Data Communications and Networking Fourth Edition

Forouzan

Chapter 15

Connecting LANs, Backbone Networks, and Virtual LANs

Copyright © The McGraw-Hill Companies, Inc. Permission required for reproduction or display.

15-1 CONNECTING DEVICES

In this section, we divide connecting devices into five different categories based on the layer in which they operate in a network.

Topics discussed in this section:

Passive Hubs
Active Hubs
Bridges
Two-Layer Switches
Routers
Three-Layer Switches
Gateways

Figure 15.1 *Five categories of connecting devices*

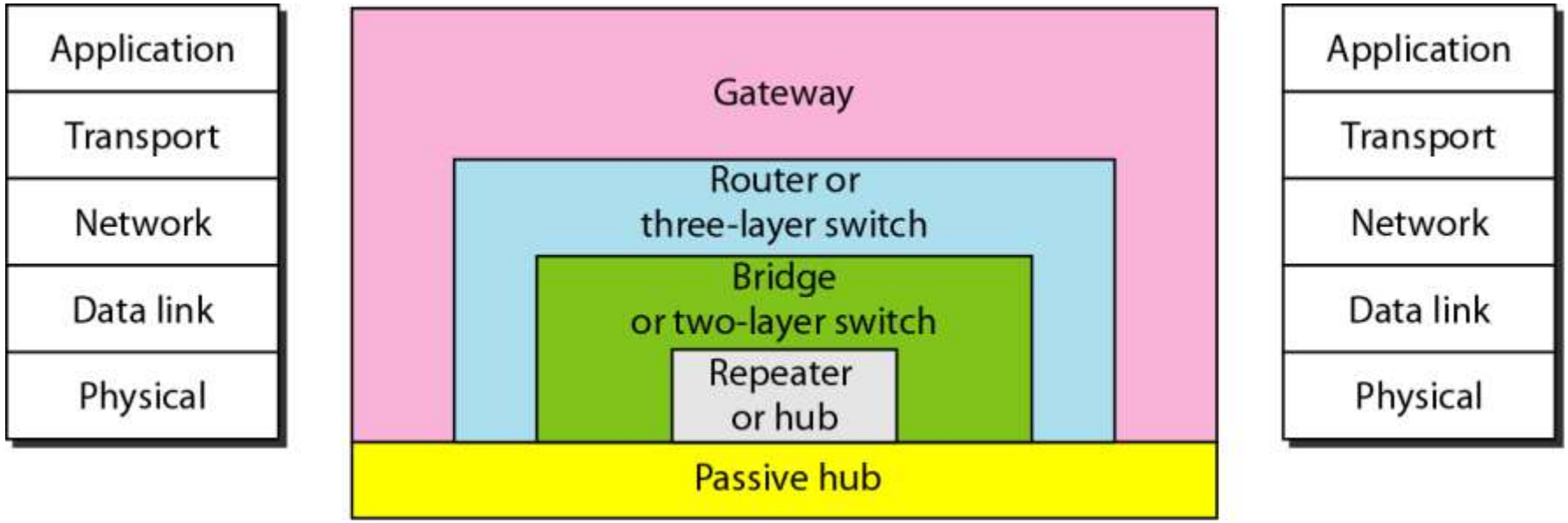
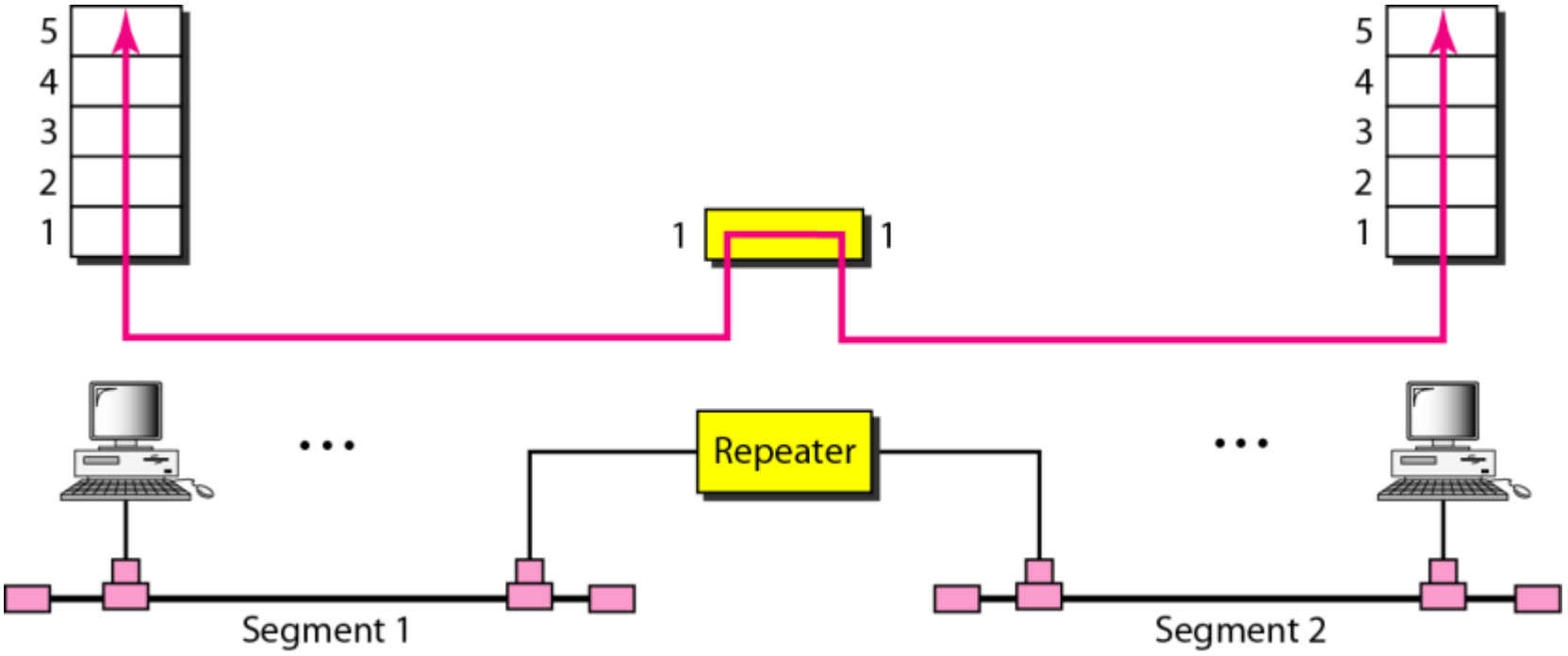
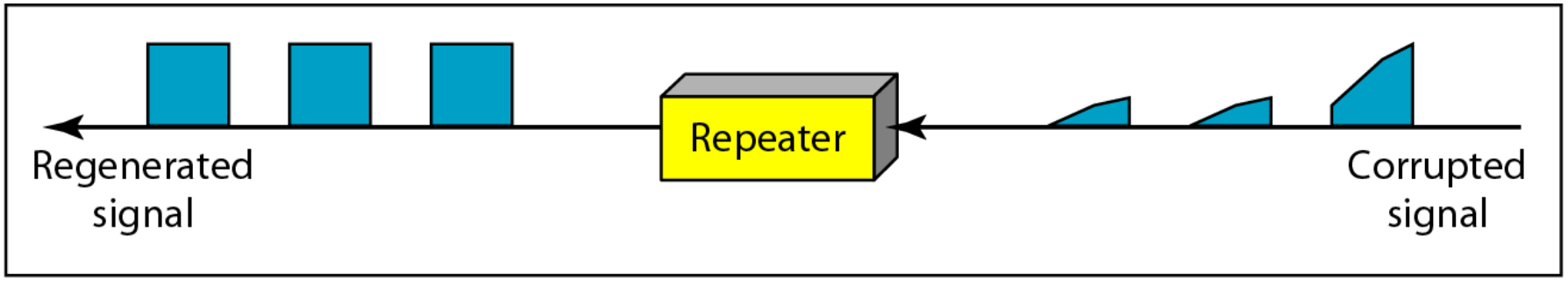


Figure 15.2 *A repeater connecting two segments of a LAN*

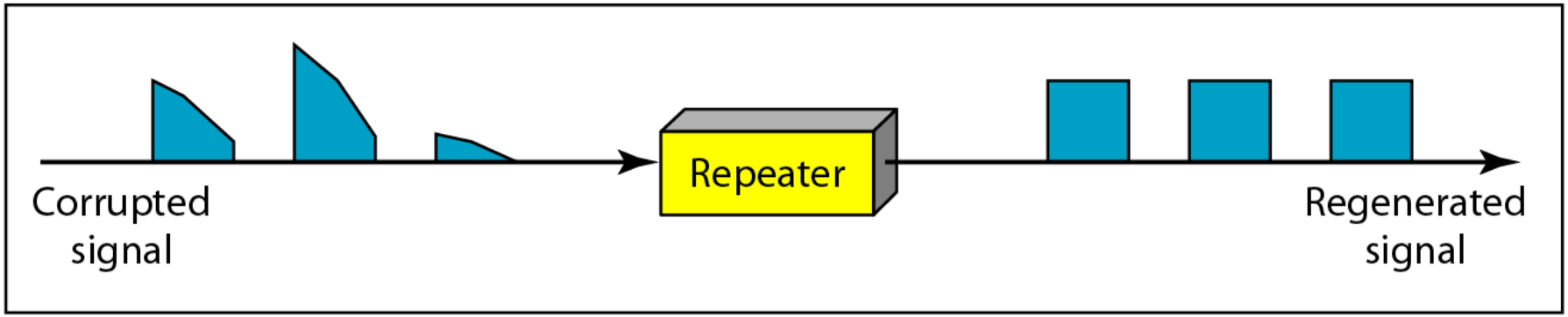


A repeater connects segments of a LAN.
A repeater forwards every frame – there is no filtering.
A repeater is a regenerator, not an amplifier.

Figure 15.3 *Function of a repeater*

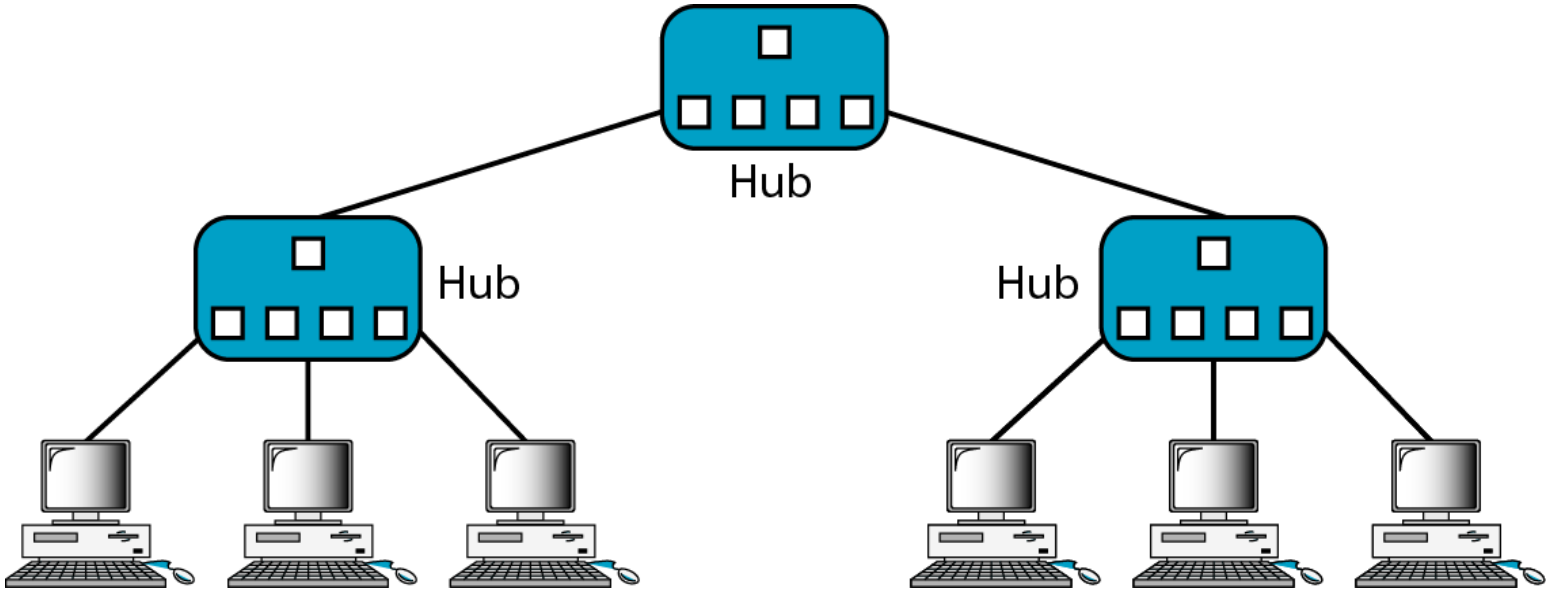


a. Right-to-left transmission.



b. Left-to-right transmission.

Figure 15.4 *A hierarchy of hubs*

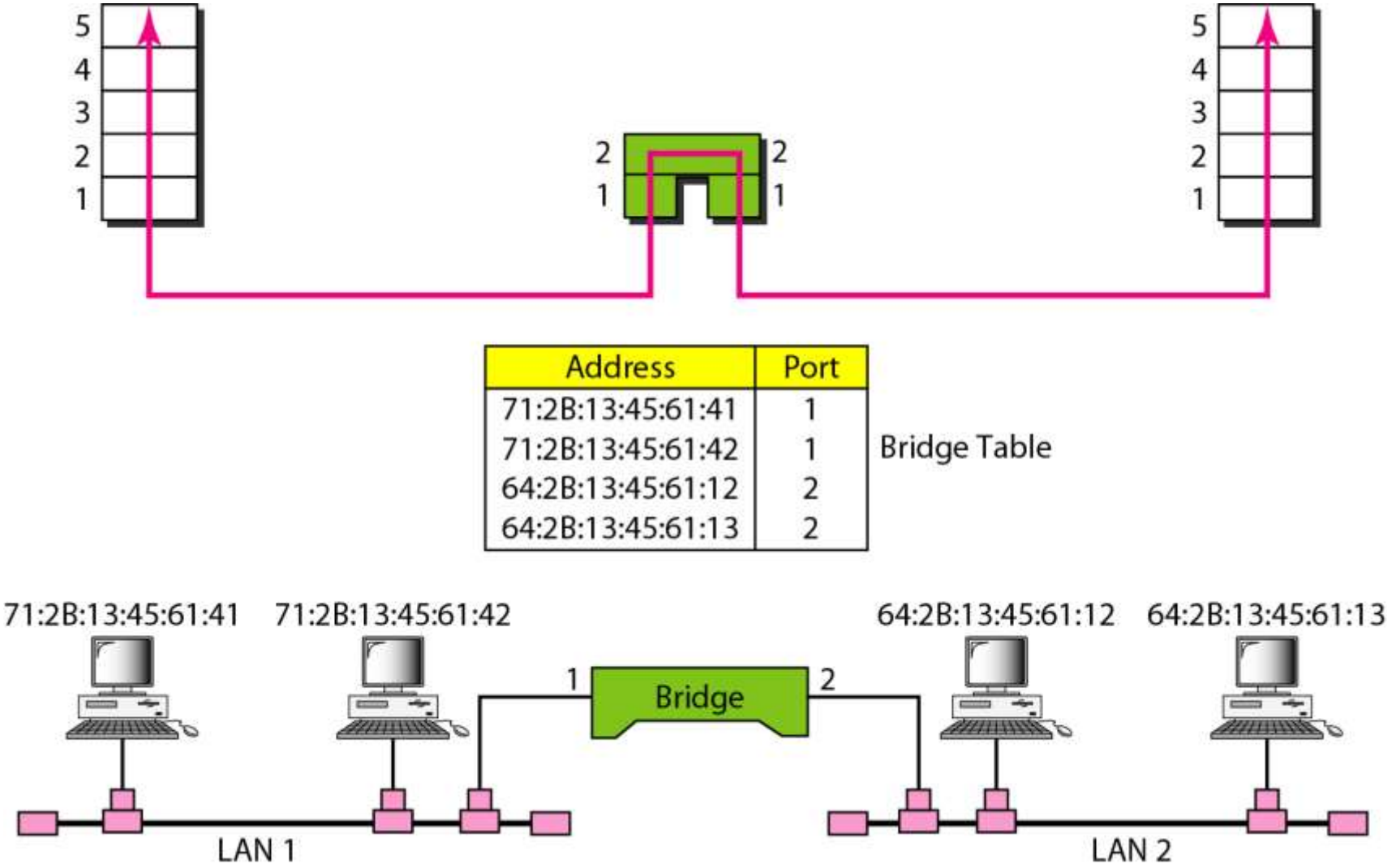


A hub is a multi-port repeater, used in star-wired LANs (Ethernet).
Because of the amount of traffic and collisions, hubs can only be used in small network configurations.

Note

A bridge has a table used in filtering decisions.

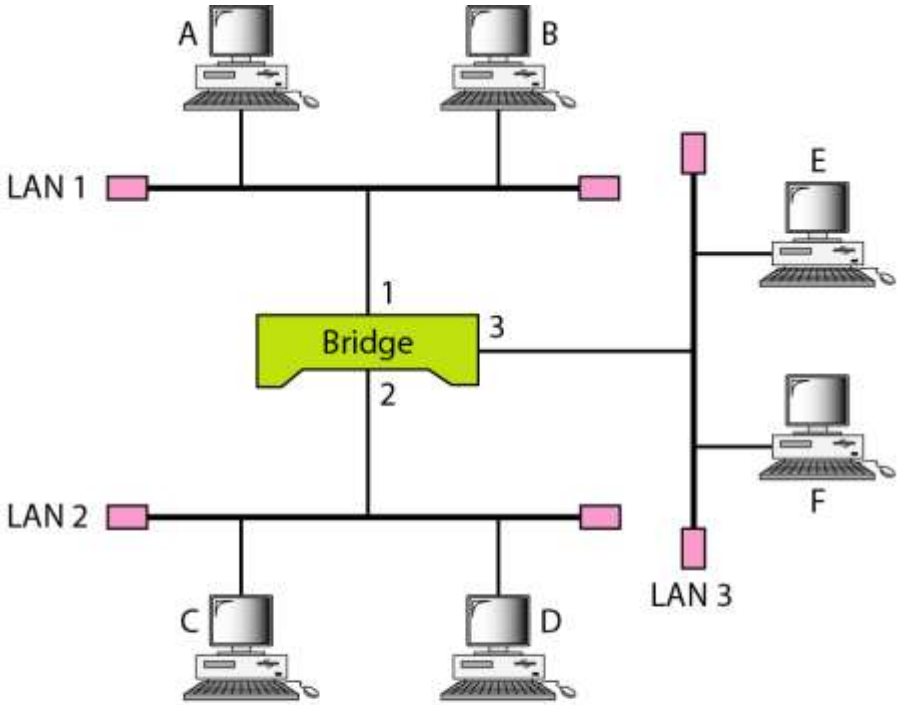
Figure 15.5 *A bridge connecting two LANs*



Note

A bridge does not change the physical (MAC) addresses in a frame.

Figure 15.6 *A learning bridge and the process of learning*



Address	Port

a. Original

Address	Port
A	1

b. After A sends a frame to D

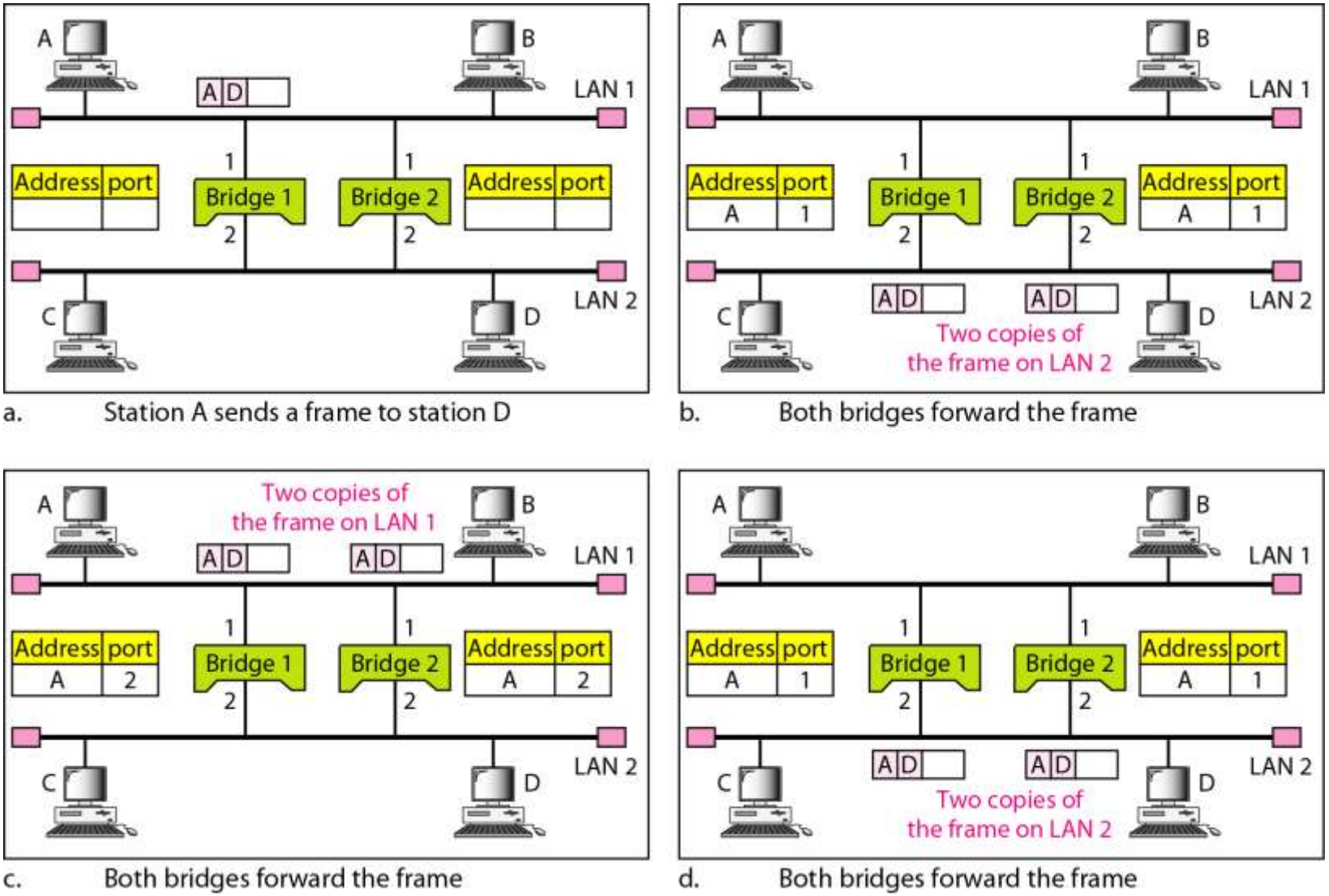
Address	Port
A	1
E	3

c. After E sends a frame to A

Address	Port
A	1
E	3
B	1

d. After B sends a frame to C

Figure 15.7 *Loop problem in a learning bridge*



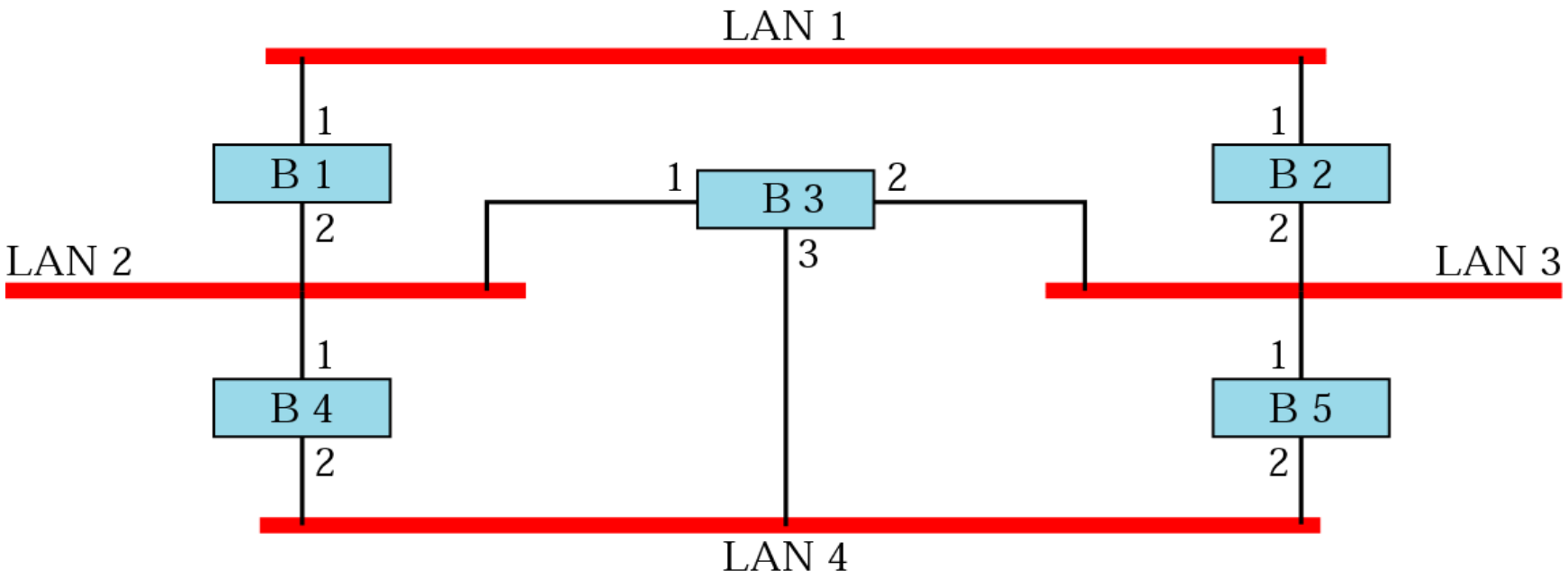
a. Station A sends a frame to station D

b. Both bridges forward the frame

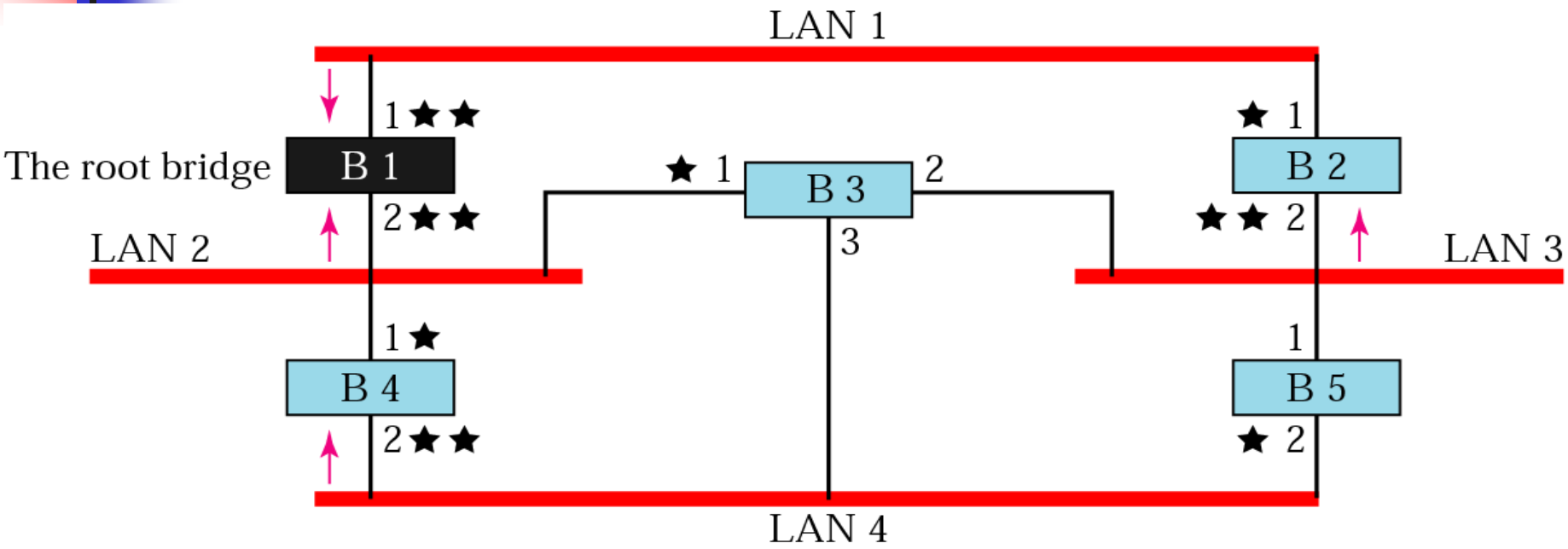
c. Both bridges forward the frame

d. Both bridges forward the frame

What happens if you have a loop of bridges/switches in your LAN?



Applying spanning tree



Step 1: Every bridge has an ID. Select the bridge with smallest ID. This is the *root bridge*.
Step 2: Mark one port of each bridge (except root bridge) as the *root port*. Root port is the port with least-cost path from the bridge to the root bridge (marked with 1 star).
Step 3: For each LAN, choose a *designated bridge*. A designated bridge has the least-cost path between the LAN and root bridge (the arrows). Mark the corresponding port that connects the LAN to its designated bridge the *designated port* (two stars).

Step 4: Mark the root port and designated port as *forwarding ports*, the others as *blocking ports* (every port with 1 or 2 stars keep, ports with no stars drop). Note - there is only 1 path between any two bridges.

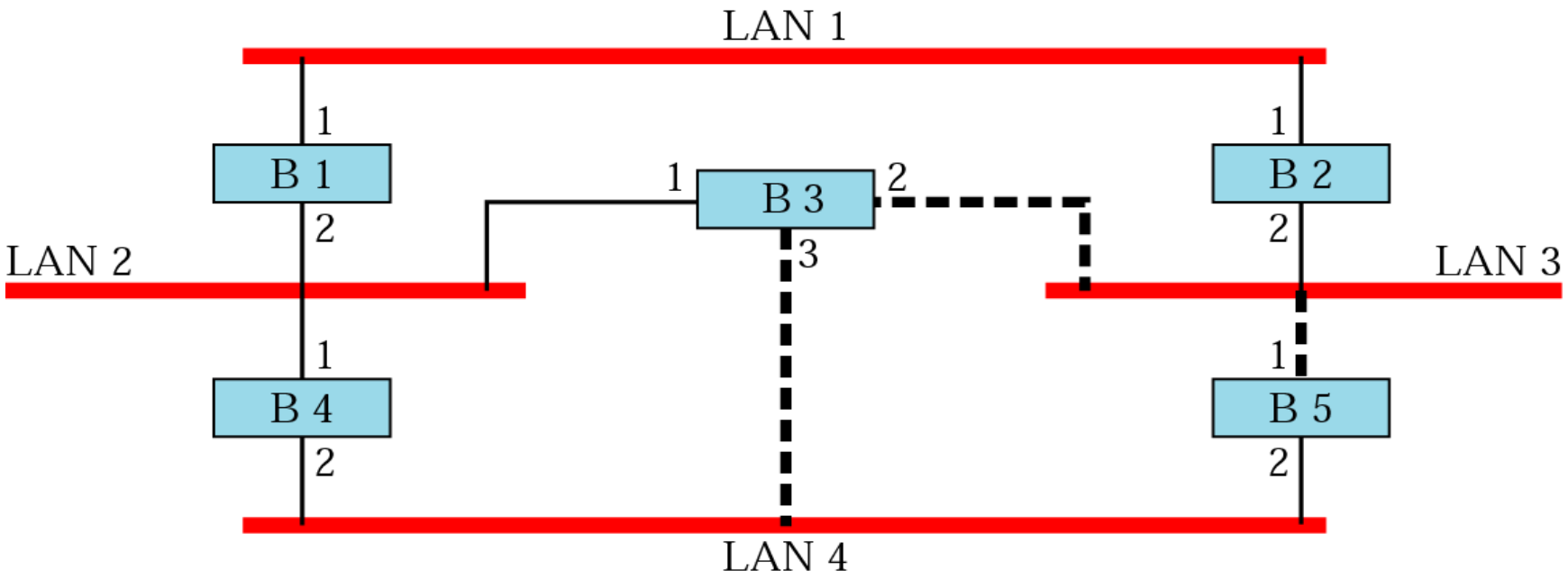
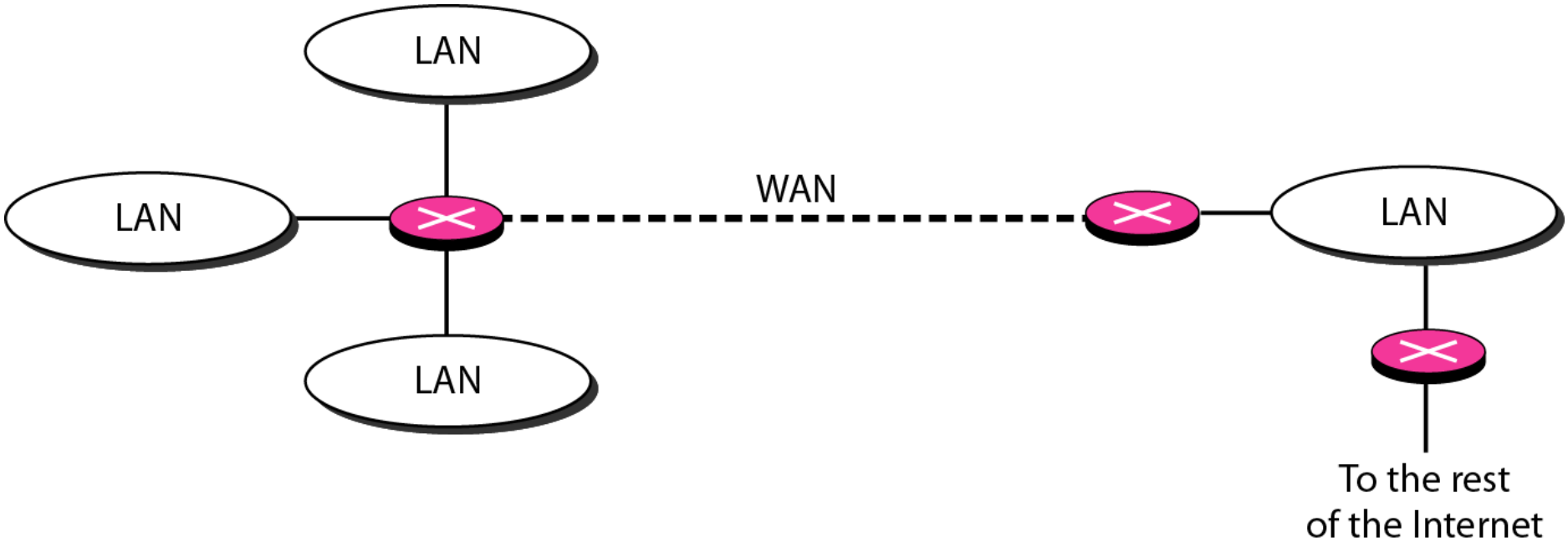


Figure 15.11 *Routers connecting independent LANs and WANs*



15-2 BACKBONE NETWORKS

A backbone network allows several LANs to be connected. In a backbone network, no station is directly connected to the backbone; the stations are part of a LAN, and the backbone connects the LANs.

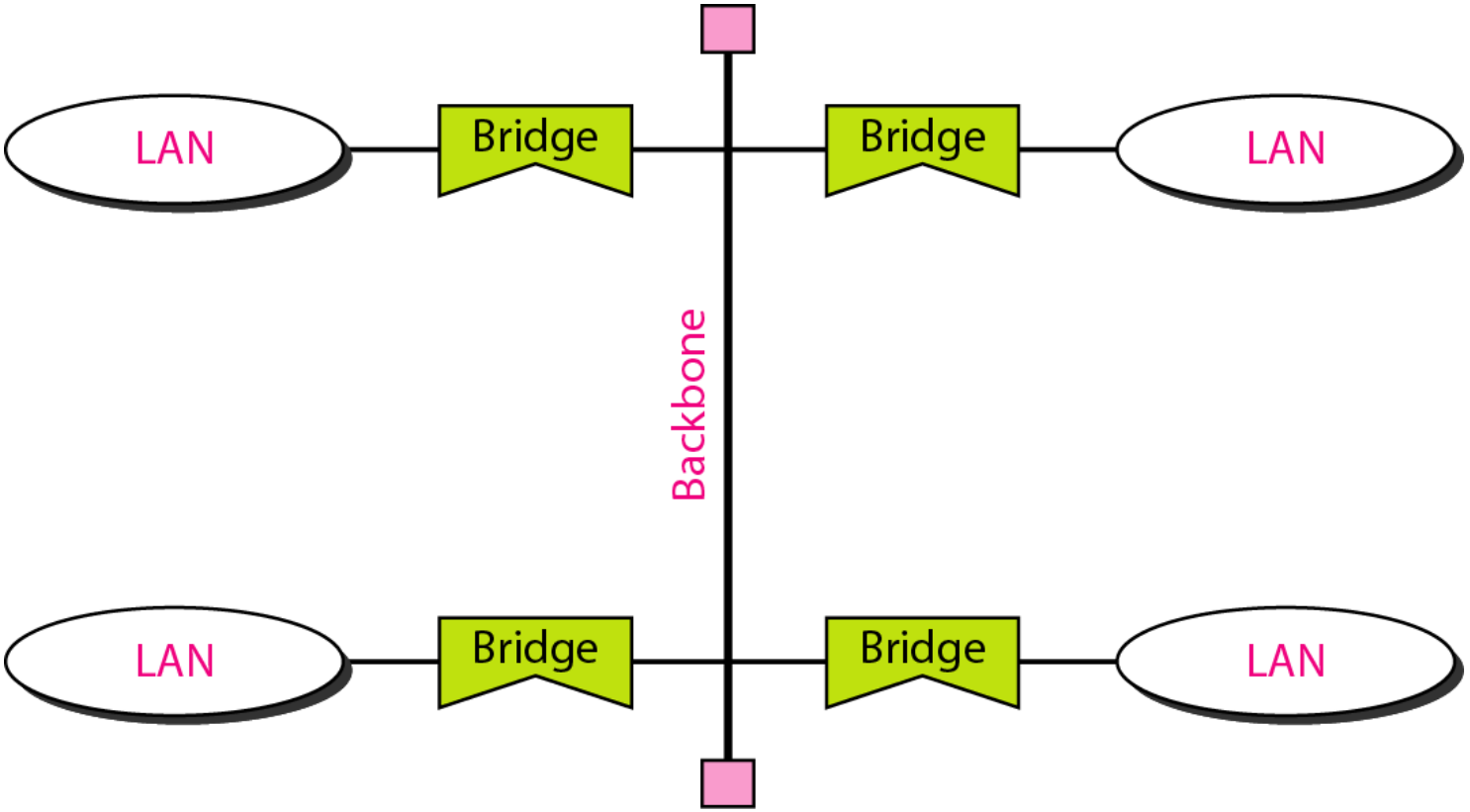
Topics discussed in this section:

Bus Backbone
Star Backbone
Connecting Remote LANs

Note

In a bus backbone, the topology of the backbone is a bus.

Figure 15.12 *Bus backbone*



Note

In a star backbone, the topology of the backbone is a star; the backbone is just one switch.

Figure 15.13 *Star backbone*

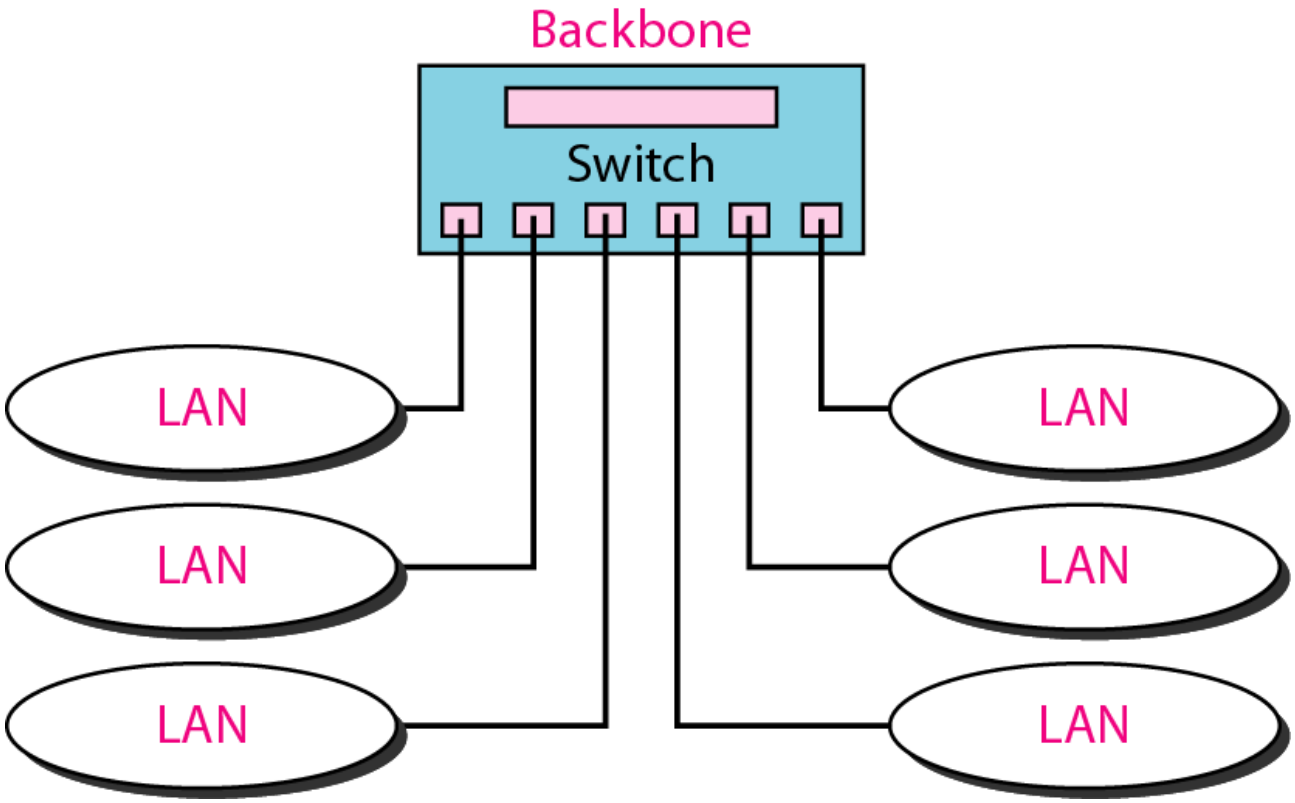
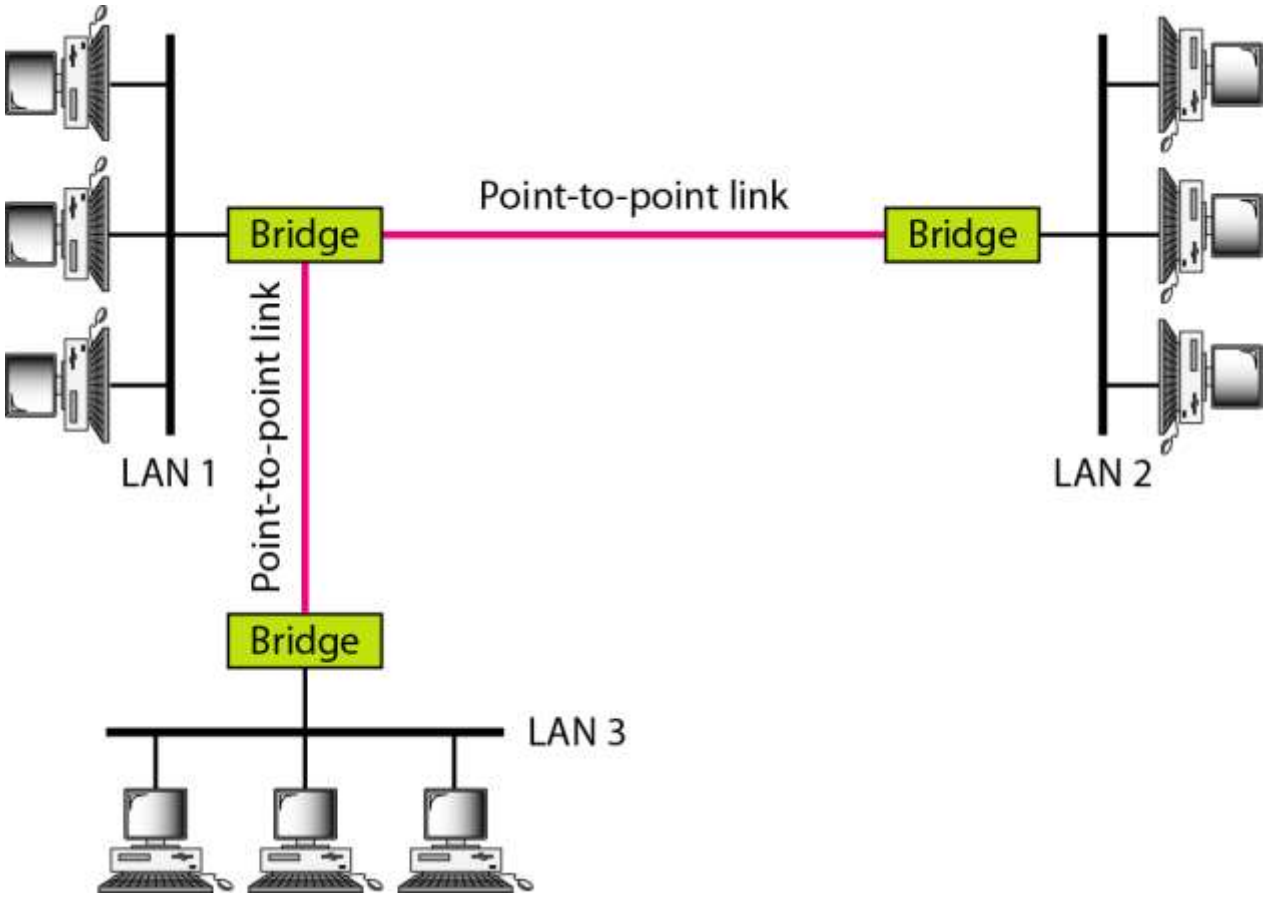


Figure 15.14 *Connecting remote LANs with bridges*



Note

A point-to-point link acts as a LAN in a remote backbone connected by remote bridges.

15-3 VIRTUAL LANs

We can roughly define a virtual local area network (VLAN) as a local area network configured by software, not by physical wiring.

Topics discussed in this section:

Membership
Configuration
Communication between Switches
IEEE Standard
Advantages

Figure 15.15 *A switch connecting three LANs*

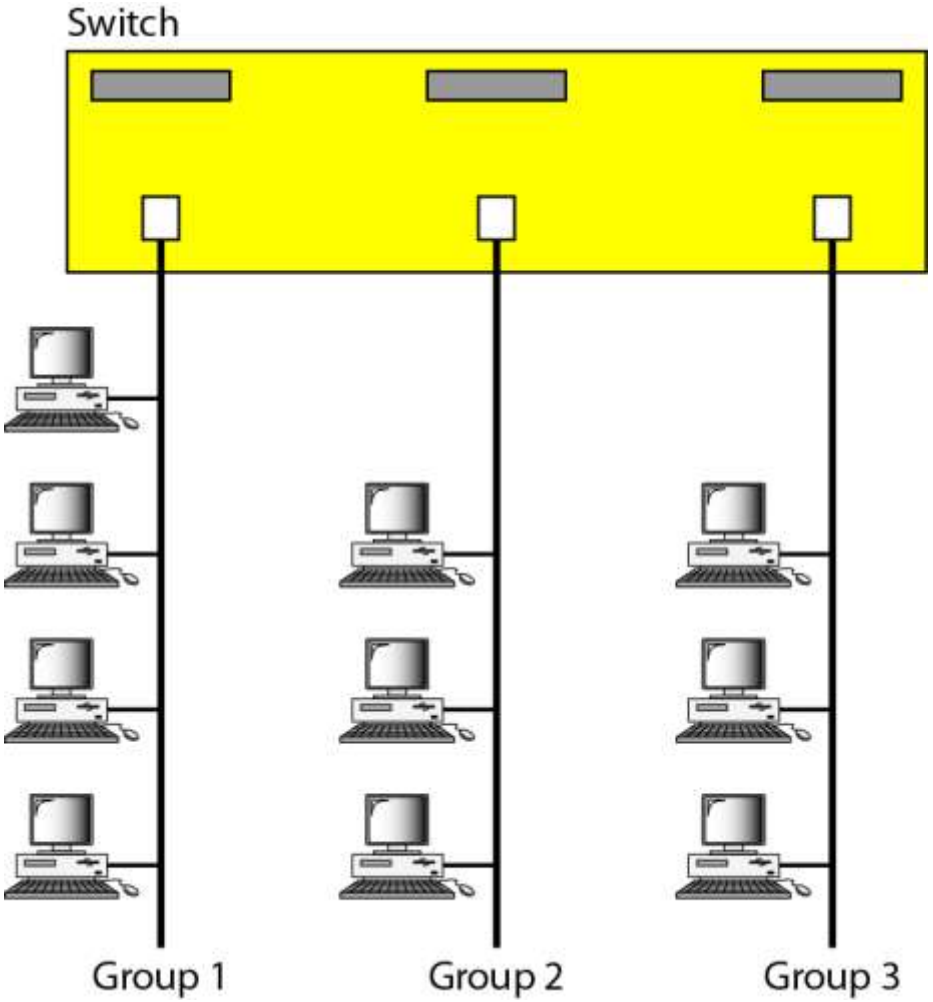


Figure 15.16 *A switch using VLAN software*

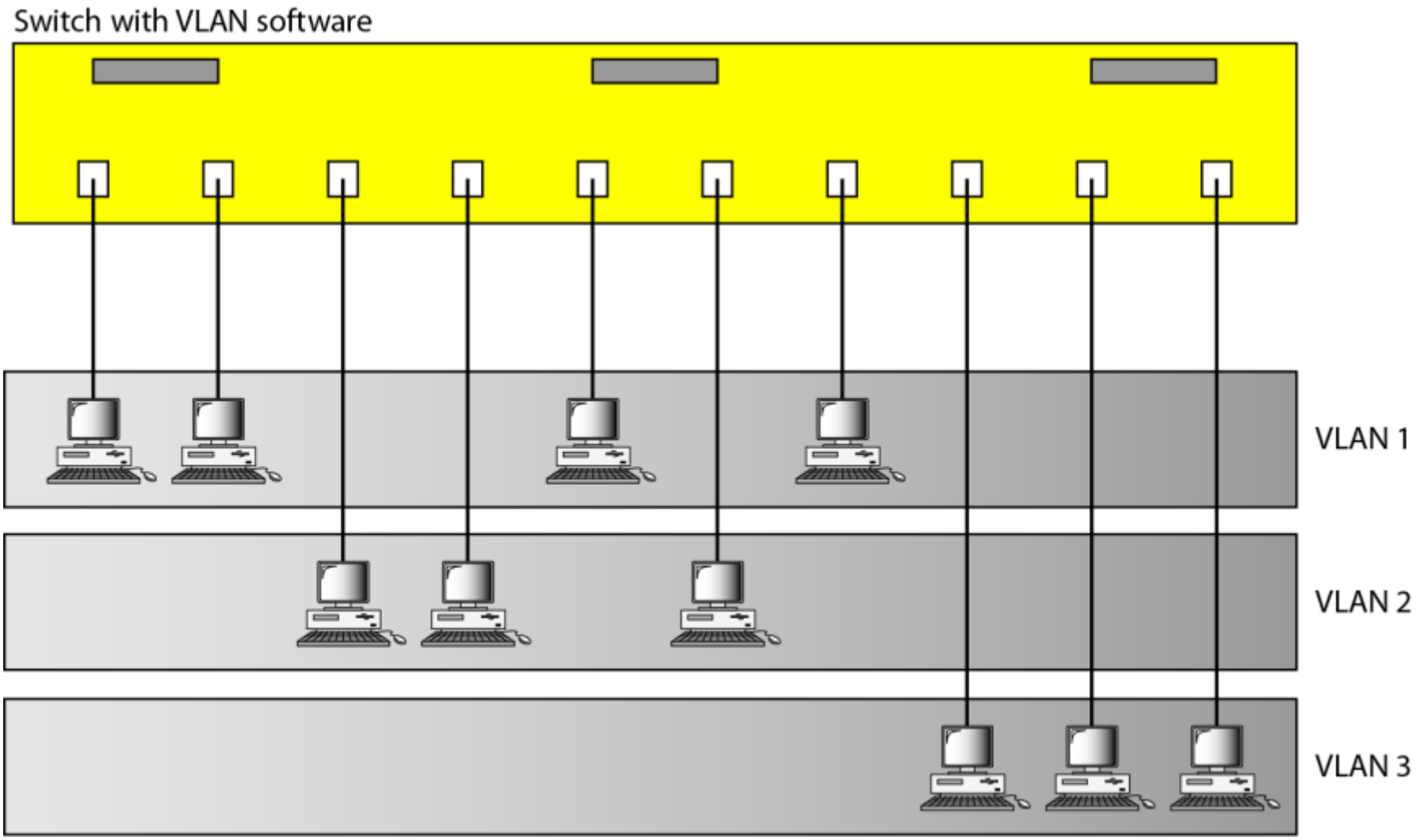
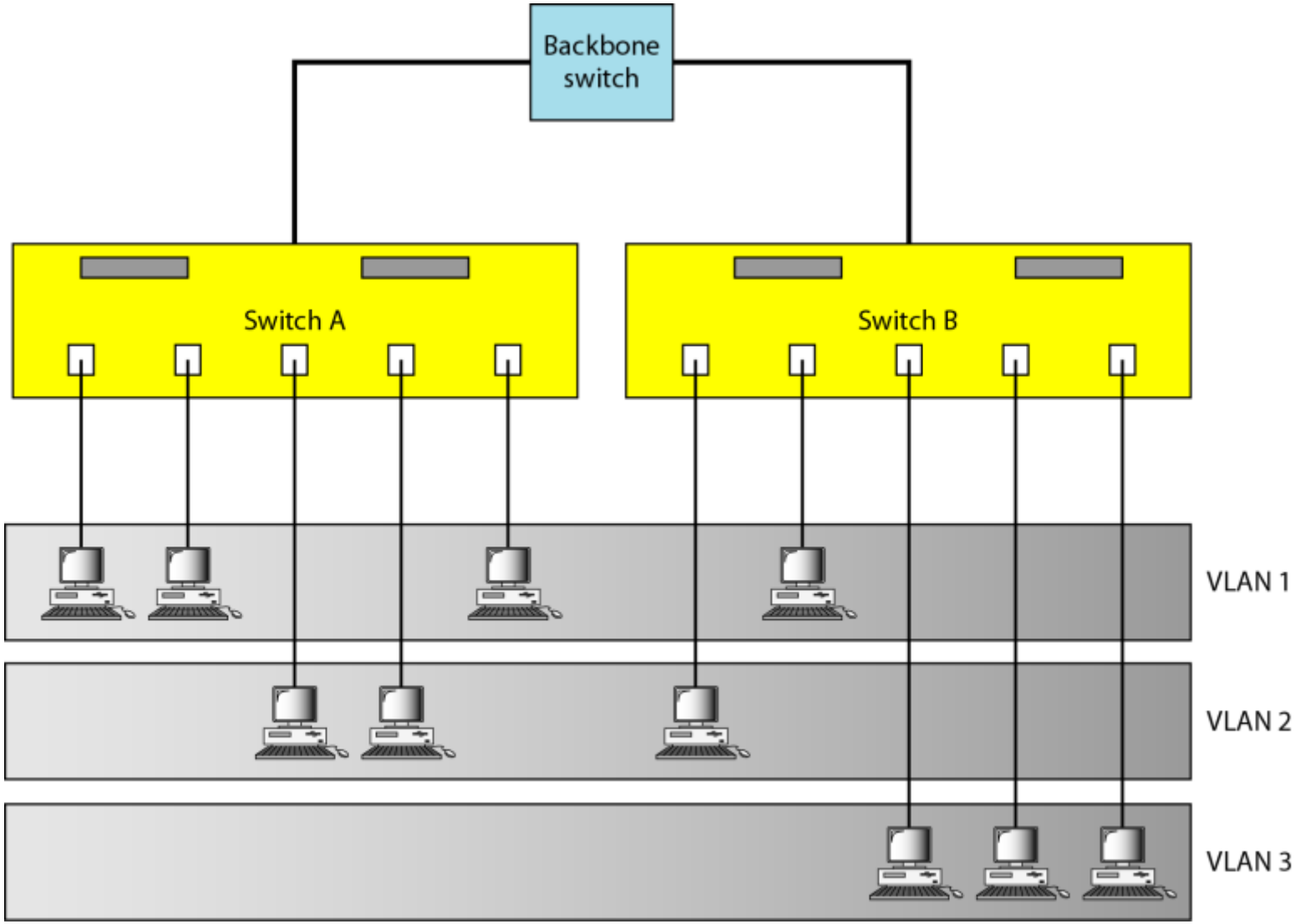


Figure 15.17 *Two switches in a backbone using VLAN software*



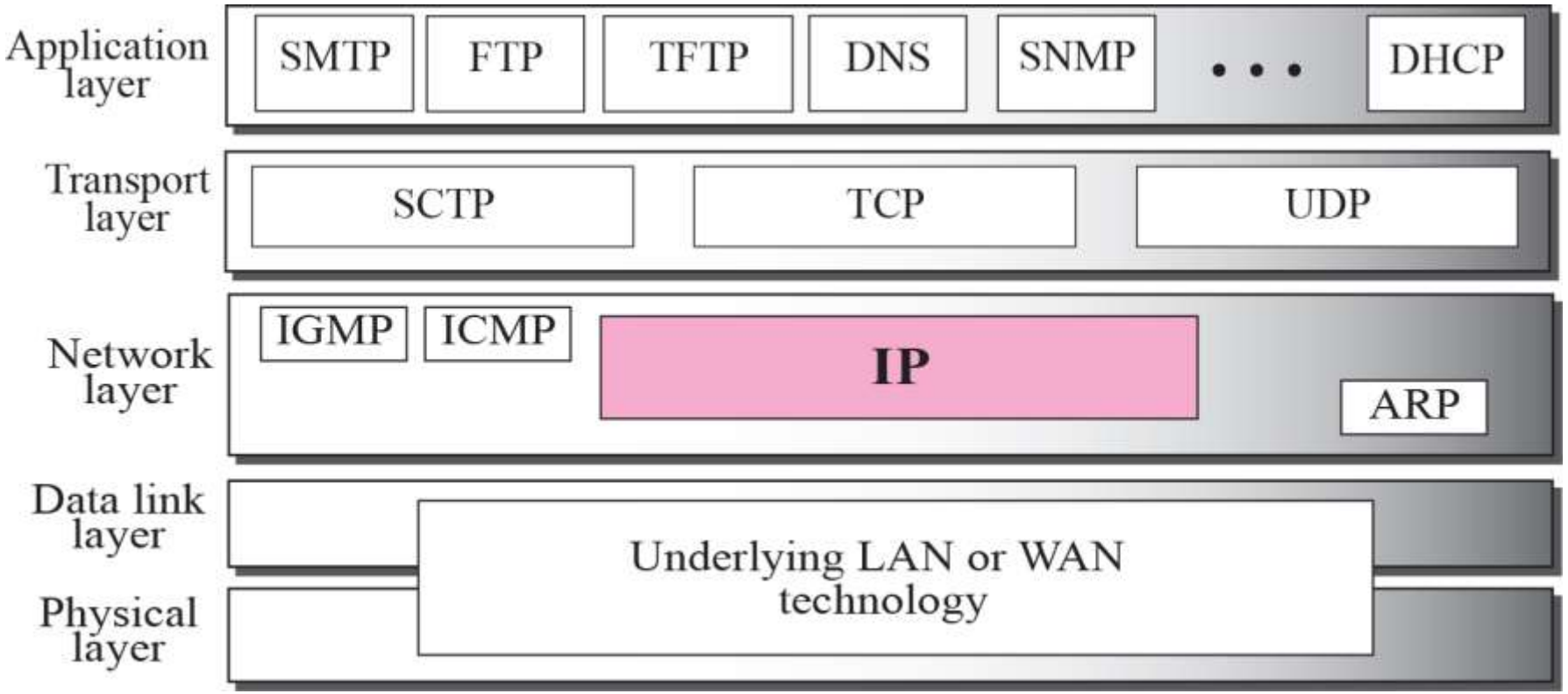
Note

VLANs create broadcast domains.

INTRODUCTION

The Internet Protocol (IP) is the transmission mechanism used by the TCP/IP protocols at the network layer.

Figure 7.1 *Position of IP in TCP/IP protocol suite*



The Internet Protocol (IP) is the method or protocol by which data is sent from one computer to another on the Internet. Each computer (known as a host) on the Internet has at least one IP address that uniquely identifies it from all other computers on the Internet

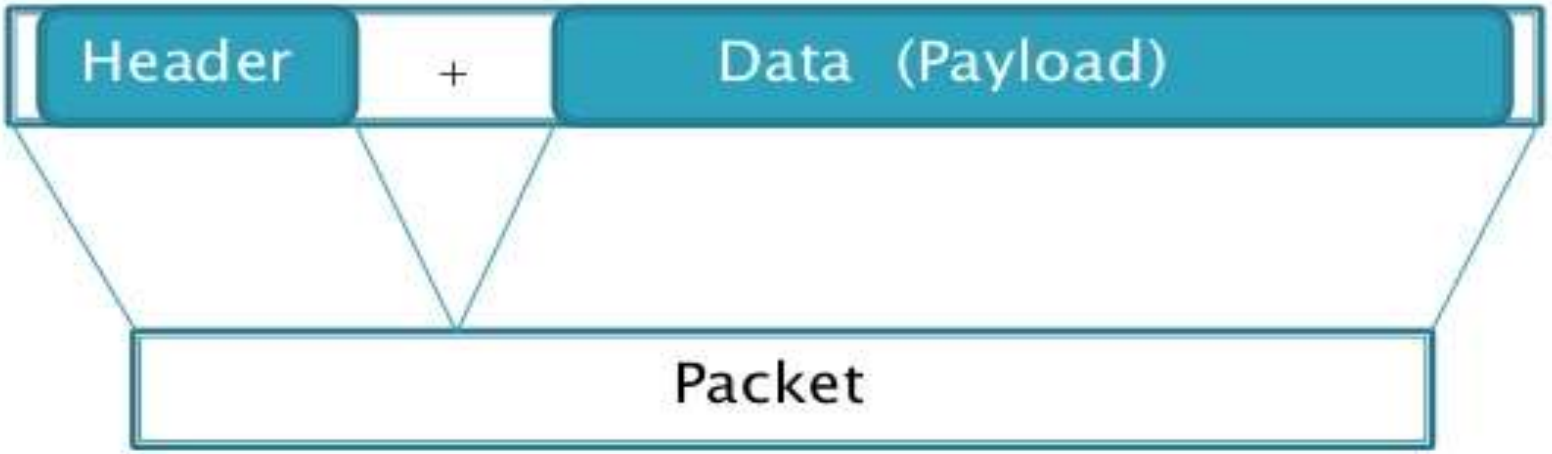
IP has the task of delivering packets from the source host to the destination host solely based on the IP addresses in the packet headers. For this purpose, IP defines packet structures that encapsulate the data to be delivered. It also defines addressing methods that are used to label the datagram with source and destination information.

Purpose of the IP….

- The Internet Protocol defines the basic unit of data transfer (IP Datagram)
- IP software performs the routing function
- IP includes a set of rules that process the idea of unreliable packet delivery.
  - How hosts and routers should process packets
  - How & when error messages should be generated
  - The Conditions under which packets can be discarded.

Construction of Datagrams….

- Each #datagram has two components
  - Header
  - Payload

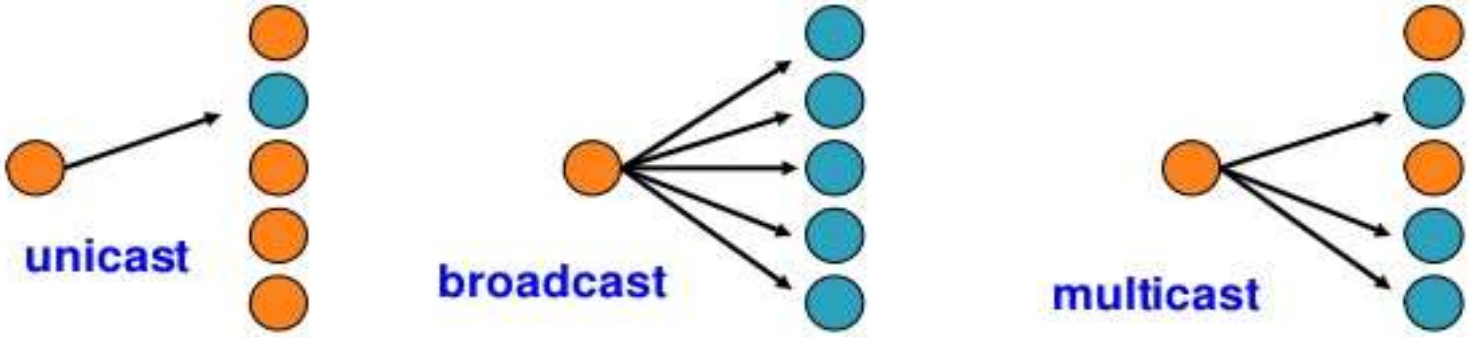


IP Service

- Delivery service of IP is minimal.
- IP provides an unreliable connectionless best effort service
  - Unreliable : IP doesn't make an attempt to recover lost packets
  - Connectionless : Each packet is handled independently
  - Best Effort : IP doesn't make guarantees on the service (No through output , No delay guarantee…)

IP Service (Cont.…)

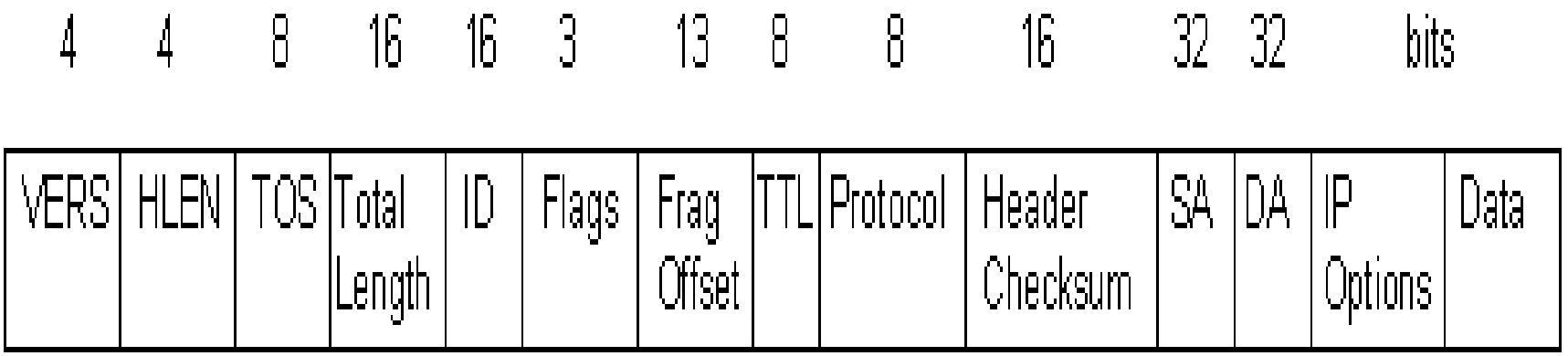
- IP supports the following services
  - One-to-one (unicast)
  - One-to-all (broadcast)
  - One-to-several (multicast)



DATAGRAMS

Packets in the network (internet) layer are called *datagrams*. A datagram is a variable-length packet consisting of two parts: header and data. The header is 20 to 60 bytes in length and contains information essential to routing and delivery. It is customary in TCP/IP to show the header in 4-byte sections. A brief description of each field is in order.

IP DATAGAM


4 4 8 16 16 3 13 8 8 16 32 32 bits
VERS
HLEN
TOS
Total Length
ID
Flags
Frag Offset
TTL
Protocol
Header Checksum
SA
DA
IP Options
Data

VERS - is the IP version number (currently binary **0100** (4), but can now also be version 6). All nodes must use the same version.
HLEN - header length in 32-bit words, so if the number is 6, then 6 x 32 bit words are in the header i.e. 24 bytes. The maximum size is 15 x 32-bit words which is 60 bytes. The minimum size is 20 bytes or 5 x 32-bit words.

Type of Service - is how the datagram should be used, e.g. delay, precedence, reliability, minimum cost, throughput etc. This TOS field is now used by **Differentiated Services** and is called the **Diff Serv Code Point (DSCP)**.

Total Length - is the number of octets that the IP datagram takes up including the header. The maximum size that an IP datagram can be is 65,535 octet

Identification - The Identification is a unique number assigned to a datagram fragment to help in the reassembly of fragmented datagrams.
Flags - Bit 0 is always 0 and is reserved. Bit 1 indicates whether a datagram can be fragmented (0) or not (1). Bit 2 indicates to the receiving unit whether the fragment is the last one in the datagram (1) or if there are still more fragments to come (0).

Frag Offset - in units of 8 octets (64 bits) this specifies a value for each data fragment in the reassembly process. Different sized Maximum Transmission Units (MTUs) can be used throughout the Internet.
TTL - the time that the datagram is allowed to exist on the network. A router that processes the packet decrements this by one. Once the value reaches 0, the packet is discarded.

Protocol - Layer 4 protocol sending the datagram, UDP uses the number 17, TCP uses 6, ICMP uses 1, IGRP uses 88 and OSPF uses 89.
Header Checksum - error control for the header *only*.

IP Options - this field is for testing, debugging and security.
Padding - there is padding added sometimes just to make sure that the datagram is confined within a 32 bit boundary in multiples of 32 bits.

SA: SOURCE ADDESS
This is IP address of the sender IP datagram
DA: Destination address
This is the IP address of the reciever

Chapter 19
Network Layer: Logical Addressing

Stephen Kim

IPV4 ADDRESSES

An IPv4 address is a 32-bit address that uniquely and universally defines the connection of a device (for example, a computer or a router) to the Internet.

Ipv4 addresses are **unique**. Each address defines **one , and only one connection** to the Internet. Two devices on the Intrenet can never have the same address at the same time.

EX: If network layer has **M** connections to the Internet ,It needs to have **M** addresses.

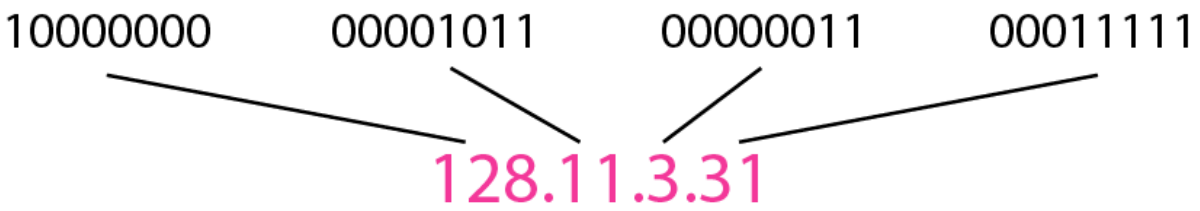
IPv4 addresses are univesal in the sense that the addressing system must be accepted by any host that wants to be connected to the Internet.

ADDESS SPACE

- A Protocol such as IPv4 that defines addresses has an address space.
- An address space is the total number of addresses used by the protocol.
- If a protocol has N-bits to define an address, the address space is 2 power of N

IPv4 Address

- The IPv4 addresses are unique and universal.
- An IPv4 address is 32 bits long.
  - The address space of IPv4 is 2^{32} (4,294,967,296)
  - Notation.
    - Binary notation
    - Dotted-decimal notation



Example 19.1

Change the following IPv4 addresses from binary notation to dotted-decimal notation.

a. 10000001 00001011 00001011 11101111

b. 11000001 10000011 00011011 11111111

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. 193.131.27.255

Example 19.2

Change the following IPv4 addresses from dotted-decimal notation to binary notation.

a. 111.56.45.78

b. 221.34.7.82

Solution

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

a. 01101111 00111000 00101101 01001110

b. 11011101 00100010 00000111 01010010

Example 19.3

Find the error, if any, in the following IPv4

a. 111.56.045.78

b. 221.34.7.8.20

c. 75.45.301.14

d. 11100010.23.14.67

Solution

a. There must be no leading zero (045).

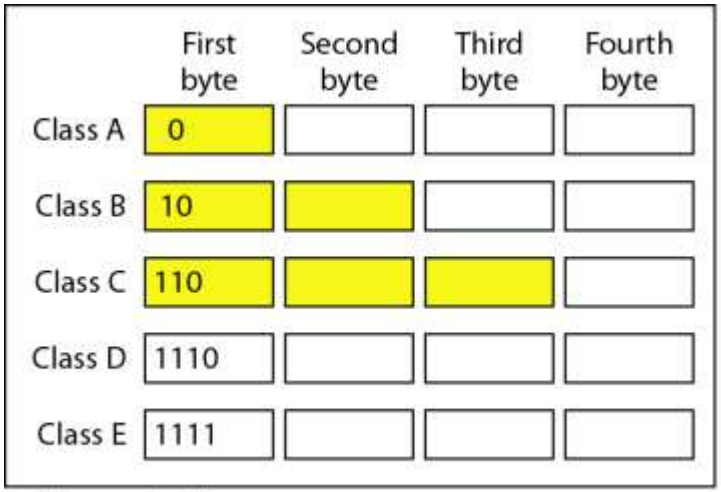
b. There can be no more than four numbers.

c. Each number needs to be less than or equal to 255.

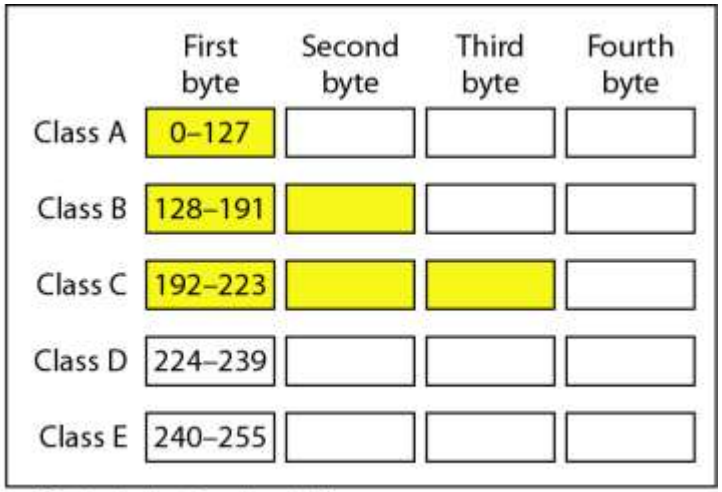
d. A mixture of binary notation and dotted-decimal notation is not allowed.

Classful Addressing

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



a. Binary notation



b. Dotted-decimal notation

Example 19.4

Find the class of each address.

a. **0**0000001 00001011 00001011 11101111

b. **110**00001 10000011 00011011 11111111

c. **14**.23.120.8

d. **252**.5.15.111

Solution

a. ***The first bit is 0. This is a class A address.***

b. ***The first 2 bits are 1; the third bit is 0. This is a class C address.***

c. ***The first byte is 14; the class is A.***

d. ***The first byte is 252; the class is E.***

Classes and Blocks

- The classful addressing wastes a large part of the address space.
  - Class A:
  - Class B:
  - Class C:
  - Class D:

Class	*Number of Blocks*	*Block Size*	*Application*
A	128	16,777,216	Unicast
B	16,384	65,536	Unicast
C	2,097,152	256	Unicast
D	1	268,435,456	Multicast
E	1	268,435,456	Reserved

Structure of IPv4 Address

- Consists of Net ID and Host ID.

Class	*Binary*	*Dotted-Decimal*	*CIDR*
A	11111111 00000000 00000000 00000000	255.0.0.0	/8
B	11111111 11111111 00000000 00000000	255.255.0.0	/16
C	11111111 11111111 11111111 00000000	255.255.255.0	/24

- Mask
  - 32-bit number of contiguous 1's followed by contiguous 0's.
  - To help to find the net ID and the host ID.

Use of IPv4 Address

- Subnetting
  - Divide a large address block into smaller sub-groups.
  - Use of flexible net mask.
- Supernetting
  - Exhausted class A and B address space
  - Huge demand for class B address space
  - To combine several contiguous address spaces into a larger single address space

Classless Addressing

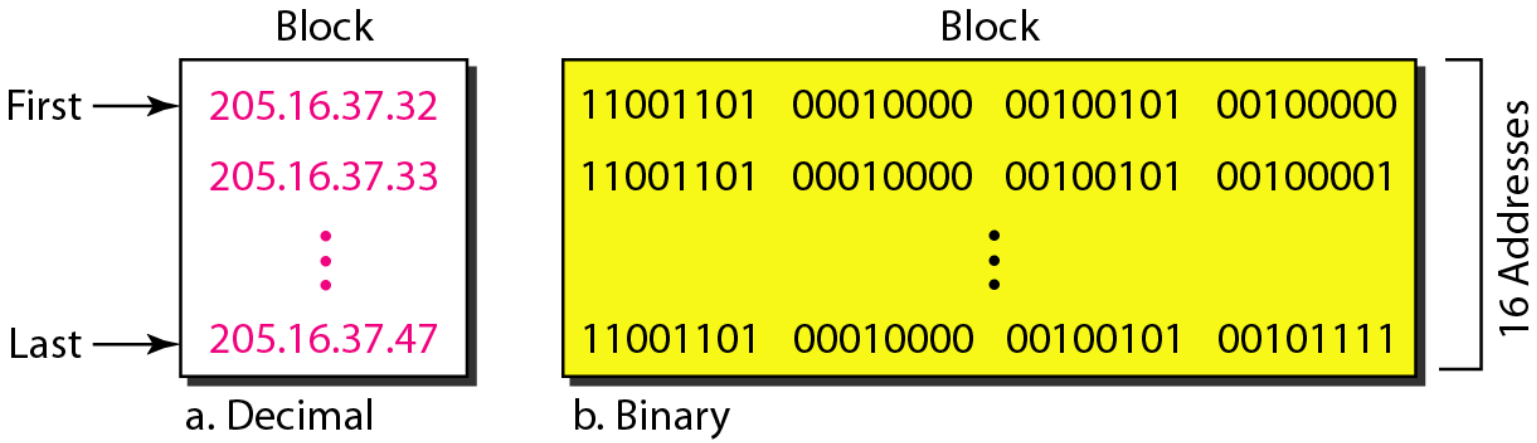
- To overcome the depletion of address space.
- Restriction
  - The addresses in a block must be contiguous.
  - The number of addresses in a block must be a power of 2.
  - The first address must be evenly divisible by the number of address.
- Mask
  - Consists of *n* consecutive 1's followed by zeros.
  - *n* can be any number b/w 0 and 32.
- Tips:
  - In IPv4 addressing, a block of addresses can be defined as x.y.z.t /n, in which x.y.z.t defines one of the addresses and the /n defines the mask.
  - The first address in the block can be found by setting the rightmost 32 − n bits to 0s.
  - The last address in the block can be found by setting the rightmost 32 − n bits to 1s.
  - The number of addresses in the block can be found by using the formula 2^{32-n}.

Example 19.5

Figure 19.3 shows a block of addresses, in both binary and dotted-decimal notation, granted to a small business that needs 16 addresses.

We can see that the restrictions are applied to this block. The addresses are contiguous. The number of addresses is a power of 2 ($16 = 2^4$), and the first address is divisible by 16. The first address, when converted to a decimal number, is 3,440,387,360, which when divided by 16 results in 215,024,210.

Figure 19.3 *A block of 16 addresses granted to a small organization*



Example 19.6

A block of addresses is granted to a small organization. We know that one of the addresses is 205.16.37.39/28. What is the first address in the block?

Solution

The binary representation of the given address is

11001101 00010000 00100101 00100111

If we set 32−28 rightmost bits to 0, we get

11001101 00010000 00100101 0010000

or

205.16.37.32.

This is actually the block shown in Figure 19.3.

Example 19.7

Find the last address for the block in Example 19.6.

Solution

The binary representation of the given address is

11001101 00010000 00100101 00100111

If we set 32 − 28 rightmost bits to 1, we get

11001101 00010000 00100101 00101111

or

205.16.37.47

This is actually the block shown in Figure 19.3.

Example 19.8

Find the number of addresses in Example 19.6.

Solution

The value of n is 28, which means that number of addresses is 2^{32-28} or 16.

Example 19.9

Another way to find the first address, the last address, and the number of addresses is to represent the mask as a 32-bit binary (or 8-digit hexadecimal) number. This is particularly useful when we are writing a program to find these pieces of information. In Example 19.5 the /28 can be represented as

11111111 11111111 11111111 11110000

(twenty-eight 1s and four 0s).

Find

a. The first address

b. The last address

c. The number of addresses.

Example 19.9 (continued)

Solution

a. The first address can be found by ANDing the given addresses with the mask. ANDing here is done bit by bit. The result of ANDing 2 bits is 1 if both bits are 1s; the result is 0 otherwise.

Address:	11001101	00010000	00100101	00100111
Mask:	**11111111**	**11111111**	**11111111**	**11110000**
First address:	11001101	00010000	00100101	00100000

Example 19.9 (continued)

b. The last address can be found by ORing the given addresses with the complement of the mask. Oring here is done bit by bit. The result of ORing 2 bits is 0 if both bits are 0s; the result is 1 otherwise. The complement of a number is found by changing each 1 to 0 and each 0 to 1.

Address:	11001101	00010000	00100101	00100111
Mask complement:	**00000000**	**00000000**	**00000000**	**00001111**
Last address:	11001101	00010000	00100101	00101111

Example 19.9 (continued)

c. The number of addresses can be found by complementing the mask, interpreting it as a decimal number, and adding 1 to it.

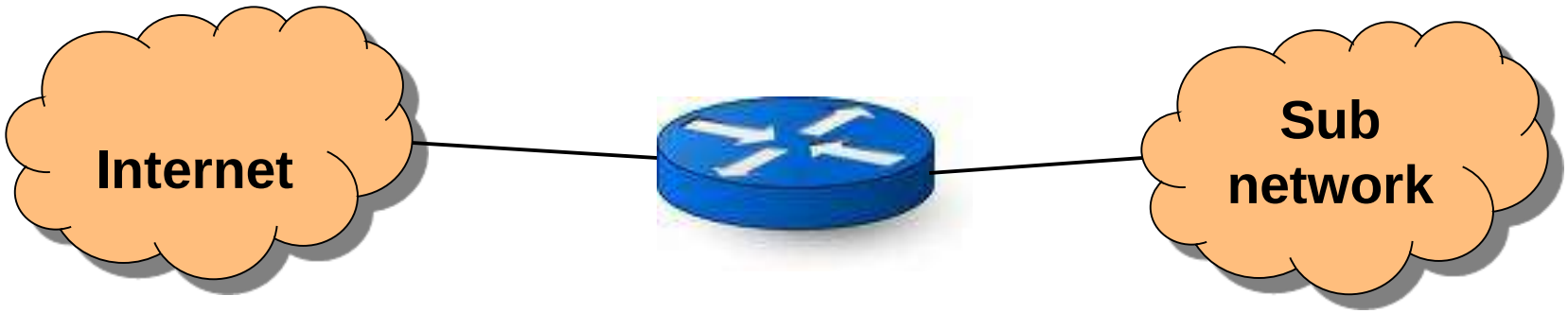
Mask complement:	00000000 00000000 00000000 00001111
Number of addresses:	15 + 1 = 16

Special Addresses

- Network address
  - The first address in a block is normally not assigned to any device; it is used as the network address that represents the organization to the rest of the world.
- Broadcast address
  - The last address in a block is used for broadcasting to all devices under the network.

Routing in IPv4

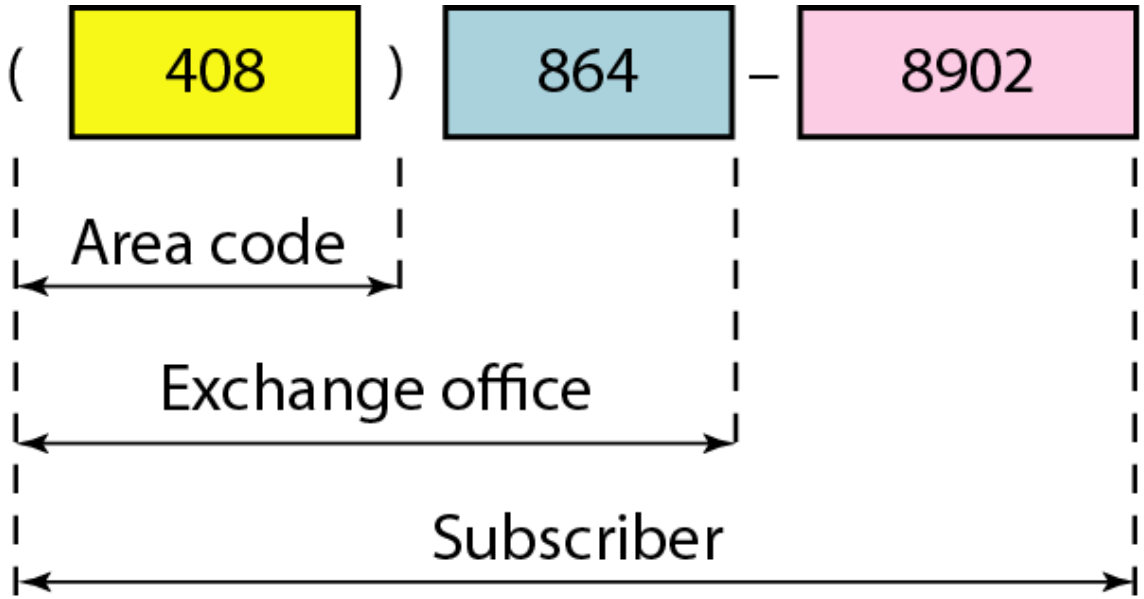
- A router has two addresses
  - An address through which the device inside of the router can be accessed.
  - Another address belongs to the granted block (sub-network).



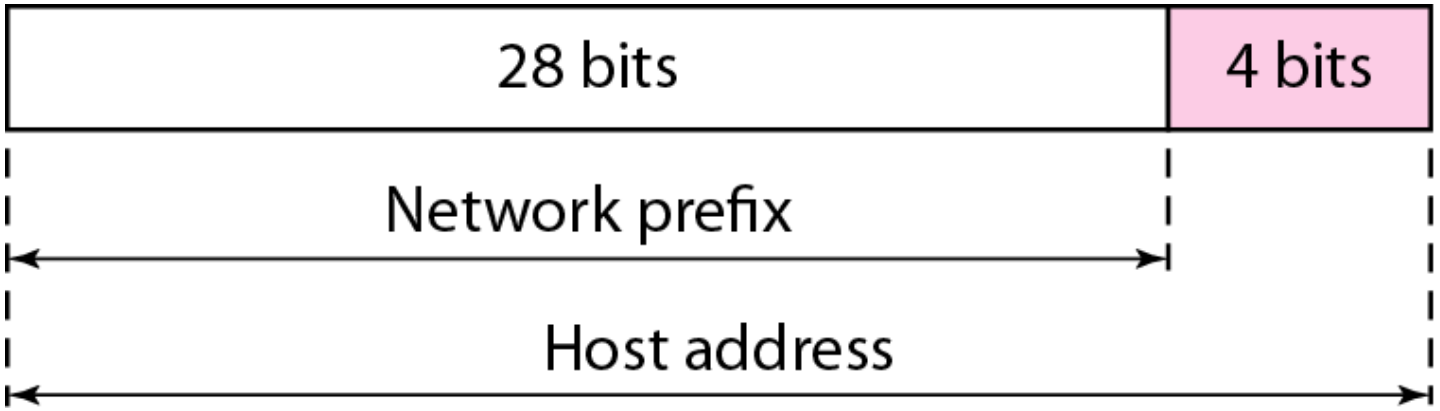
Hierarchy of IPv4 Addressing

- Each address in the block can be considered as a two-level hierarchical structure: the leftmost *n* bits (prefix) define the network; the rightmost 32 − *n* bits define the host.
- Why Hierarchy?

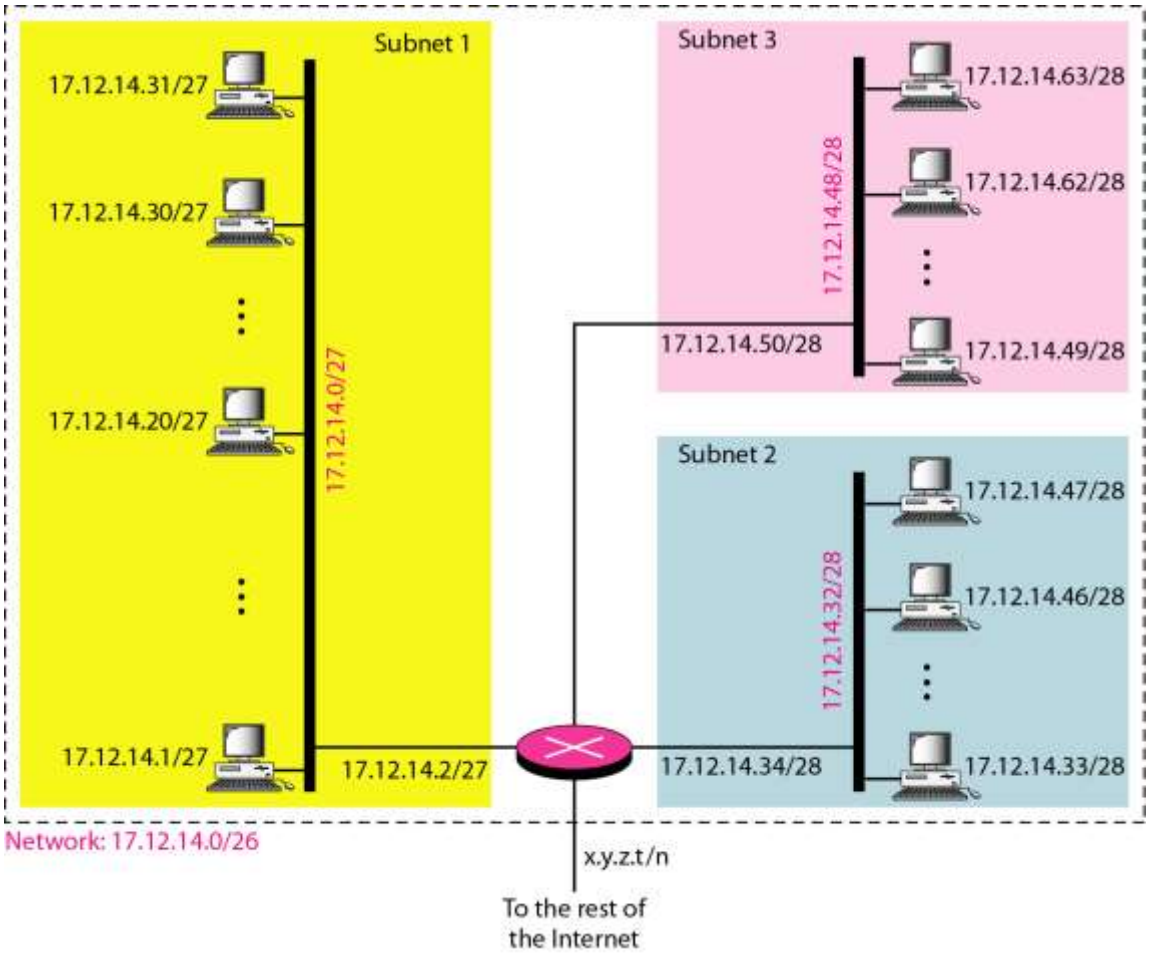
Figure 19.5 *Two levels of hierarchy in an IPv4 address*



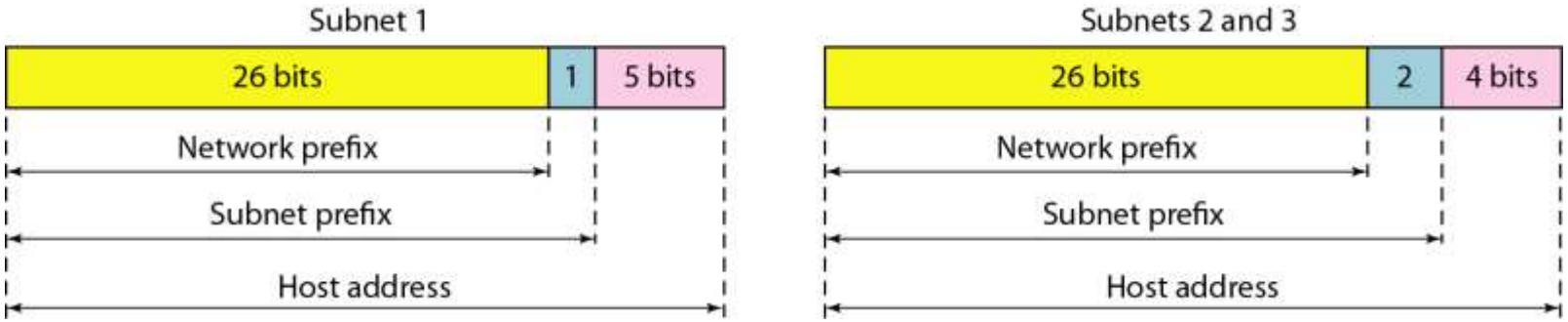
Two Level of Hierarchy



Three Level of Hierarchy



Three Level of Hierarchy



Example 19.10

An ISP is granted a block of addresses starting with 190.100.0.0/16 (65,536 addresses). The ISP needs to distribute these addresses to three groups of customers as follows:

a. The first group has 64 customers; each needs 256 addresses.

b. The second group has 128 customers; each needs 128 addresses.

c. The third group has 128 customers; each needs 64 addresses.

Design the subblocks and find out how many addresses are still available after these allocations.

Example 19.10 (continued)

Solution

Figure 19.9 shows the situation.

Group 1

For this group, each customer needs 256 addresses. This means that 8 (log2 256) bits are needed to define each host. The prefix length is then 32 − 8 = 24. The addresses are

1st Customer:	*190.100.0.0/24*	*190.100.0.255/24*
2nd Customer:	*190.100.1.0/24*	*190.100.1.255/24*
. . .		
64th Customer:	*190.100.63.0/24*	*190.100.63.255/24*

Total = 64 × 256 = 16,384

Example 19.10 (continued)

Group 2

For this group, each customer needs 128 addresses. This means that 7 (log2 128) bits are needed to define each host. The prefix length is then 32 − 7 = 25. The addresses are

1st Customer:	*190.100.64.0/25*	*190.100.64.127/25*
2nd Customer:	*190.100.64.128/25*	*190.100.64.255/25*
. . .		
128th Customer:	*190.100.127.128/25*	*190.100.127.255/25*

Total = 128 × 128 = 16,384

Example 19.10 (continued)

Group 3

For this group, each customer needs 64 addresses. This means that 6 ($\log_2 64$) bits are needed to each host. The prefix length is then 32 − 6 = 26. The addresses are

1st Customer:	*190.100.128.0/26*	*190.100.128.63/26*
2nd Customer:	*190.100.128.64/26*	*190.100.128.127/26*
. . .		
128th Customer:	*190.100.159.192/26*	*190.100.159.255/26*

Total = 128 × 64 = 8192

Number of granted addresses to the ISP: 65,536
Number of allocated addresses by the ISP: 40,960
Number of available addresses: 24,576

Network Address Translation (NAT)

- Benefits
  - Use of a single IP address among many devices in a network
  - Use of a dynamic IP address for home user for sharing
- Private Addresses

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}

Figure 19.10 *A NAT implementation*

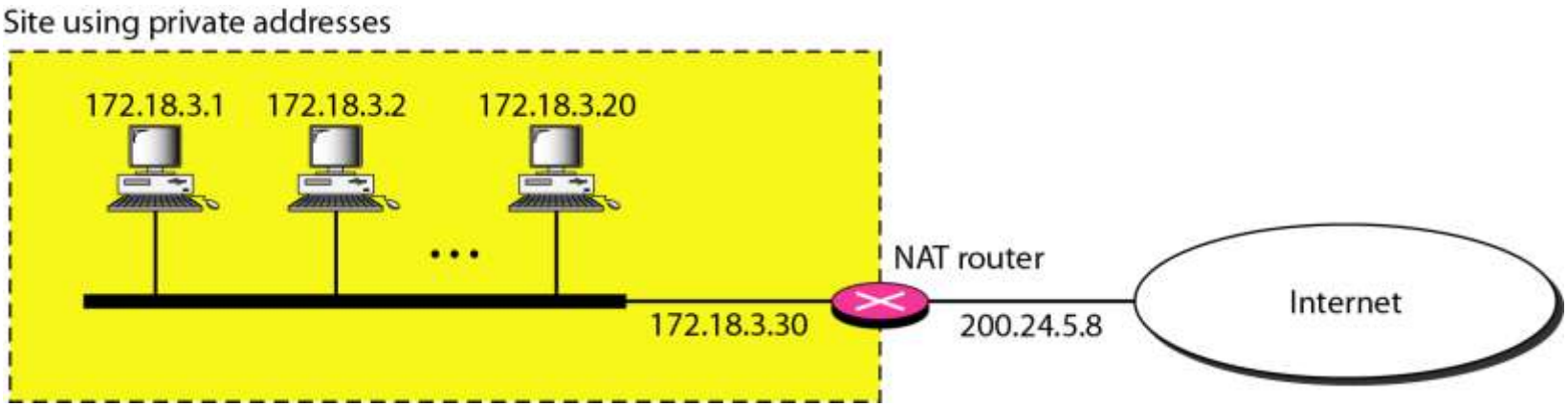


Figure 19.11 *Addresses in a NAT*

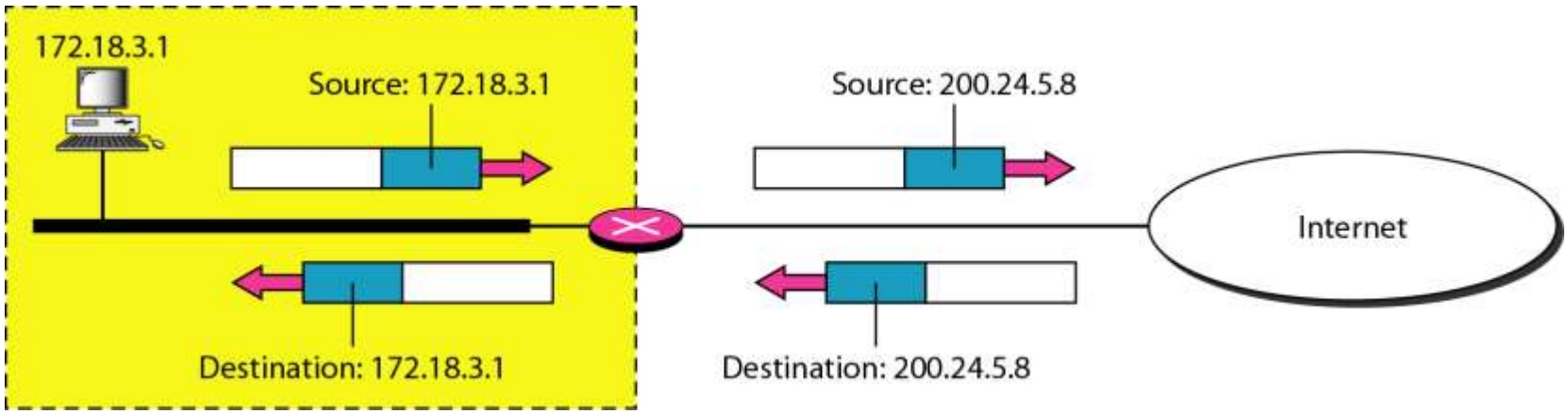


Figure 19.12 *NAT address translation*

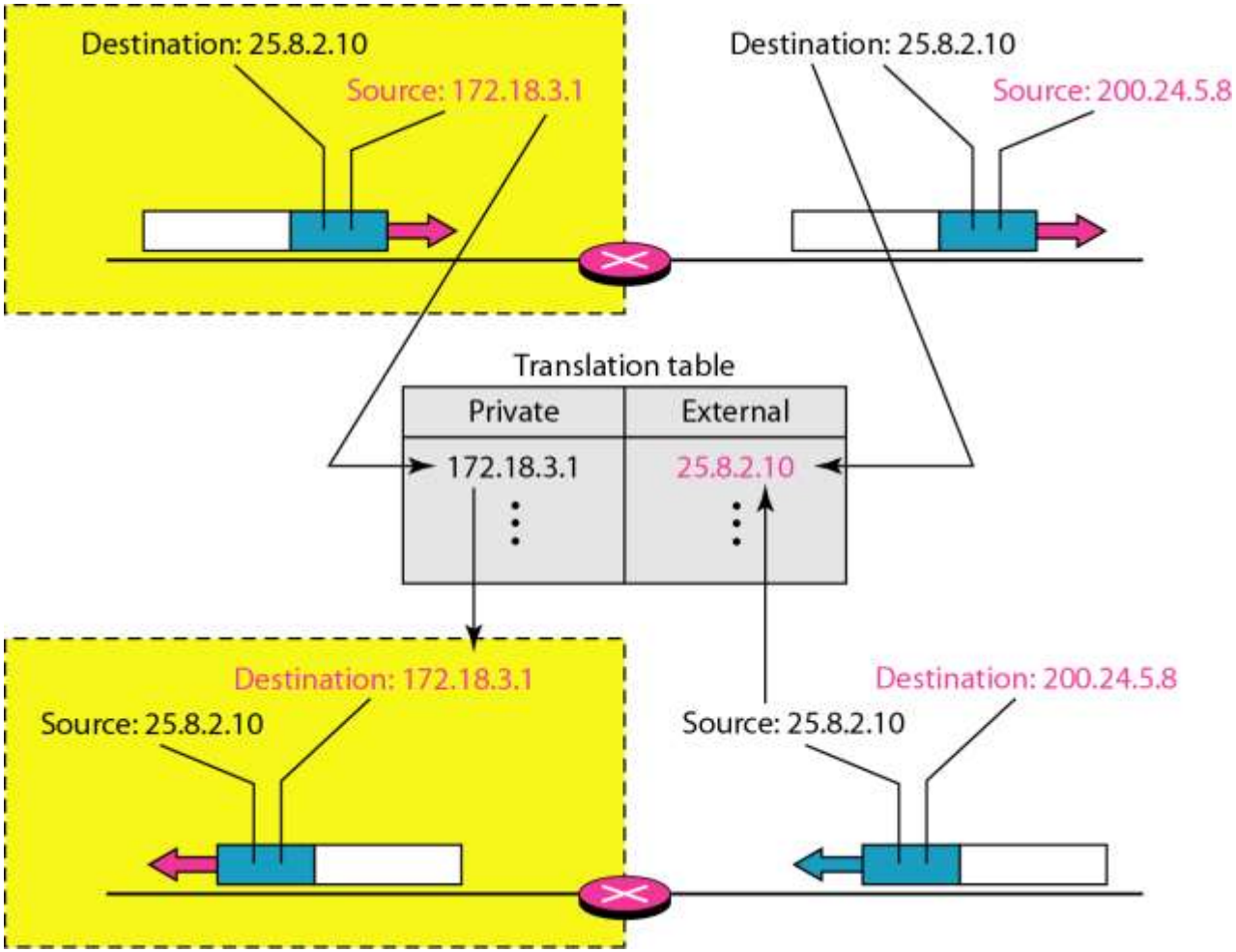
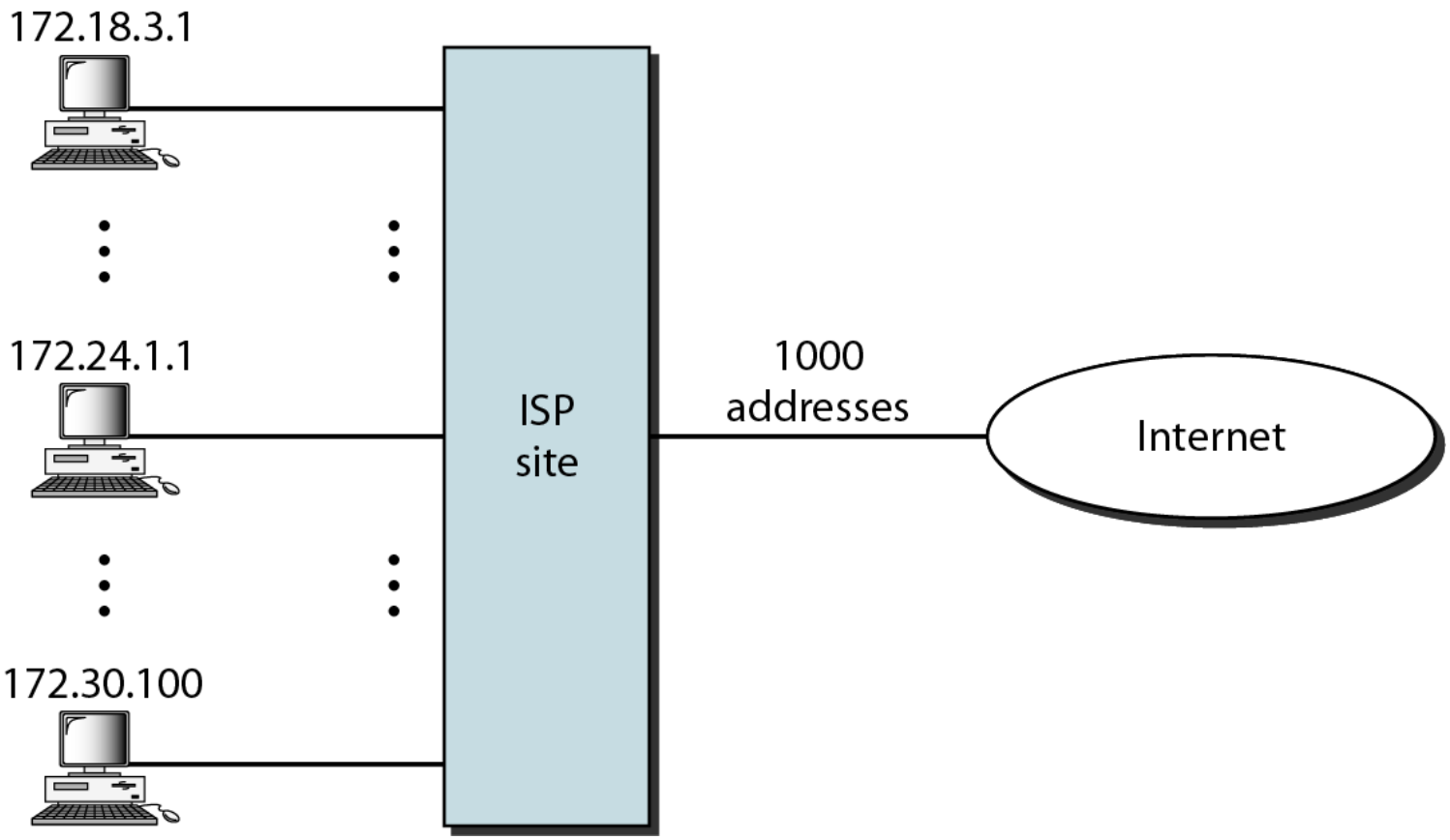


Table 19.4 *Five-column translation table*

Private Address	*Private Port*	*External Address*	*External Port*	*Transport Protocol*
172.18.3.1	1400	25.8.3.2	80	TCP
172.18.3.2	1401	25.8.3.2	80	TCP
. . .	. . .	. . .	. . .	. . .

Figure 19.13 *An ISP and NAT*

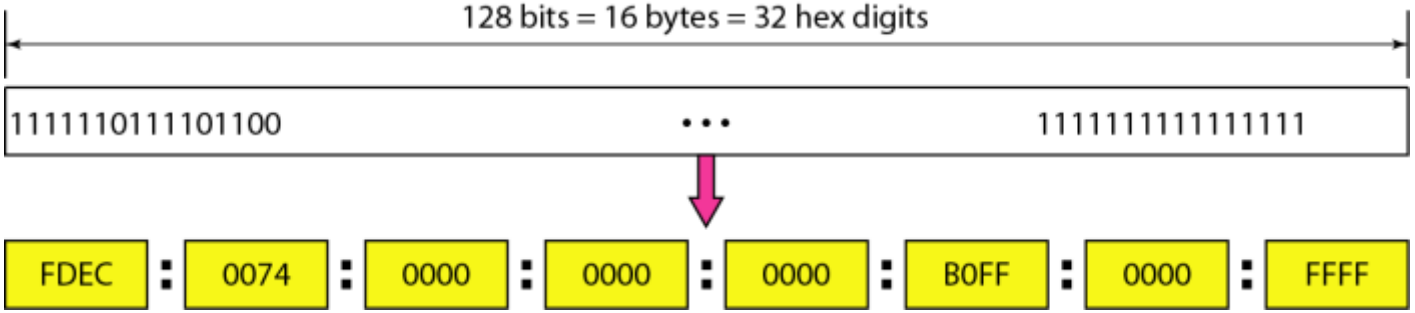


IPV6 ADDRESSES

Despite all short-term solutions, address depletion is still a long-term problem for the Internet. This and other problems in the IP protocol itself have been the motivation for IPv6.

IPv6 Address

- An IPv6 address is 128 bits long (16-byte).
- Hexadecimal Colon Notation



- Abbreviation

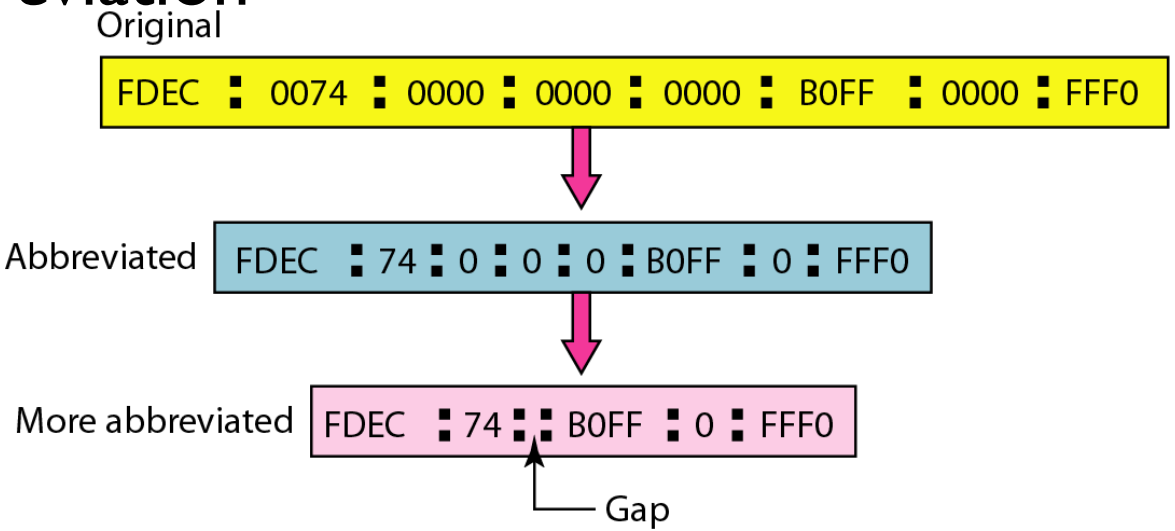
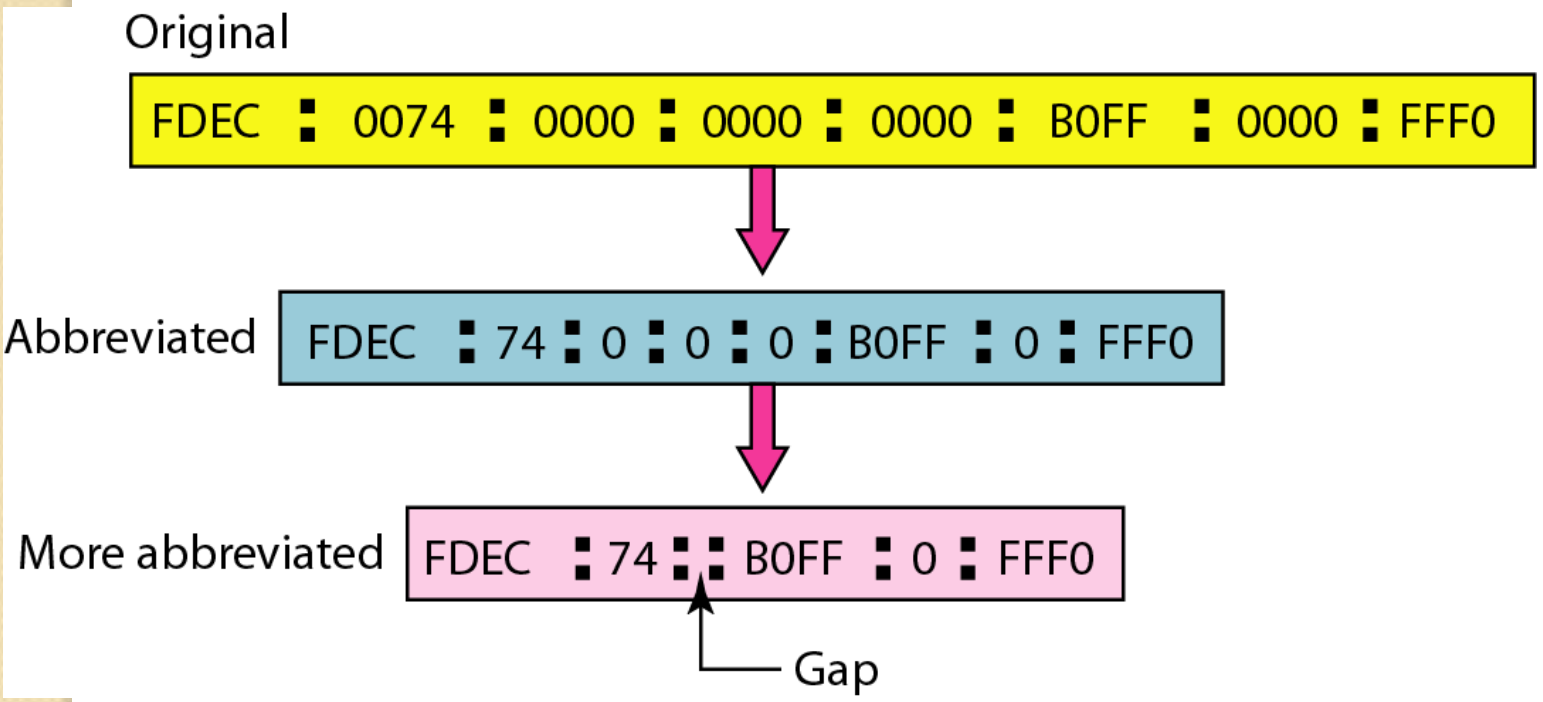


Figure 19.15 *Abbreviated IPv6 addresses*



Example 19.11

Expand the address 0:15::1:12:1213 to its original.

Solution

We first need to align the left side of the double colon to the left of the original pattern and the right side of the double colon to the right of the original pattern to find how many 0s we need to replace the double colon.

```
xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx
  0:   15:                   :    1:   12:1213
```

This means that the original address is.

```
0000:0015:0000:0000:0000:0001:0012:1213
```

Structure of IPv6 Address

- Type prefix
  - For categorization,
  - Variable length,
  - No partial conflict among the different prefix
  -

Type prefixes for IPv6 addresses

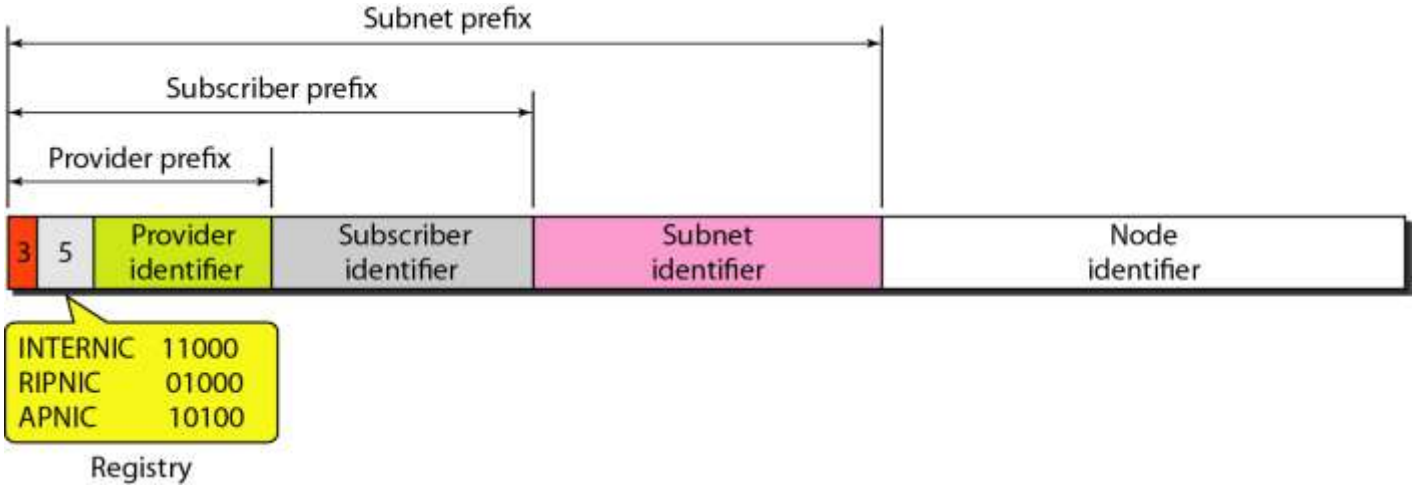
Type Prefix	*Type*	*Fraction*
0000 0000	Reserved	1/256
0000 0001	Unassigned	1/256
0000 001	ISO network addresses	1/128
0000 010	IPX (Novell) network addresses	1/128
0000 011	Unassigned	1/128
0000 1	Unassigned	1/32
0001	Reserved	1/16
001	Reserved	1/8
010	**Provider-based unicast addresses**	**1/8**

Type prefixes for IPv6 addresses

Type Prefix	*Type*	*Fraction*
011	Unassigned	1/8
100	Geographic-based unicast addresses	1/8
101	Unassigned	1/8
110	Unassigned	1/8
1110	Unassigned	1/16
1111 0	Unassigned	1/32
1111 10	Unassigned	1/64
1111 110	Unassigned	1/128
1111 1110 0	Unassigned	1/512
1111 1110 10	Link local addresses	1/1024
1111 1110 11	Site local addresses	1/1024
1111 1111	Multicast addresses	1/256

Unicast

- For a single computer
- Two types of unicast addresses
  - Geographically based
  - Provider-based
- Fields
  - Type ID (3-bit), Registry ID (5-bit), Provider ID (16-bit), Subscriber ID (24-bit), Subnet ID (32-bit), Node ID (48-bit)



Multicast address in IPv6

- For a group of hosts
- To deliver packets to each member

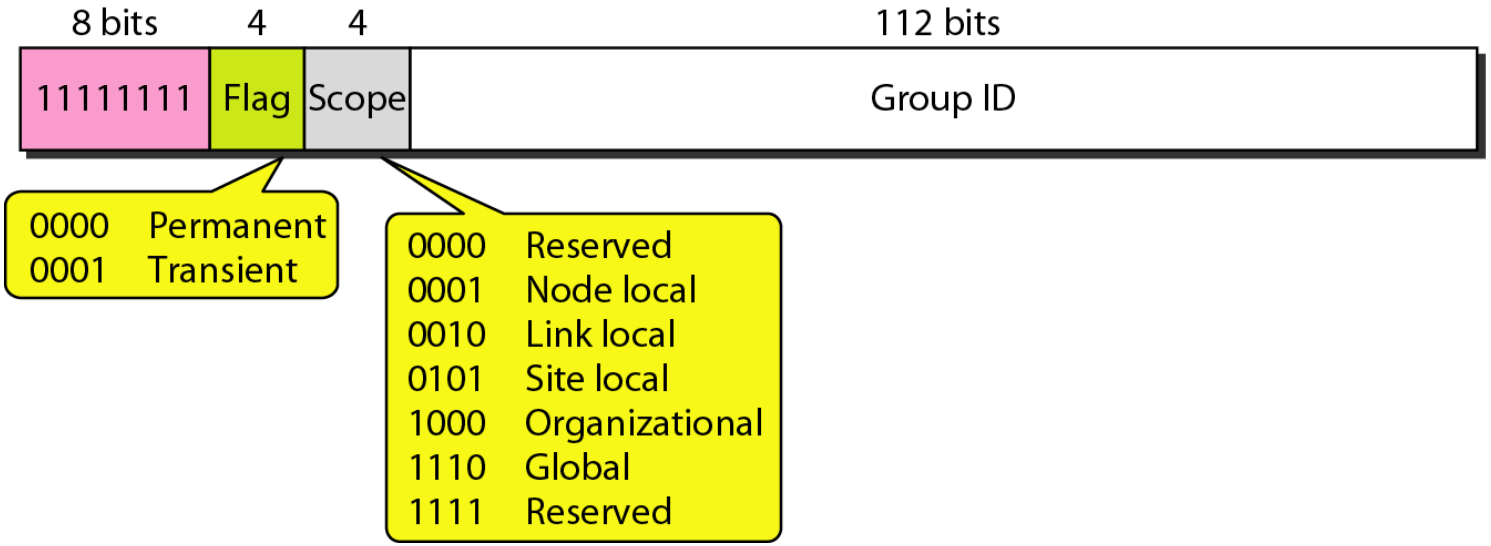
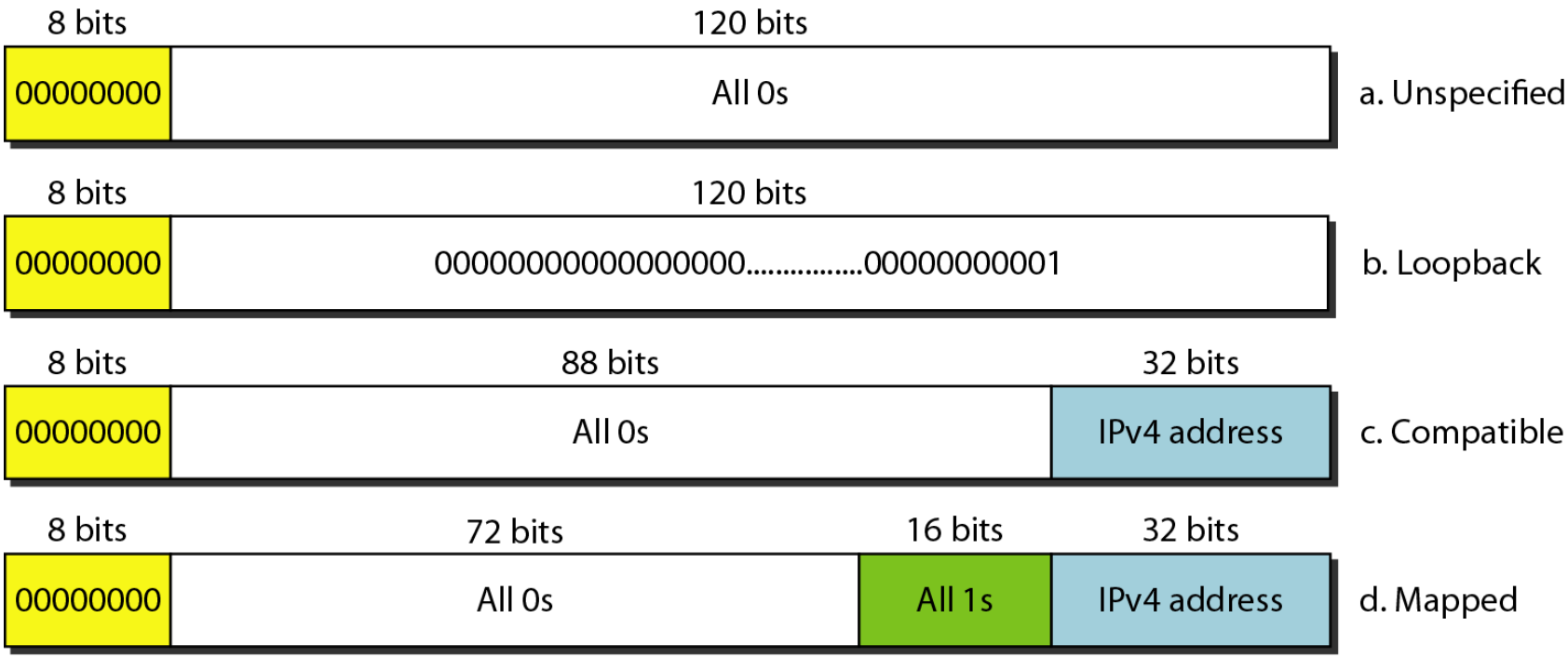
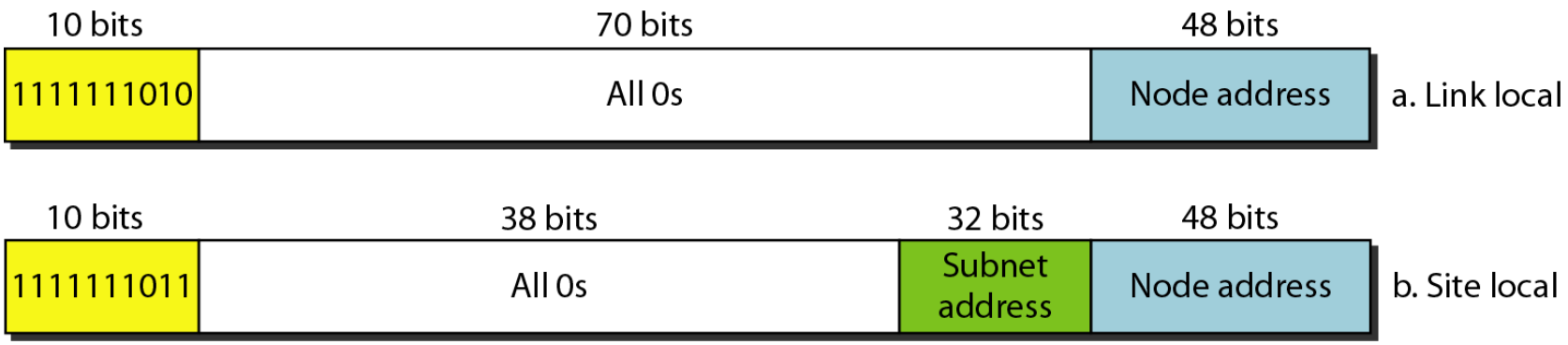


Figure 19.18 *Reserved addresses in IPv6*



Local addresses in IPv6

- to use IPv6 without connecting to the global Internet.



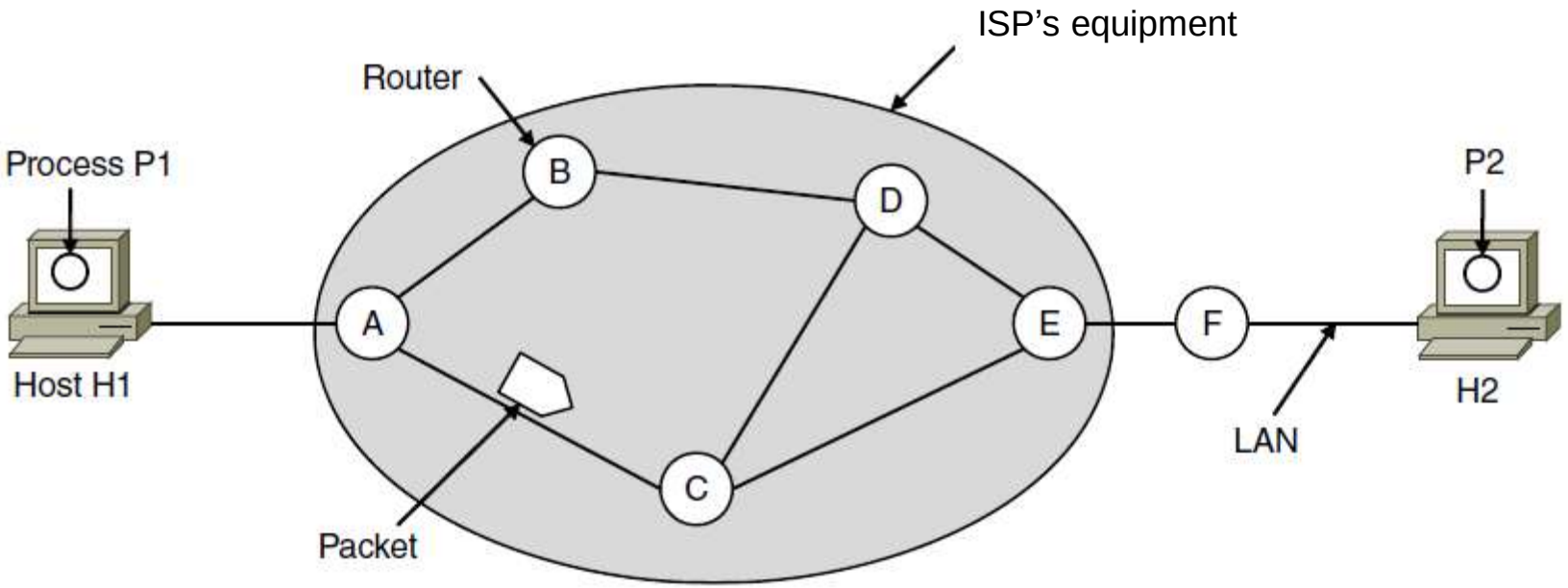
The Network Layer

Chapter 5

Network Layer Design Issues

- Store-and-forward packet switching
- Services provided to transport layer
- Implementation of connectionless service
- Implementation of connection-oriented service
- Comparison of virtual-circuit and datagram networks

Store-and-Forward Packet Switching

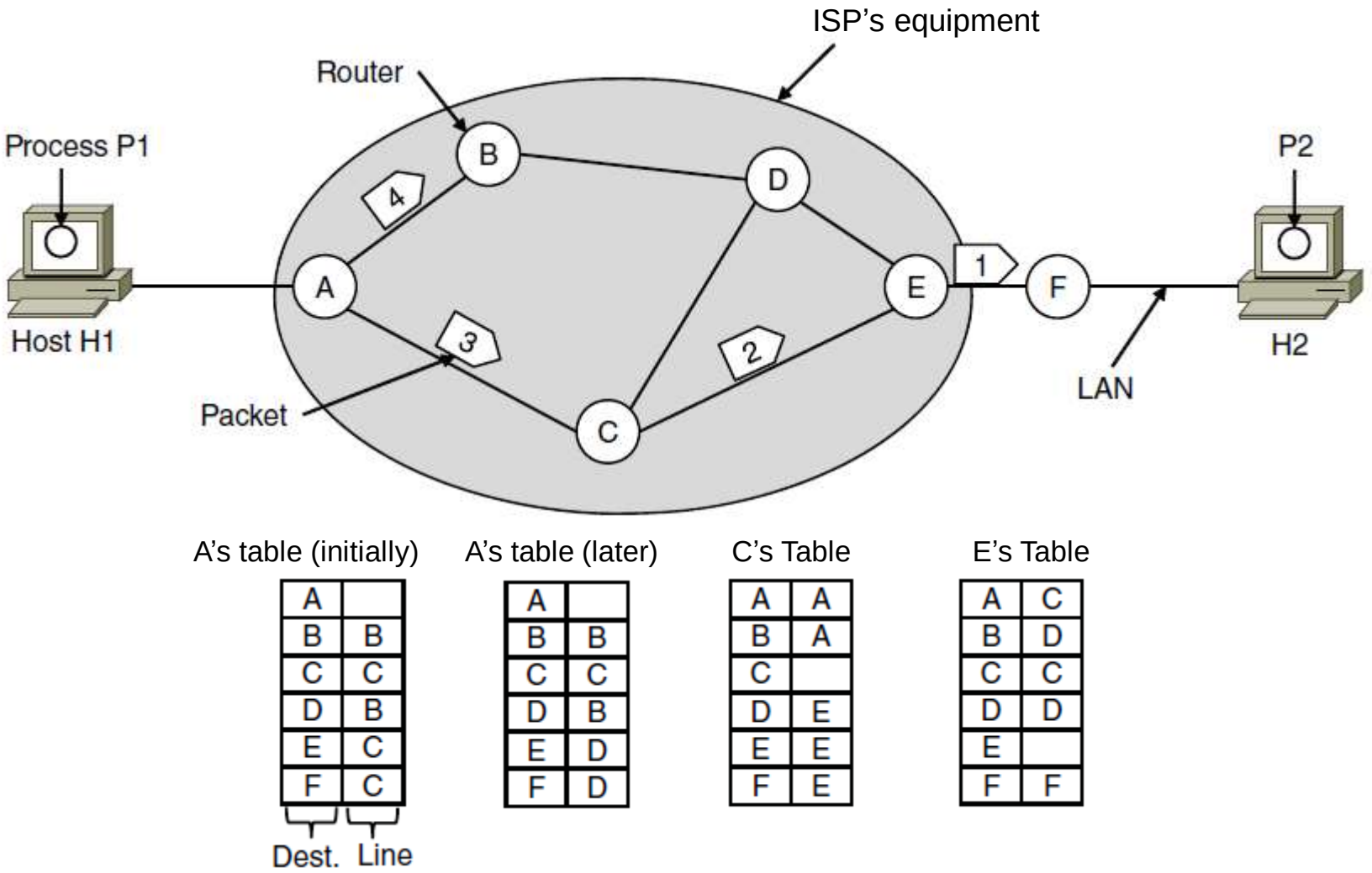


The environment of the network layer protocols.

Services Provided to the Transport Layer

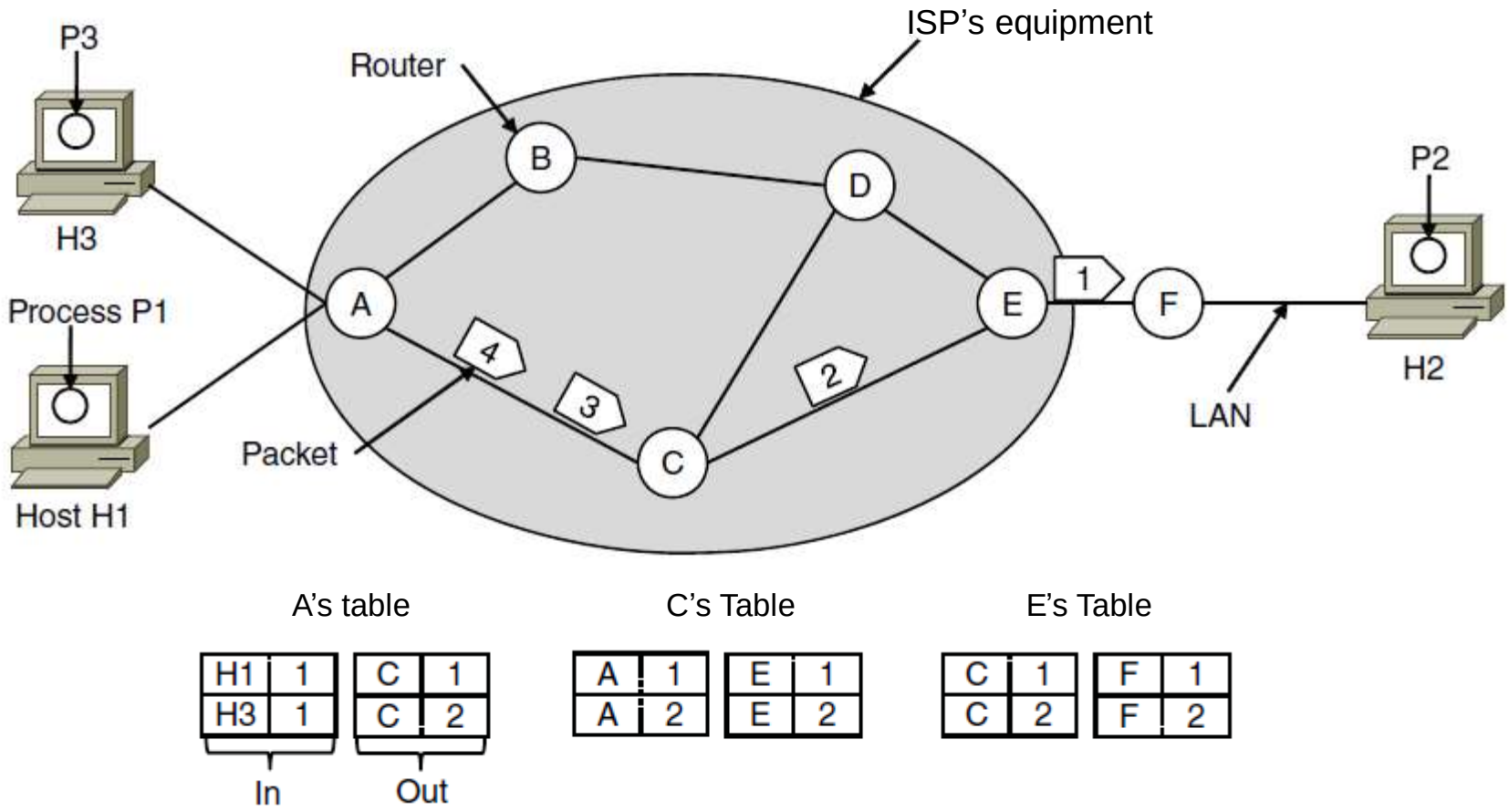
1. Services independent of router technology.
2. Transport layer shielded from number, type, topology of routers.
3. Network addresses available to transport layer use uniform numbering plan
   - even across LANs and WANs

Implementation of Connectionless Service



Routing within a datagram network

Implementation of Connection-Oriented Service



Routing within a virtual-circuit network

Computer Networks, Fifth Edition by Andrew Tanenbaum and David Wetherall, © Pearson Education-Prentice Hall, 2011

Comparison of Virtual-Circuit and Datagram Networks

Issue	Datagram network	Virtual-circuit network
Circuit setup	Not needed	Required
Addressing	Each packet contains the full source and destination address	Each packet contains a short VC number
State information	Routers do not hold state information about connections	Each VC requires router table space per connection
Routing	Each packet is routed independently	Route chosen when VC is set up; all packets follow it
Effect of router failures	None, except for packets lost during the crash	All VCs that passed through the failed router are terminated
Quality of service	Difficult	Easy if enough resources can be allocated in advance for each VC
Congestion control	Difficult	Easy if enough resources can be allocated in advance for each VC

Comparison of datagram and virtual-circuit networks

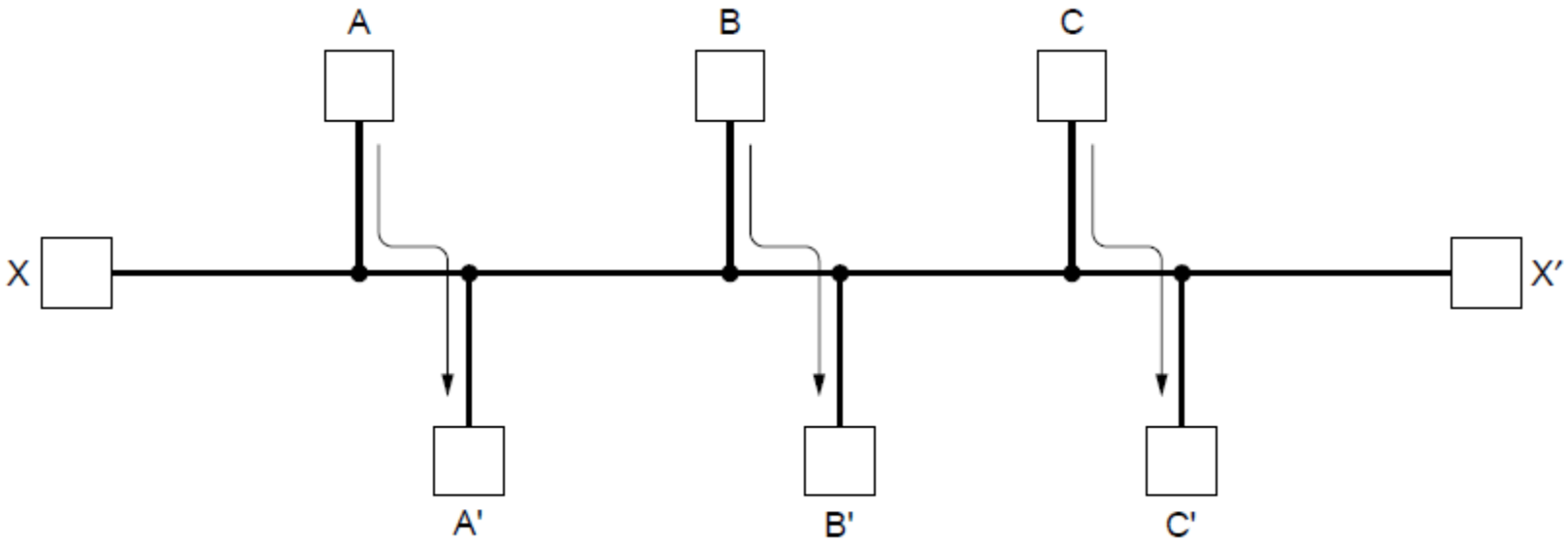
Routing Algorithms (1)

- Optimality principle
- Shortest path algorithm
- Flooding
- Distance vector routing
- Link state routing
- Routing in ad hoc networks

Routing Algorithms (2)

- Broadcast routing
- Multicast routing
- Anycast routing
- Routing for mobile hosts
- Routing in ad hoc networks

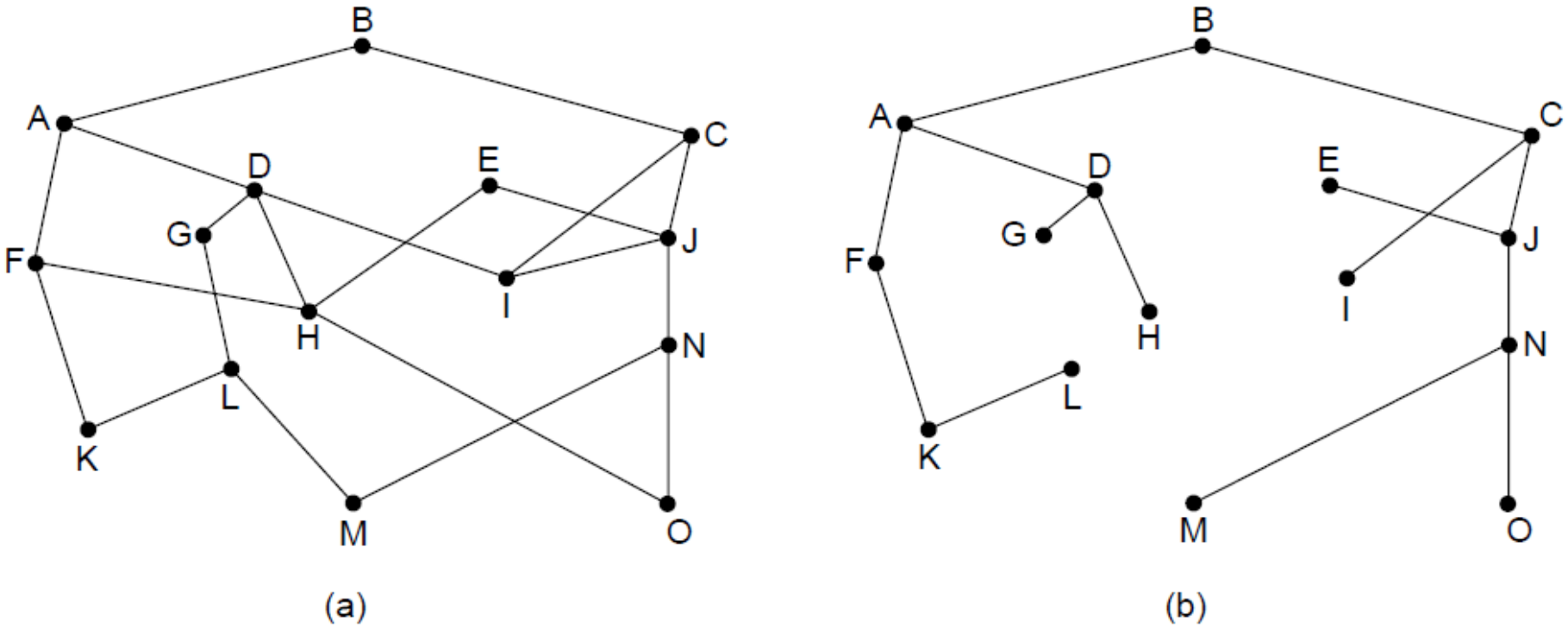
Fairness vs. Efficiency



Network with a conflict between fairness and efficiency.

Computer Networks, Fifth Edition by Andrew Tanenbaum and David Wetherall, © Pearson Education-Prentice Hall, 2011

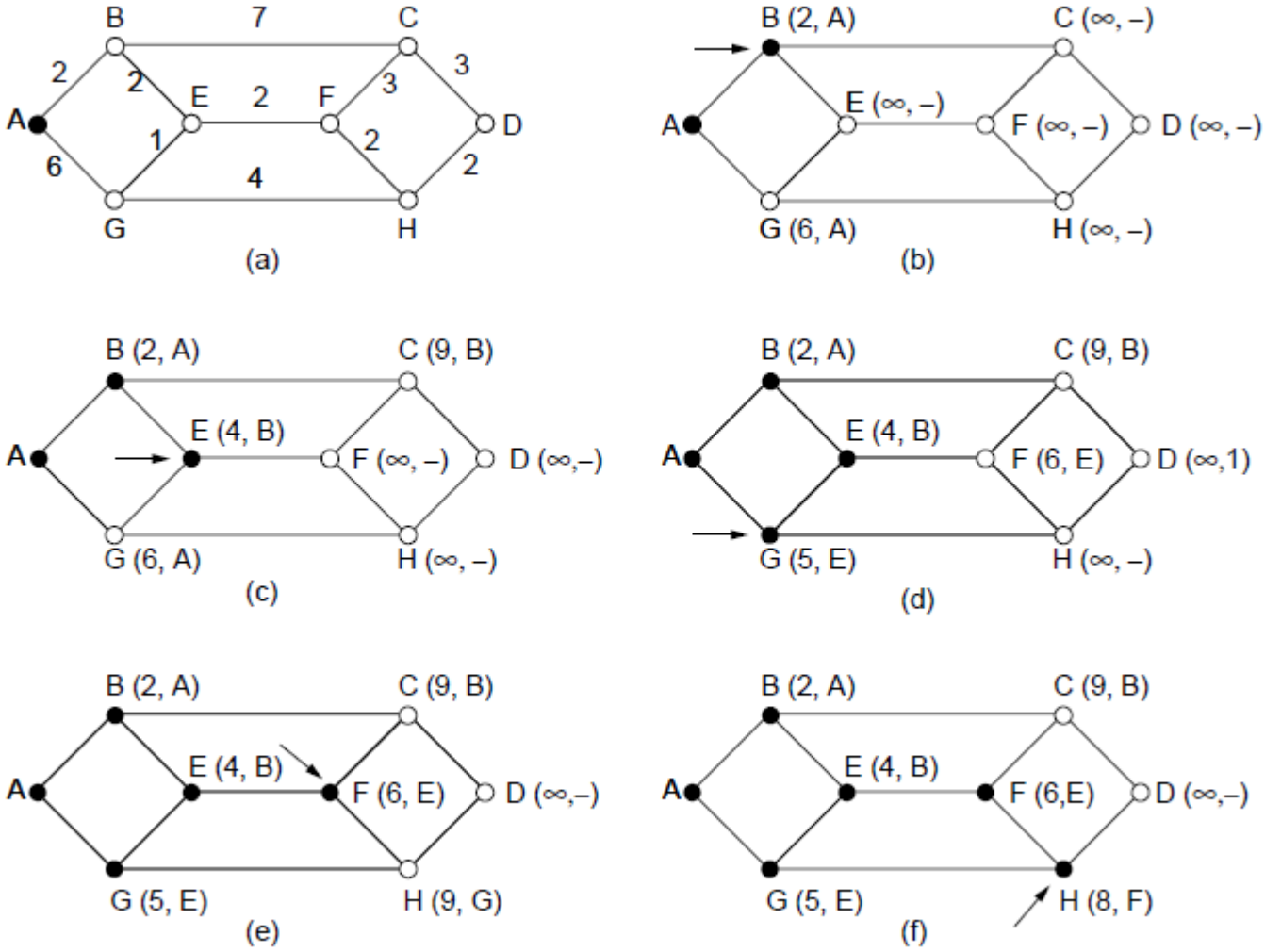
The Optimality Principle



(a) A network. (b) A sink tree for router *B.*

Computer Networks, Fifth Edition by Andrew Tanenbaum and David Wetherall, © Pearson Education-Prentice Hall, 2011

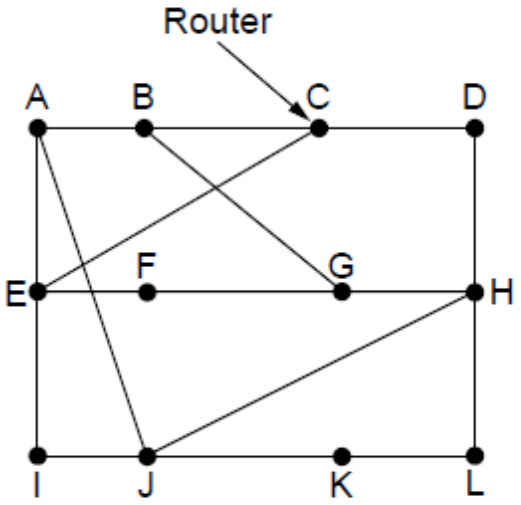
Shortest Path Algorithm (1)



The first five steps used in computing the shortest path from *A to D.* The arrows indicate the working node

Computer Networks, Fifth Edition by Andrew Tanenbaum and David Wetherall, © Pearson Education-Prentice Hall, 2011

Distance Vector Routing



(a)

New estimated delay from J

To	A	I	H	K	Line
A	0	24	20	21	8 A
B	12	36	31	28	20 A
C	25	18	19	36	28 I
D	40	27	8	24	20 H
E	14	7	30	22	17 I
F	23	20	19	40	30 I
G	18	31	6	31	18 H
H	17	20	0	19	12 H
I	21	0	14	22	10 I
J	9	11	7	10	0 –
K	24	22	22	0	6 K
L	29	33	9	9	15 K

JA delay is 8 | JI delay is 10 | JH delay is 12 | JK delay is 6

Vectors received from J's four neighbors

New routing table for J

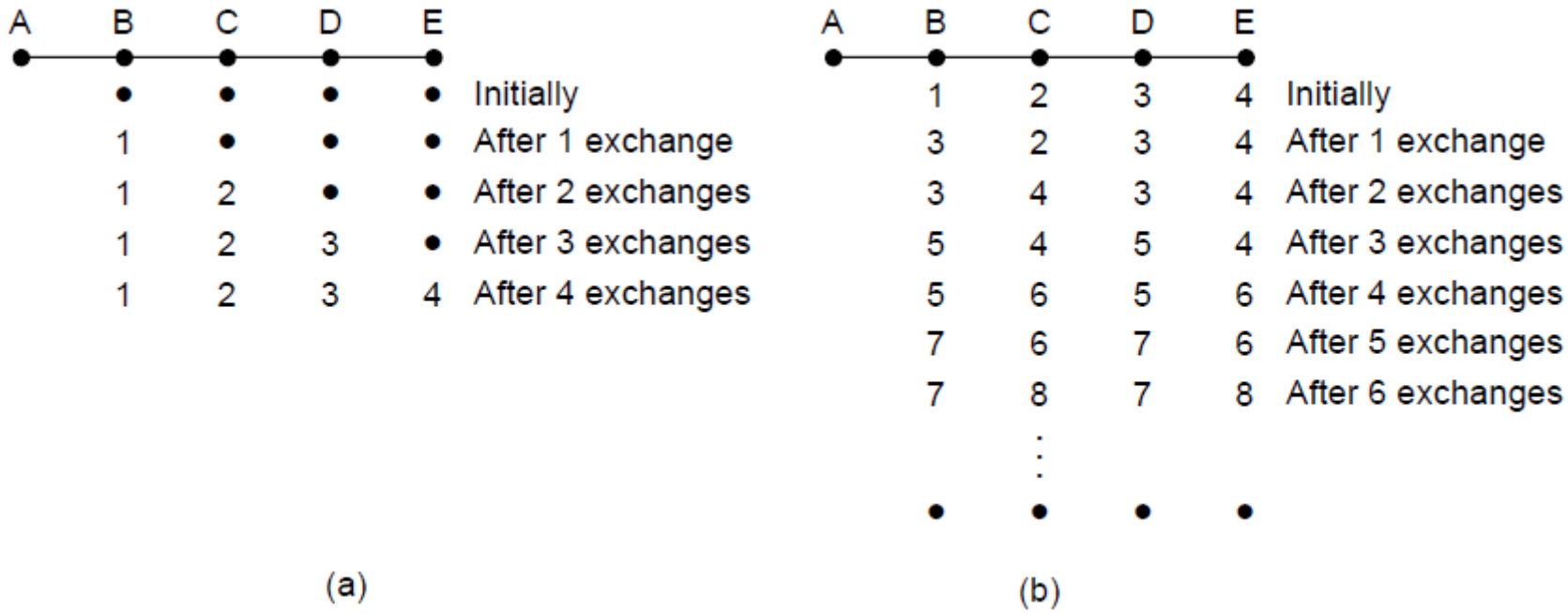
(b)

(a) A network.

(b) Input from *A, I, H, K, and the new routing table* for *J.*

Computer Networks, Fifth Edition by Andrew Tanenbaum and David Wetherall, © Pearson Education-Prentice Hall, 2011

The Count-to-Infinity Problem



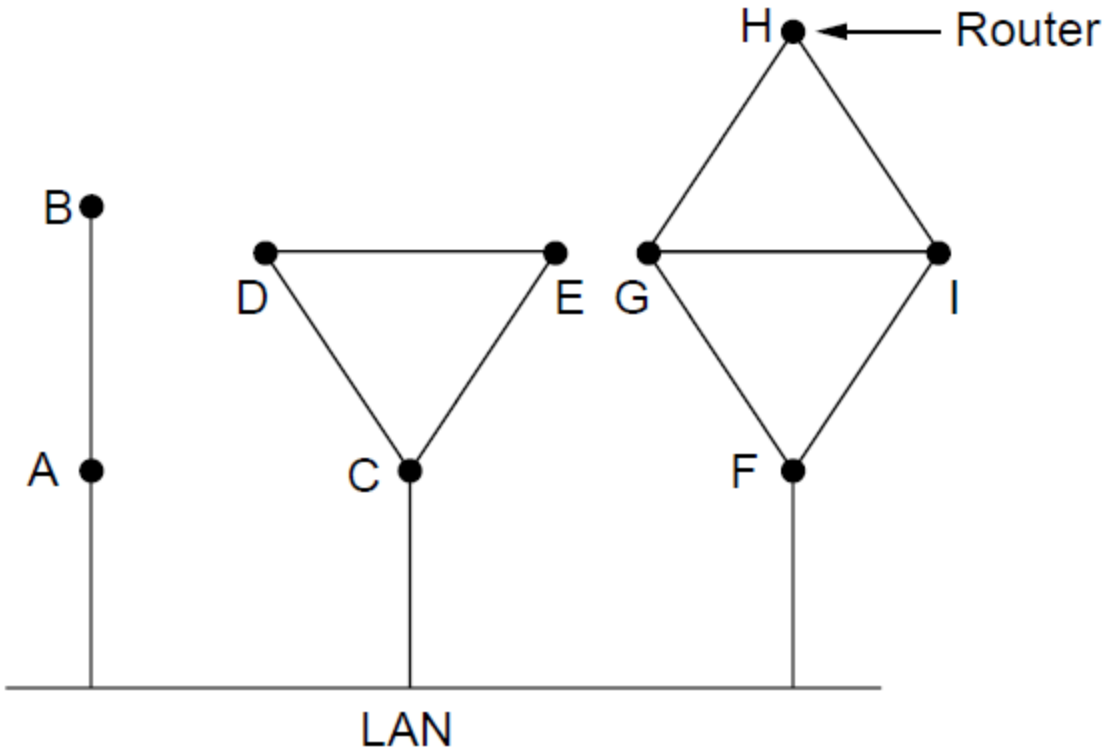
The count-to-infinity problem

Computer Networks, Fifth Edition by Andrew Tanenbaum and David Wetherall, © Pearson Education-Prentice Hall, 2011

Link State Routing

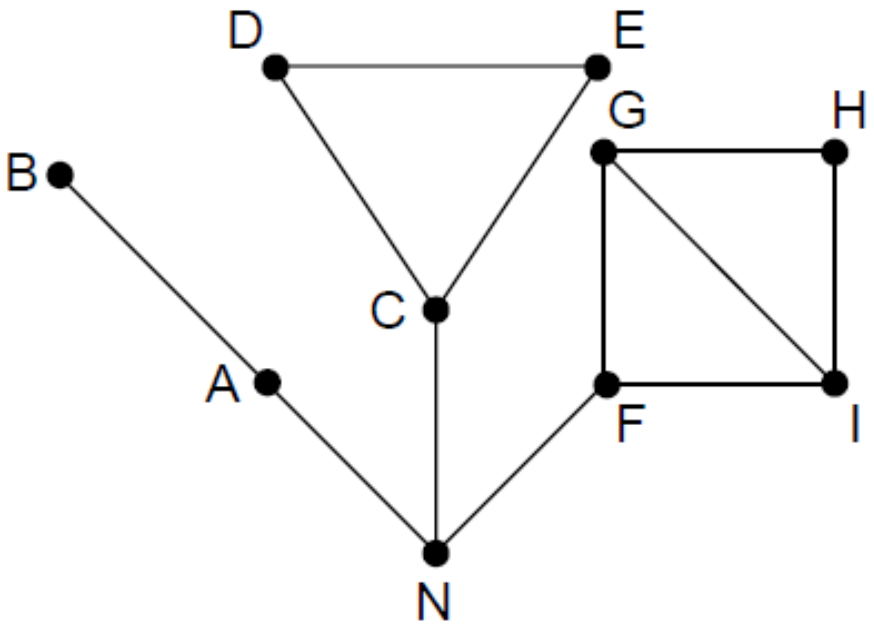
1. Discover neighbors, learn network addresses.
2. Set distance/cost metric to each neighbor.
3. Construct packet telling all learned.
4. Send packet to, receive packets from other routers.
5. Compute shortest path to every other router.

Learning about the Neighbors (1)



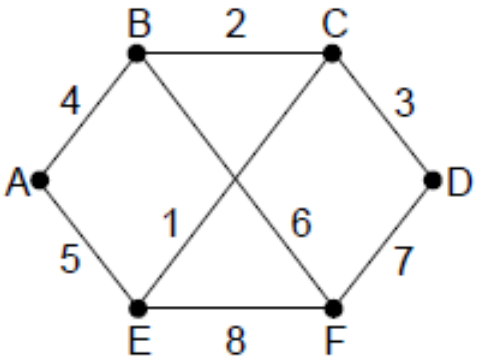
Nine routers and a broadcast LAN.

Learning about the Neighbors (2)



A graph model of previous slide.

Building Link State Packets



(a)

Link State Packets

A	B	C	D	E	F
Seq.	Seq.	Seq.	Seq.	Seq.	Seq.
Age	Age	Age	Age	Age	Age
B 4	A 4	B 2	C 3	A 5	B 6
E 5	C 2	D 3	F 7	C 1	D 7
	F 6	E 1		F 8	E 8

(b)

(a) A network. (b) The link state packets for this network.

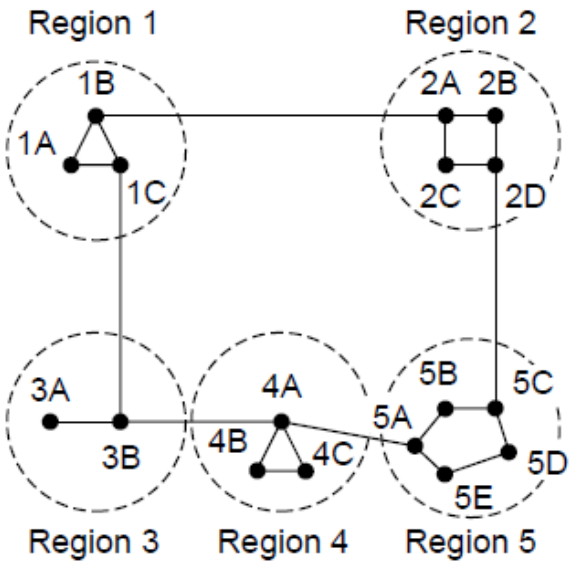
Computer Networks, Fifth Edition by Andrew Tanenbaum and David Wetherall, © Pearson Education-Prentice Hall, 2011

Distributing the Link State Packets

Source	Seq.	Age	Send flags A	Send flags C	Send flags F	ACK flags A	ACK flags C	ACK flags F	Data
A	21	60	0	1	1	1	0	0	
F	21	60	1	1	0	0	0	1	
E	21	59	0	1	0	1	0	1	
C	20	60	1	0	1	0	1	0	
D	21	59	1	0	0	0	1	1	

The packet buffer for router *B* in previous slide

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)

Hierarchical table for 1A

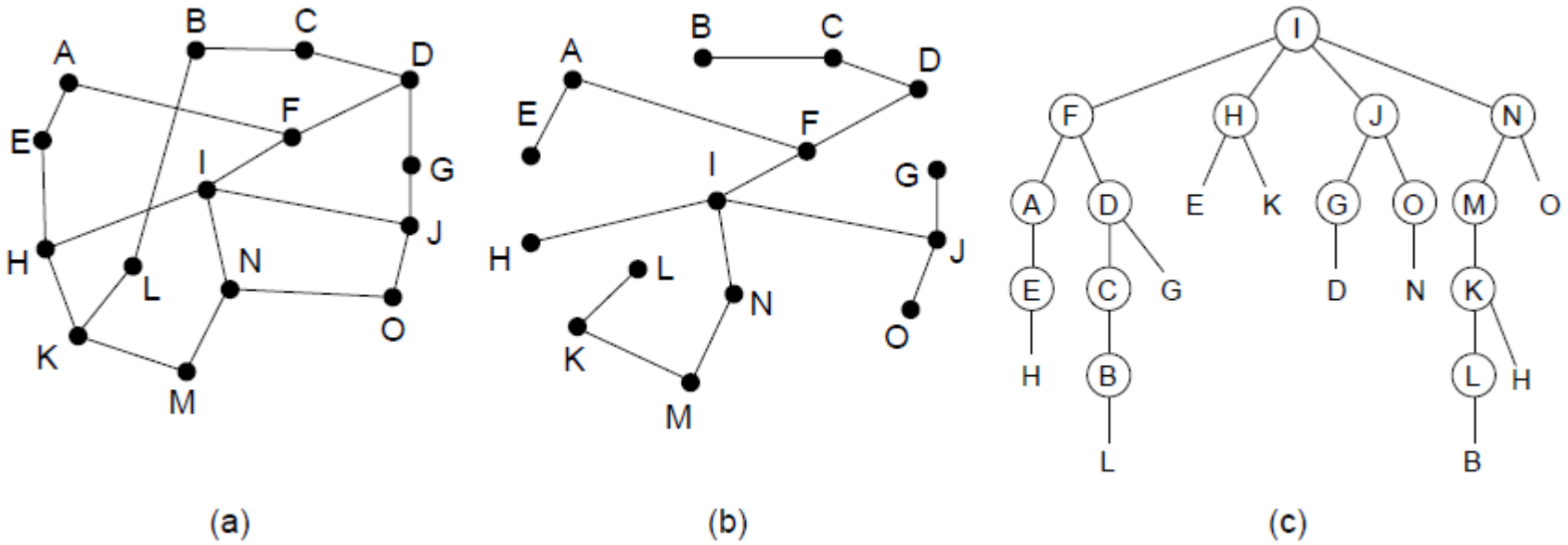
Dest.	Line	Hops
1A	–	–
1B	1B	1
1C	1C	1
2	1B	2
3	1C	2
4	1C	3
5	1C	4

(c)

Hierarchical routing.

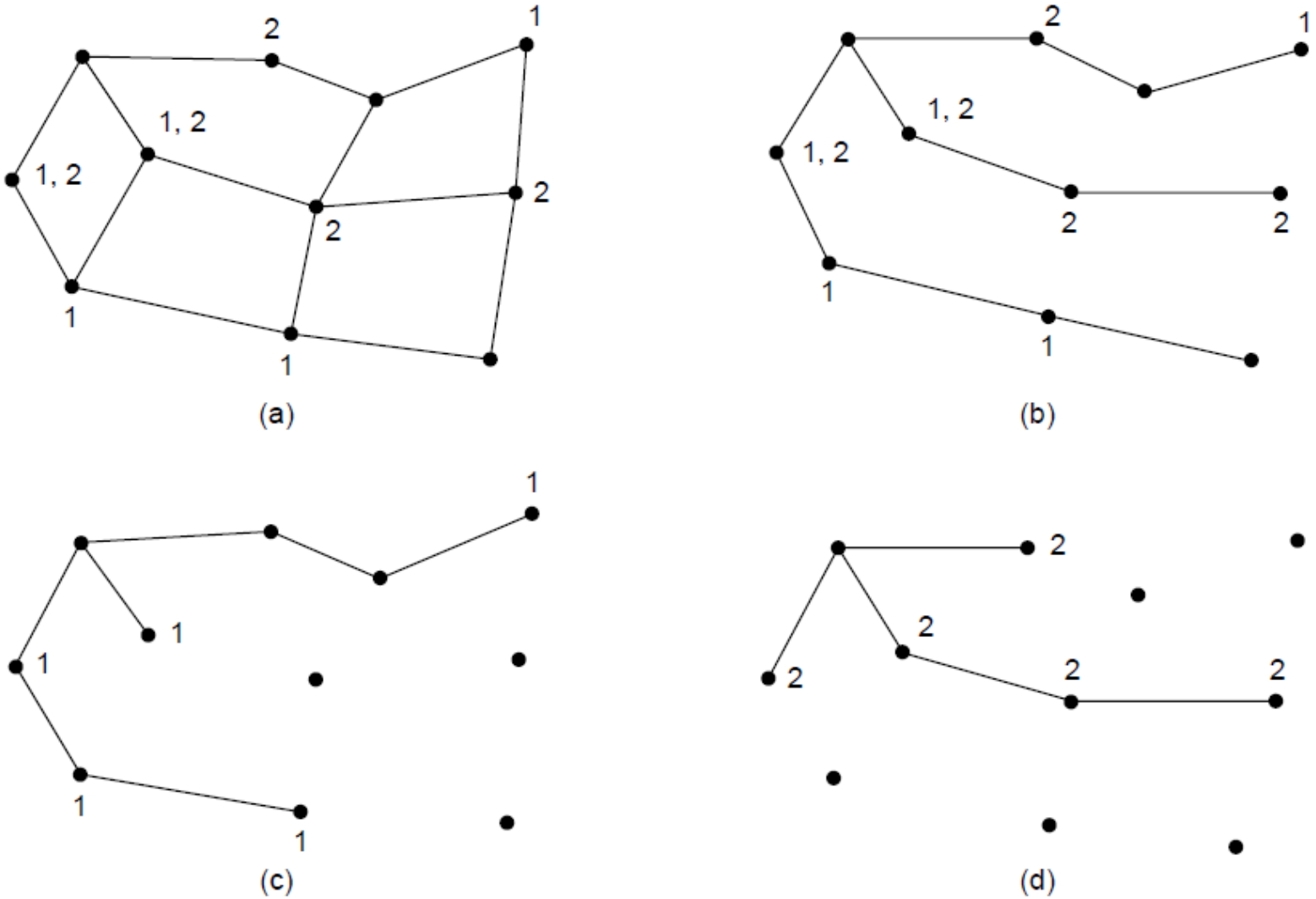
Computer Networks, Fifth Edition by Andrew Tanenbaum and David Wetherall, © Pearson Education-Prentice Hall, 2011

Broadcast Routing



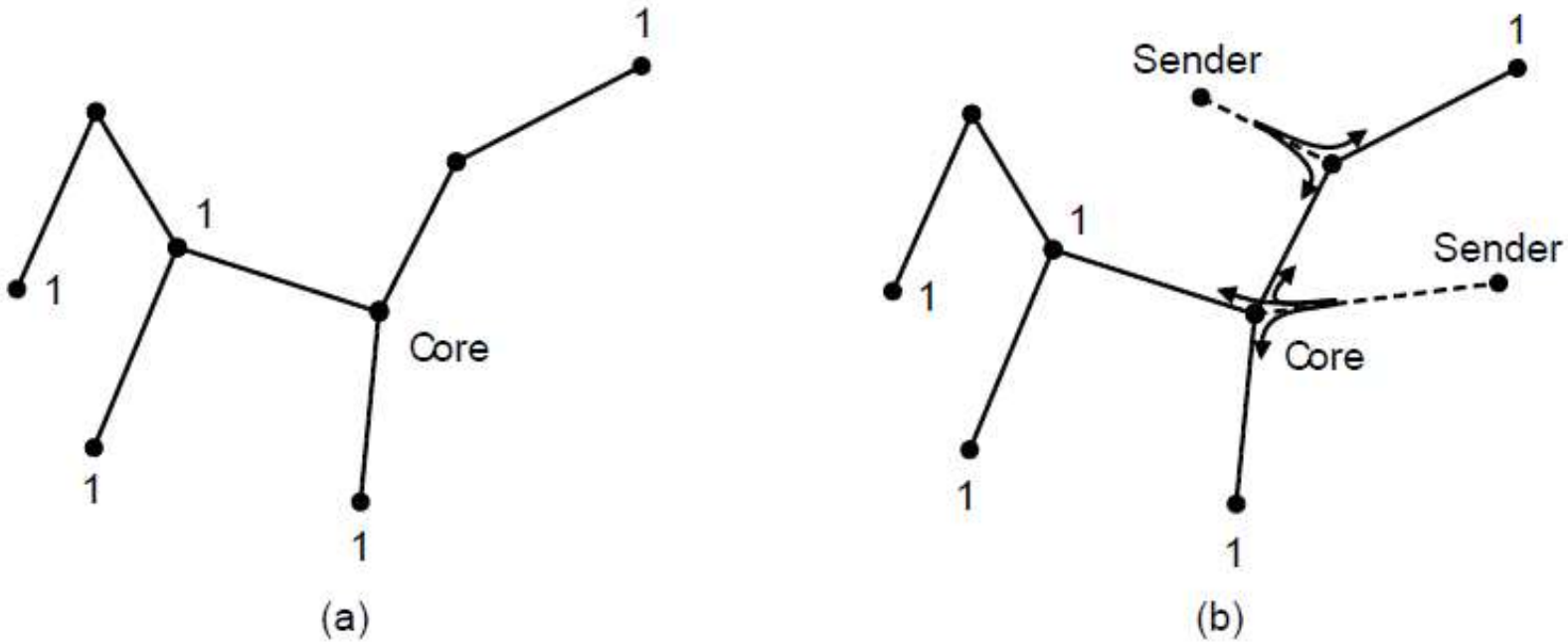
Reverse path forwarding. (a) A network. (b) A sink tree. (c) The tree built by reverse path forwarding.

Multicast Routing (1)



(a) A network. (b) A spanning tree for the leftmost router. (c) A multicast tree for group 1. (d) A multicast tree for group 2.

Multicast Routing (2)

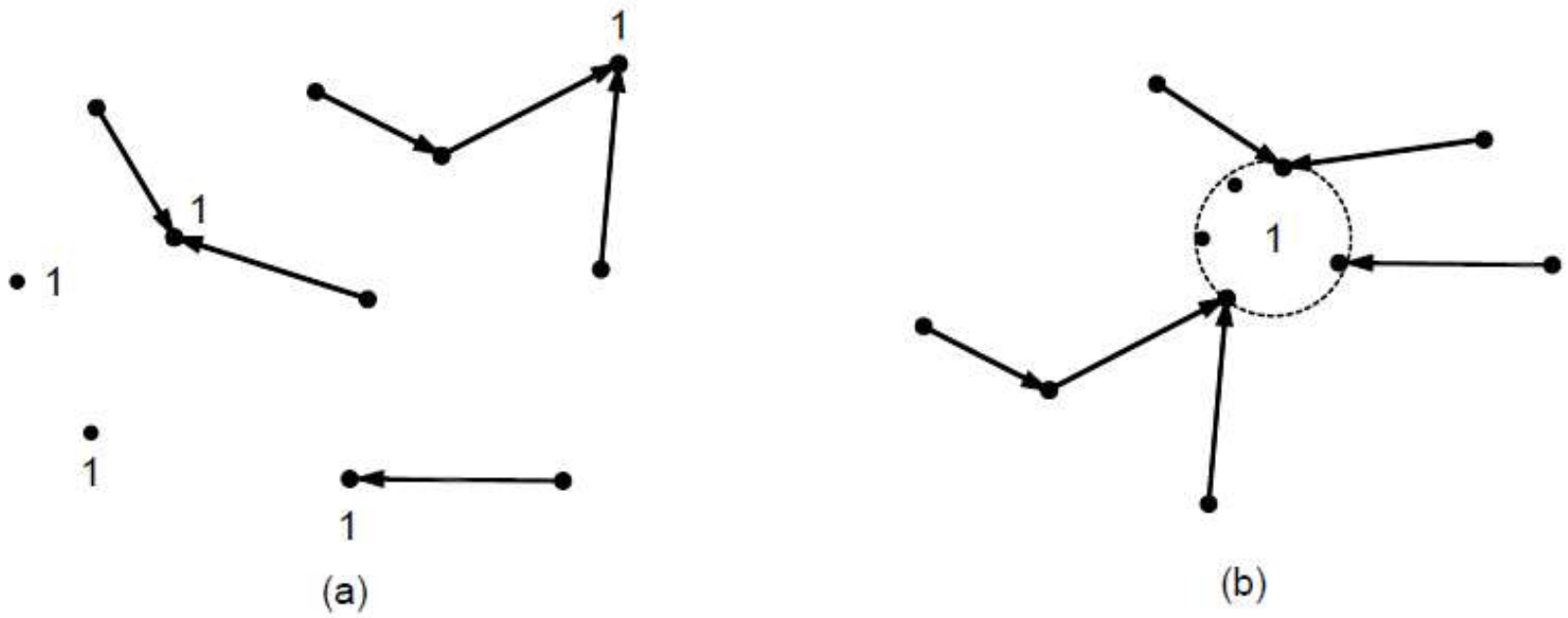


(a) Core-based tree for group 1.

(b) Sending to group 1.

Computer Networks, Fifth Edition by Andrew Tanenbaum and David Wetherall, © Pearson Education-Prentice Hall, 2011

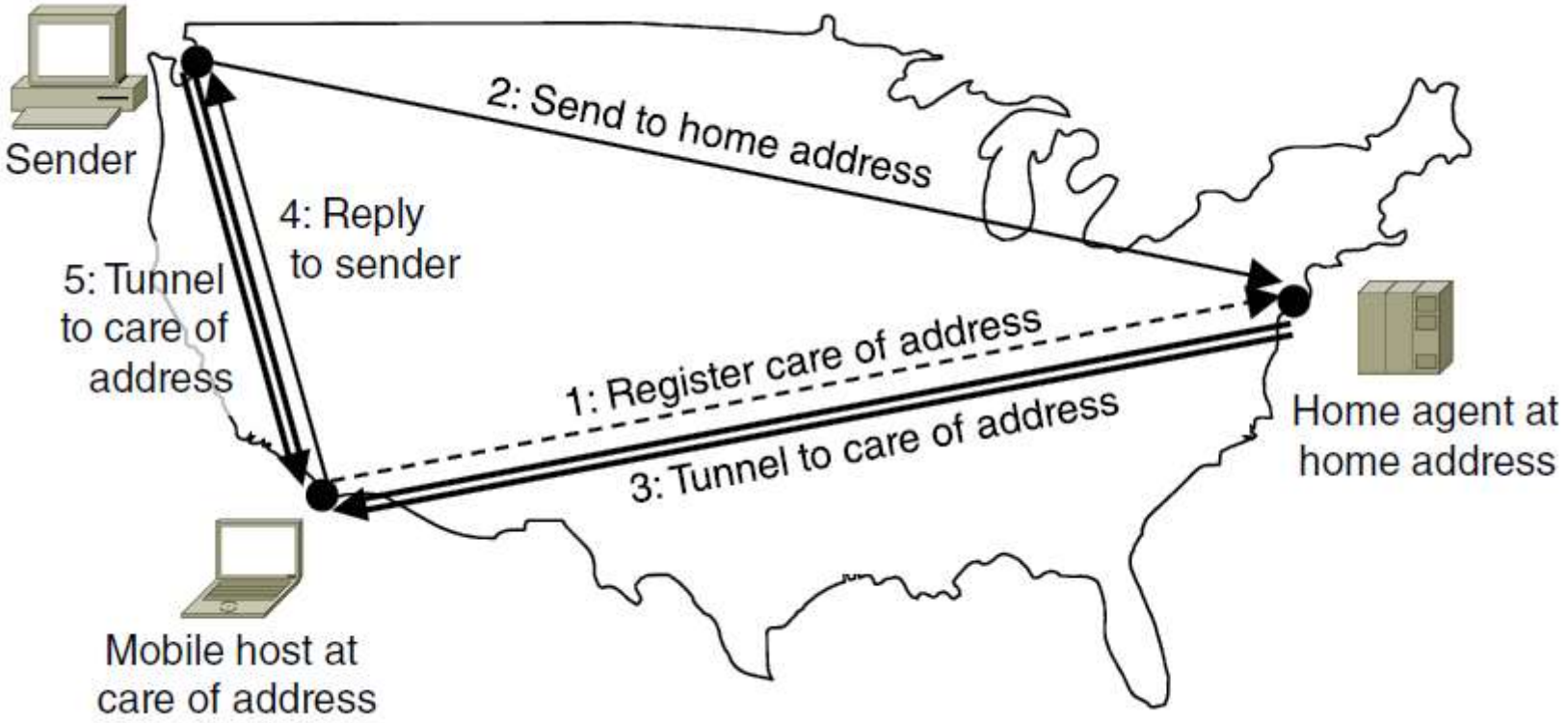
Anycast Routing



(a) Anycast routes to group 1.

(b) Topology seen by the routing protocol.

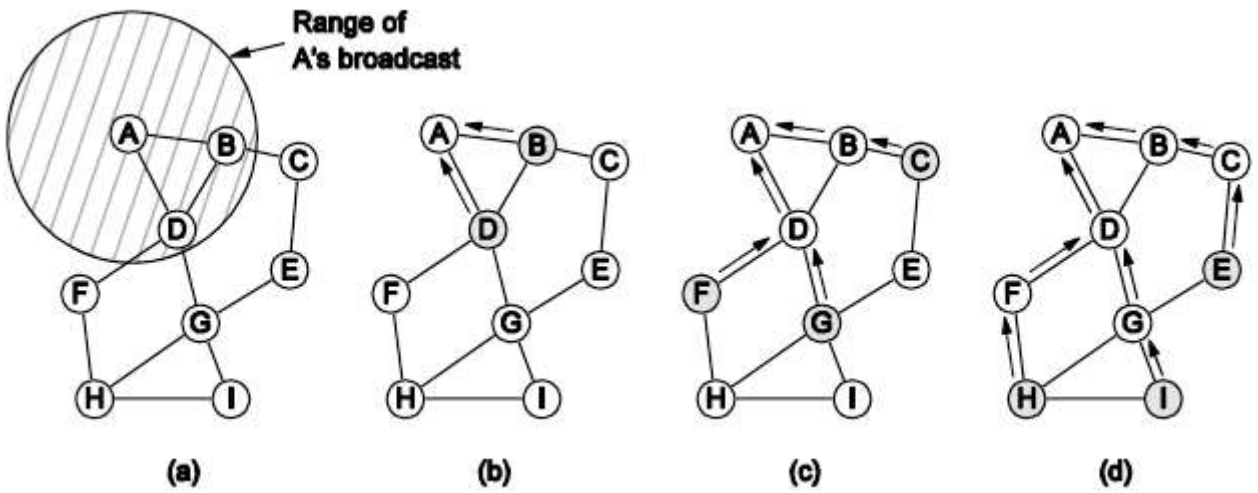
Routing for Mobile Hosts



Packet routing for mobile hosts

Computer Networks, Fifth Edition by Andrew Tanenbaum and David Wetherall, © Pearson Education-Prentice Hall, 2011

Routing in Ad Hoc Networks



(a) Range of A's broadcast.

(b) After B and D receive it.

(c) After C, F, and G receive it.

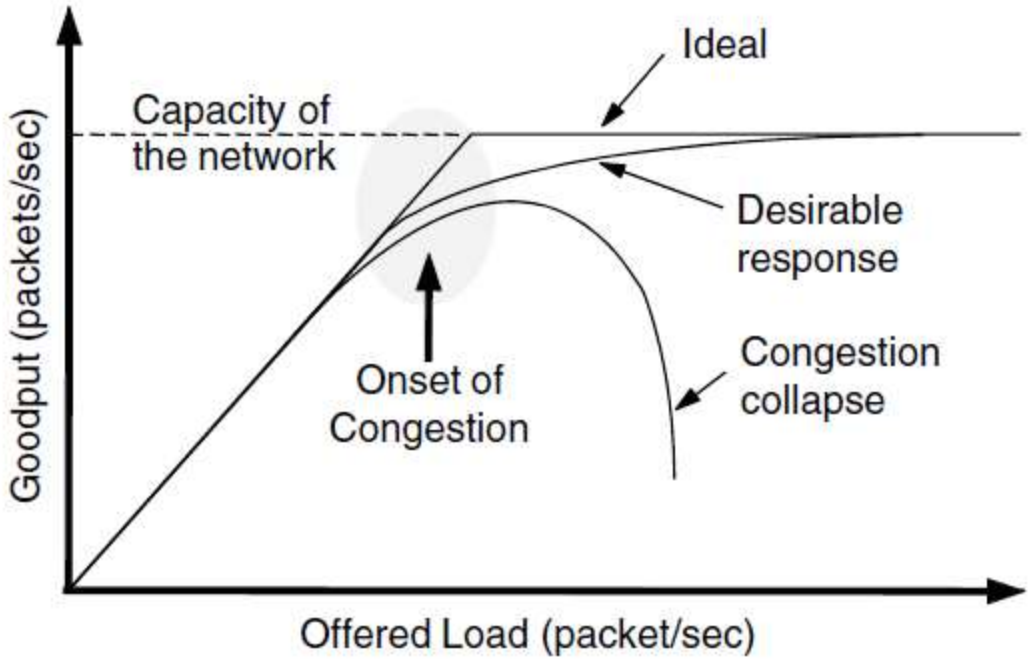
(d) After E, H, and I receive it.

The shaded nodes are new recipients. The dashed lines show possible reverse routes. The solid lines show the discovered route.

Congestion Control Algorithms (1)

- Approaches to congestion control
- Traffic-aware routing
- Admission control
- Traffic throttling
- Load shedding

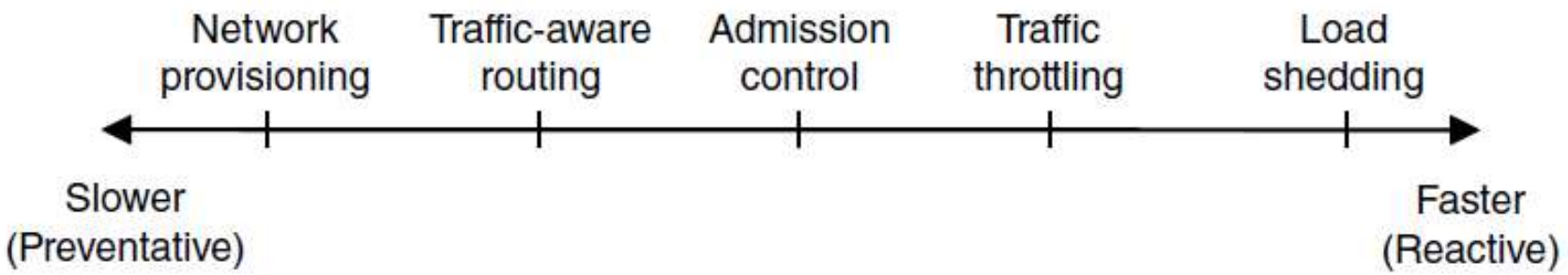
Congestion Control Algorithms (2)



When too much traffic is offered, congestion sets in and performance degrades sharply.

Computer Networks, Fifth Edition by Andrew Tanenbaum and David Wetherall, © Pearson Education-Prentice Hall, 2011

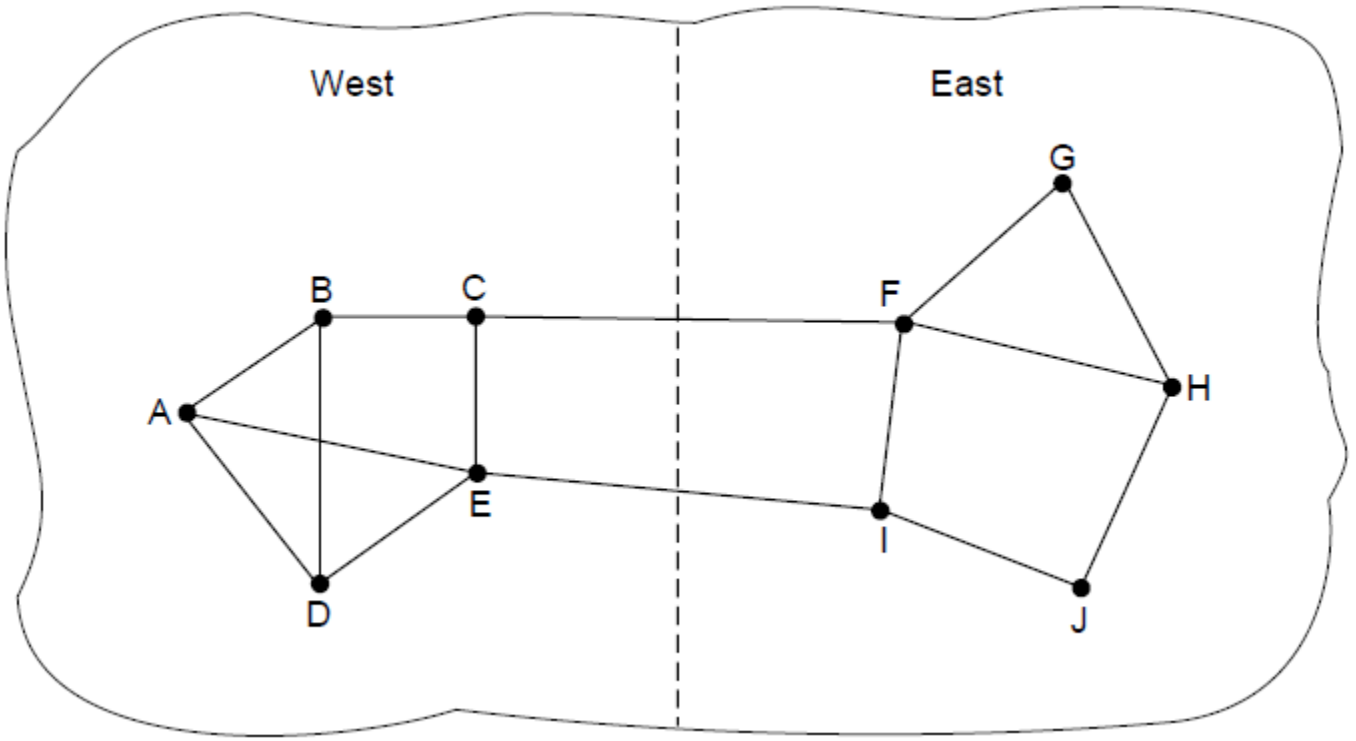
Approaches to Congestion Control



Timescales of approaches to congestion control

Computer Networks, Fifth Edition by Andrew Tanenbaum and David Wetherall, © Pearson Education-Prentice Hall, 2011

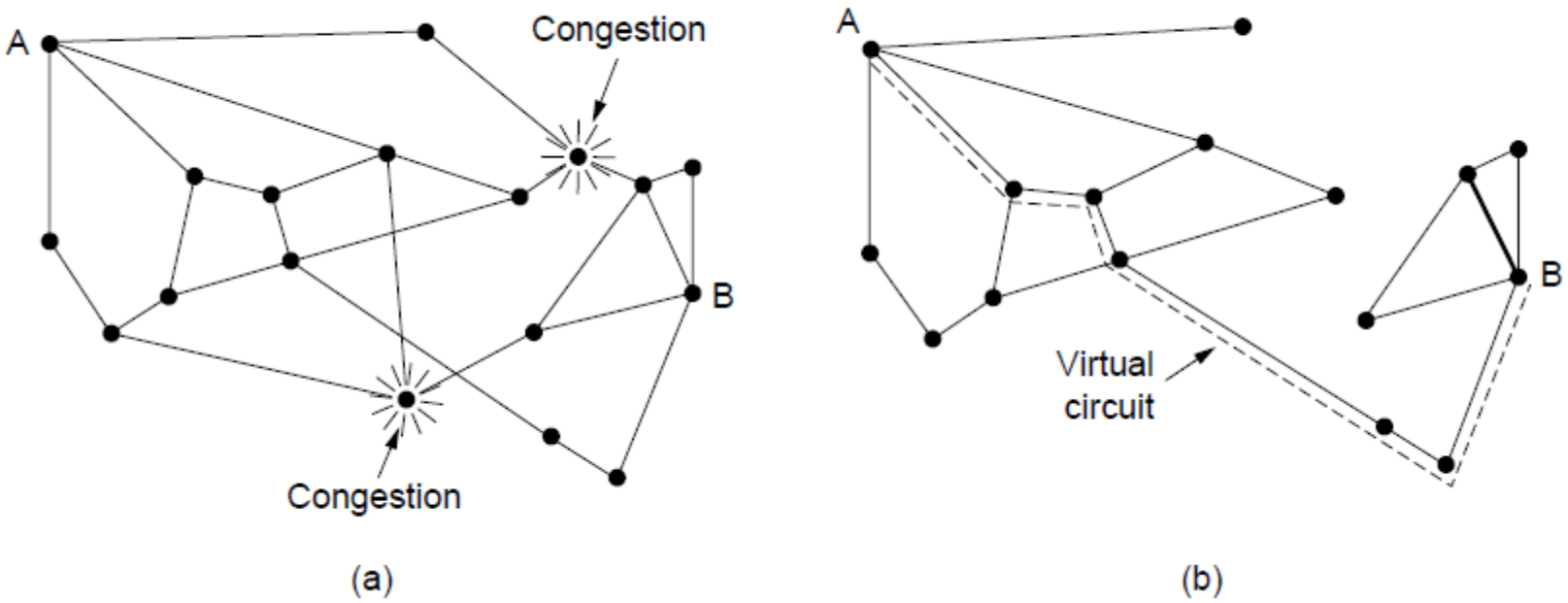
Traffic-Aware Routing



A network in which the East and West parts are connected by two links.

Computer Networks, Fifth Edition by Andrew Tanenbaum and David Wetherall, © Pearson Education-Prentice Hall, 2011

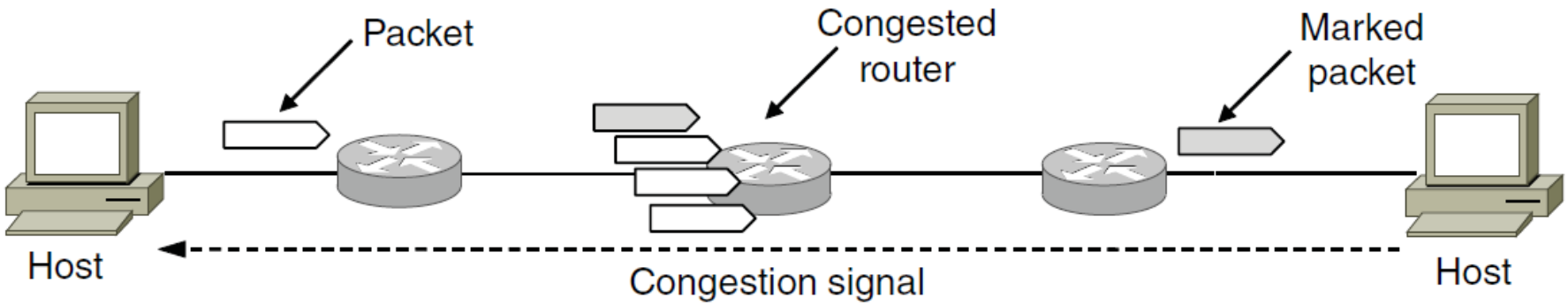
Traffic Throttling (1)



(a) A congested network. (b) The portion of the network that is not congested. A virtual circuit from A to B is also shown.

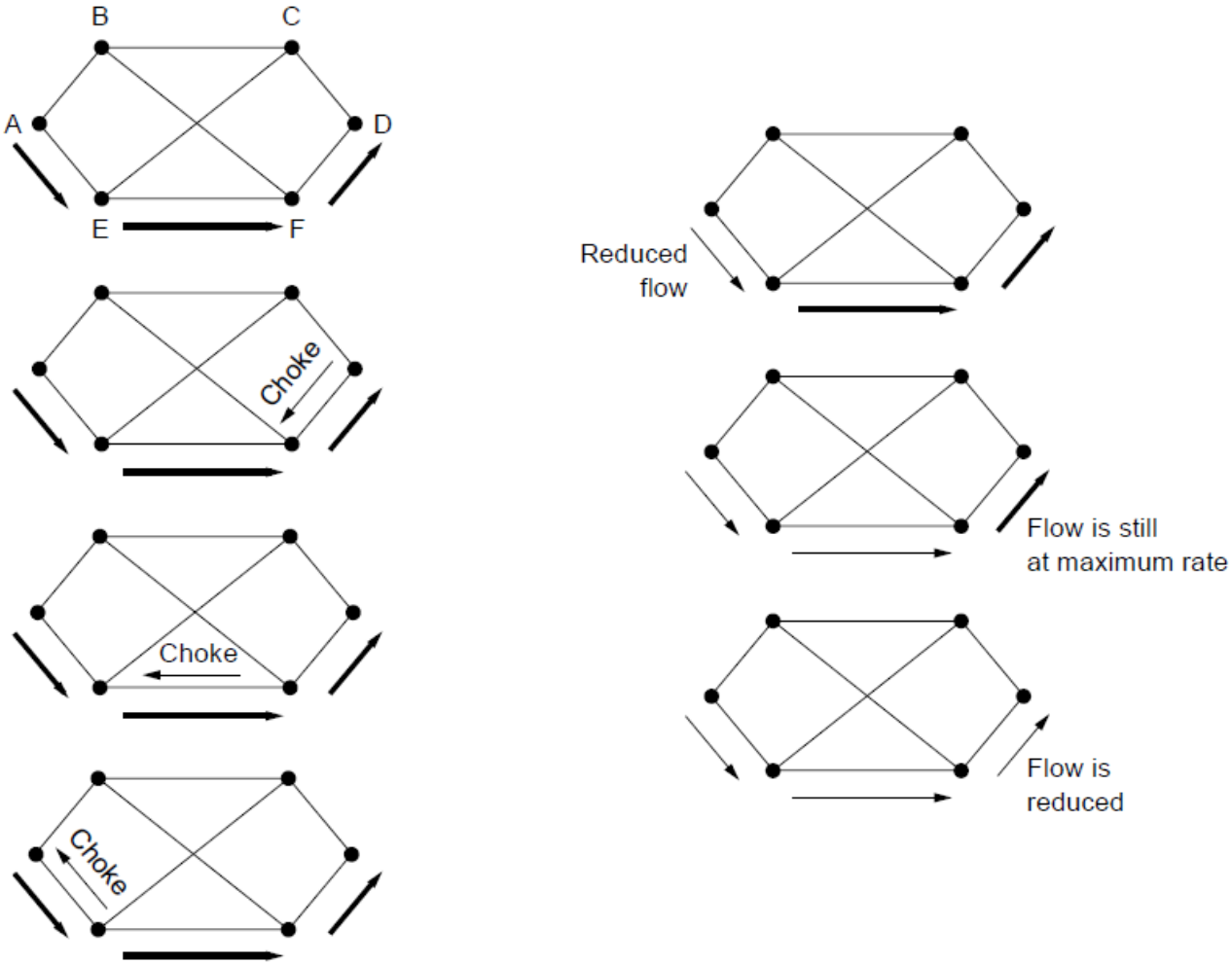
Computer Networks, Fifth Edition by Andrew Tanenbaum and David Wetherall, © Pearson Education-Prentice Hall, 2011

Traffic Throttling (2)



Explicit congestion notification

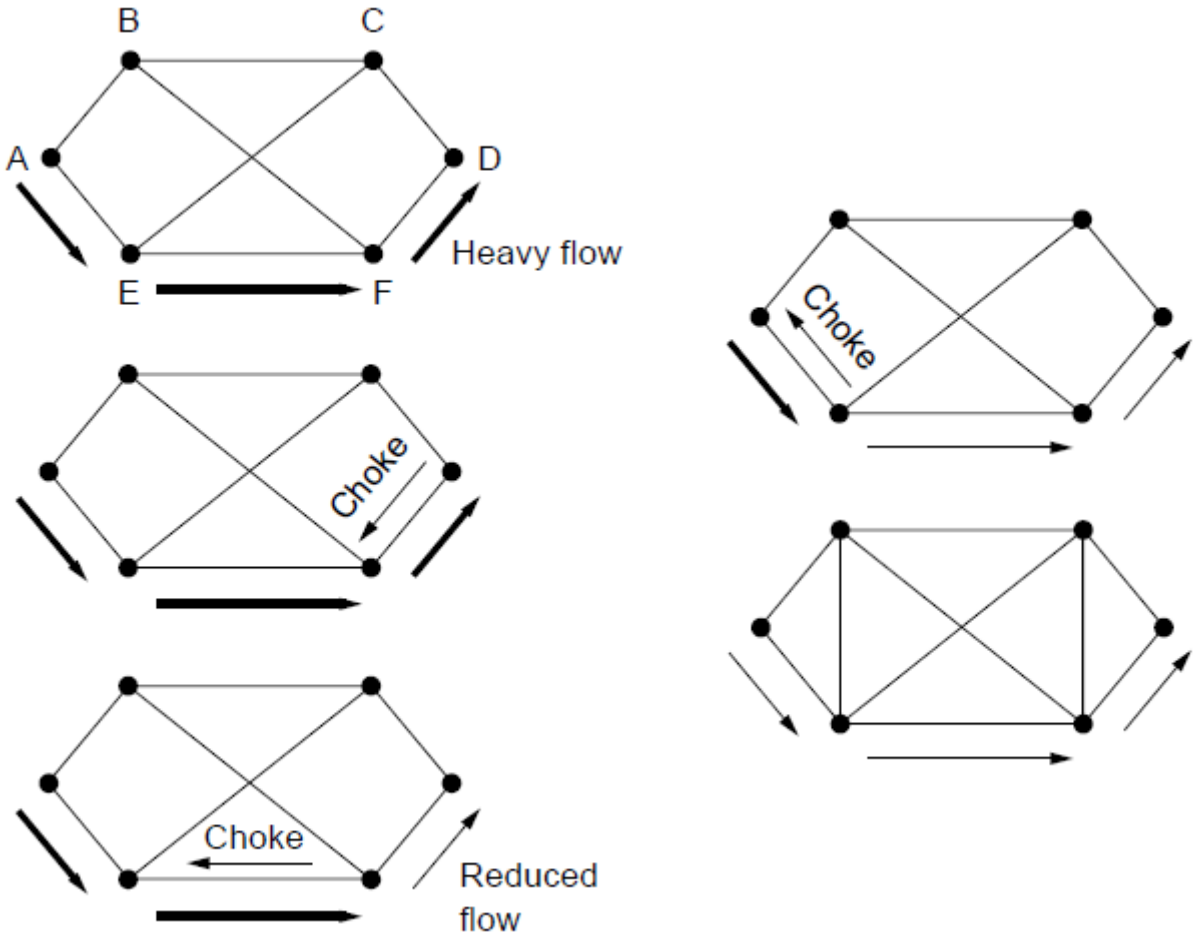
Load Shedding (1)



A choke packet that affects only the source..

Computer Networks, Fifth Edition by Andrew Tanenbaum and David Wetherall, © Pearson Education-Prentice Hall, 2011

Load Shedding (2)



A choke packet that affects each hop it passes through.

Computer Networks, Fifth Edition by Andrew Tanenbaum and David Wetherall, © Pearson Education-Prentice Hall, 2011