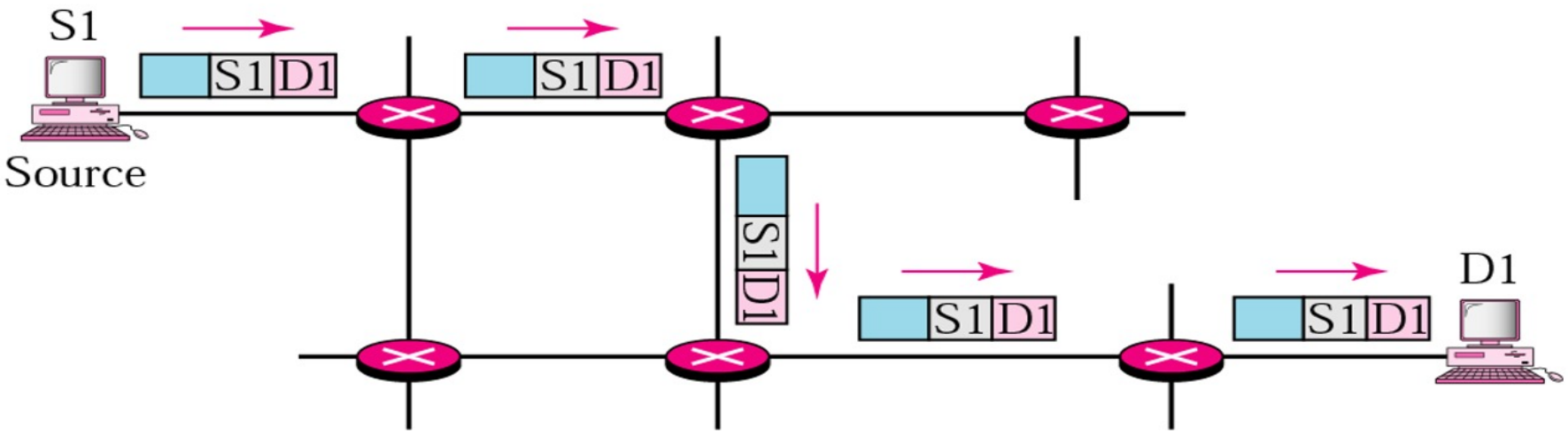
Unicast and Multicast Routing:

Routing Protocols



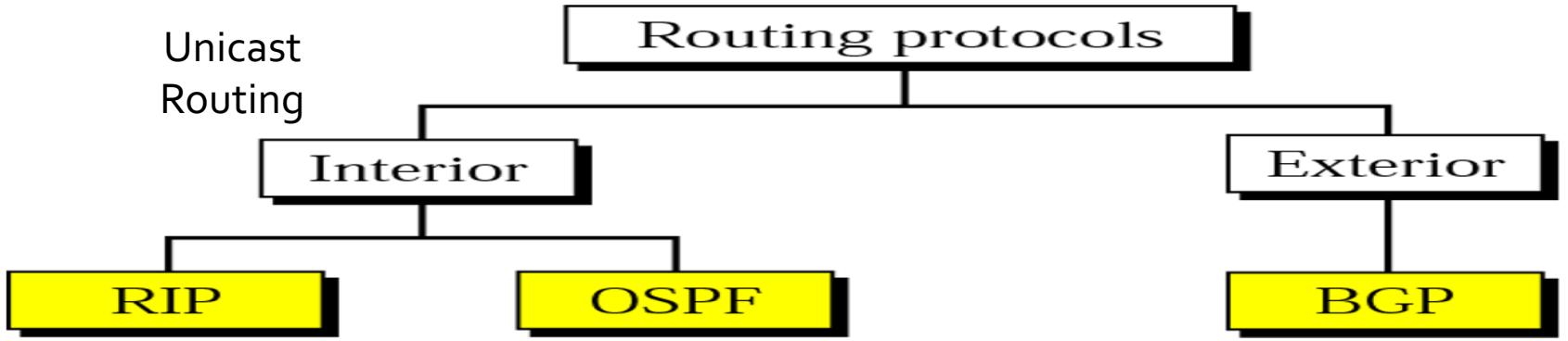
Note:

In unicast routing, the router forwards the received packet through only one of its ports.



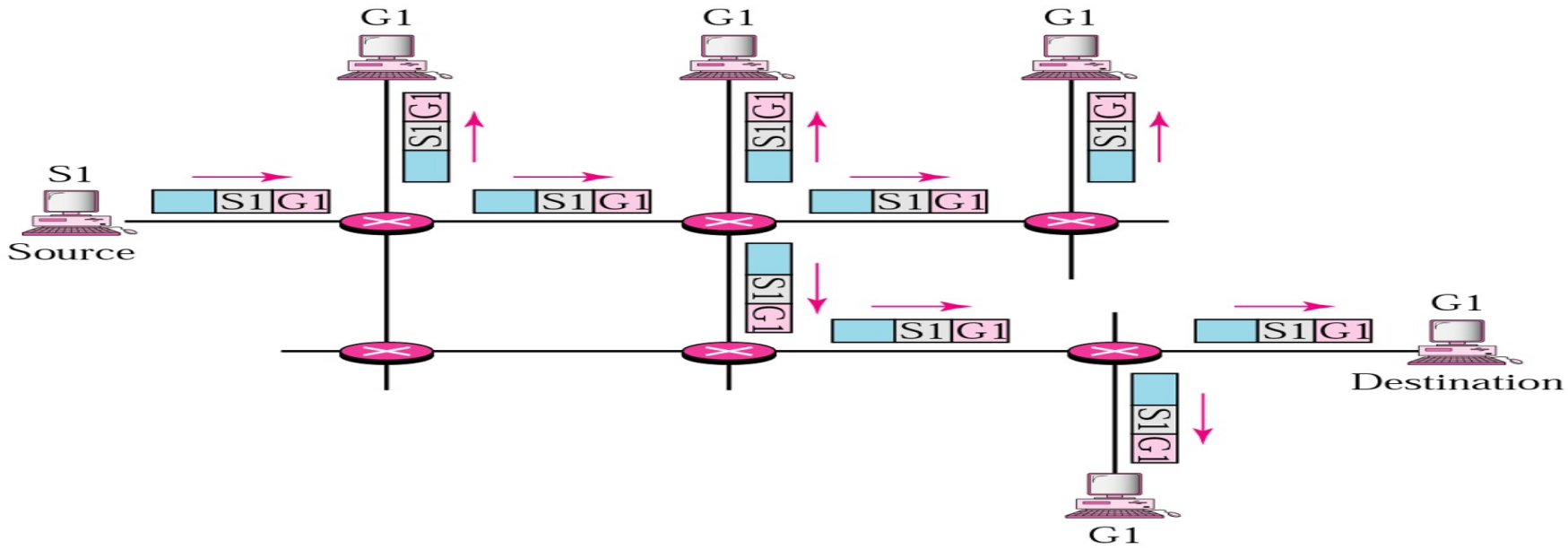
Network Layer (Multicast Routing)

Unicast Routing



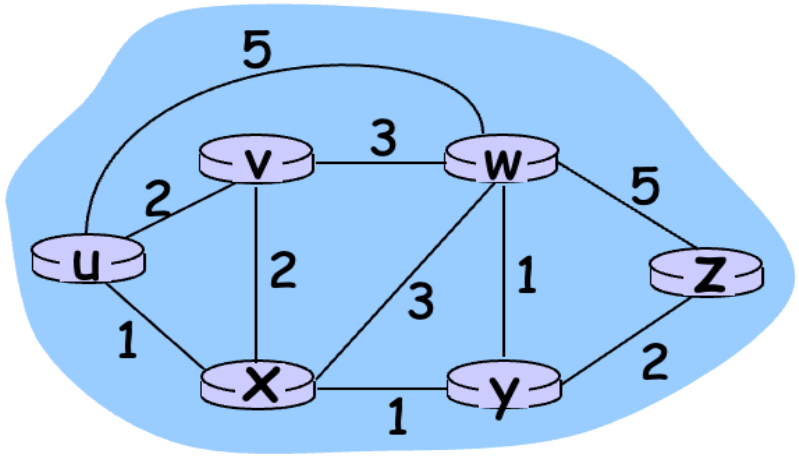
Note:

In multicast routing, the router may forward the received packet through several of its ports.



Network Layer (Graph Abstraction)

Graph abstraction



Graph: G = (N,E)

N = set of routers = { u, v, w, x, y, z }

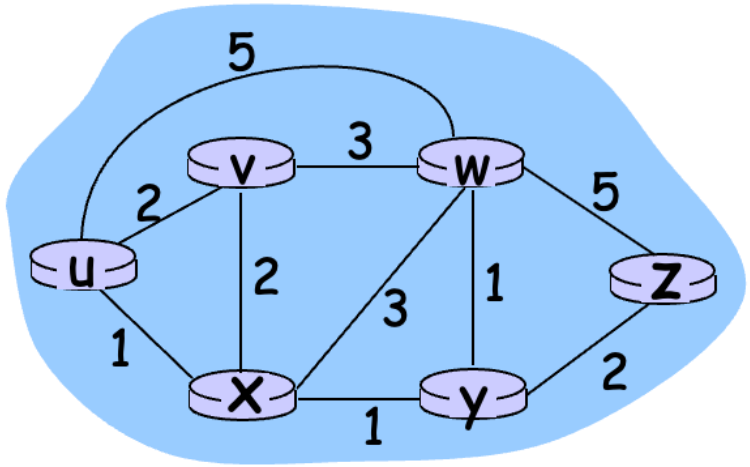
E = set of links ={ (u,v), (u,x), (v,x), (v,w), (x,w), (x,y), (w,y), (w,z), (y,z) }

Remark: Graph abstraction is useful in other network contexts

Example: P2P, where N is set of peers and E is set of TCP connections

Network Layer (Graph Abstraction-Costs)

Graph abstraction: costs



- c(x,x') = cost of link (x,x')
  - e.g., c(w,z) = 5
- cost could always be 1, or inversely related to bandwidth, or inversely related to congestion

Cost of path $(x_1, x_2, x_3, \ldots, x_p) = c(x_1,x_2) + c(x_2,x_3) + \ldots + c(x_{p-1},x_p)$

Question: What's the least-cost path between u and z ?

Routing algorithm: algorithm that finds least-cost path

Network Layer (Routing Algorithm Classification)

Routing Algorithm classification

Global or decentralized information?

Global:

- all routers have complete topology, link cost info
- "link state" algorithms

Decentralized:

- router knows physically-connected neighbors, link costs to neighbors
- iterative process of computation, exchange of info with neighbors
- "distance vector" algorithms

Static or dynamic?

Static:

- routes change slowly over time

Dynamic:

- routes change more quickly
  - periodic update
  - in response to link cost changes

Network Layer (Link State Routing- Dijkstra's Algorithm)

A Link-State Routing Algorithm

Dijkstra's algorithm

- net topology, link costs known to all nodes
  - accomplished via "link state broadcast"
  - all nodes have same info
- computes least cost paths from one node ('source") to all other nodes
  - gives forwarding table for that node
- iterative: after k iterations, know least cost path to k dest.'s

Notation:

- c(x,y): link cost from node x to y; = ∞ if not direct neighbors
- D(v): current value of cost of path from source to dest. v
- p(v): predecessor node along path from source to v
- N': set of nodes whose least cost path definitively known

Network Layer (Dijkstra's Algorithm)

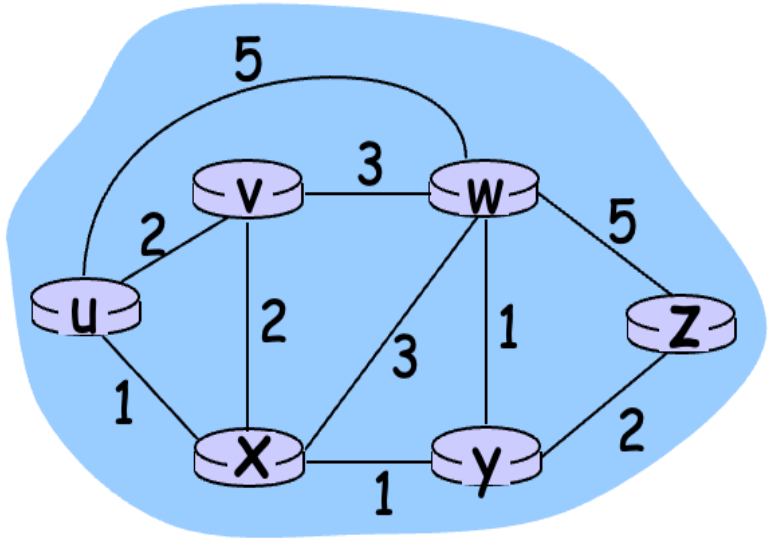
Dijsktra's Algorithm

```
1  Initialization:
2    N' = {u}
3    for all nodes v
4      if v adjacent to u
5          then D(v) = c(u,v)
6      else D(v) = ∞
7
8   Loop
9     find w not in N' such that D(w) is a minimum
10     add w to N'
11     update D(v) for all v adjacent to w and not in N' :
12        D(v) = min( D(v), D(w) + c(w,v) )
13     /* new cost to v is either old cost to v or known
14      shortest path cost to w plus cost from w to v */
15  until all nodes in N'
```

Dijkstra's algorithm: example

Network Layer
(Dijkstra's Algorithm)

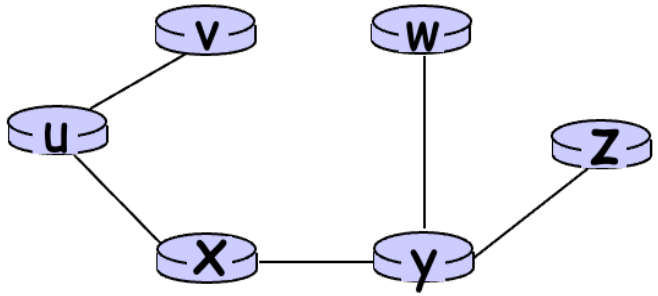
Step	N'	D(v),p(v)	D(w),p(w)	D(x),p(x)	D(y),p(y)	D(z),p(z)
0	u	2,u	5,u	1,u	∞	∞
1	ux	2,u	4,x		2,x	∞
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			4,y
4	uxyvw					4,y
5	uxyvwz					



Network Layer (Dijkstra's Algorithm)

Dijkstra's algorithm: example (2)

Resulting shortest-path tree from u:



Resulting forwarding table in u:

destination	link
v	(u,v)
x	(u,x)
y	(u,x)
w	(u,x)
z	(u,x)

Algorithm complexity: n nodes

- each iteration: need to check all nodes, w, not in N
- $n(n+1)/2$ comparisons: $O(n^2)$
- more efficient implementations possible: O(nlogn)

Network Layer (Distance Vector Routing- Bellman Ford Algorithm)

Distance Vector Algorithm

Bellman-Ford Equation (dynamic programming)

Define

$d_x(y)$:= cost of least-cost path from x to y

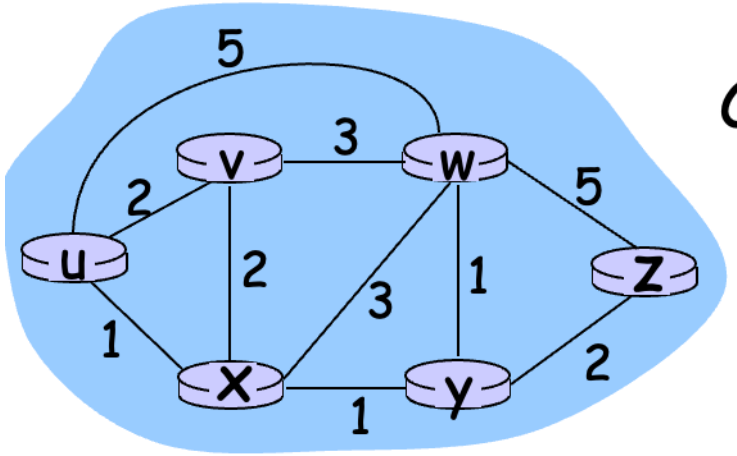
Then

$$d_x(y) = \min_v \{c(x,v) + d_v(y)\}$$

where min is taken over all neighbors v of x

Network Layer
(Bellman Ford Algorithm)

Bellman-Ford example



Clearly, $d_v(z) = 5$, $d_x(z) = 3$, $d_w(z) = 3$

B-F equation says:

$$d_u(z) = \min \{ c(u,v) + d_v(z), \; c(u,x) + d_x(z), \; c(u,w) + d_w(z) \}$$
$$= \min \{2 + 5, \; 1 + 3, \; 5 + 3\} = 4$$

Node that achieves minimum is next hop in shortest path ➜ forwarding table

Network Layer (Distance Vector Algorithm)

Distance Vector Algorithm

- $D_x(y)$ = estimate of least cost from x to y
- Distance vector: $D_x = [D_x(y): y \in N]$
- Node x knows cost to each neighbor v: $c(x,v)$
- Node x maintains $D_x = [D_x(y): y \in N]$
- Node x also maintains its neighbors' distance vectors
  - For each neighbor v, x maintains $D_v = [D_v(y): y \in N]$

Network Layer (Distance Vector Algorithm)

Distance vector algorithm (4)

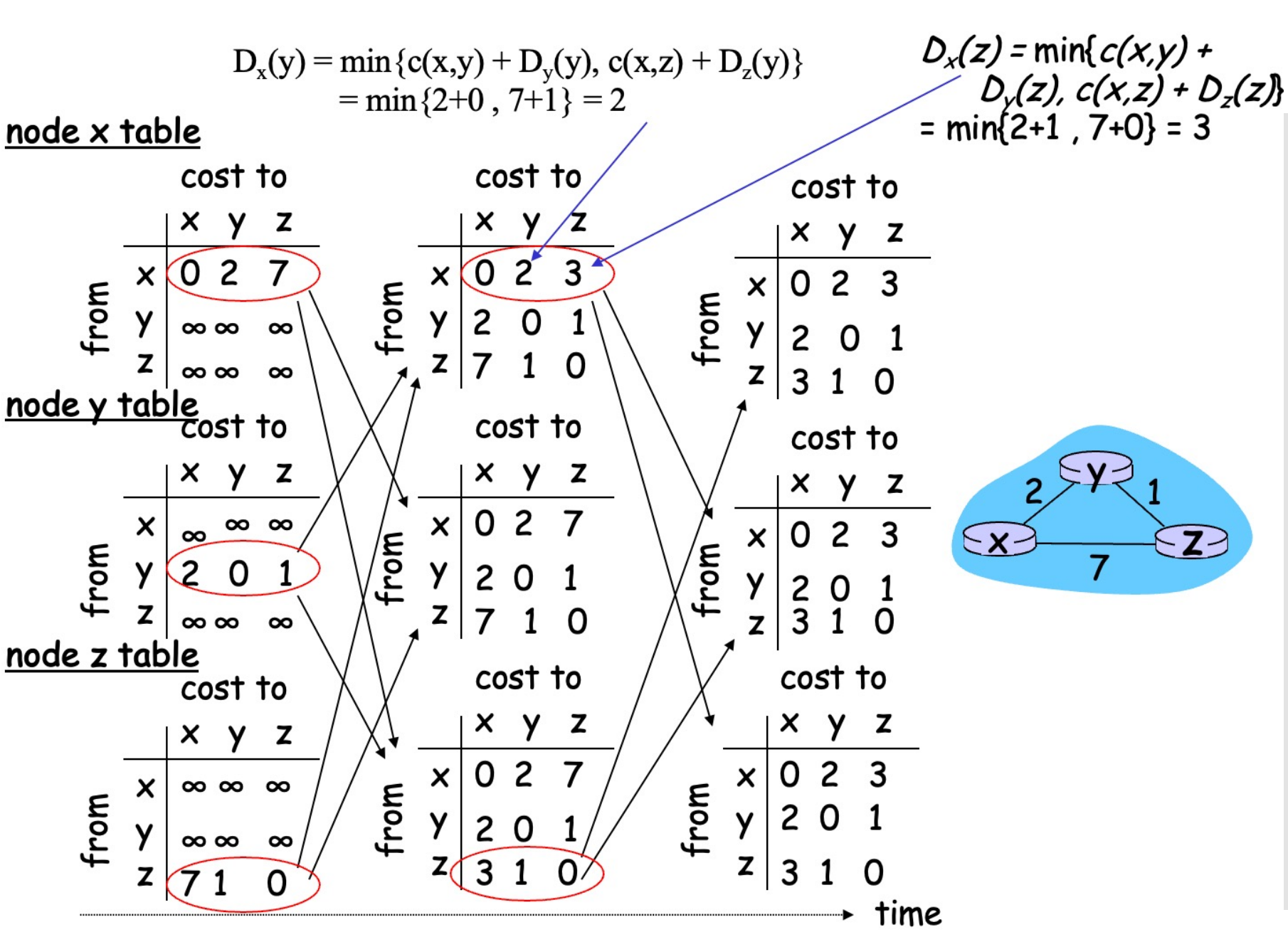
Basic idea:

- Each node periodically sends its own distance vector estimate to neighbors
- When a node x receives new DV estimate from neighbor, it updates its own DV using B-F equation:

$$D_x(y) \leftarrow min_v\{c(x,v) + D_v(y)\} \quad \text{for each node } y \in N$$

- Under minor, natural conditions, the estimate $D_x(y)$ *converge to the actual least cost* $d_x(y)$

Network Layer (Distance Vector Algorithm)

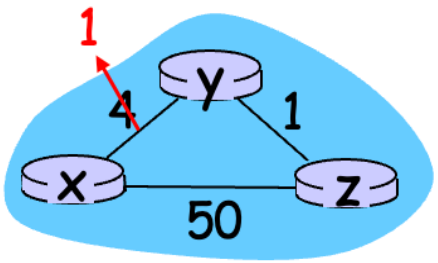


Network Layer (Distance Vector Algorithm)

Distance Vector: link cost changes

Link cost changes:

- node detects local link cost change
- updates routing info, recalculates distance vector
- if DV changes, notify neighbors



"good news travels fast"

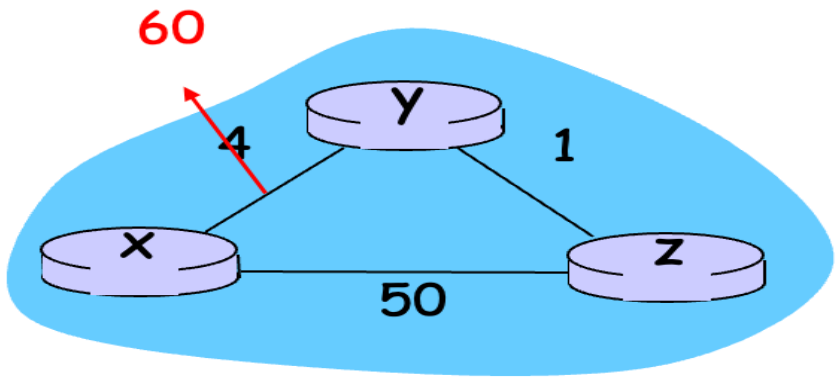
At time t_0, y detects the link-cost change, updates its DV, and informs its neighbors.

At time t_1, z receives the update from y and updates its table. It computes a new least cost to x and sends its neighbors its DV.

At time t_2, y receives z's update and updates its distance table. y's least costs do not change and hence y does *not* send any message to z.

Network Layer (Distance Vector Algorithm)

Distance Vector: link cost changes



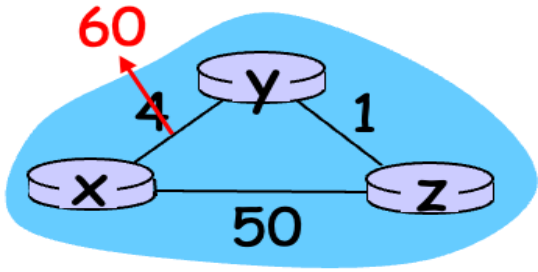
What happens now ?

Network Layer (Distance Vector Algorithm)

Distance Vector: link cost changes

Link cost changes:

- good news travels fast
- bad news travels slow - "count to infinity" problem!
- 44 iterations before algorithm stabilizes: see text



Poissoned reverse:

- If Z routes through Y to get to X :
  - Z tells Y its (Z's) distance to X is infinite (so Y won't route to X via Z)
- will this completely solve count to infinity problem?

Network Layer
(Comparison of DV and LS Routing)

Comparison of LS and DV algorithms

Message complexity

- LS: with n nodes, E links, O(nE) msgs sent
- DV: exchange between neighbors only
  - convergence time varies

Speed of Convergence

- LS: $O(n^2)$ algorithm requires O(nE) msgs
  - may have oscillations
- DV: convergence time varies
  - may be routing loops
  - count-to-infinity problem

Robustness: what happens if router malfunctions?

LS:

- node can advertise incorrect *link* cost
- each node computes only its *own* table

DV:

- DV node can advertise incorrect *path* cost
- each node's table used by others
  - error propagate thru network

Network Layer (Host and Network Specific Routing)

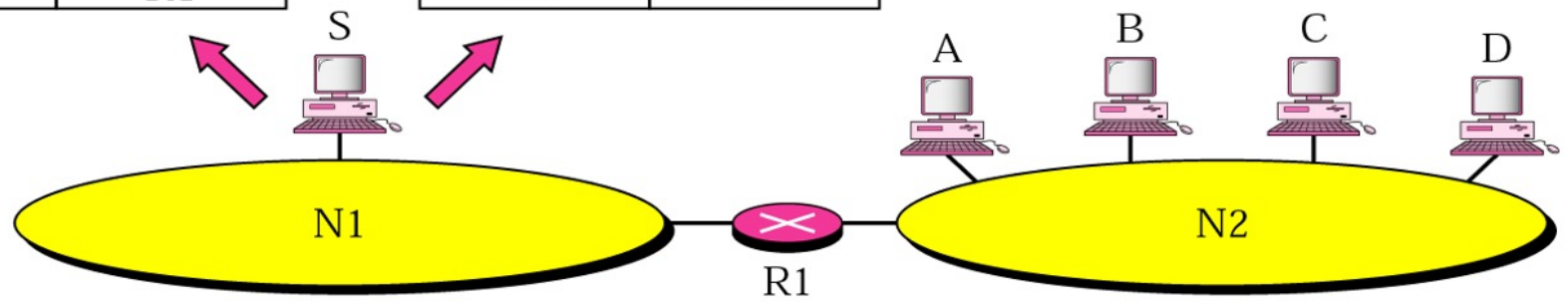
Host and Network Specific Routing

Routing table for host S based on host-specific routing

Destination	Next Hop
A	R1
B	R1
C	R1
D	R1

Routing table for host S based on network-specific routing

Destination	Next Hop
N2	R1



Network Layer (Next Hop Routing)

Routing table for host A

Destination	Route
Host B	R1, R2, Host B

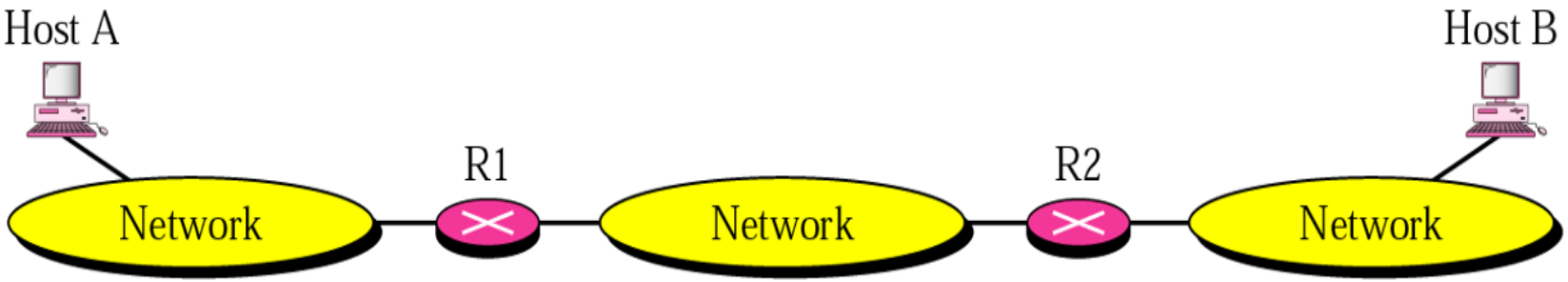
Routing table for R1

Destination	Route
Host B	R2, Host B

Routing table for R2

Destination	Route
Host B	Host B

a. Routing tables based on route



Routing table for host A

Destination	Next Hop
Host B	R1

Routing table for R1

Destination	Next Hop
Host B	R2

Routing table for R2

Destination	Next Hop
Host B	—

b. Routing tables based on next hop

Network Layer (Hierarchical Routing)

Hierarchical Routing

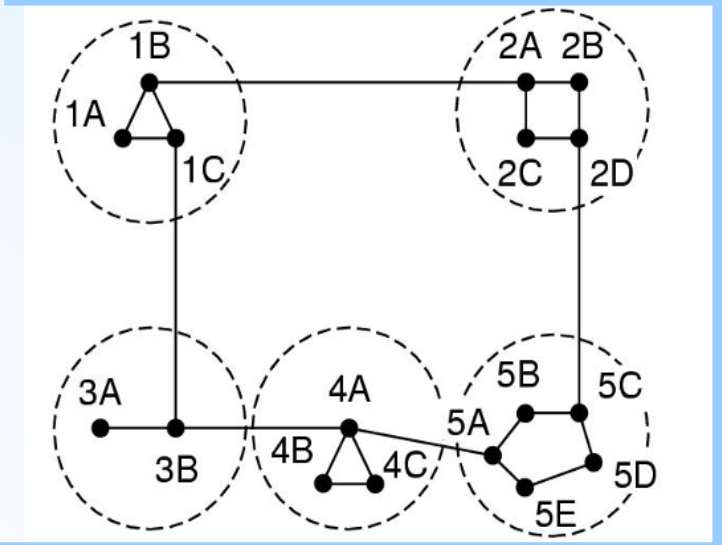
- aggregate routers into regions, "autonomous systems" (AS)
- routers in same AS run same routing protocol
  - "intra-AS" routing protocol
  - routers in different AS can run different intra-AS routing protocol

Gateway router

- Direct link to router in another AS

Network Layer (Hierarchical Routing)

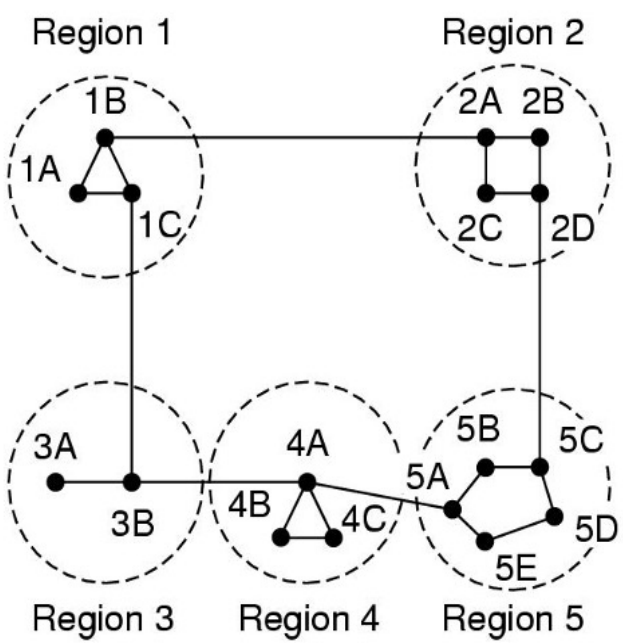
Hierarchical routing



- When network size increases…
  - Larger stables
  - More CPU time needed to compute …
  - More bandwidth needed

Hierarchical routing

Network Layer (Hierarchical Routing)



(a)

Full table for 1A

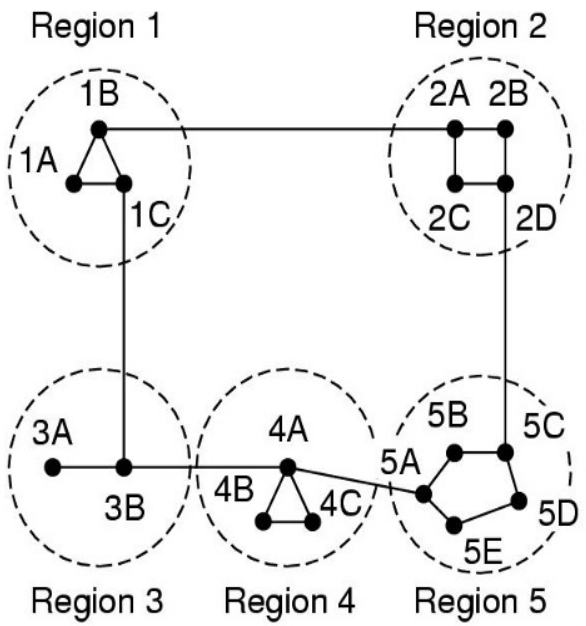
Dest.	Line	Hops
1A	–	–
1B	1B	1
1C	1C	1
2A	1B	2
2B	1B	3
2C	1B	3
2D	1B	4
3A	1C	3
3B	1C	2
4A	1C	3
4B	1C	4
4C	1C	4
5A	1C	4
5B	1C	5
5C	1B	5
5D	1C	6
5E	1C	5

(b)

- Solution?

Hierarchical routing

Network Layer (Hierarchical Routing)



- Routers grouped in regions
- Each routers knows how to reach:
  - Other routers in its own group
  - Other regions

Full table for 1A

Dest.	Line	Hops
1A	–	–
1B	1B	1
1C	1C	1
2A	1B	2
2B	1B	3
2C	1B	3
2D	1B	4
3A	1C	3
3B	1C	2
4A	1C	3
4B	1C	4
4C	1C	4
5A	1C	4
5B	1C	5
5C	1B	5
5D	1C	6
5E	1C	5

(b)

Hierarchical table for 1A

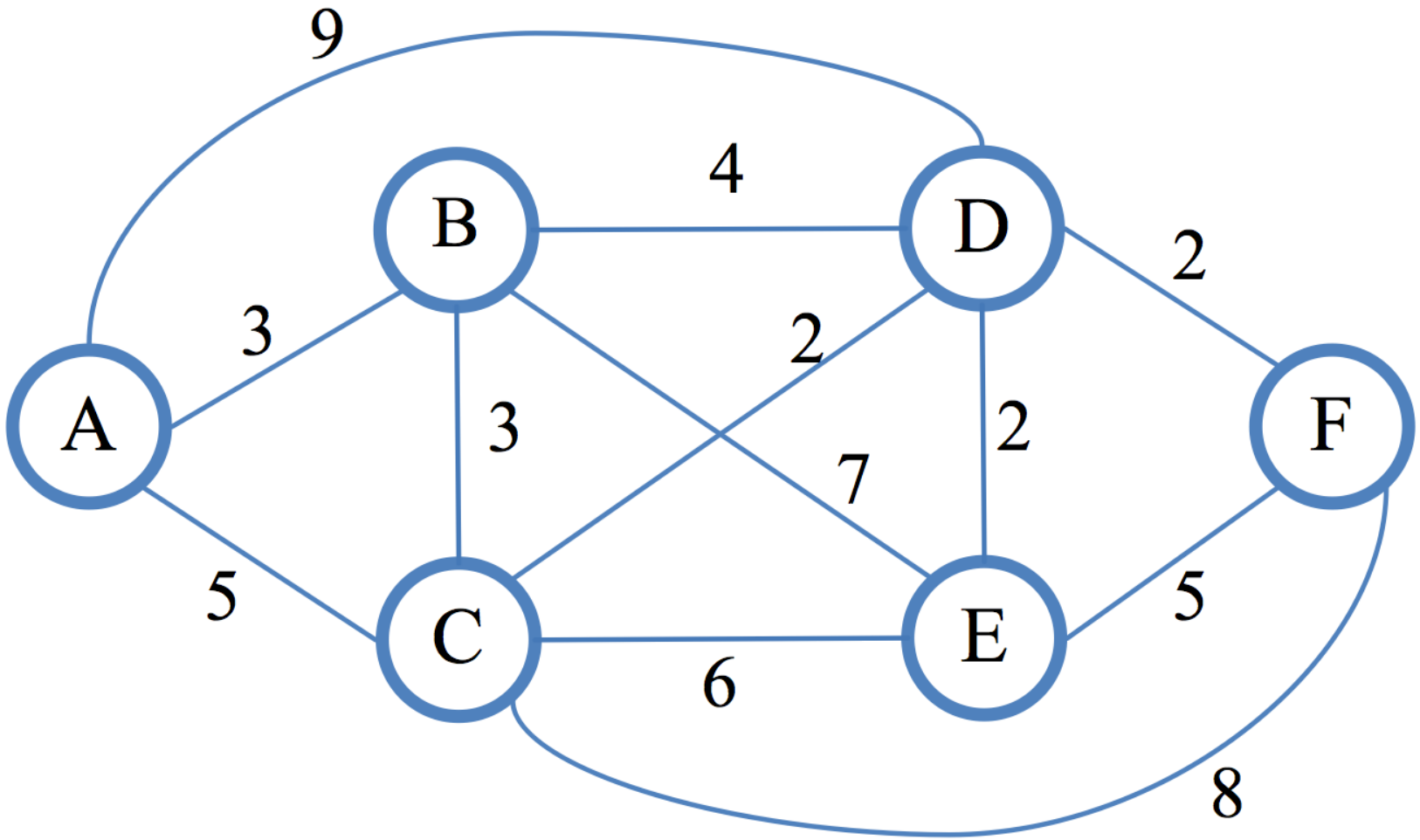
Dest.	Line	Hops
1A	–	–
1B	1B	1
1C	1C	1
2	1B	2
3	1C	2
4	1C	3
5	1C	4

\+ Smaller tables

– Longer paths

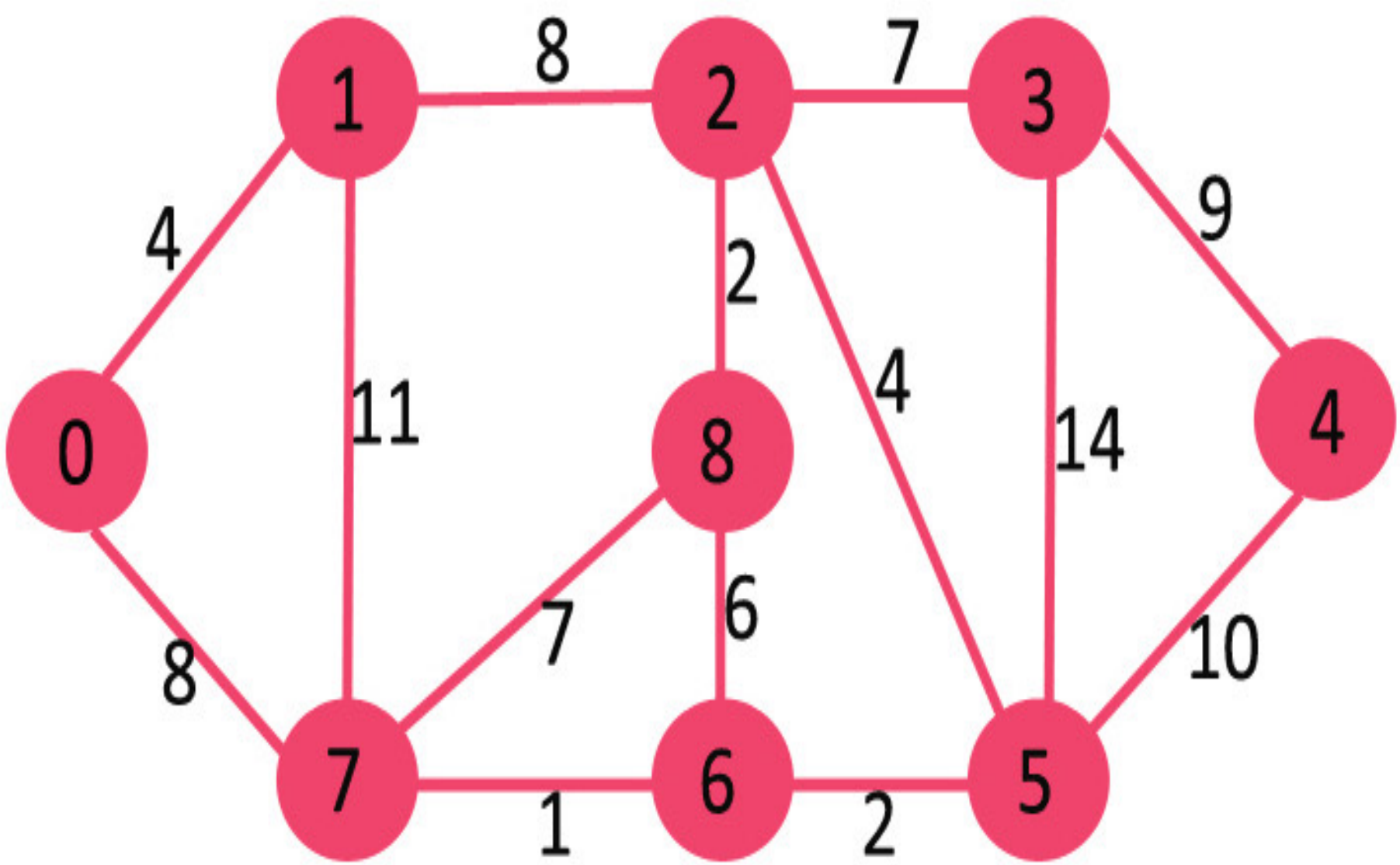
Solve this example

(Shortest Path A-F)



Solve this example

(Shortest Path 0-4)



Network Layer

Chapter 4: Network Layer

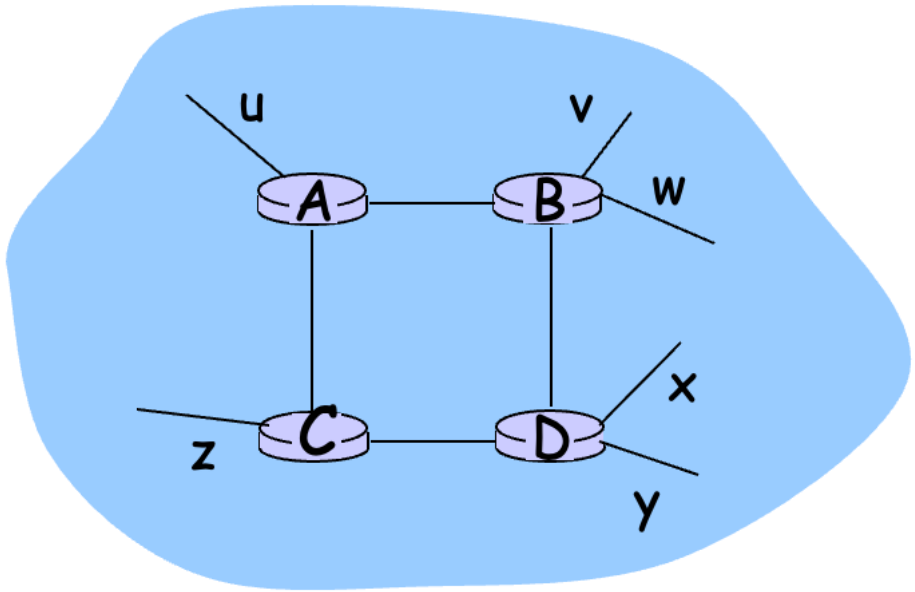
- 4. 1 Introduction
- 4.2 Virtual circuit and datagram networks
- 4.3 What's inside a router
- 4.4 IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- 4.5 Routing algorithms
  - Link state
  - Distance Vector
  - Hierarchical routing
- 4.6 Routing in the Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Broadcast and multicast routing

Network Layer

(RIP-Routing Information Protocol)

RIP (Routing Information Protocol)

- Distance vector algorithm
- Included in BSD-UNIX Distribution in 1982
- Distance metric: # of hops (max = 15 hops)



From router A to subsets:

destination	hops
u	1
v	2
w	2
x	3
y	3
z	2

Network Layer (RIP-Routing Information Protocol)

RIP advertisements

- Distance vectors: exchanged among neighbors every 30 sec via Response Message (also called advertisement)
- Each advertisement: list of up to 25 destination nets within AS

- RIP is a Dynamic Routing Protocol that uses hop count as a routing metric to find the best path between the source and the destination network.

- It is a distance vector routing protocol that has an AD value of 120 and uses port number 520.

Network Layer

(RIP-Routing Information Protocol)

Hop Count

- Hop count is the number of routers occurring between the source and destination network.
- The path with the lowest hop count is considered the best route to reach a network and, therefore placed in the routing table.
- RIP prevents routing loops by limiting the number of hops allowed in a path from source to destination.
- The maximum hop count allowed for RIP is 15, and a hop count of 16 is considered as network unreachable.

Network Layer

(RIP-Routing Information Protocol)

RIP: Link Failure and Recovery

If no advertisement heard after 180 sec --> neighbor/link declared dead

- routes via neighbor invalidated
- new advertisements sent to neighbors
- neighbors in turn send out new advertisements (if tables changed)
- link failure info quickly propagates to entire net
- poison reverse used to prevent ping-pong loops (infinite distance = 16 hops)

Network Layer

(RIP-Routing Information Protocol)

Features of RIP

1. Updates of the network are exchanged periodically.
2. Updates (routing information) are always broadcast.
3. Full routing tables are sent in updates.
4. Routers always trust routing information received from neighbour routers. This is also known as *Routing on* rumors.

RIP versions:

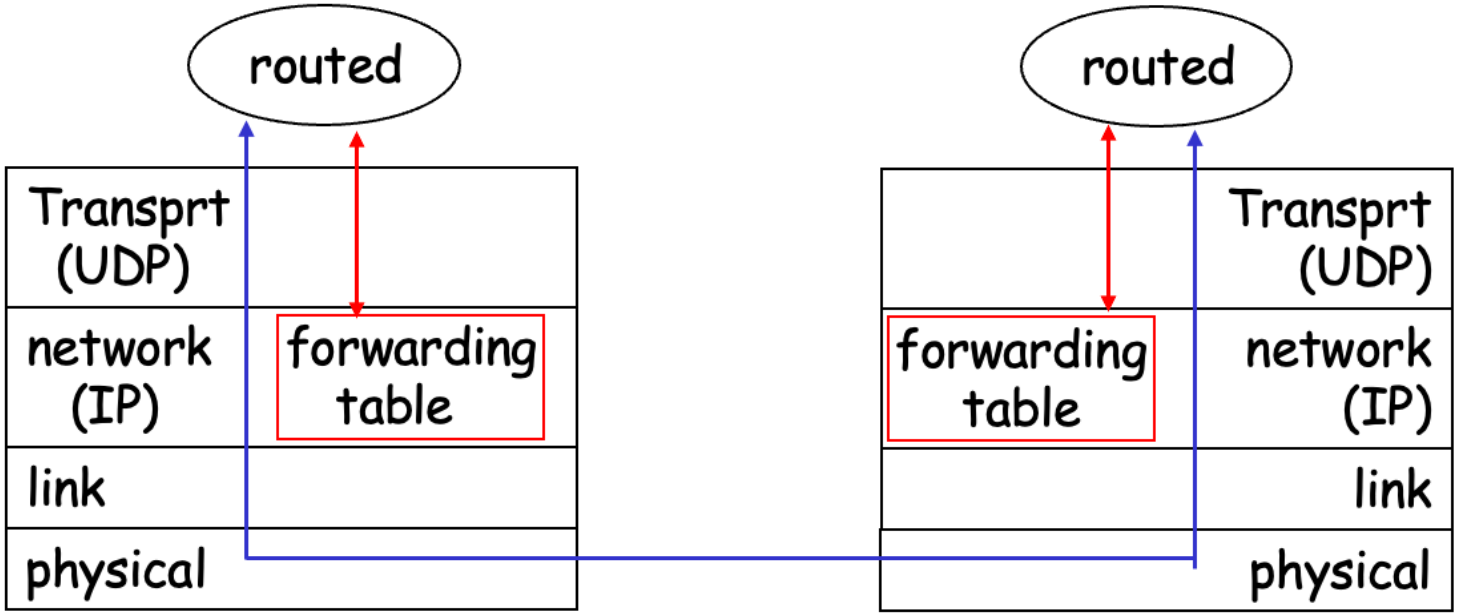
There are three versions of routing information protocol – **RIP Version1**, **RIP Version2**, and **RIPng**.

Network Layer

(RIP-Routing Information Protocol)

RIP Table processing

- RIP routing tables managed by application-level process called route-d (daemon)
- advertisements sent in UDP packets, periodically repeated



Network Layer

(RIP-Routing Information Protocol)

Advantages of RIP :

- **Simplicity:** RIP is a relatively simple protocol to configure and manage, making it an ideal choice for small to medium-sized networks with limited resources.
- **Easy implementation:** RIP is easy to implement, as it does not require much technical expertise to set up and maintain.
- **Convergence:** RIP is known for its fast convergence time, meaning it can adapt to network topology changes and route packets efficiently.
- **Automatic updates:** RIP automatically updates routing tables at regular intervals, ensuring that the most up-to-date information is being used to route packets.
- **Low bandwidth overhead:** RIP uses a relatively low amount of bandwidth to exchange routing information, making it an ideal choice for networks with limited bandwidth.
- **Compatibility:** RIP is compatible with many different types of routers and network devices, making it easy to integrate into existing networks.

Network Layer

(RIP-Routing Information Protocol)

Disadvantages of RIP:

•**Limited scalability:** RIP has limited scalability and may not be the best choice for larger networks with complex topologies. RIP can only support up to 15 hops, which may not be sufficient for larger networks.

•**Slow convergence:** While RIP is known for its fast convergence time, it can be slower to converge than other routing protocols. This can lead to delays and inefficiencies in network performance.

•**Routing loops:** RIP can sometimes create routing loops, which can cause network congestion and reduce overall network performance.

•**Limited support for load balancing:** RIP does not support sophisticated load balancing, which can result in suboptimal routing paths and uneven network traffic distribution.

•**Security vulnerabilities:** RIP does not provide any native security features, making it vulnerable to attacks such as spoofing and tampering.

•**Inefficient bandwidth use:** RIP uses a lot of bandwidth for periodic updates, which can be inefficient in networks with limited bandwidth.

Network Layer

(OSPF- Open Shortest Path First Protocol)

(OSPF- Open Shortest Path First Protocol)

- Open Shortest Path First (OSPF) is a link-state routing protocol that is used to find the best path between the source and the destination router using its own (Shortest Path First).
- OSPF is developed by the Internet Engineering Task Force (IETF) as one of the Interior Gateway Protocols (IGP), i.e, the protocol that aims at moving the packet within a large autonomous system or routing domain.
- It is a network layer protocol that works on protocol number 89 and uses AD value 110.
- OSPF uses multicast address 224.0.0.5 for normal communication and 224.0.0.6 for update to designated router(DR)/Backup Designated Router (BDR).

Network Layer

(OSPF- Open Shortest Path First Protocol)

OSPF supports/provides/advantages –

- Both IPv4 and IPv6 routed protocols
- Load balancing with equal-cost routes for the same destination
- VLSM and route summarization
- Unlimited hop counts
- Trigger updates for fast convergence
- A loop-free topology using SPF algorithm.
- Run-on most routers
- Classless protocol

Network Layer

(OSPF- Open Shortest Path First Protocol)

OSPF messages –

OSPF uses certain messages for the communication between the routers operating OSPF.

Hello message – These are keep-alive messages used for neighbor discovery /recovery. These are exchanged every 10 seconds. This includes the following information: Router I'd, Hello/dead interval, Area I'd, Router priority, DR and BDR IP address, and authentication data.

Database Description (DBD) – It is the OSPF route of the router. This contains the topology of an AS or an area (routing domain).

Link state request (LSR) – When a router receives DBD, it compares it with its own DBD. If the DBD received has some more updates than its own DBD, then LSR is being sent to its neighbor.

Link state update (LSU) – When a router receives LSR, it responds with an LSU message containing the details requested.

Link-state acknowledgment – This provides reliability to the link-state exchange process. It is sent as the acknowledgment of LSU.

Link state advertisement (LSA) – It is an OSPF data packet that contains link-state routing information, shared only with the routers to which adjacency has been formed.

Network Layer

(OSPF- Open Shortest Path First Protocol)

OSPF (Open Shortest Path First)

- "open": publicly available
- Uses Link State algorithm
  - LS packet dissemination
  - Topology map at each node
  - Route computation using Dijkstra's algorithm
- OSPF advertisement carries one entry per neighbor router
- Advertisements disseminated to entire AS (via flooding)
  - Carried in OSPF messages directly over IP (rather than TCP or UDP

Network Layer

(OSPF- Open Shortest Path First Protocol)

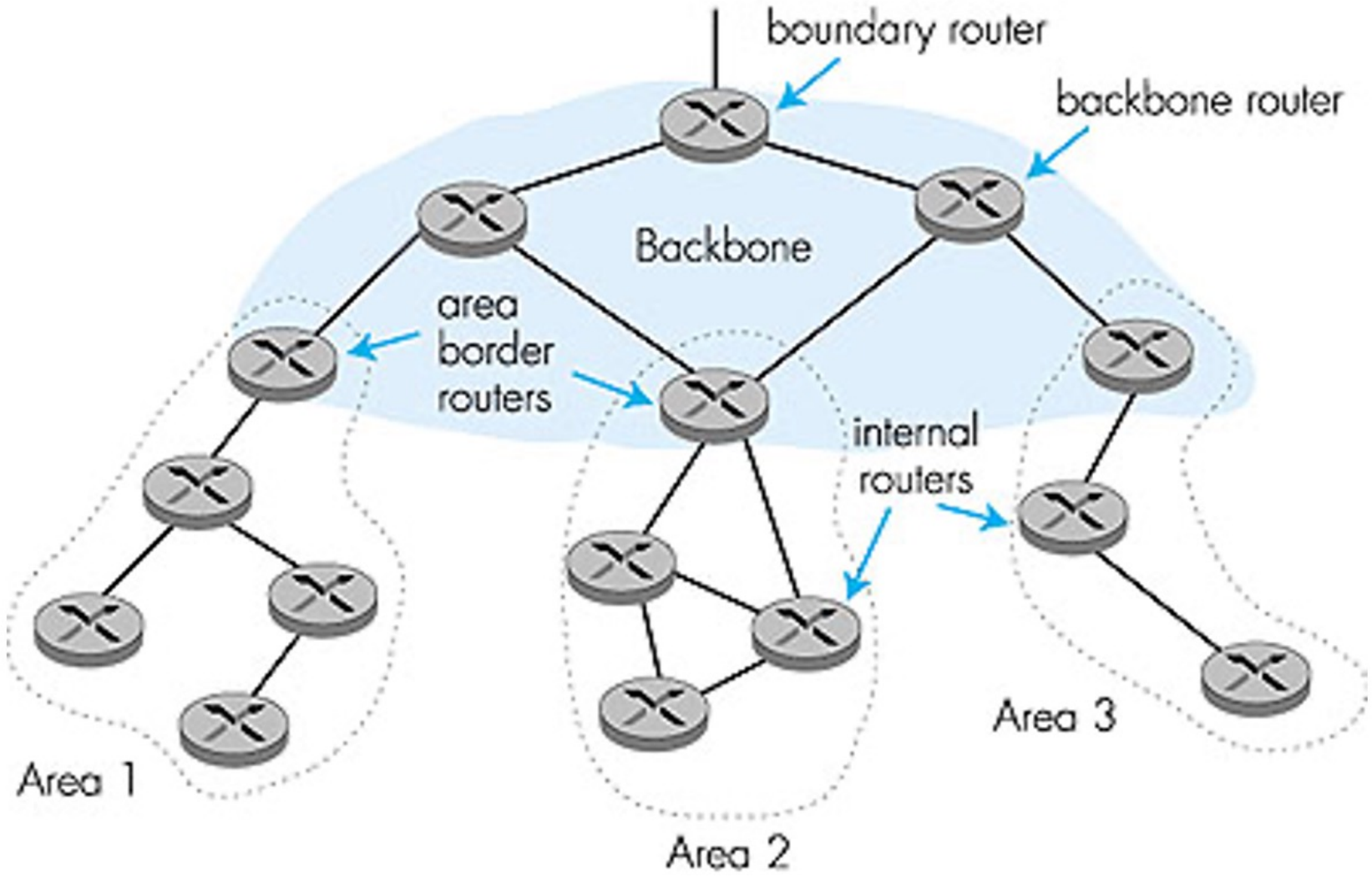
OSPF "advanced" features (not in RIP)

- Security: all OSPF messages authenticated (to prevent malicious intrusion)
- Multiple same-cost paths allowed (only one path in RIP)
- For each link, multiple cost metrics for different TOS (e.g., satellite link cost set "low" for best effort; high for real time)
- Integrated uni- and multicast support:
  - Multicast OSPF (MOSPF) uses same topology data base as OSPF
- Hierarchical OSPF in large domains.

Network Layer

(Hierarchical OSPF- Open Shortest Path First Protocol)

Hierarchical OSPF



Network Layer

(Hierarchical OSPF- Open Shortest Path First Protocol)

Hierarchical OSPF

- Two-level hierarchy: local area, backbone.
  - Link-state advertisements only in area
  - each nodes has detailed area topology; only know direction (shortest path) to nets in other areas.
- Area border routers: "summarize" distances to nets in own area, advertise to other Area Border routers.
- Backbone routers: run OSPF routing limited to backbone.
- Boundary routers: connect to other AS's.

Network Layer

(BGP- Border Gateway Protocol)

Chapter 4: Network Layer

- 4. 1 Introduction
- 4.2 Virtual circuit and datagram networks
- 4.3 What's inside a router
- 4.4 IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- 4.5 Routing algorithms
  - Link state
  - Distance Vector
  - Hierarchical routing
- 4.6 Routing in the Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Broadcast and multicast routing

Network Layer

(BGP- Border Gateway Protocol)

(BGP- Border Gateway Protocol—Port 179)

- Border Gateway Protocol (BGP) is used to Exchange routing information for the internet and is the protocol used between ISPs, which are different ASes.
- The protocol can connect together any internetwork of an autonomous system using an arbitrary topology.
- The only requirement is that each AS has at least one router that can run BGP, which is a router connected to at least one other AS's BGP router.
- BGP's main function is to exchange network reachability information with other BGP systems.
- Border Gateway Protocol constructs an autonomous systems graph based on the information exchanged between BGP routers.

Network Layer

(BGP- Border Gateway Protocol)

Internet inter-AS routing: BGP

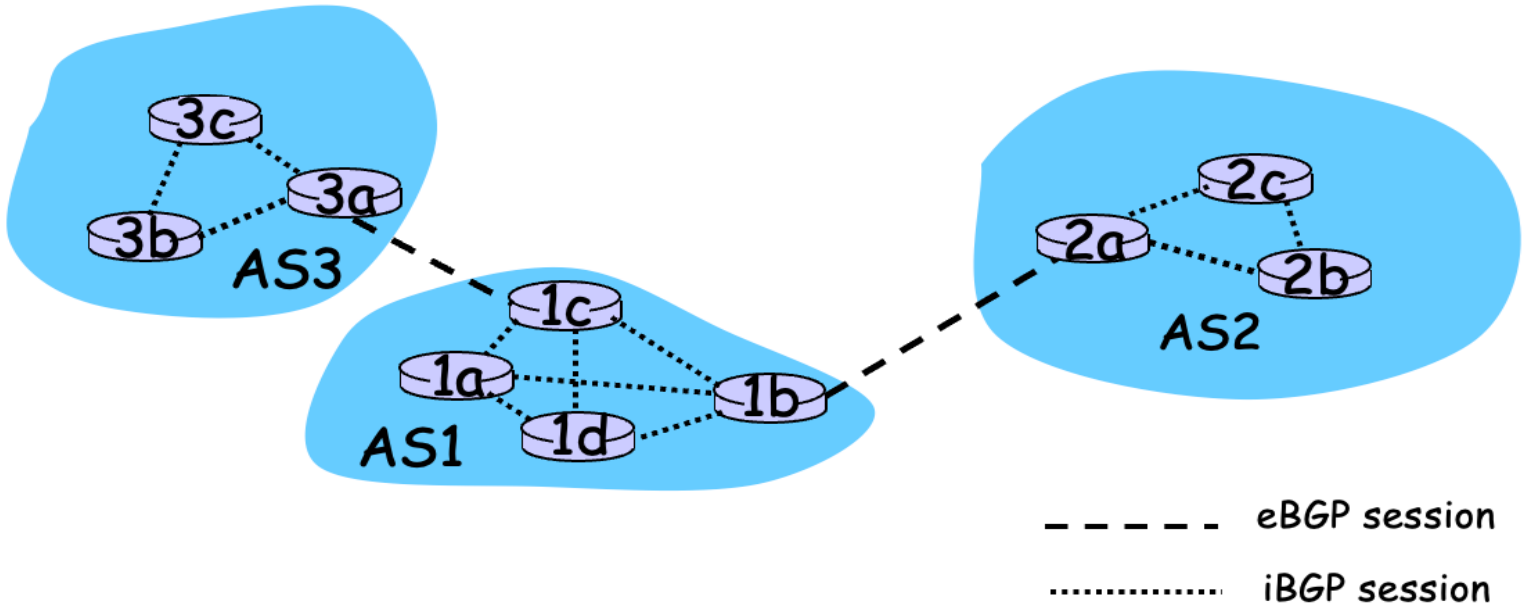
- BGP (Border Gateway Protocol): *the* de facto standard
- BGP provides each AS a means to:
  1. Obtain subnet reachability information from neighboring ASs.
  2. Propagate the reachability information to all routers internal to the AS.
  3. Determine "good" routes to subnets based on reachability information and policy.
- Allows a subnet to advertise its existence to rest of the Internet: *"I am here"*

Network Layer

(BGP- Border Gateway Protocol)

BGP basics

- Pairs of routers (BGP peers) exchange routing info over semi-permanent TCP conctns: BGP sessions
- Note that BGP sessions do not correspond to physical links.
- When AS2 advertises a prefix to AS1, AS2 is *promising* it will forward any datagrams destined to that prefix.
  - AS2 can aggregate prefixes in its advertisement

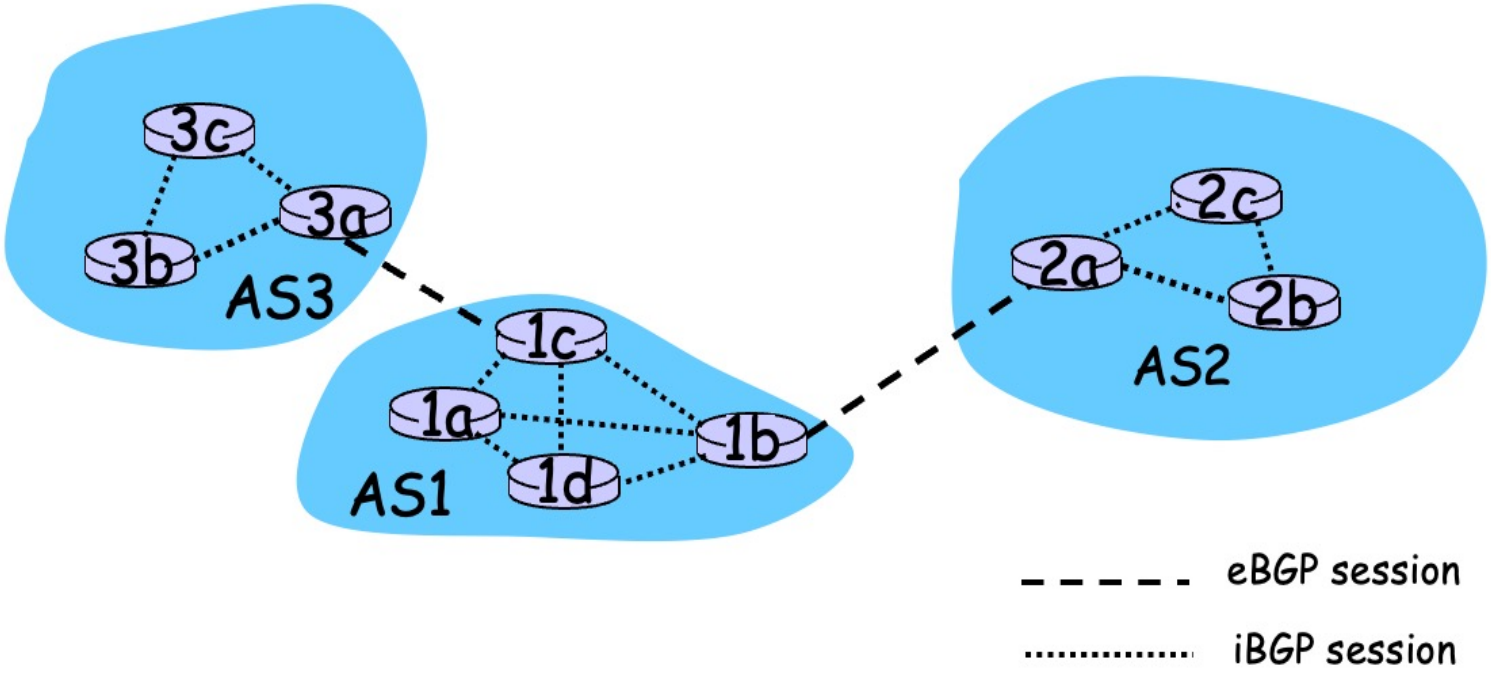


Network Layer

(BGP- Border Gateway Protocol)

Distributing reachability info

- With eBGP session between 3a and 1c, AS3 sends prefix reachability info to AS1.
- 1c can then use iBGP do distribute this new prefix reach info to all routers in AS1
- 1b can then re-advertise the new reach info to AS2 over the 1b-to-2a eBGP session
- When router learns about a new prefix, it creates an entry for the prefix in its forwarding table.



Network Layer

(BGP- Border Gateway Protocol)

BGP messages

- BGP messages exchanged using TCP.
- BGP messages:
  - OPEN: opens TCP connection to peer and authenticates sender
  - UPDATE: advertises new path (or withdraws old)
  - KEEPALIVE keeps connection alive in absence of UPDATES; also ACKs OPEN request
  - NOTIFICATION: reports errors in previous msg; also used to close connection

Network Layer

(BGP- Border Gateway Protocol)

Characteristics of Border Gateway Protocol (BGP):

- **Inter-Autonomous System Configuration:** BGP's main role is to communicate between two autonomous systems.
- BGP supports the Next-Hop Paradigm.
- Coordination among multiple BGP speakers within the AS (Autonomous System).
- **Path Information:** BGP advertisement also includes path information and the reachable destination and next destination pair.
- **Policy Support:** BGP can implement policies that the administrator can configure. For example, a router running BGP can be configured to distinguish between the routes known within the AS and those known from outside the AS.
- Runs Over TCP.
- BGP conserves network Bandwidth.
- BGP supports CIDR.
- BGP also supports Security.

Functionality of Border Gateway Protocol (BGP):

Network Layer

(BGP- Border Gateway Protocol)

BGP peers perform 3 functions, which are given below.

1. The first function consists of initial peer acquisition and authentication, both the peers established a TCP connection and perform a message exchange that guarantees both sides have agreed to communicate.

2. The second function mainly focus on sending negative or positive reach-ability information.

3. The third function verifies that the peers and the network connection between them are functioning correctly.

BGP Route Information Management Functions:

Network Layer

(BGP- Border Gateway Protocol)

- **Route Storage:** Each BGP stores information about how to reach other networks.
- **Route Update:** In this task, Special techniques are used to determine when and how to use the information received from peers to properly update the routes.
- **Route Selection:** Each BGP uses the information in its route databases to select good routes to each network on the internet network.
- **Route advertisement:** Each BGP speaker regularly tells its peer what is known about various networks and methods to reach them.

Network Layer

(Broadcast and Multicast Routing)

Chapter 4: Network Layer

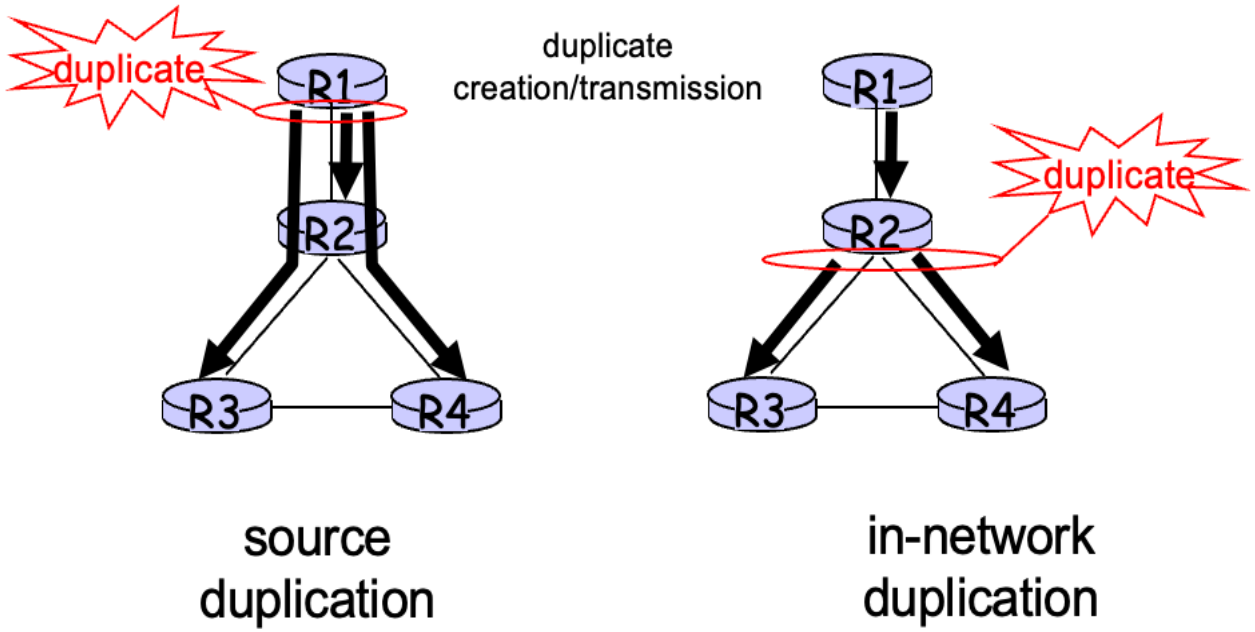
- 4. 1 Introduction
- 4.2 Virtual circuit and datagram networks
- 4.3 What's inside a router
- 4.4 IP: Internet Protocol
  - Datagram format
  - IPv4 addressing
  - ICMP
  - IPv6
- 4.5 Routing algorithms
  - Link state
  - Distance Vector
  - Hierarchical routing
- 4.6 Routing in the Internet
  - RIP
  - OSPF
  - BGP
- 4.7 Broadcast and multicast routing

Network Layer

(Broadcast and Multicast Routing)

Broadcast Routing

- Deliver packets from source to all other nodes
- Source duplication is inefficient:



Network Layer

(Broadcast and Multicast Routing)

In-network duplication

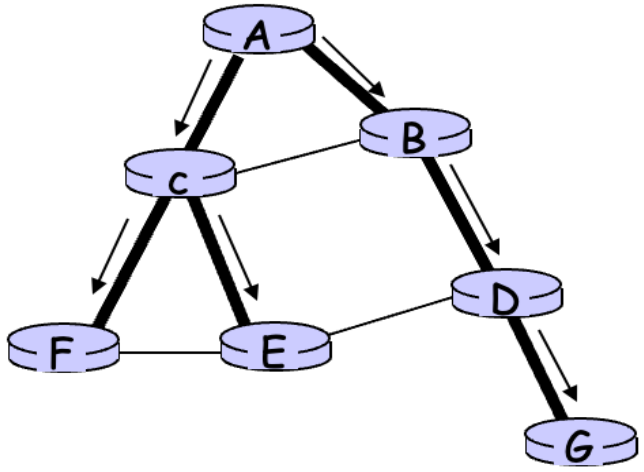
- Flooding: when node receives brdcst pckt, sends copy to all neighbors
  - Problems: cycles & broadcast storm
- Controlled flooding: node only brdcsts pkt if it hasn't brdcst same packet before
  - Node keeps track of pckt ids already brdcsted
  - Or reverse path forwarding (RPF): only forward pckt if it arrived on shortest path between node and source
- Spanning tree
  - No redundant packets received by any node

Network Layer

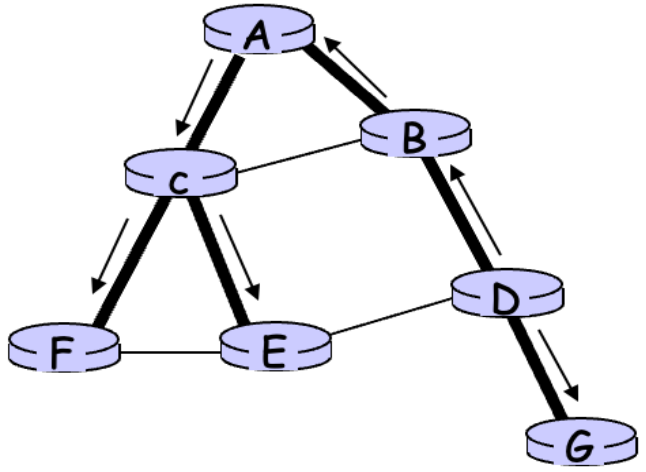
(Broadcast and Multicast Routing)

Spanning Tree

- First construct a spanning tree
- Nodes forward copies only along spanning tree



(a) Broadcast initiated at A



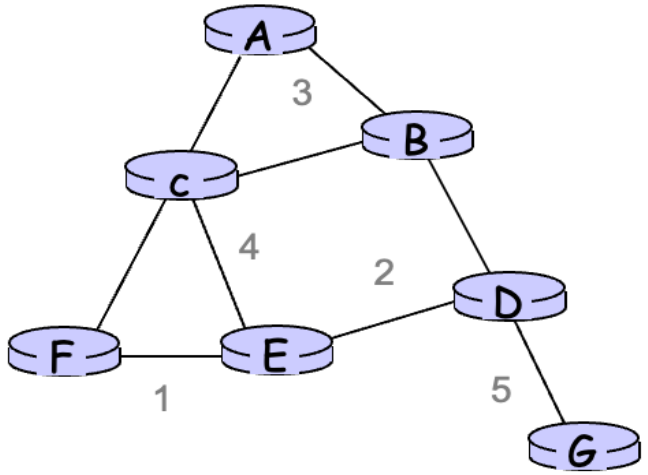
(b) Broadcast initiated at D

Network Layer

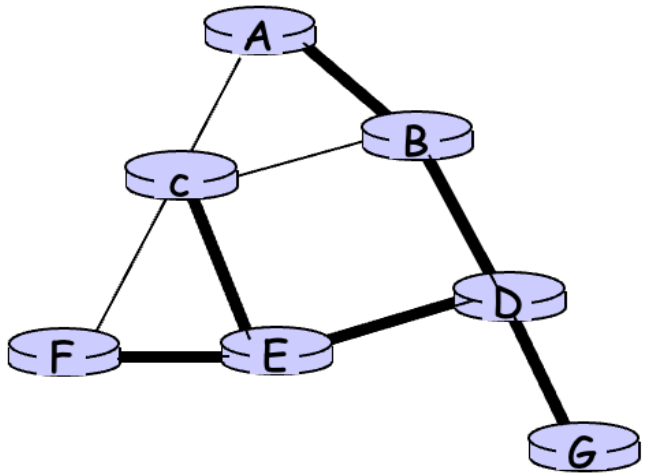
(Broadcast and Multicast Routing)

Spanning Tree: Creation

- Center node
- Each node sends unicast join message to center node
  - Message forwarded until it arrives at a node already belonging to spanning tree



(a) Stepwise construction of spanning tree



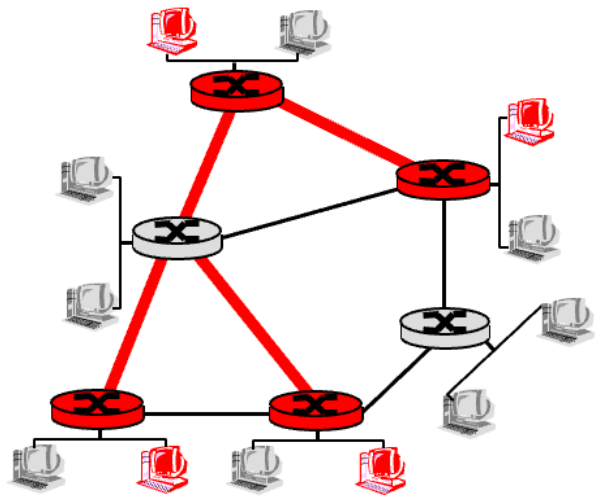
(b) Constructed spanning tree

Network Layer

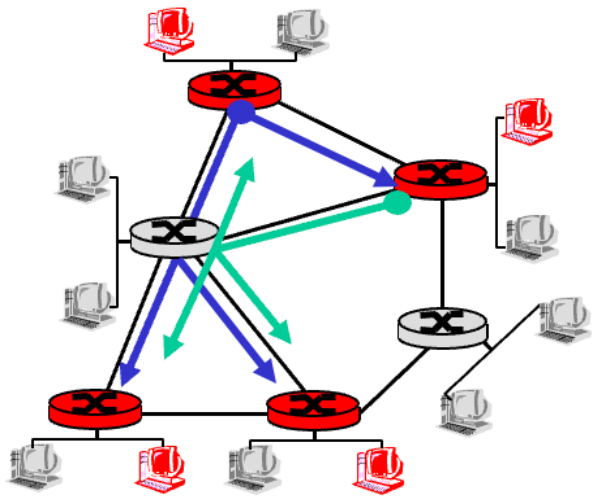
(Broadcast and Multicast Routing)

Multicast Routing: Problem Statement

- *Goal:* find a tree (or trees) connecting routers having local mcast group members
  - *tree:* not all paths between routers used
  - *source-based:* different tree from each sender to rcvrs
  - *shared-tree:* same tree used by all group members



Shared tree



Source-based trees

Network Layer

(Broadcast and Multicast Routing)

Approaches for building mcast trees

Approaches:

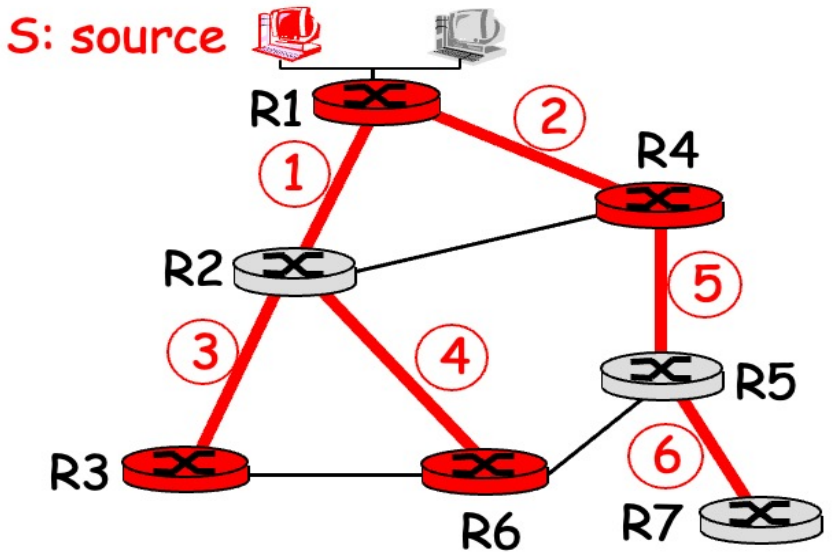
- source-based tree: one tree per source
  - shortest path trees
  - reverse path forwarding
- group-shared tree: group uses one tree
  - minimal spanning (Steiner)
  - center-based trees

Network Layer

(Broadcast and Multicast Routing)

Shortest Path Tree

- mcast forwarding tree: tree of shortest path routes from source to all receivers
  - Dijkstra's algorithm



LEGEND

router with attached group member

router with no attached group member

i — link used for forwarding, i indicates order link added by algorithm

Network Layer

(Broadcast and Multicast Routing)

Internet Multicasting Routing: DVMRP

- DVMRP: distance vector multicast routing protocol, RFC1075
- *flood and prune:* reverse path forwarding, source-based tree
  - RPF tree based on DVMRP's own routing tables constructed by communicating DVMRP routers
  - no assumptions about underlying unicast
  - initial datagram to mcast group flooded everywhere via RPF
  - routers not wanting group: send upstream prune msgs