Network Layer: Delivery, Forwarding, and Routing

22-1　DELIVERY

The network layer supervises the handling of the packets by the underlying physical networks. We define this handling as the delivery of a packet.

Direct Versus Indirect Delivery

The delivery of a packet to its final destination is accomplished by using two different methods of delivery, **direct and indirect**.
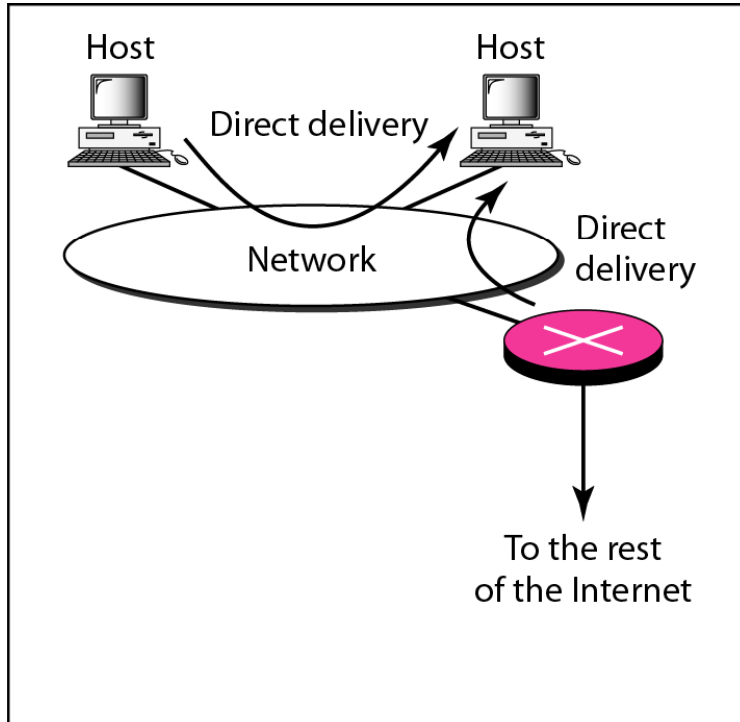
Direct Delivery

In a direct delivery, the final destination of the packet is a host connected to the same physical network as the deliverer.

Direct delivery occurs when the source and destination of the packet are located on the same physical network.
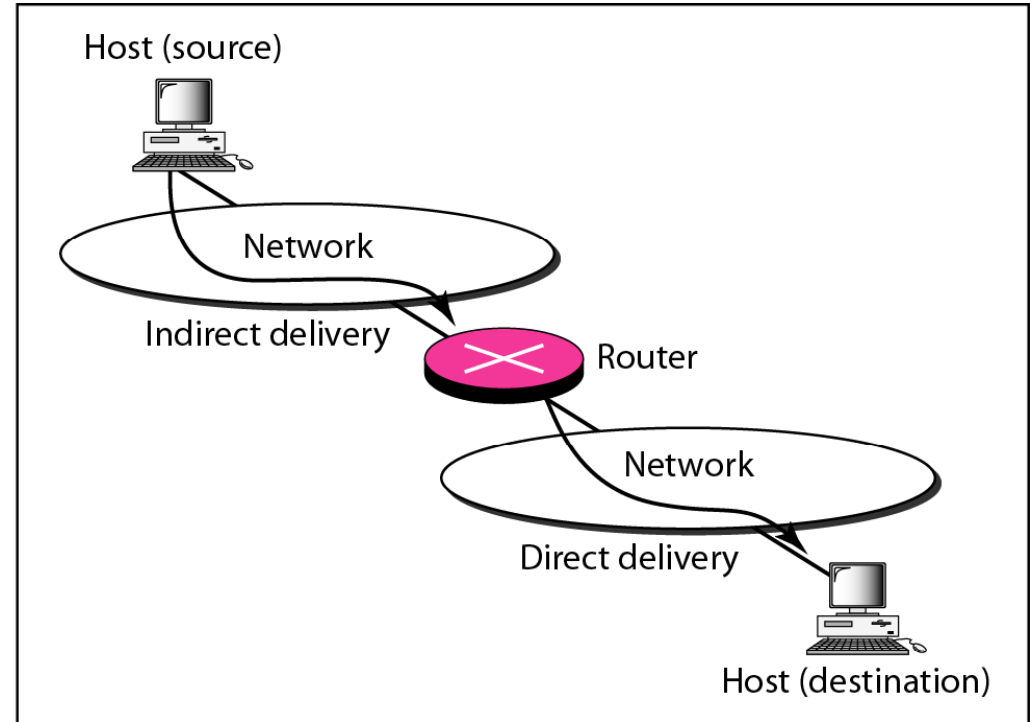
The sender can easily determine if the delivery is direct.

It can extract the network address of the destination and compare this address with the addresses of the networks to which it is connected. If a match is found, the delivery is direct.

Figure 22.1 *Direct and indirect delivery*

a. Direct delivery

b. Indirect and direct delivery

Indirect Delivery

If the destination host is not on the same network as the deliverer, the packet is delivered indirectly.

In an indirect delivery, the packet goes from router to router until it reaches the one connected to the same physical network as its final destination.

22-2　FORWARDING

Forwarding means to place the packet in its route to its destination. Forwarding requires a host or a router to have a *routing table**. When a host has a packet to send or when a router has received a packet to be forwarded, it looks at this table to find the route to the final destination.*

Topics discussed in this section:

Forwarding Techniques
Forwarding Process
Routing Table

Routing table

For a network router to know where to send packets of data it receives, it uses a **routing table**.

The routing table contains a list of specific routing destinations, and when the router receives a packet of data, it references the routing table to know where to send that data.

The routing table may also contain information on how far each destination is from the router. In essence, a routing table is a map for the router.

The table is usually stored in a router or the network computer as a database or file.

This information helps to **find the best possible path.**

The routing table has at least 3 fields: **the destination network id, cost of the path, next hop or address to send the packet**.

However, this simple solution is impossible today in an Internetwork such as the Internet because the number of entries needed in the routing table would make table lookups inefficient.

Forwarding Techniques

Several techniques can make the size of the routing table manageable and also handle issues such as security.

Next-Hop Method Versus Route Method

One technique to reduce the contents of a routing table is called the next-hop method.

In this technique, the routing table holds only the address of the next hop instead of information about the complete route (route method).

Figure 22.2 *Route method versus next-hop method*

a. Routing tables based on route

Destination	Route
Host B	R1, R2, host B

Routing table for host A

Destination	Route
Host B	R2, host B

Routing table for R1

Destination	Route
Host B	Host B

Routing table for R2

b. Routing tables based on next hop

Destination	Next hop
Host B	R1

Destination	Next hop
Host B	R2

Destination	Next hop
Host B	---

Host A

Host B

R1

R2

Network — R1 — Network — R2 — Network
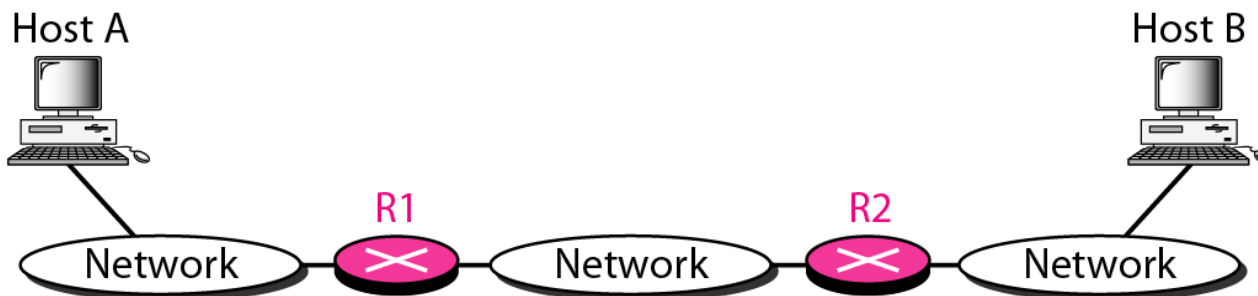
Network-Specific Method Versus Host-Specific Method

A second technique to reduce the routing table and simplify the searching process is called the network-specific method.

Here, instead of having an entry for every destination host connected to the same physical network (host-specific method), we have only one entry that defines the address of the destination network itself.

In other words, we treat all hosts connected to the same network as one single entity.

For example, if 1000 hosts are attached to the same network, only one entry exists in the routing table instead of 1000.
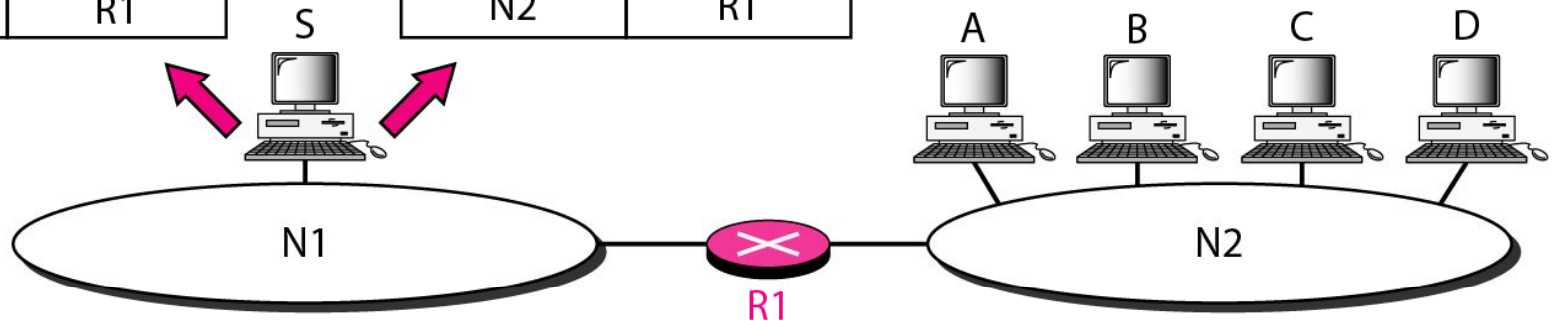
Figure 22.3 *Host-specific versus network-specific method*

Routing table for host S based on host-specific method

Destination	Next hop
A	R1
B	R1
C	R1
D	R1

Routing table for host S based on network-specific method

Destination	Next hop
N2	R1

Default Method

Another technique to simplify routing is called the default method.

In Figure 22.4 host A is connected to a network with two routers. Router Rl routes the packets to hosts connected to network N2. However, for the rest of the Internet, router R2 is used.

So instead of listing all networks in the entire Internet, host A can just have one entry called the *default*.

Figure 22.4 *Default method*

Destination	Next hop
N2	R1
Any other	R2

Routing table
for host A

Host A

N1

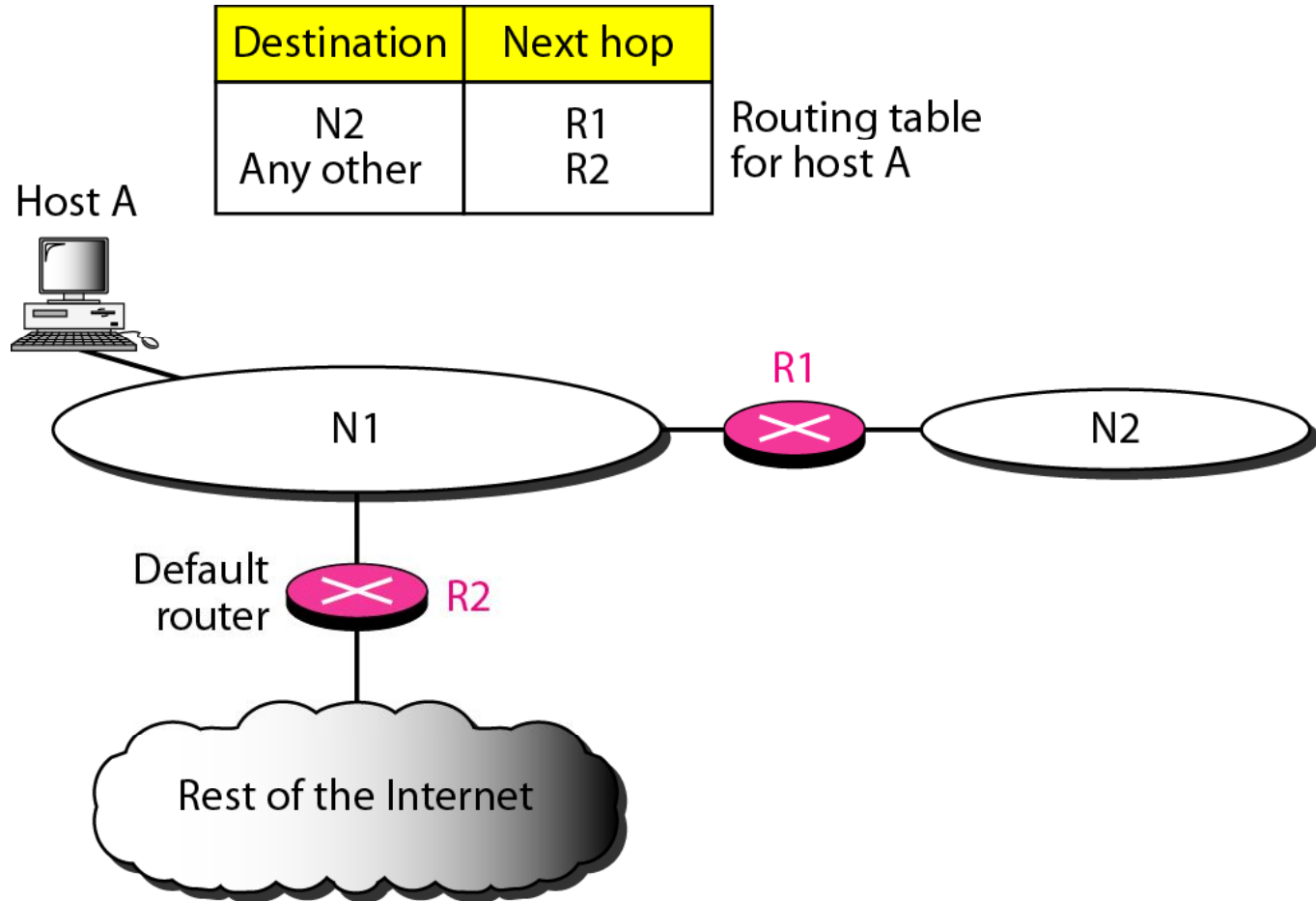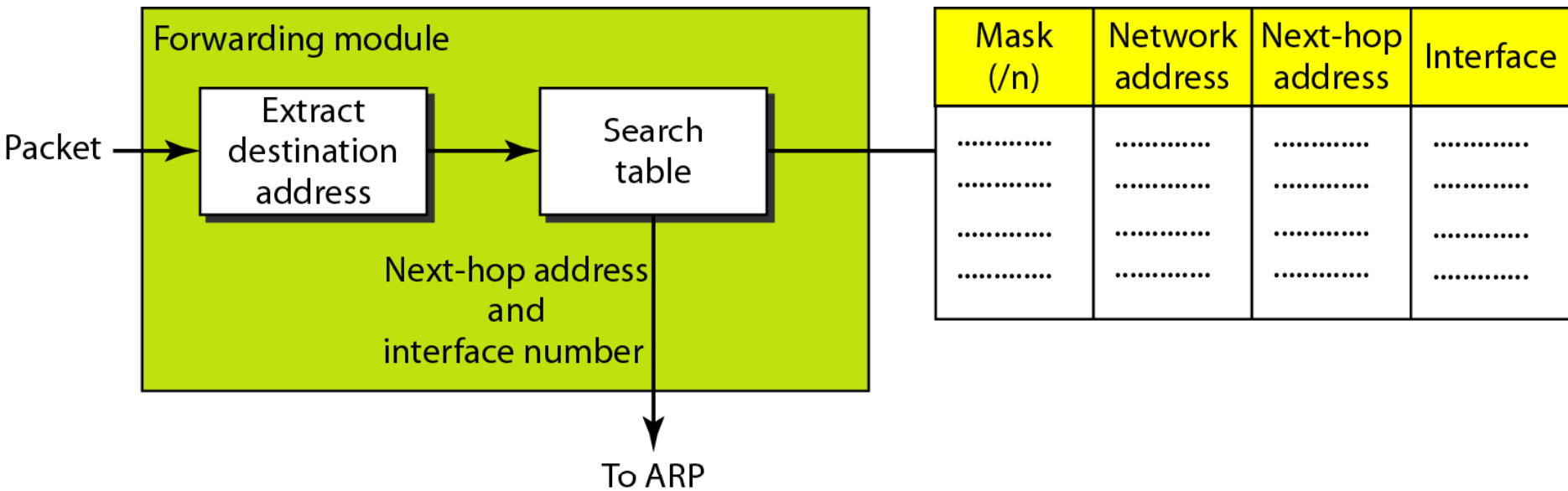R1

N2

Default
router

R2

Rest of the Internet

Figure 22.5 *Simplified forwarding module in classless address*

Note

In classless addressing, we need at least four columns in a routing table.

Example 22.1

Make a routing table for router R1, using the configuration in Figure 22.6.

Solution
Table 22.1 shows the corresponding table.
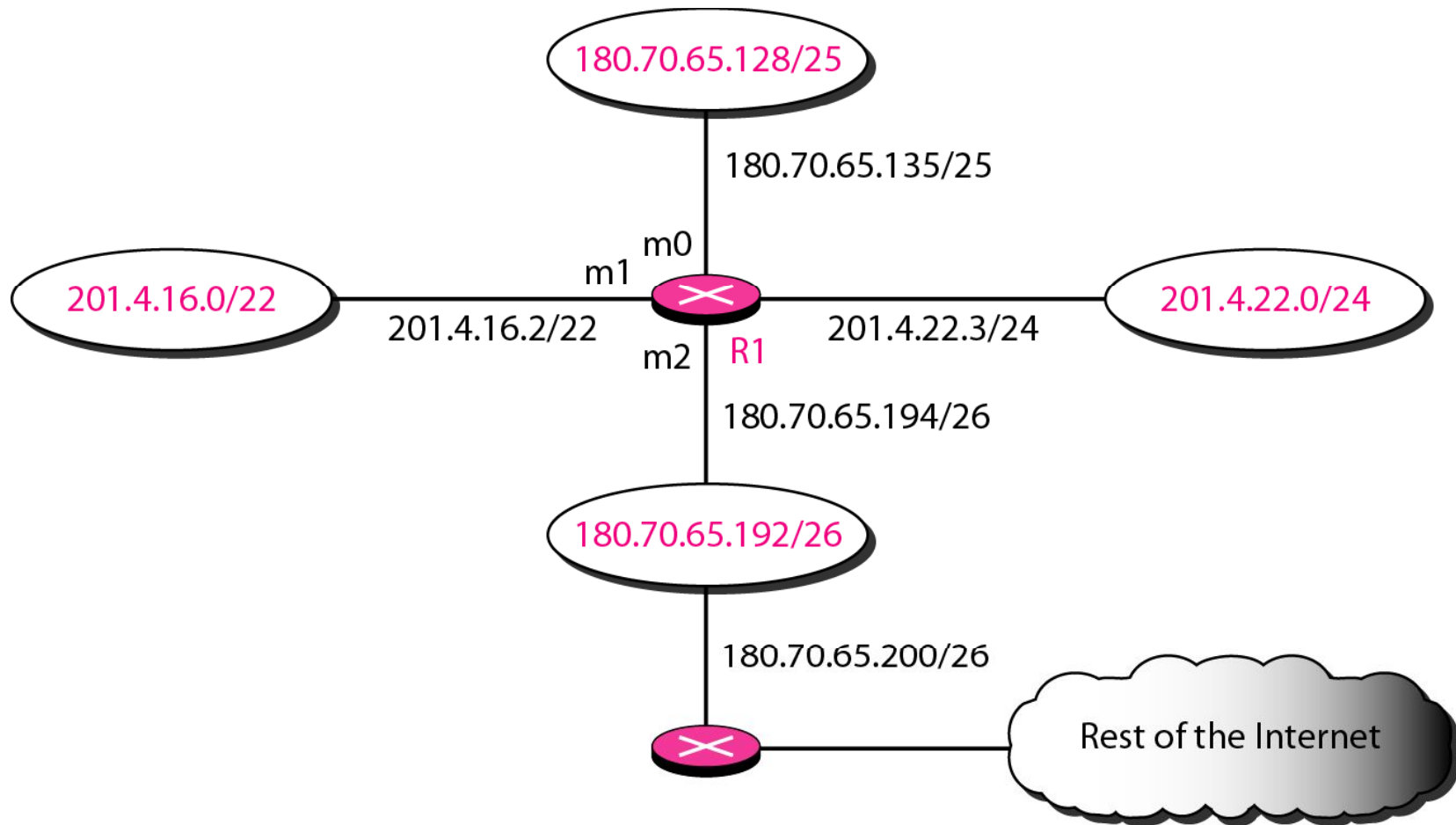
Figure 22.6 *Configuration for Example 22.1*

180.70.65.128/25

180.70.65.135/25

m0

m1

201.4.16.0/22

201.4.16.2/22

201.4.22.3/24

201.4.22.0/24

m2   R1

180.70.65.194/26

180.70.65.192/26

180.70.65.200/26

Rest of the Internet

Table 22.1 *Routing table for router R1 in Figure 22.6*

Mask	Network Address	Next Hop	Interface
/26	180.70.65.192	—	m2
/25	180.70.65.128	—	m0
/24	201.4.22.0	—	m3
/22	201.4.16.0	....	m1
Any	Any	180.70.65.200	m2

Example 22.2

Show the forwarding process if a packet arrives at R1 in Figure 22.6 with the destination address 180.70.65.140.

Solution

The router performs the following steps:

1. The first mask (/26) is applied to the destination address. The result is 180.70.65.128, which does not match the corresponding network address.

2. The second mask (/25) is applied to the destination address. The result is 180.70.65.128, which matches the corresponding network address. The next-hop address and the interface number m0 are passed to ARP for further processing.

Example 22.3

Show the forwarding process if a packet arrives at R1 in Figure 22.6 with the destination address 201.4.22.35.

Solution

The router performs the following steps:

1. The first mask (/26) is applied to the destination address. The result is 201.4.22.0, which does not match the corresponding network address.

2. The second mask (/25) is applied to the destination address. The result is 201.4.22.0, which does not match the corresponding network address (row 2).

Example 22.3 (continued)

3. ***The third mask (/24) is applied to the destination address. The result is 201.4.22.0, which matches the corresponding network address. The destination address of the packet and the interface number m3 are passed to ARP.***
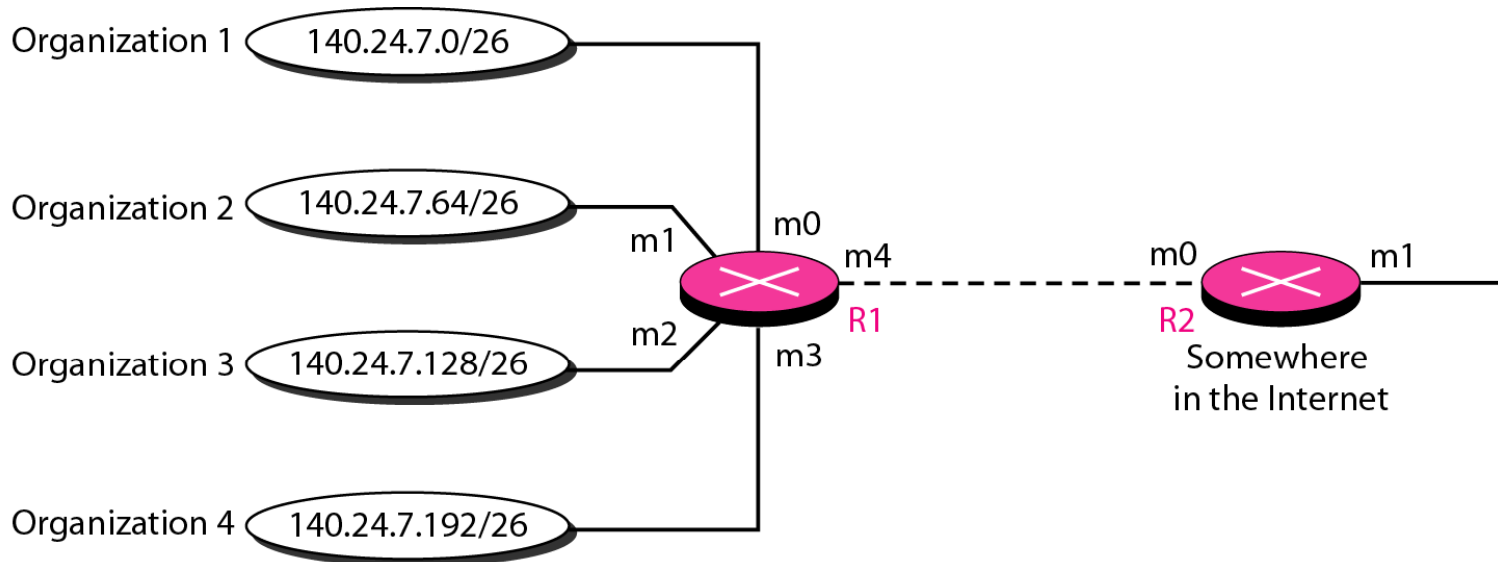
Example 22.4

Show the forwarding process if a packet arrives at R1 in Figure 22.6 with the destination address 18.24.32.78.

Solution

This time all masks are applied, one by one, to the destination address, but no matching network address is found. When it reaches the end of the table, the module gives the next-hop address 180.70.65.200 and interface number m2 to ARP. This is probably an outgoing package that needs to be sent, via the default router, to someplace else in the Internet.

Figure 22.7   *Address aggregation*

Organization 1   140.24.7.0/26
Organization 2   140.24.7.64/26
Organization 3   140.24.7.128/26
Organization 4   140.24.7.192/26

m1 m0 m4   R1   m0   R2   m1
m2
m3

Somewhere
in the Internet

Mask	Network address	Next-hop address	Interface
/26	140.24.7.0	----------	m0
/26	140.24.7.64	----------	m1
/26	140.24.7.128	----------	m2
/26	140.24.7.192	----------	m3
/0	0.0.0.0	Default	m4

Routing table for R1

Mask	Network address	Next-hop address	Interface
/24	140.24.7.0	----------	m0
/0	0.0.0.0	Default	m1

Routing table for R2

Figure 22.8 Longest mask matching
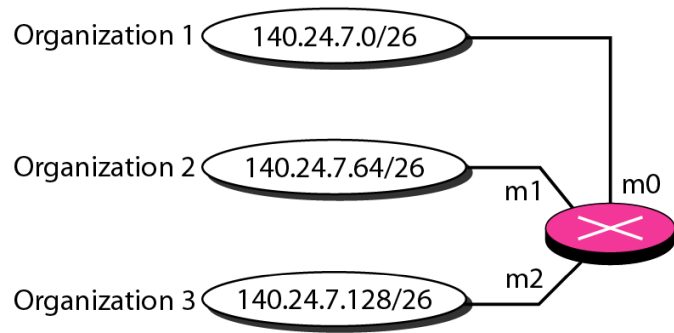
Routing table for R2

Mask	Network address	Next-hop address	Interface
/26	140.24.7.192	---------	m1
/24	140.24.7.0	---------	m0
/??	???????	?????????	m1
/0	0.0.0.0	Default	m2

Organization 1   140.24.7.0/26

Organization 2   140.24.7.64/26

Organization 3   140.24.7.128/26

m1   m0

m3   m0   m2

m2

R1   R2

m1

m2

To other networks

m1

R3

m0

140.24.7.192/26

Organization 4

Mask	Network address	Next-hop address	Interface
/26	140.24.7.0	---------	m0
/26	140.24.7.64	---------	m1
/26	140.24.7.128	---------	m2
/0	0.0.0.0	Default	m3

Routing table for R1
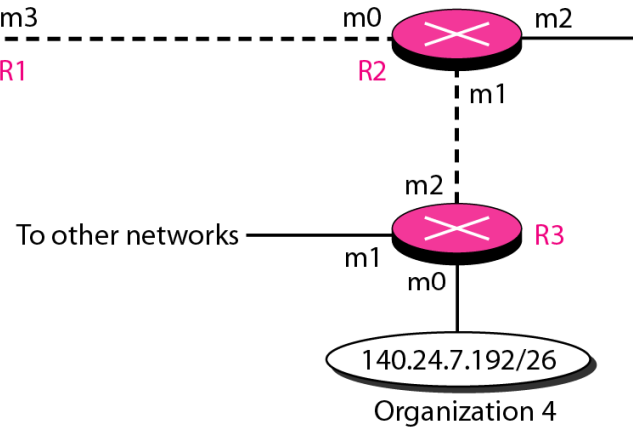
Mask	Network address	Next-hop address	Interface
/26	140.24.7.192	---------	m0
/??	???????	?????????	m1
/0	0.0.0.0	Default	m2

Routing table for R3

Example 22.5

As an example of hierarchical routing, let us consider Figure 22.9. A regional ISP is granted 16,384 addresses starting from 120.14.64.0. The regional ISP has decided to divide this block into four subblocks, each with 4096 addresses. Three of these subblocks are assigned to three local ISPs; the second subblock is reserved for future use. Note that the mask for each block is /20 because the original block with mask /18 is divided into 4 blocks.

The first local ISP has divided its assigned subblock into 8 smaller blocks and assigned each to a small ISP. Each small ISP provides services to 128 households, each using four addresses.

Example 22.5 (continued)

The second local ISP has divided its block into 4 blocks and has assigned the addresses to four large organizations.

The third local ISP has divided its block into 16 blocks and assigned each block to a small organization. Each small organization has 256 addresses, and the mask is /24.

There is a sense of hierarchy in this configuration. All routers in the Internet send a packet with destination address 120.14.64.0 to 120.14.127.255 to the regional ISP.

Figure 22.9   *Hierarchical routing with ISPs*
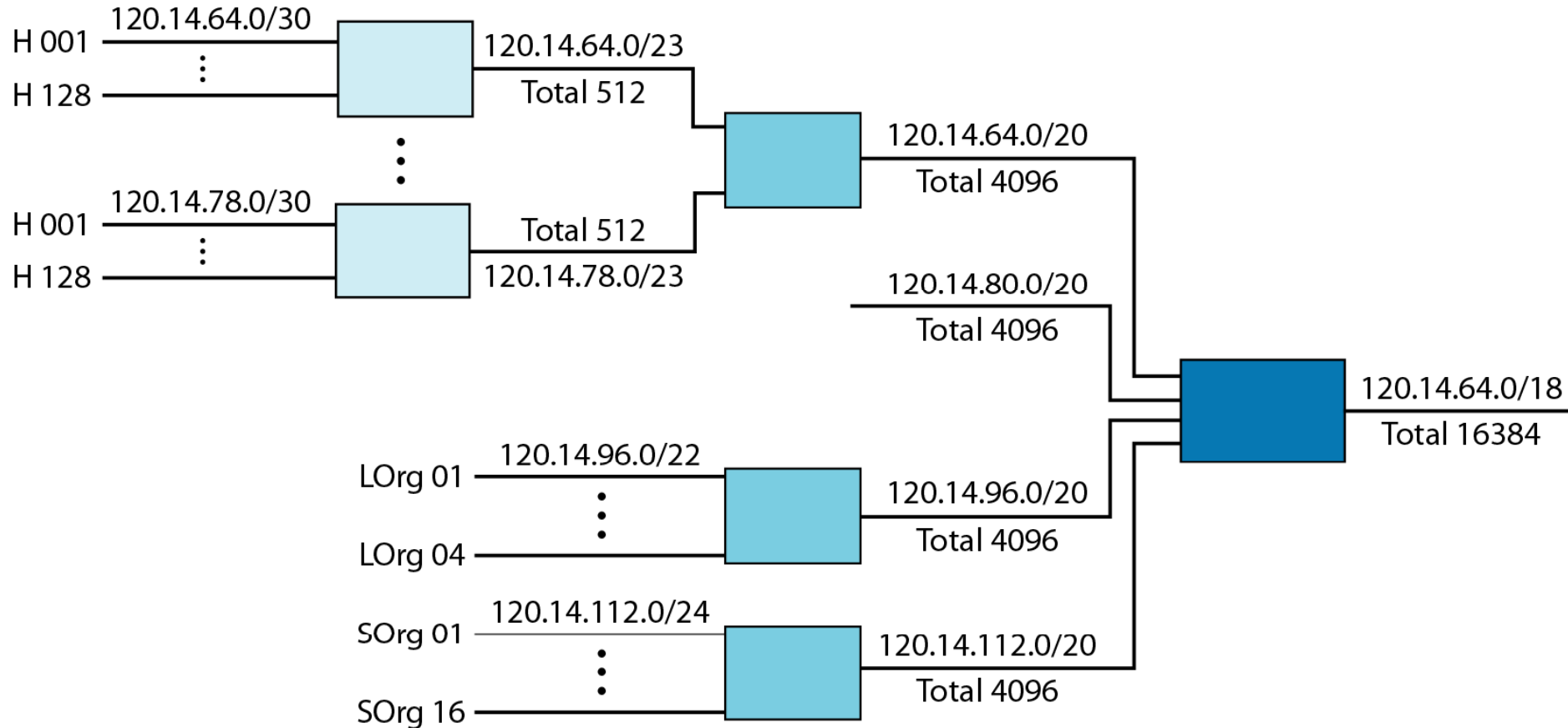
Figure 22.10 *Common fields in a routing table*

Mask	Network address	Next-hop address	Interface	Flags	Reference count	Use
....	....	....	....	....	....	....

Example 22.6

One utility that can be used to find the contents of a routing table for a host or router is netstat in UNIX or LINUX. The next slide shows the list of the contents of a default server. We have used two options, r and n. The option r indicates that we are interested in the routing table, and the option n indicates that we are looking for numeric addresses. Note that this is a routing table for a host, not a router. Although we discussed the routing table for a router throughout the chapter, a host also needs a routing table.

Example 22.6 (continued)

```
$ netstat -rn
Kernel IP routing table
Destination          Gateway              Mask                 Flags          Iface
153.18.16.0          0.0.0.0              255.255.240.0        U              eth0
127.0.0.0            0.0.0.0              255.0.0.0            U              lo
0.0.0.0              153.18.31.254        0.0.0.0              UG             eth0
```

The destination column here defines the network address. The term gateway used by UNIX is synonymous with router. This column actually defines the address of the next hop. The value 0.0.0.0 shows that the delivery is direct. The last entry has a flag of G, which means that the destination can be reached through a router (default router). The Iface defines the interface.

Example 22.6 (continued)

*More information about the IP address and physical address of the server can be found by using the **ifconfig** command on the given interface (eth0).*

```
$ ifconfig eth0
eth0   Link encap:Ethernet  HWaddr 00:B0:D0:DF:09:5D
inet addr:153.18.17.11  Bcast:153.18.31.255  Mask:255.255.240.0
. . .
```

Figure 22.11 *Configuration of the server for Example 22.6*

eth0

00:B0:D0:DF:09:5D

153.18.17.11/20

153.18.16.0/20

153.18.31.254/20

Default router

Rest of the Internet
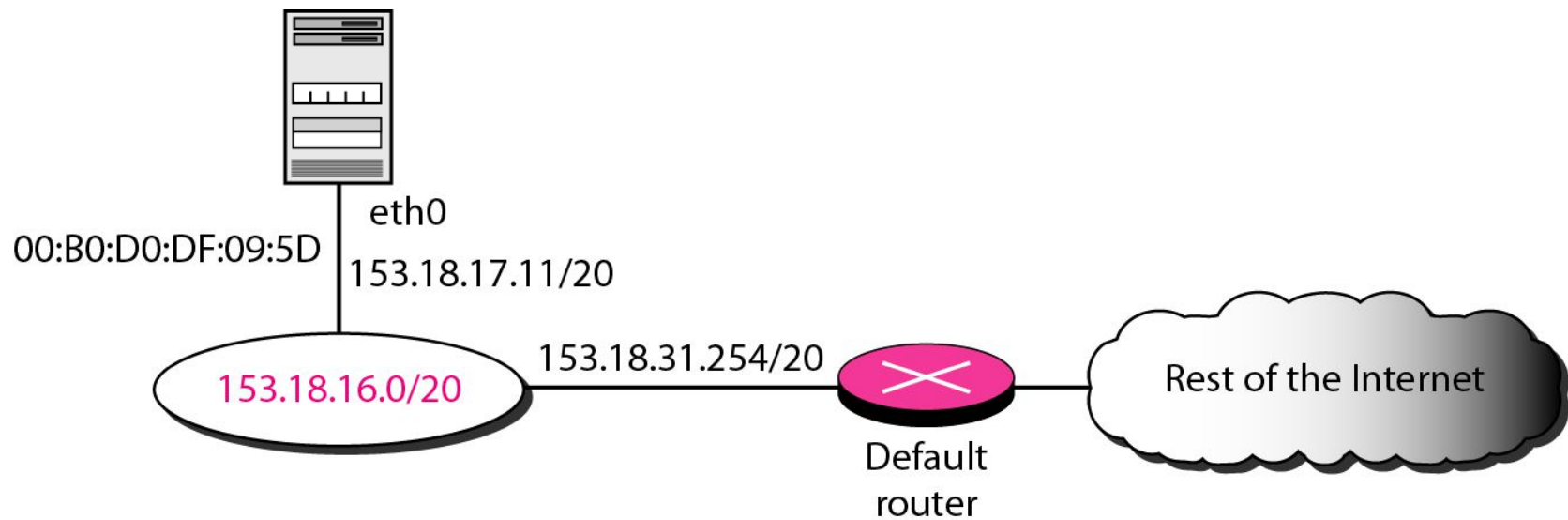
22-3   UNICAST ROUTING PROTOCOLS

A routing table can be either static or dynamic. A static table is one with manual entries. A dynamic table is one that is updated automatically when there is a change somewhere in the Internet. A routing protocol is a combination of rules and procedures that lets routers in the Internet inform each other of changes.

Topics discussed in this section:

Optimization
Intra- and Interdomain Routing
Distance Vector Routing and RIP
Link State Routing and OSPF
Path Vector Routing and BGP

Figure 22.12   *Autonomous systems*
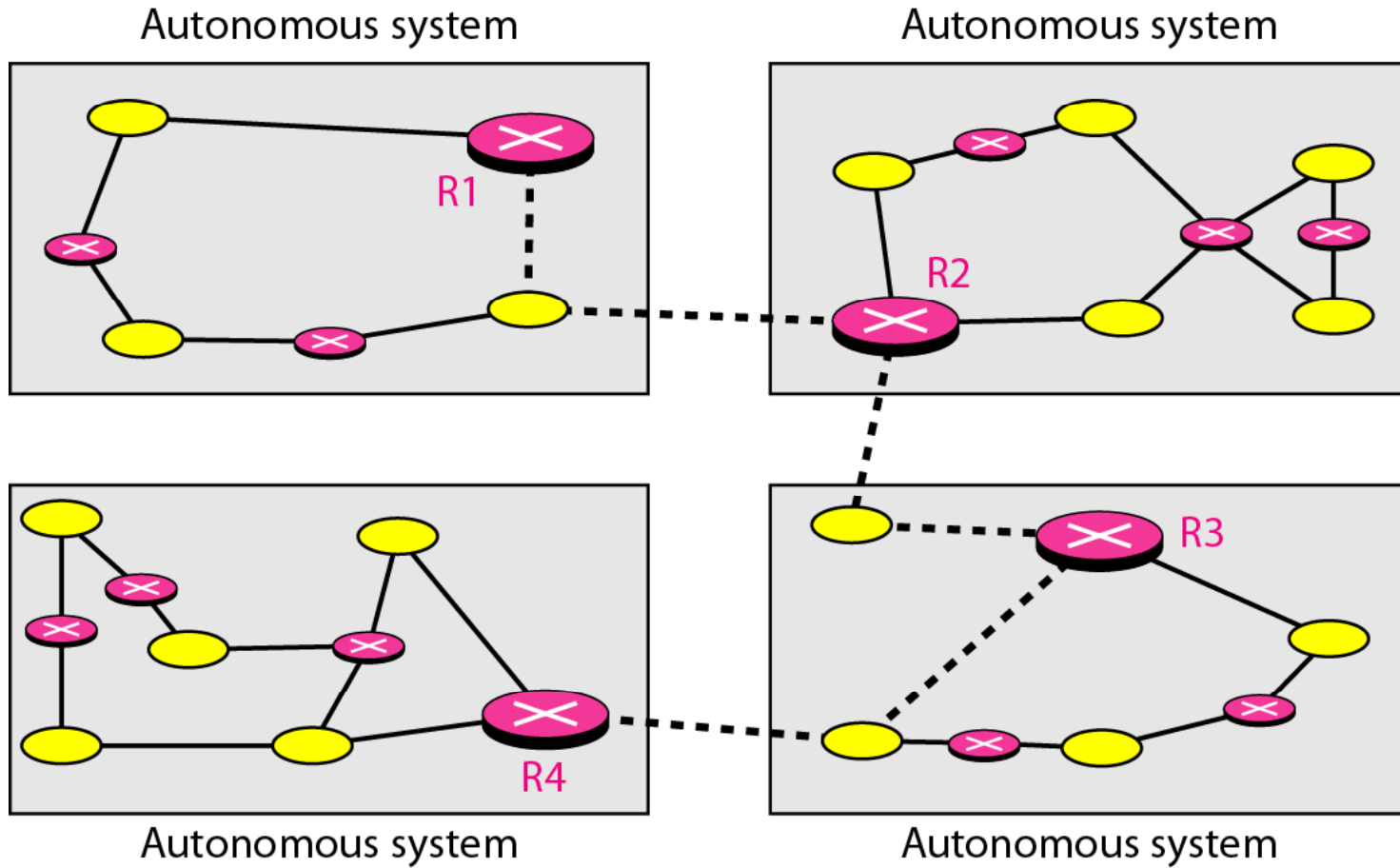
Figure 22.13 *Popular routing protocols*
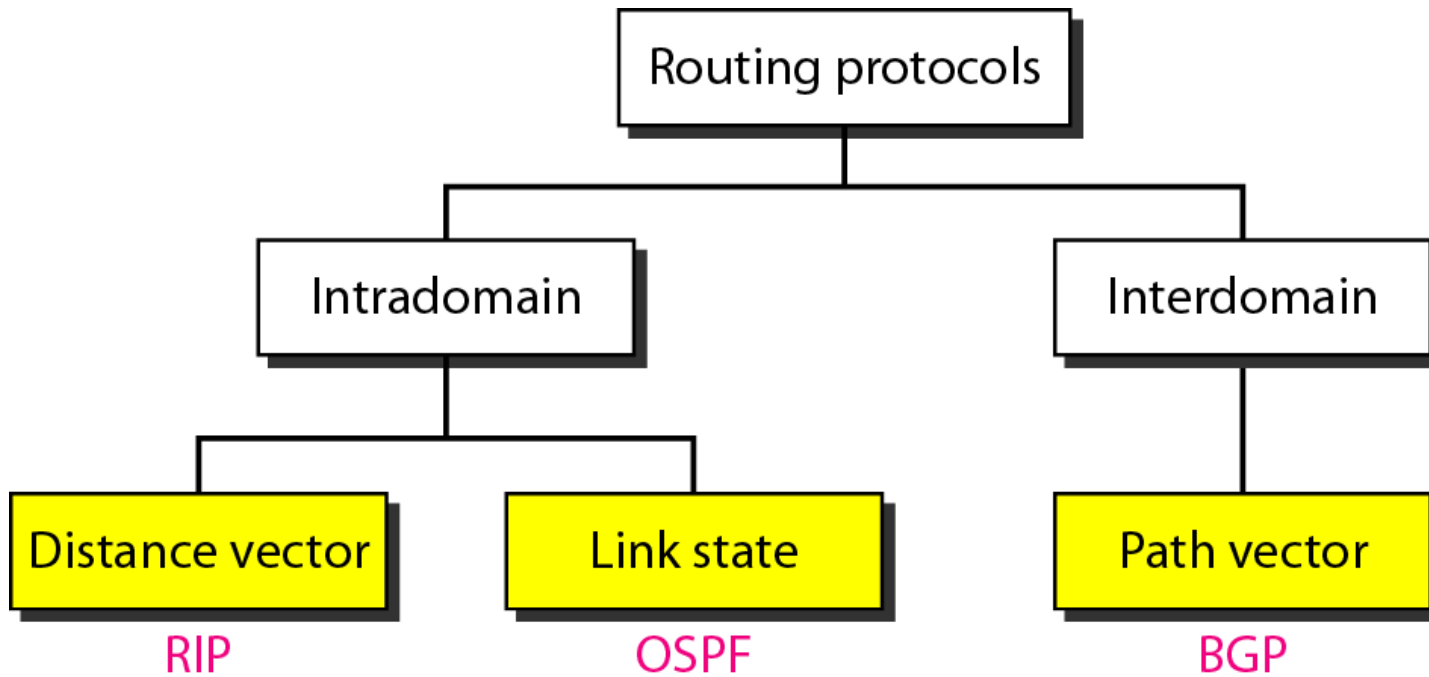
Figure 22.14 *Distance vector routing tables*

A's table

To	Cost	Next
A	0	—
B	5	—
C	2	—
D	3	—
E	6	C

B's table

To	Cost	Next
A	5	—
B	0	—
C	4	—
D	8	A
E	3	—

D's table

To	Cost	Next
A	3	—
B	8	A
C	5	A
D	0	—
E	9	A

C's table

To	Cost	Next
A	2	—
B	4	—
C	0	—
D	5	A
E	4	—

E's table

To	Cost	Next
A	6	C
B	3	—
C	4	—
D	9	C
E	0	—
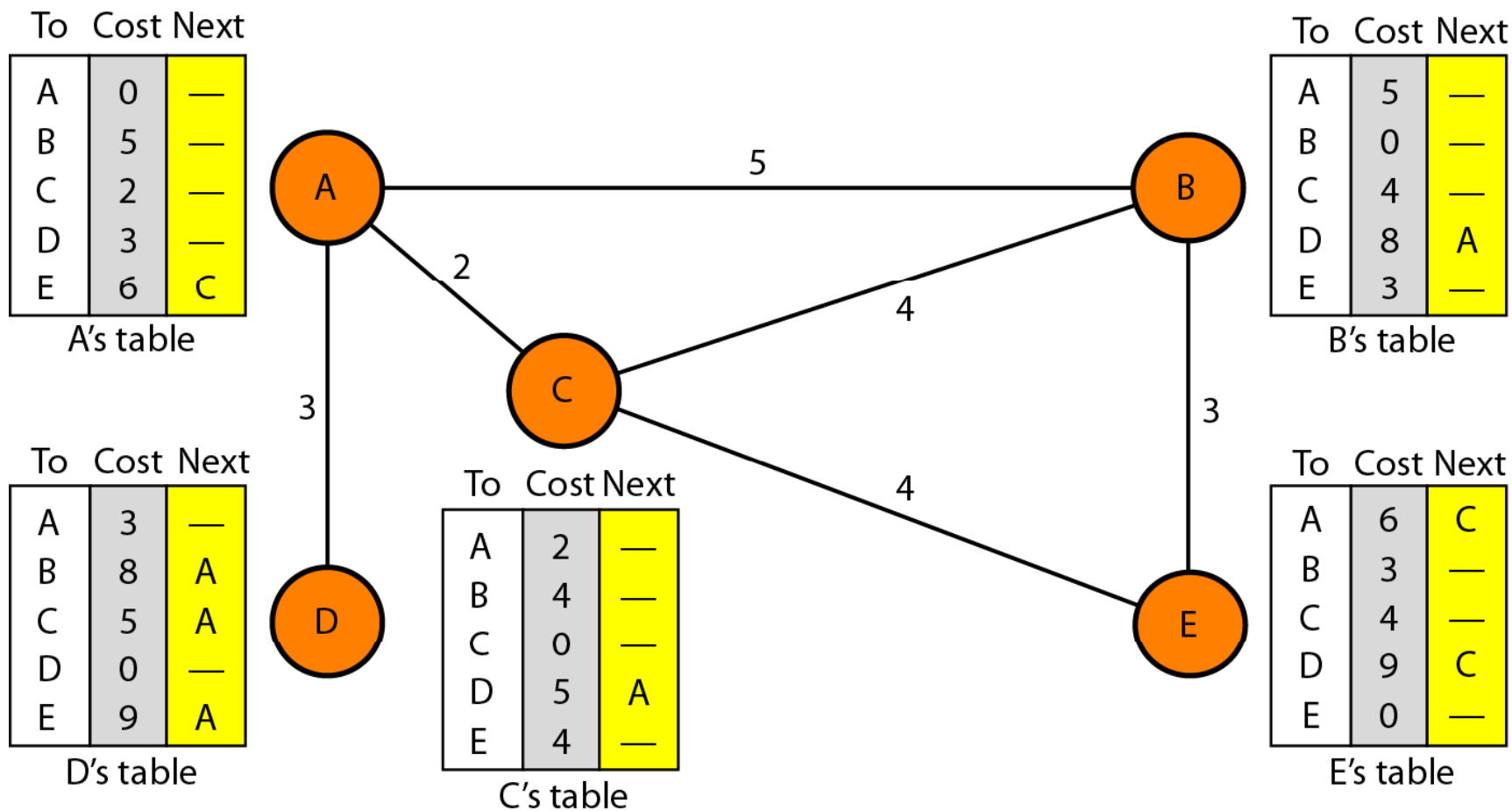
Figure 22.15 *Initialization of tables in distance vector routing*

A's table

To	Cost	Next
A	0	—
B	5	—
C	2	—
D	3	—
E	∞	

B's table

To	Cost	Next
A	5	—
B	0	—
C	4	—
D	∞	
E	3	—

D's table

To	Cost	Next
A	3	—
B	∞	
C	∞	
D	0	—
E	∞	

C's table

To	Cost	Next
A	2	—
B	4	—
C	0	—
D	∞	
E	4	—

E's table

To	Cost	Next
A	∞	
B	3	B
C	4	C
D	∞	
E	0	D
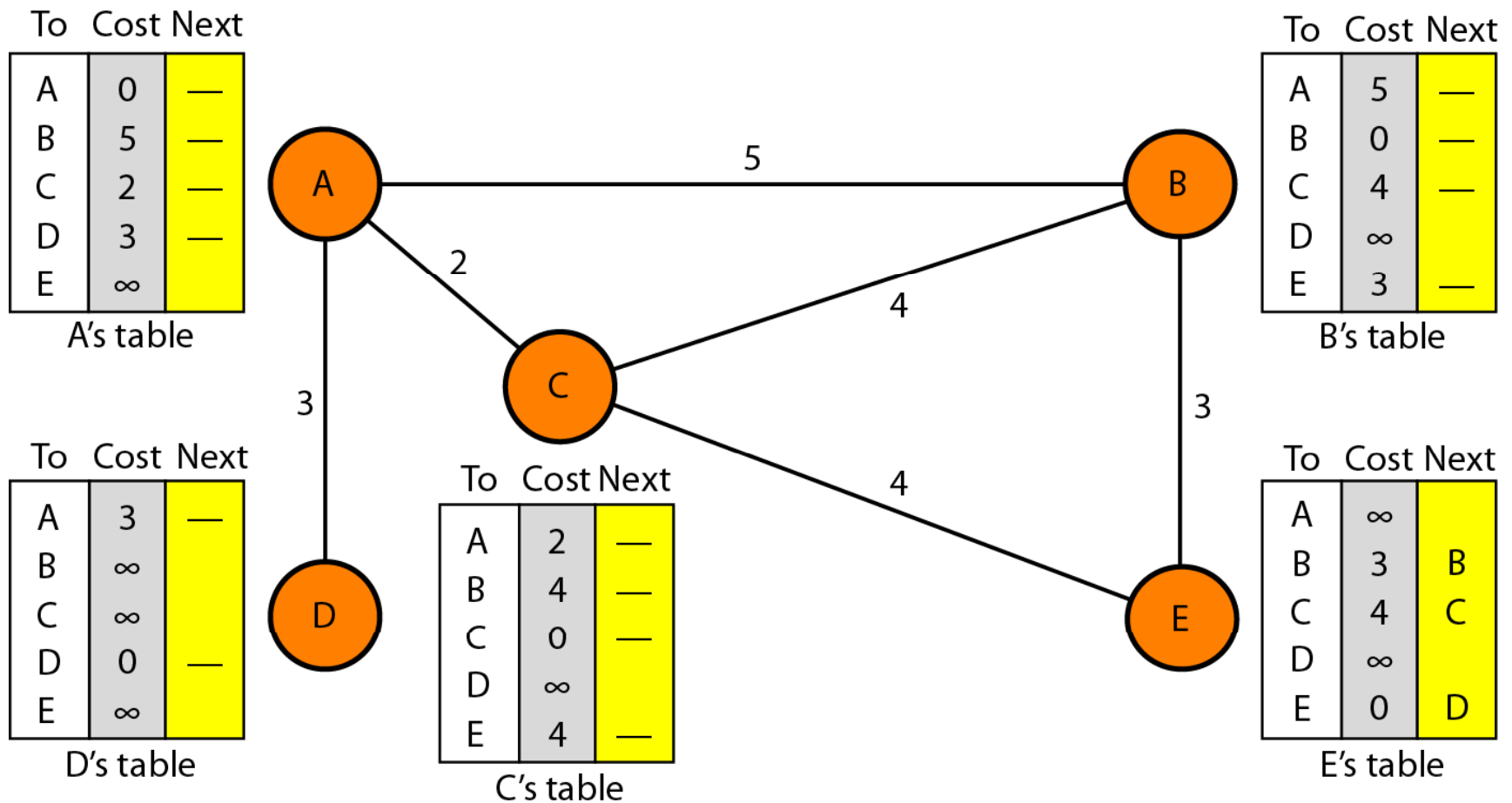
Note

In distance vector routing, each node shares its routing table with its immediate neighbors periodically and when there is a change.

Figure 22.16 *Updating in distance vector routing*

Received from C

To	Cost
A	2
B	4
C	0
D	∞
E	4

A's modified table

To	Cost	Next
A	4	C
B	6	C
C	2	C
D	∞	C
E	6	C

Compare

A's old table

To	Cost	Next
A	0	—
B	5	—
C	2	—
D	3	—
E	∞	

A's new table

To	Cost	Next
A	0	—
B	5	—
C	2	—
D	3	—
E	6	C
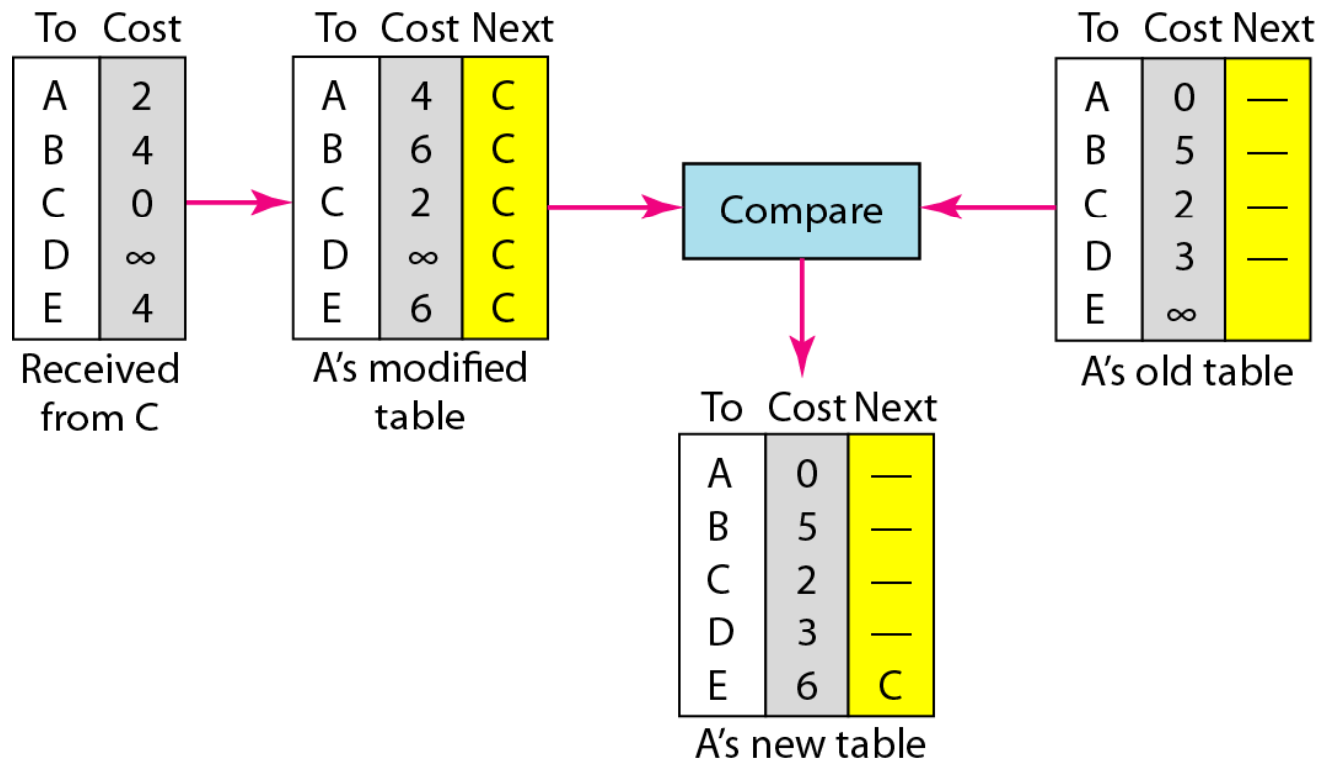
Figure 22.17 *Two-node instability*
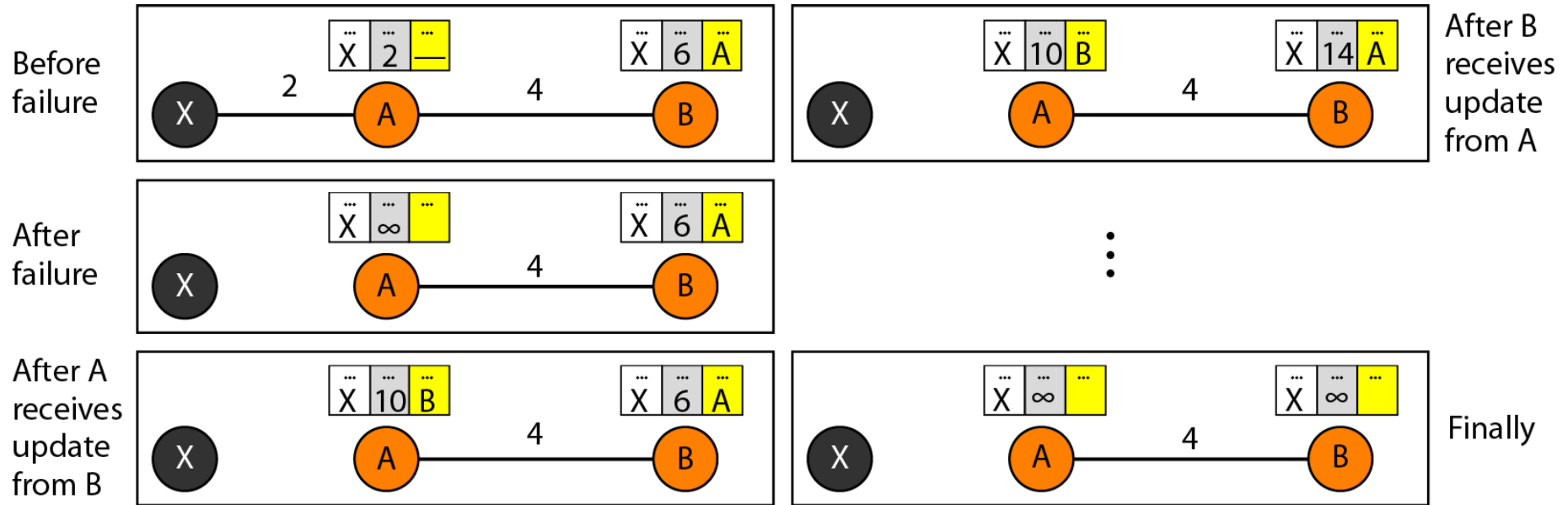
Figure 22.18 *Three-node instability*

Before failure

After A sends
the route to B
and C, but the
packet to C
is lost

After B sends
the route to A

After C sends
the route to B

22.41
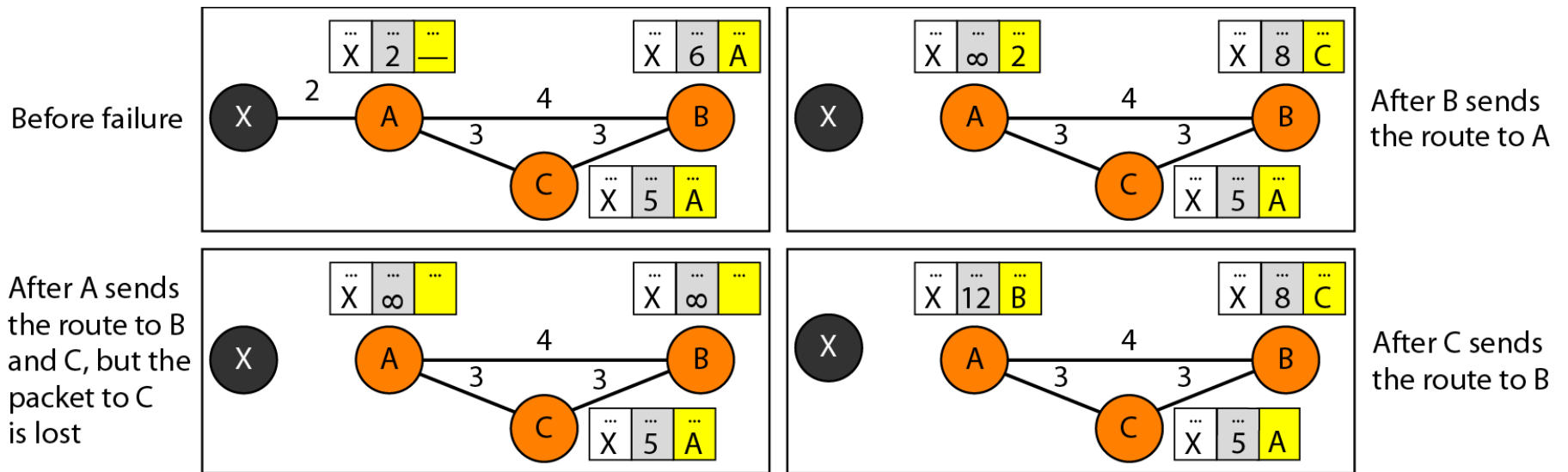
Figure 22.19 *Example of a domain using RIP*

Dest.	Hop	Next
130.10.0.0	1	———
130.11.0.0	2	130.10.0.2
195.2.4.0	1	———
195.2.5.0	1	———
195.2.6.0	2	195.2.5.2
205.5.5.0	3	130.10.0.2
205.5.6.0	3	130.10.0.2

R2 Table

Dest.	Hop	Next
130.10.0.0	2	195.2.5.1
130.11.0.0	3	195.2.5.1
195.2.4.0	2	195.2.5.1
195.2.5.0	1	———
195.2.6.0	1	———
205.5.5.0	4	195.2.5.1
205.5.6.0	4	195.2.5.1

R3 Table

Dest.	Hop	Next
130.10.0.0	1	———
130.11.0.0	1	———
195.2.4.0	2	130.10.0.1
195.2.5.0	2	130.10.0.1
195.2.6.0	3	130.10.0.1
205.5.5.0	2	130.11.0.1
205.5.6.0	2	130.11.0.1

R1 Table

Dest.	Hop	Next
130.10.0.0	2	130.11.0.2
130.11.0.0	1	———
195.2.4.0	3	130.11.0.2
195.2.5.0	3	130.11.0.2
195.2.6.0	4	130.11.0.2
205.5.5.0	1	———
205.5.6.0	1	———

R4 Table

195.2.4.0
130.10.0.0
130.11.0.0
205.5.5.0
195.2.4.1
130.10.0.2
R1
205.5.5.1
R4
130.10.0.1
130.11.0.1
R2
195.2.5.1
130.11.0.2
195.2.6.1
205.5.6.1
195.2.5.0
195.2.5.2
R3
195.2.6.1
205.5.6.0
195.2.6.0

Figure 22.20 *Concept of link state routing*
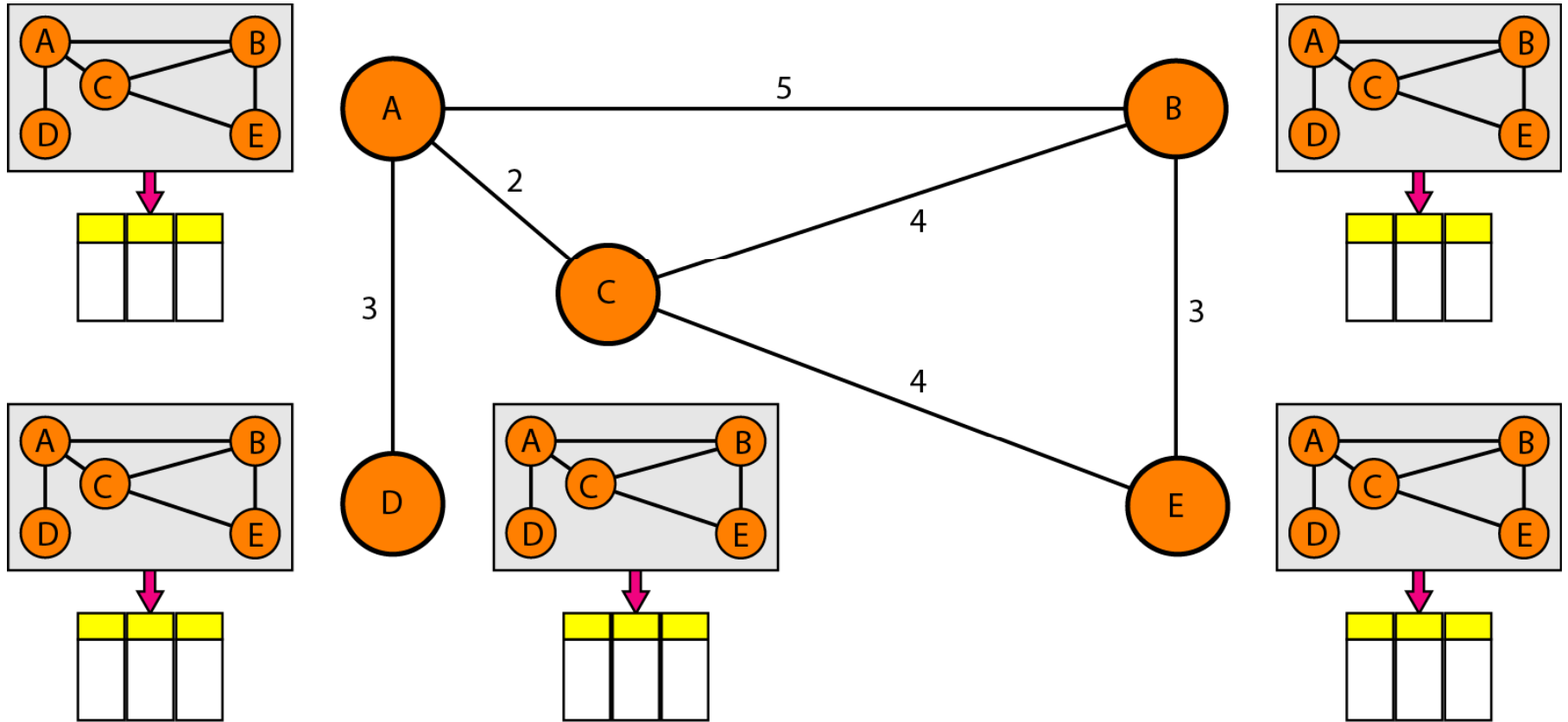
Figure 22.21 *Link state knowledge*

A's states of links

B's states of links

C's states of links

D's states of links

E's states of links
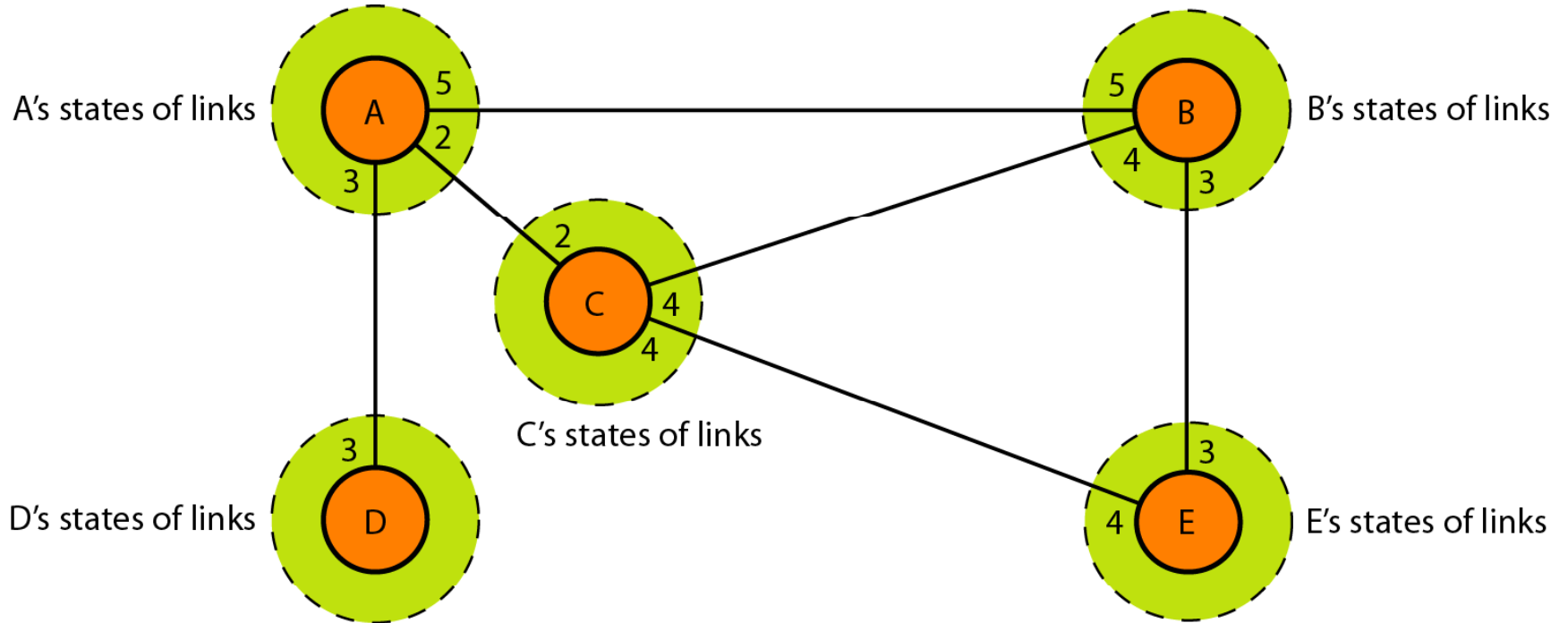
Figure 22.22 *Dijkstra algorithm*

22.45
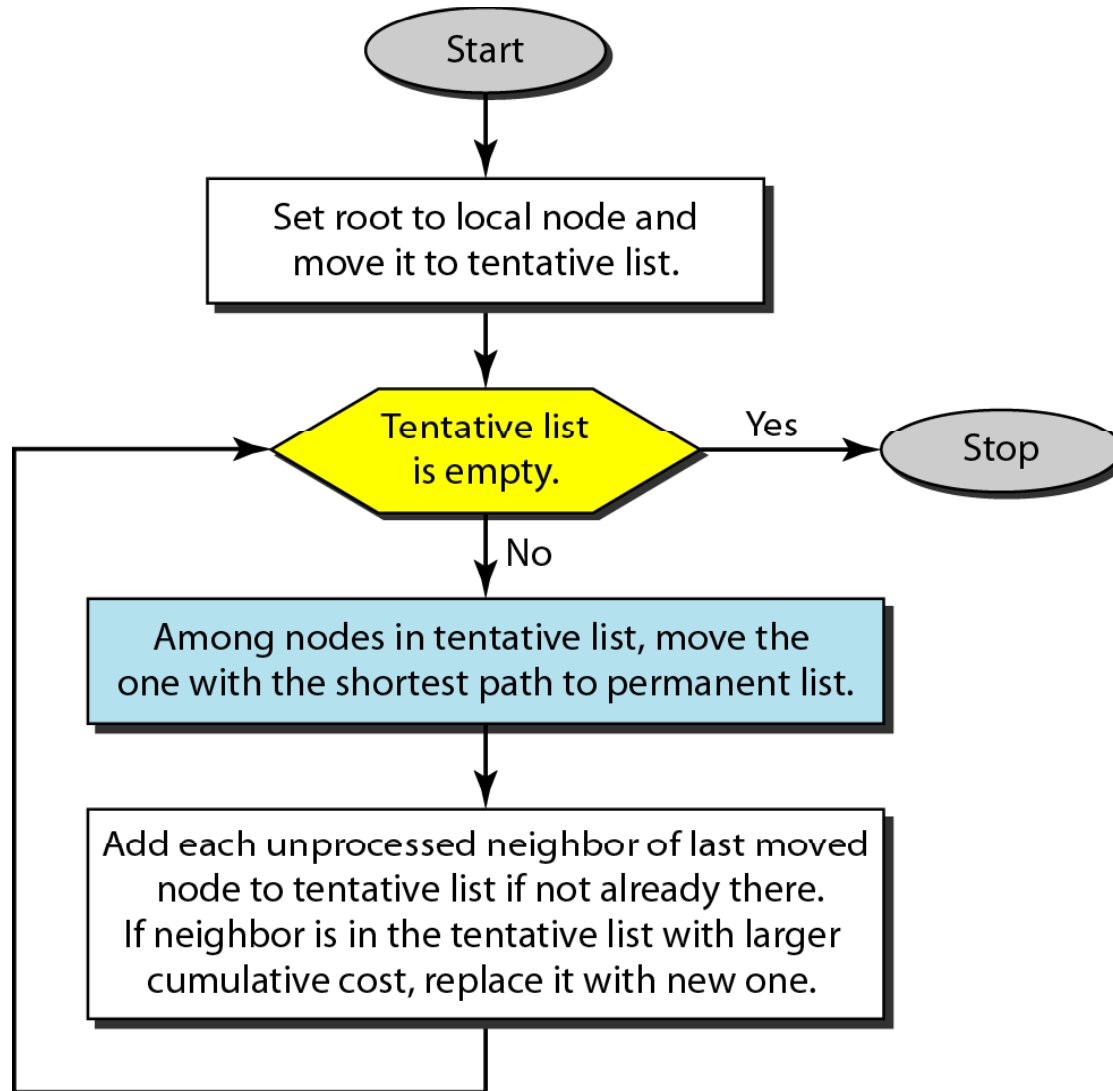
Figure 22.23 *Example of formation of shortest path tree*

Topology

1. Set root to A and move A to tentative list.

2. Move A to permanent list and add B, C, and D to tentative list.

3. Move C to permanent and add E to tentative list.

4. Move D to permanent list.

5. Move B to permanent list.

6. Move E to permanent list (tentative list is empty).
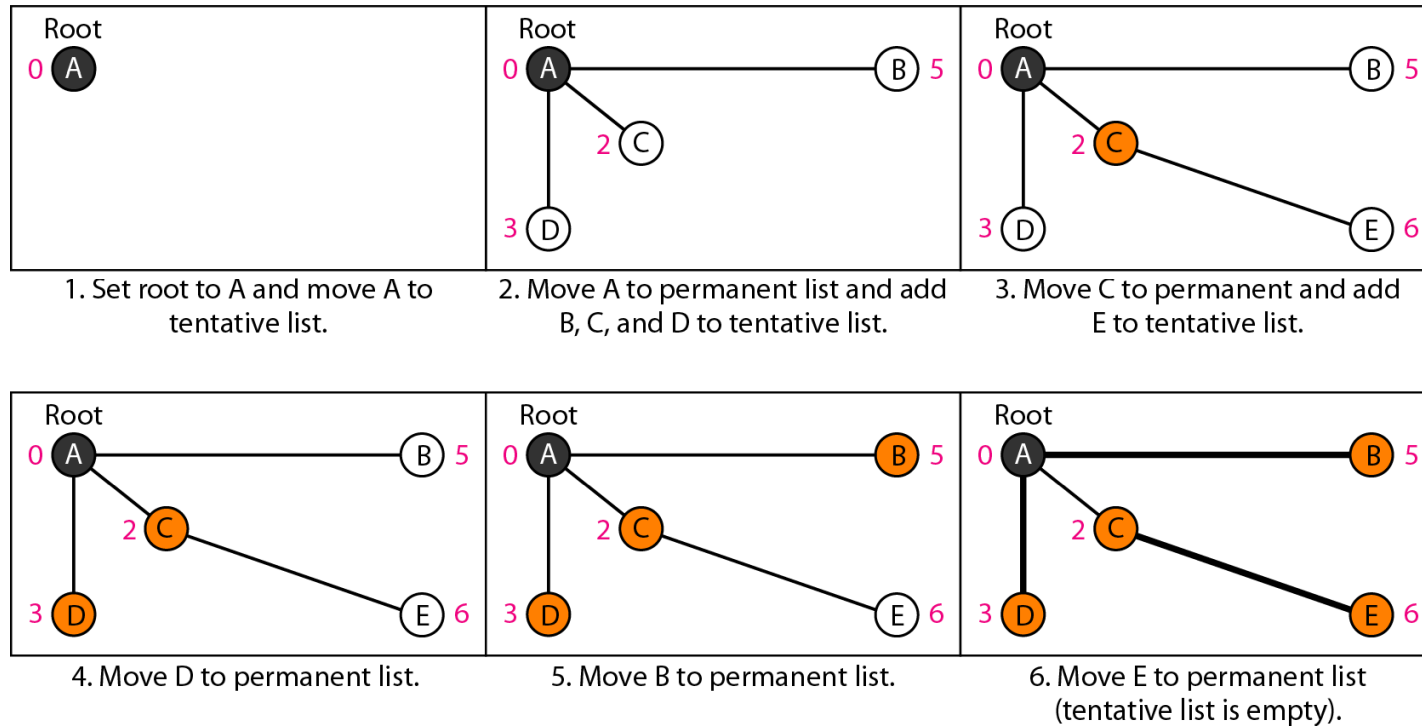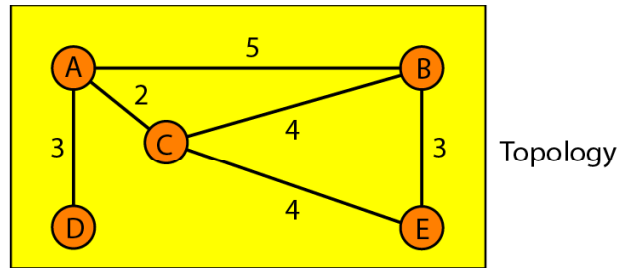
Table 22.2 *Routing table for node A*

Node	Cost	Next Router
A	0	—
B	5	—
C	2	—
D	3	—
E	6	C

Figure 22.24 *Areas in an autonomous system*
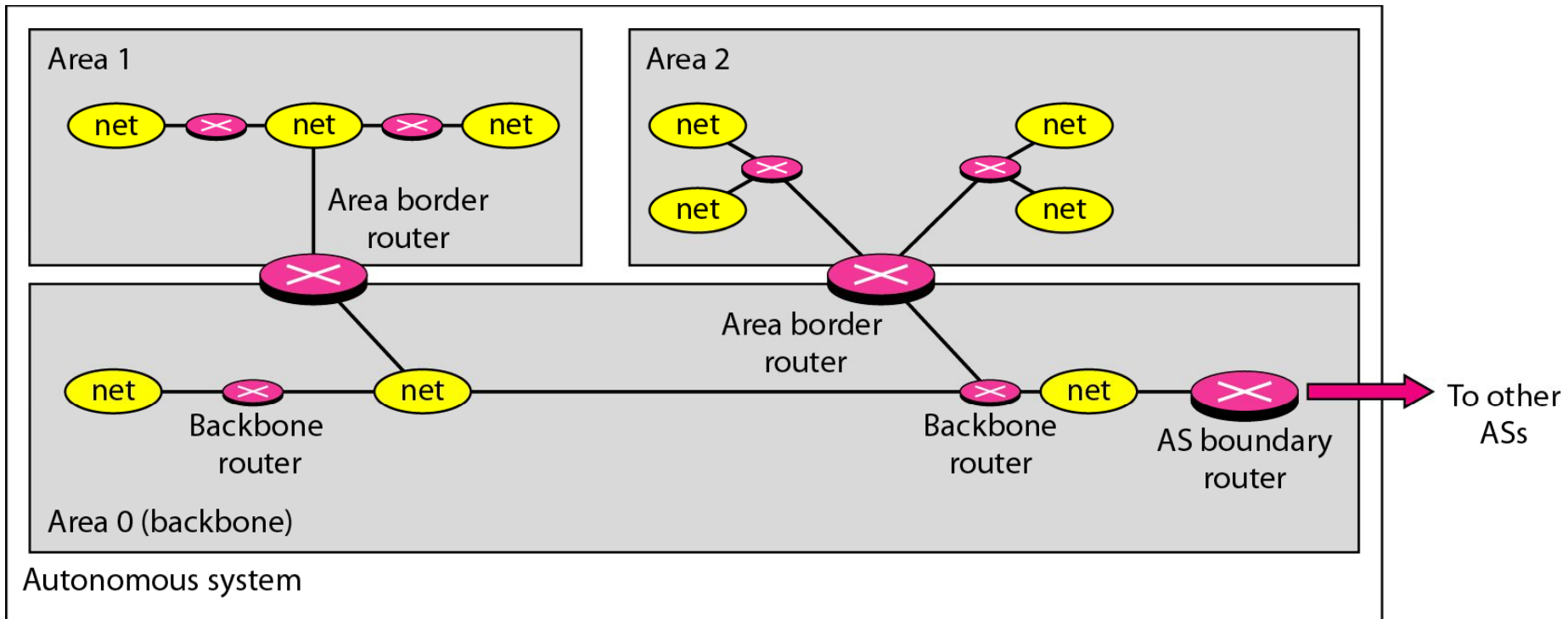
Figure 22.25 *Types of links*
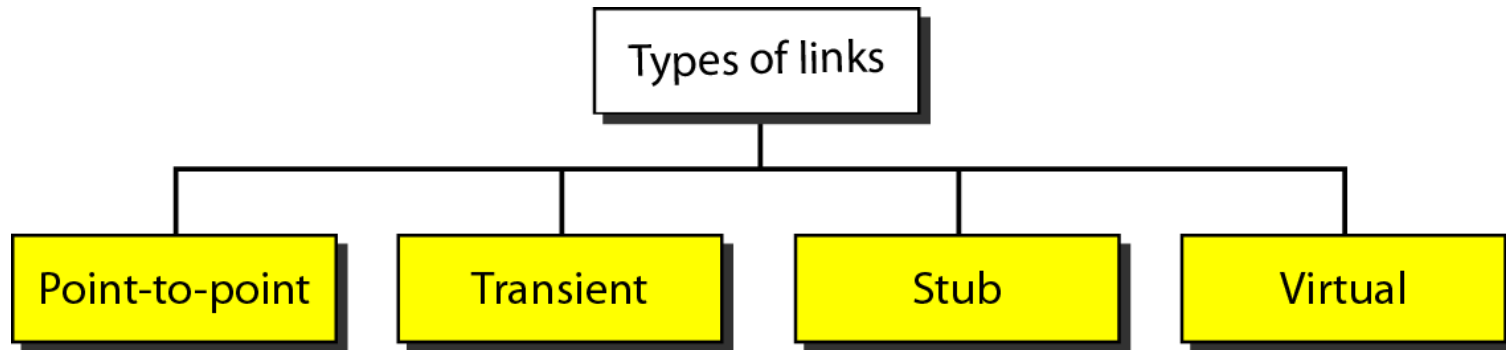
Figure 22.26 *Point-to-point link*

Point-to-point network
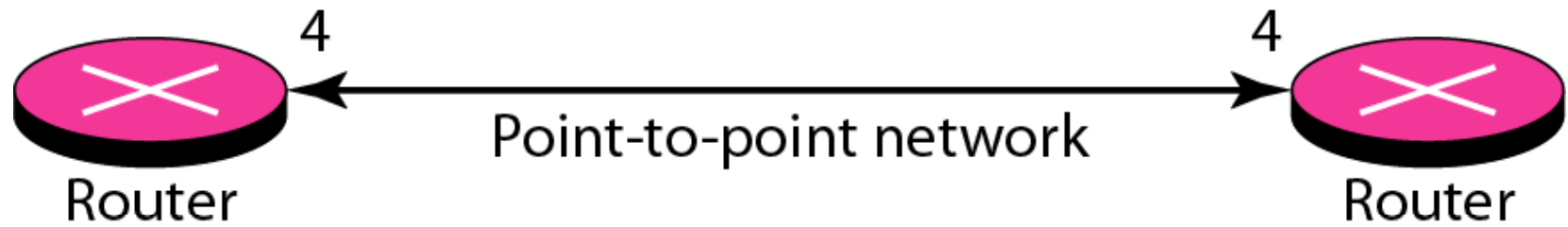
Figure 22.27 Transient link
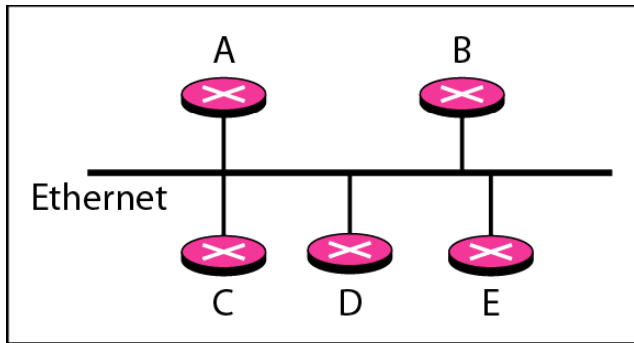
a. Transient network
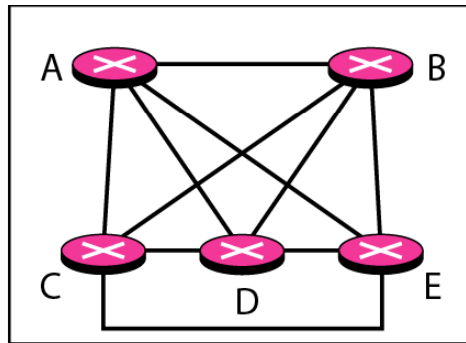
b. Unrealistic representation
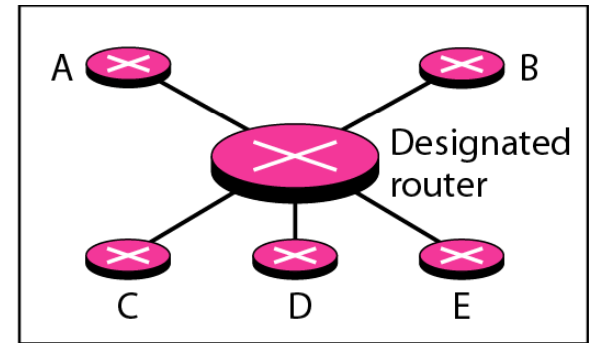
c. Realistic representation

Figure 22.28 *Stub link*
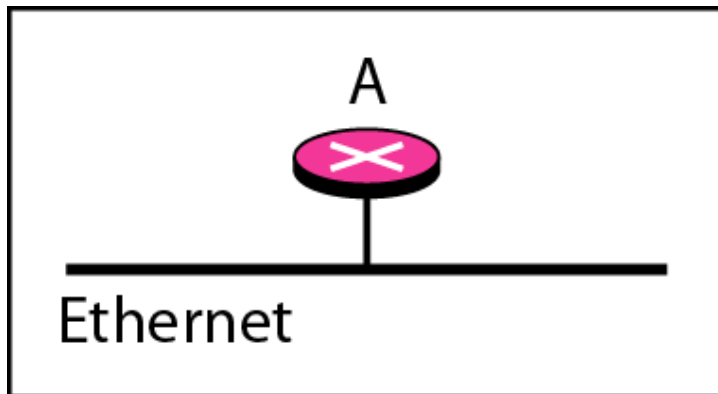
a. Stub network

b. Representation

Figure 22.29 *Example of an AS and its graphical representation in OSPF*
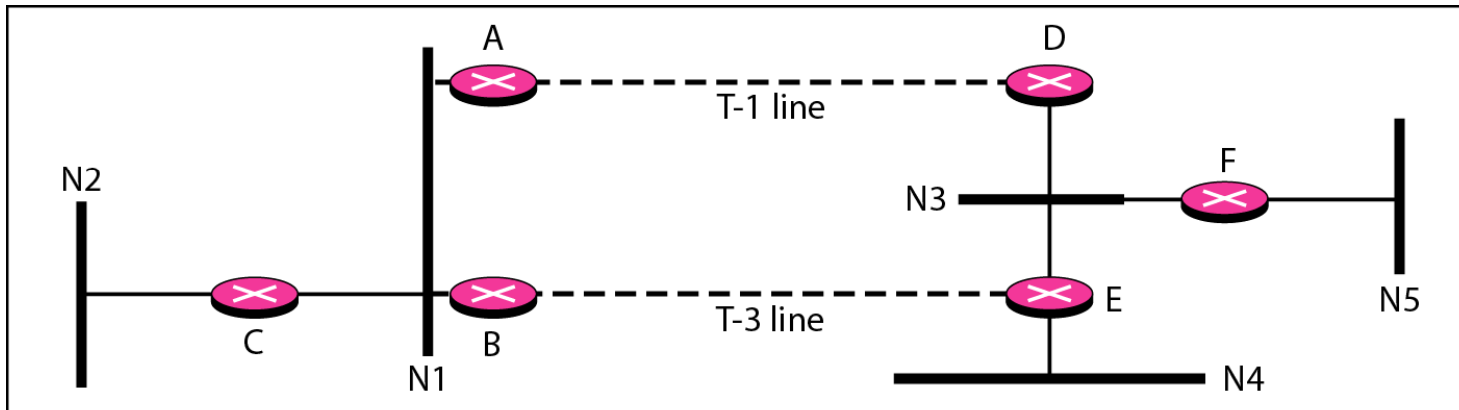
a. Autonomous system
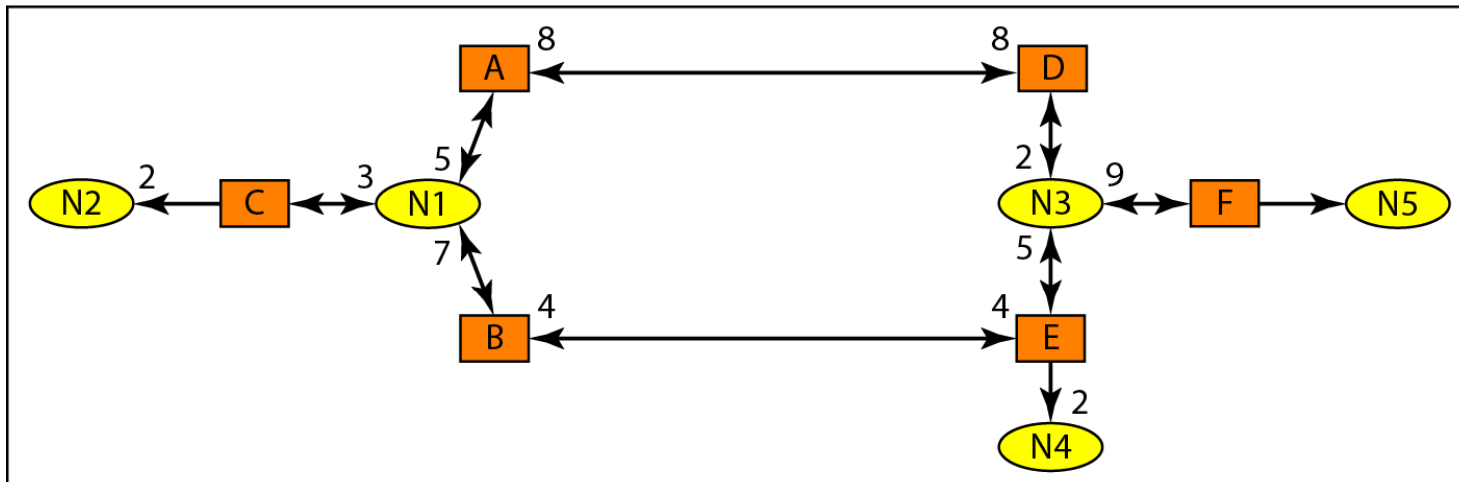
b. Graphical representation

Figure 22.30 *Initial routing tables in path vector routing*
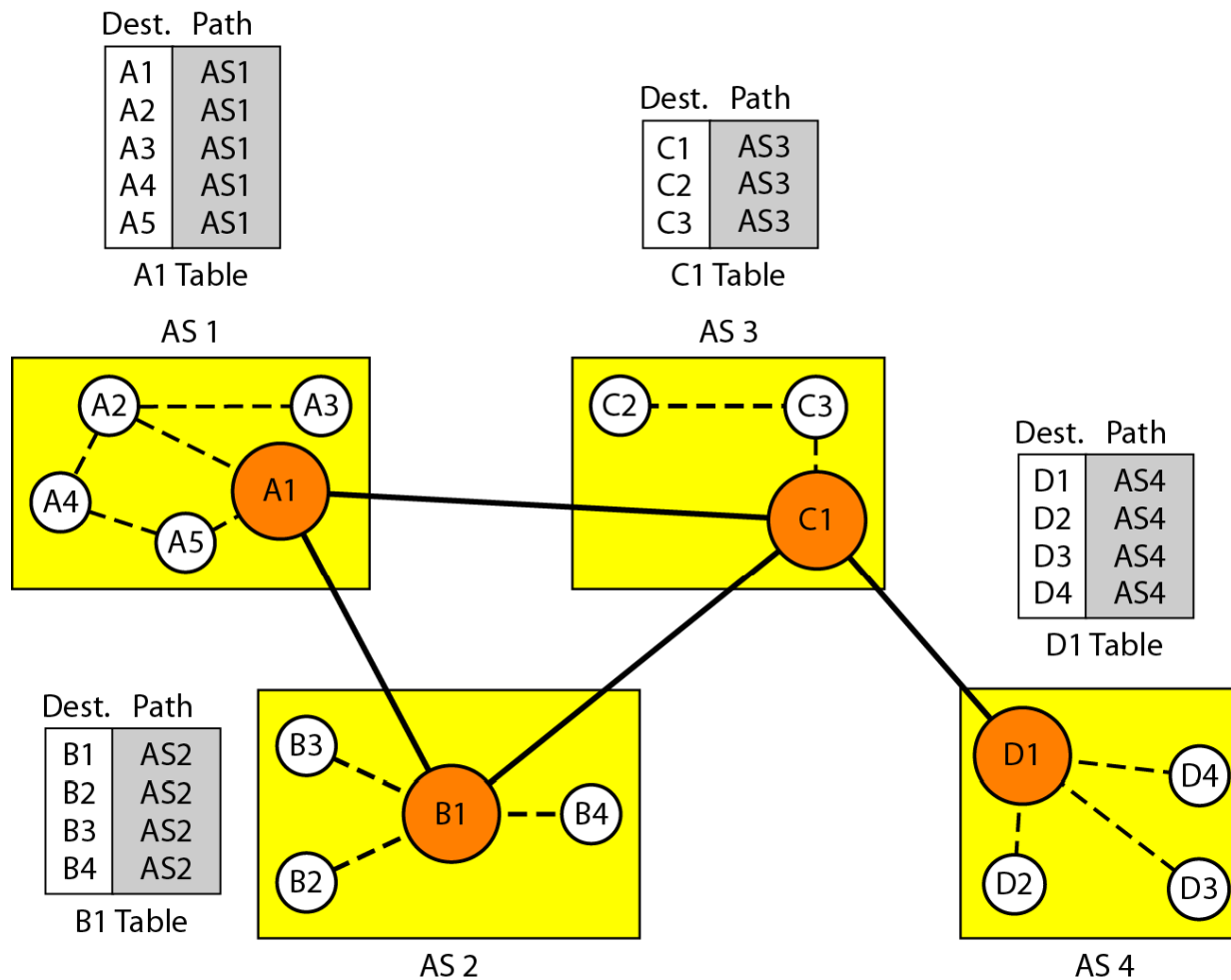
Figure 22.31 Stabilized tables for three autonomous systems

Dest.	Path
A1 ... A5	AS1 ... AS1
B1 ... B4	AS1-AS2 ... AS1-AS2
C1 ... C3	AS1-AS3 ... AS1-AS3
D1 ... D4	AS1-AS2-AS4 ... AS1-AS2-AS4

A1 Table

Dest.	Path
A1 ... A5	AS2-AS1 ... AS2-AS1
B1 ... B4	AS2 ... AS2
C1 ... C3	AS2-AS3 ... AS2-AS3
D1 ... D4	AS2-AS3-AS4 ... AS2-AS3-AS4

B1 Table

Dest.	Path
A1 ... A5	AS3-AS1 ... AS3-AS1
B1 ... B4	AS3-AS2 ... AS3-AS2
C1 ... C3	AS3 ... AS3
D1 ... D4	AS3-AS4 ... AS3-AS4

C1 Table

Dest.	Path
A1 ... A5	AS4-AS3-AS1 ... AS4-AS3-AS1
B1 ... B4	AS4-AS3-AS2 ... AS4-AS3-AS2
C1 ... C3	AS4-AS3 ... AS4-AS3
D1 ... D4	AS4 ... AS4

D1 Table

Figure 22.32 *Internal and external BGP sessions*
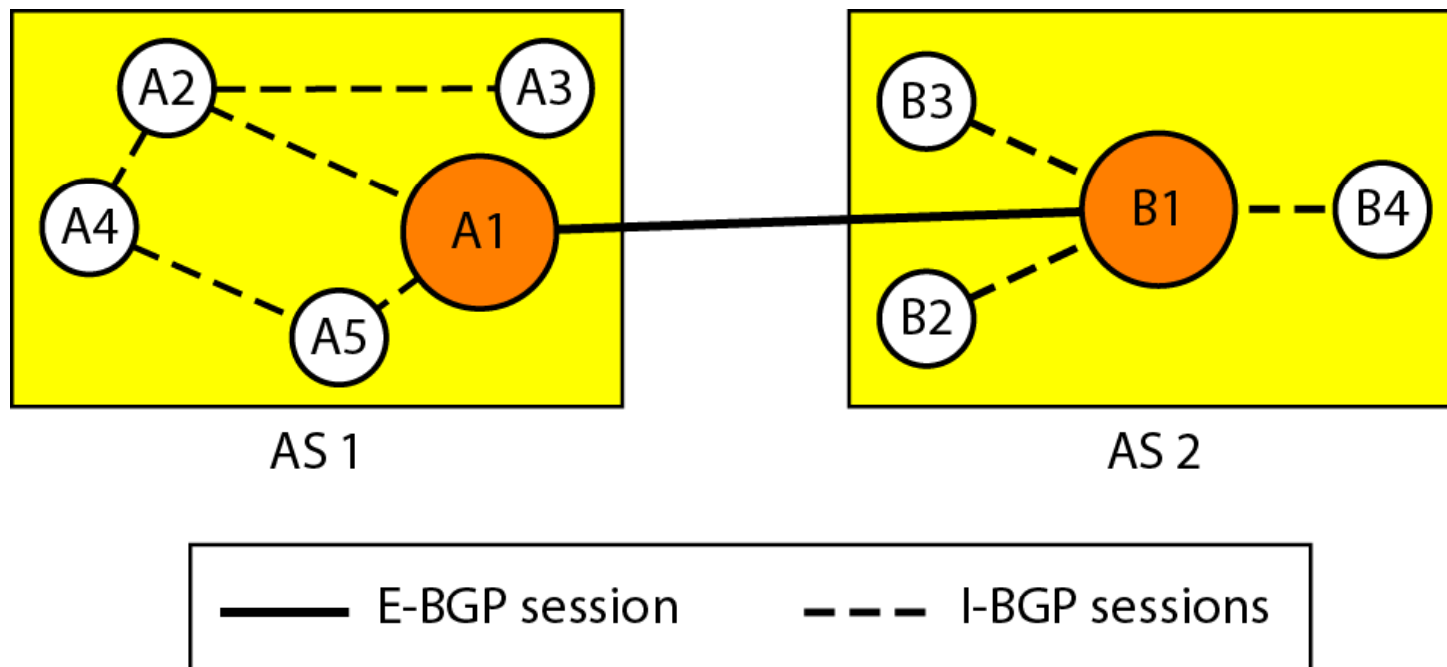
22-4   MULTICAST ROUTING PROTOCOLS

In this section, we discuss multicasting and multicast routing protocols.

<u>*Topics discussed in this section:*</u>

**Unicast, Multicast, and Broadcast
Applications
Multicast Routing
Routing Protocols**

22.57

Figure 22.33 *Unicasting*
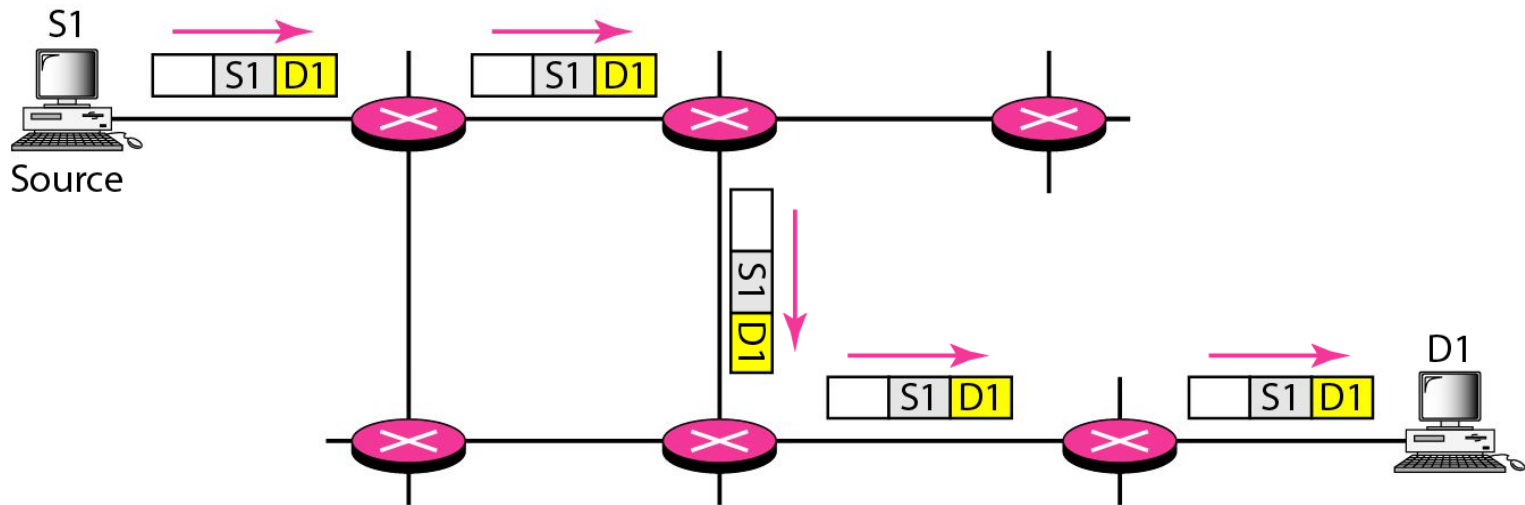
Note

In unicasting, the router forwards the received packet through only one of its interfaces.

Figure 22.34 *Multicasting*
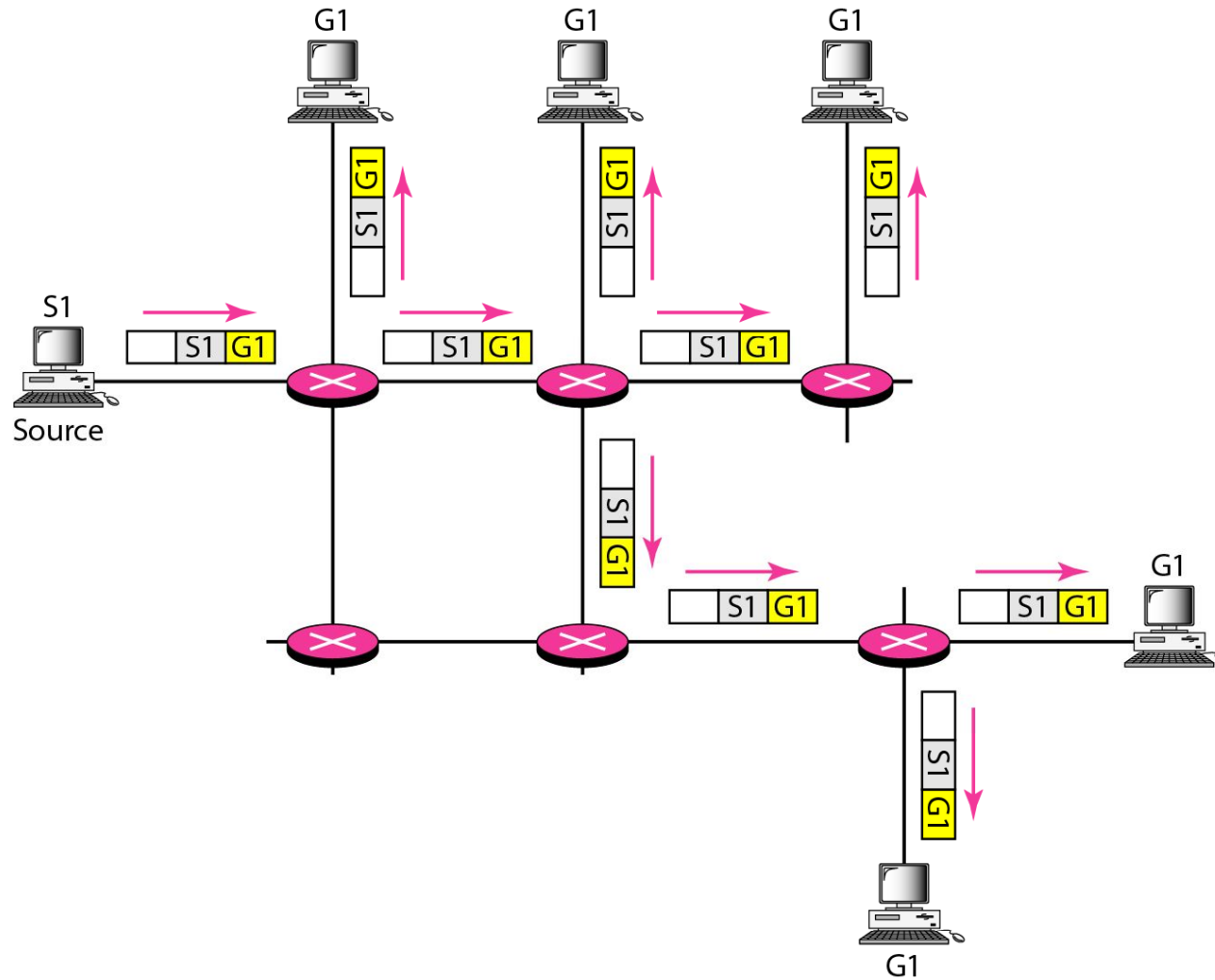
In multicasting, the router may forward the received packet through several of its interfaces.

Figure 22.35 *Multicasting versus multiple unicasting*
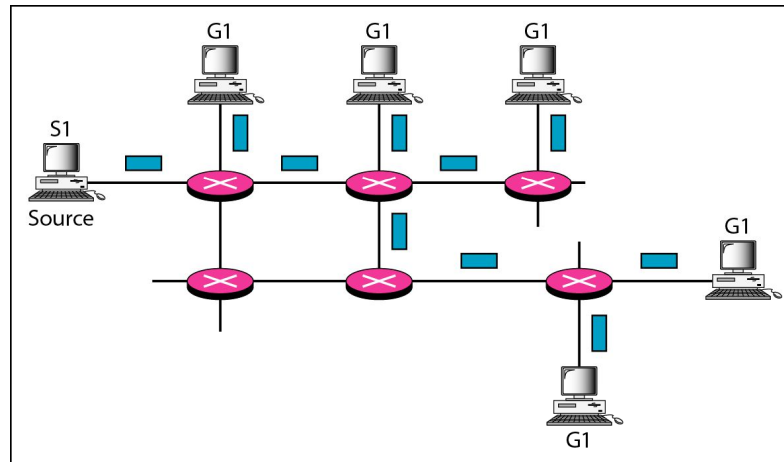
a. Multicasting
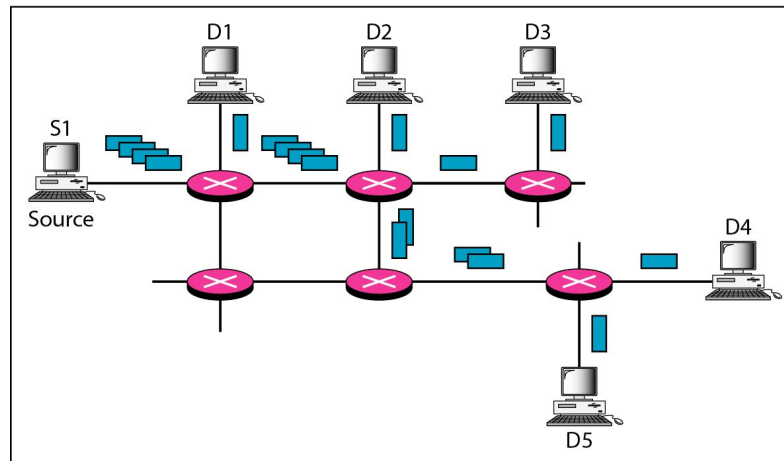
b. Multiple unicasting

Emulation of multicasting through multiple unicasting is not efficient and may create long delays, particularly with a large group.

In unicast routing, each router in the domain has a table that defines a shortest path tree to possible destinations.

Figure 22.36 *Shortest path tree in unicast routing*
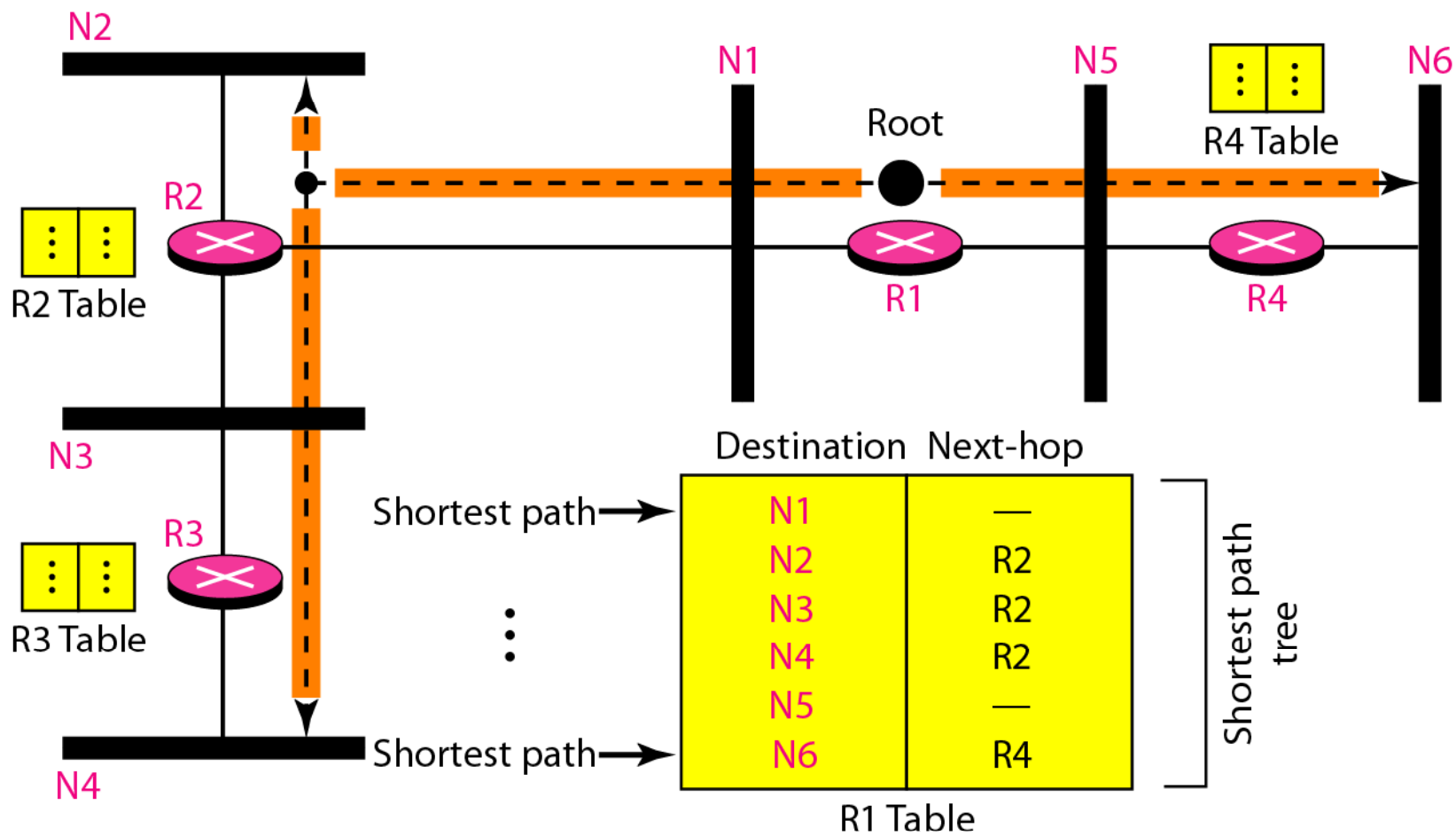
Note

In multicast routing, each involved
router needs to construct
a shortest path tree for each group.

Figure 22.37 *Source-based tree approach*

G1, G2

G1, G2   G3   G1, G4, G5

R2

R4 Table

R2 Table   R1   R4

G3, G5

R3

R3 Table

G1, G2, G4

Destination   Next hop

Shortest path tree →

Destination	Next hop
G1	—, R2, R4
G2	—, R2
G3	—, R2
G4	R2, R4
G5	R2, R4

Shortest path tree →

R1 Table

Five shortest path trees
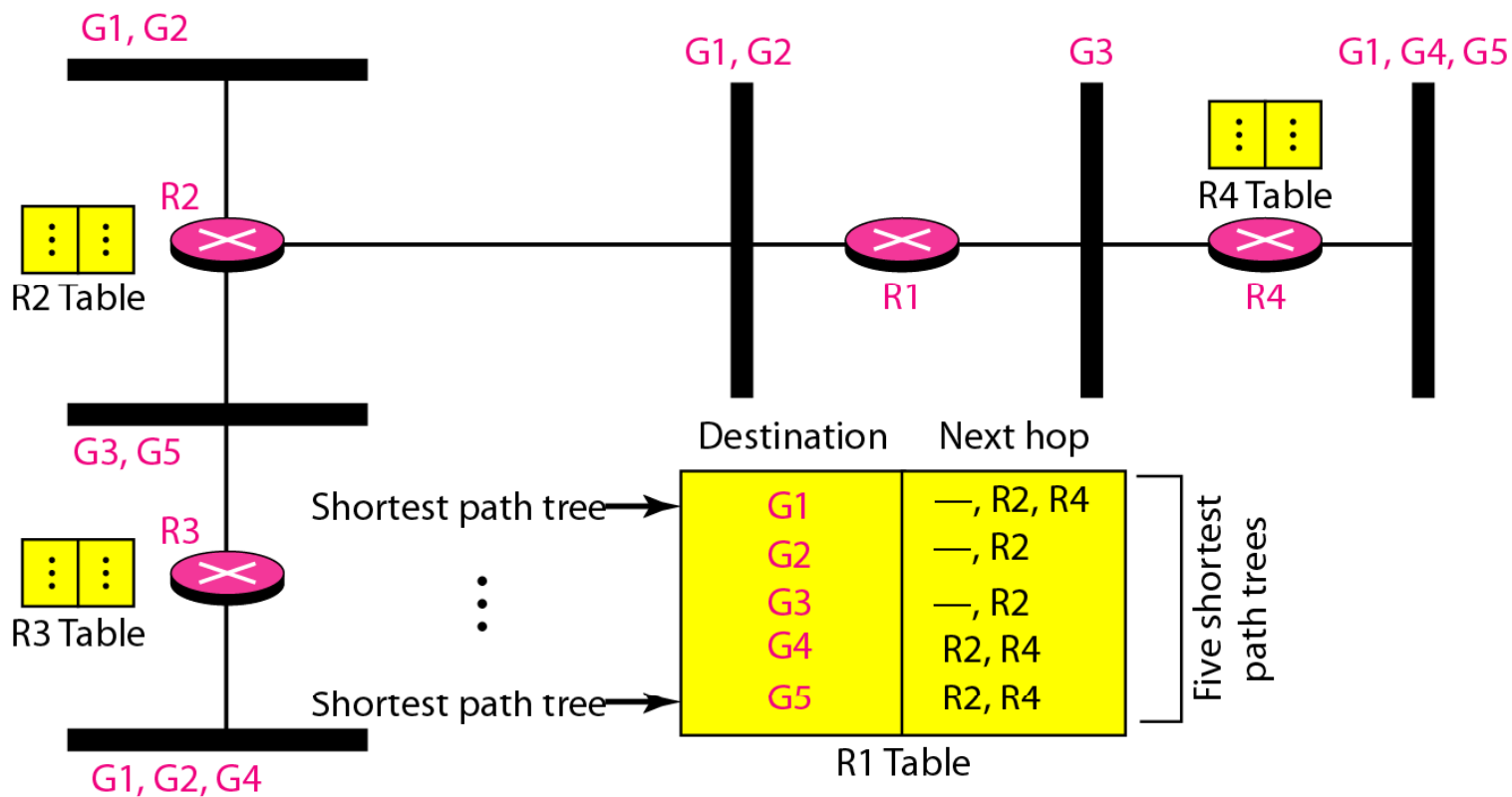
In the source-based tree approach, each router needs to have one shortest path tree for each group.

Figure 22.38 *Group-shared tree approach*

Core router

Destination	Next hop
G1	—, R2, R3, R4
G2	—, R2, R3
G3	—
G4	R3, R4
G5	—, R4

Shortest path tree →

Shortest path tree →

Core router table

Five shortest path trees
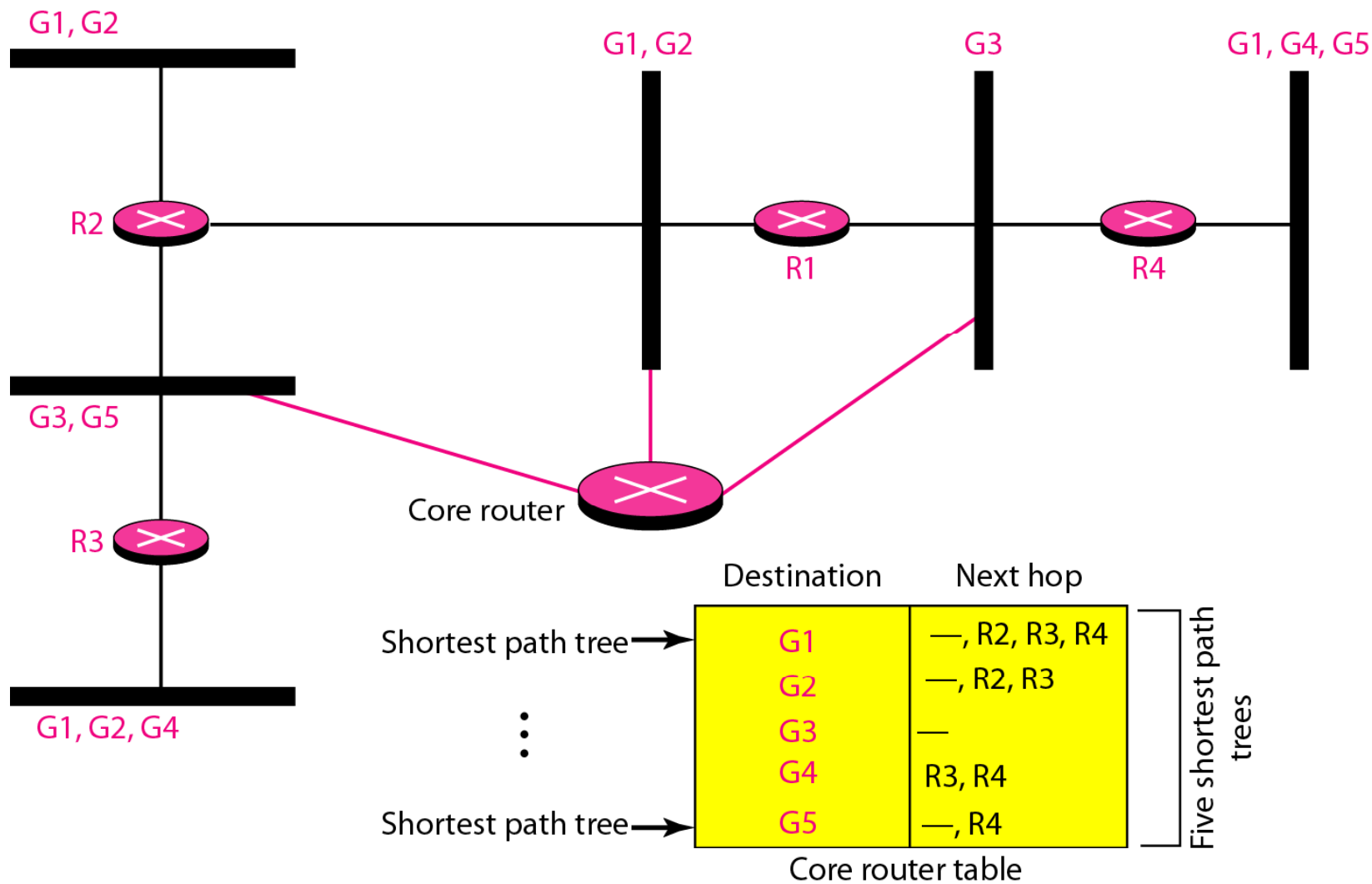
In the group-shared tree approach, only the core router, which has a shortest path tree for each group, is involved in multicasting.

Figure 22.39 *Taxonomy of common multicast protocols*
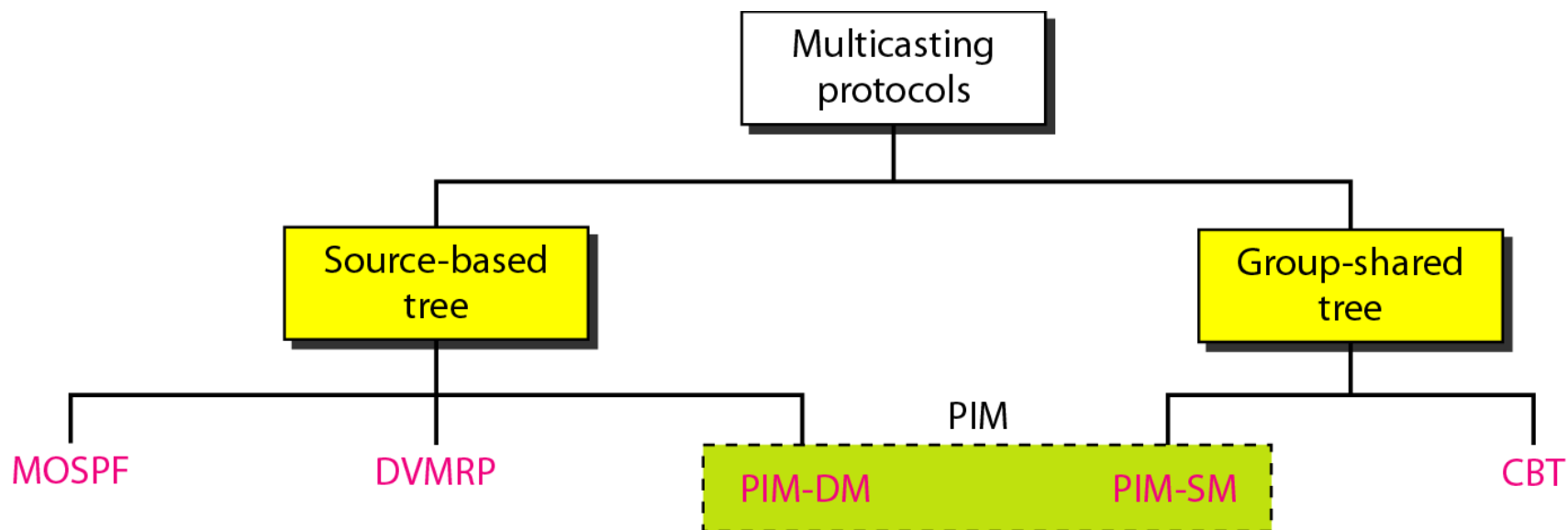
Note

Multicast link state routing uses the source-based tree approach.

Note

Flooding broadcasts packets, but creates loops in the systems.

RPF eliminates the loop in the flooding process.

Figure 22.40 *Reverse path forwarding (RPF)*
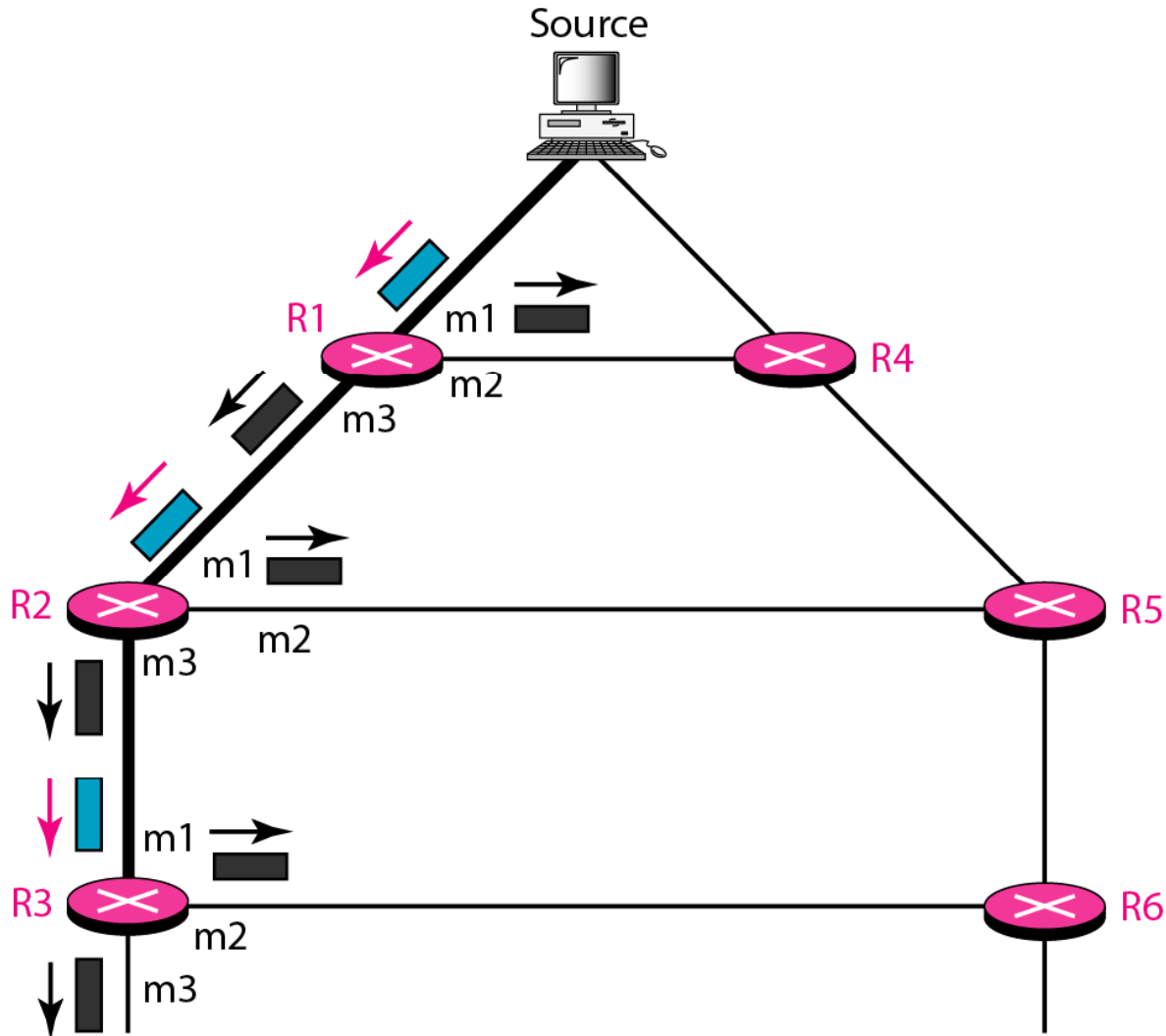
Figure 22.41 *Problem with RPF*
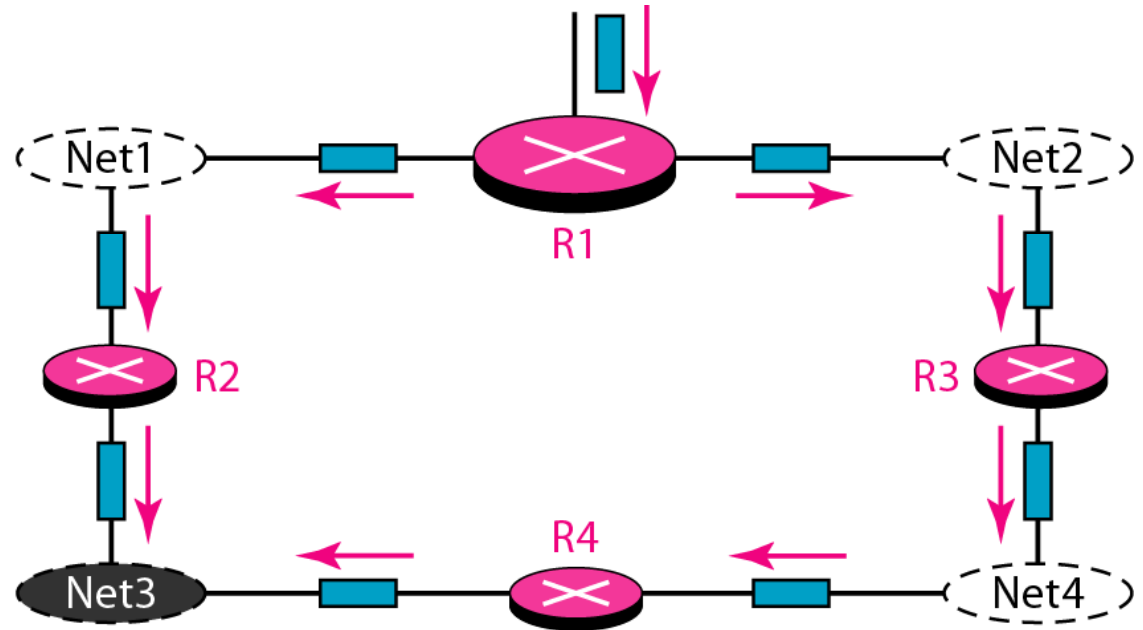
Net3 receives two copies of the packet

Figure 22.42 *RPF Versus RPB*
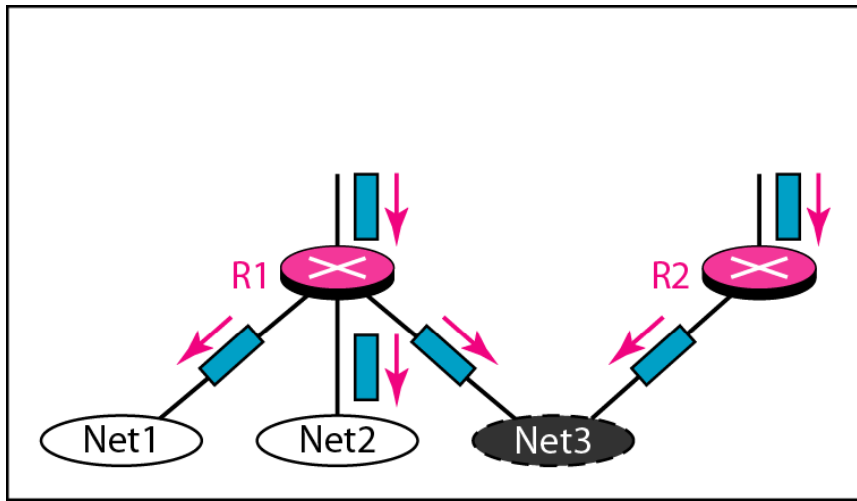
a. RPF
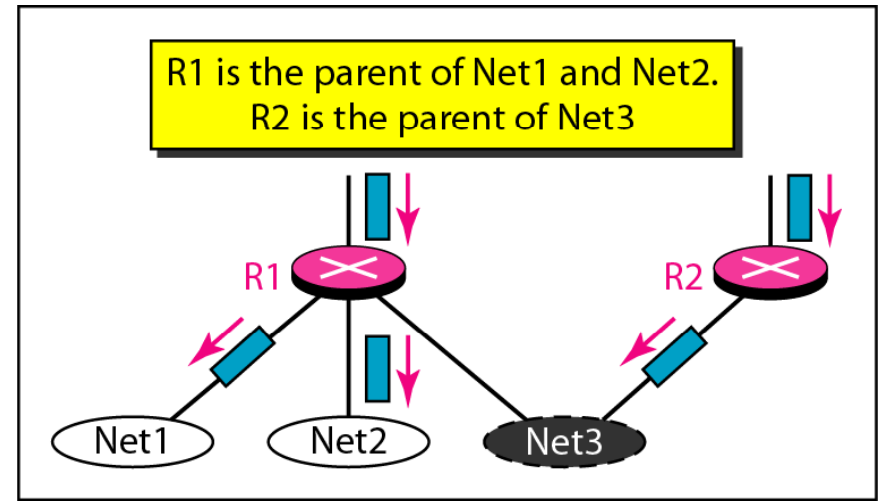
b. RPB

RPB creates a shortest path broadcast tree from the source to each destination. It guarantees that each destination receives one and only one copy of the packet.

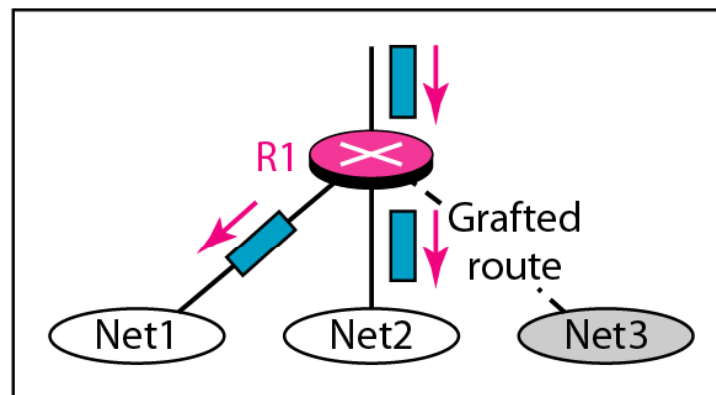# Figure 22.43 *RPF, RPB, and RPM*
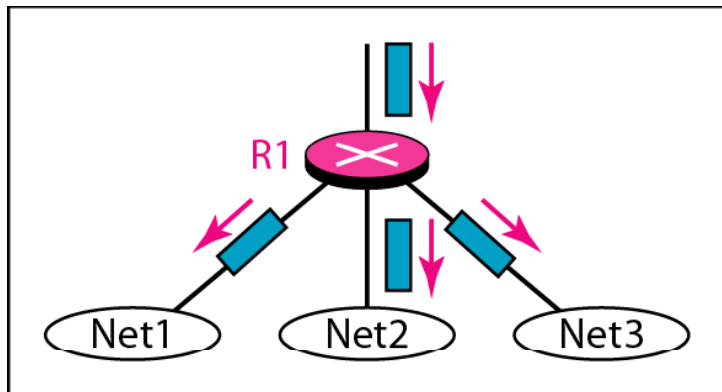
a.RPF
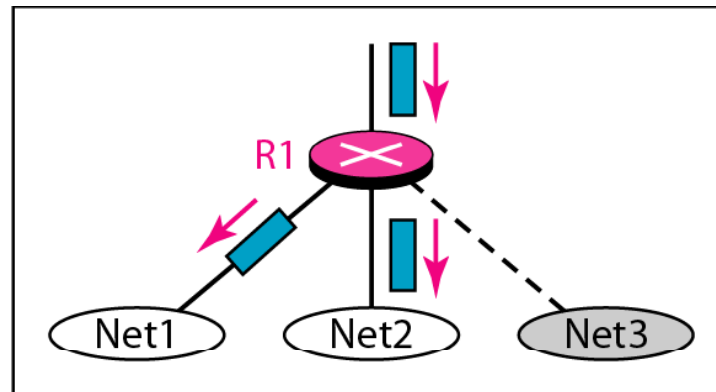
b. RPB
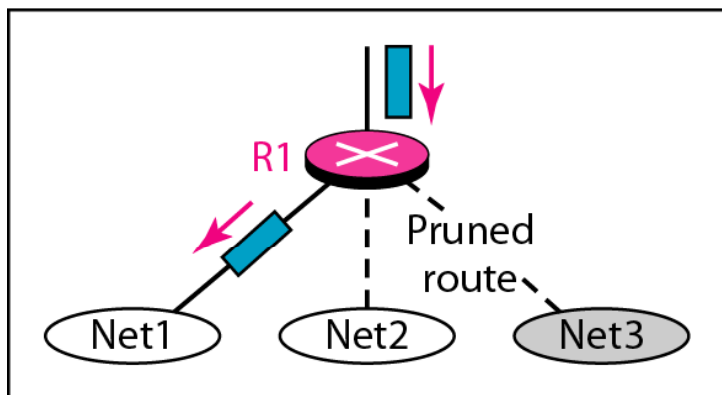
c. RPM (after pruning)

d. RPM (after grafting)

RPM adds pruning and grafting to RPB to create a multicast shortest path tree that supports dynamic membership changes.

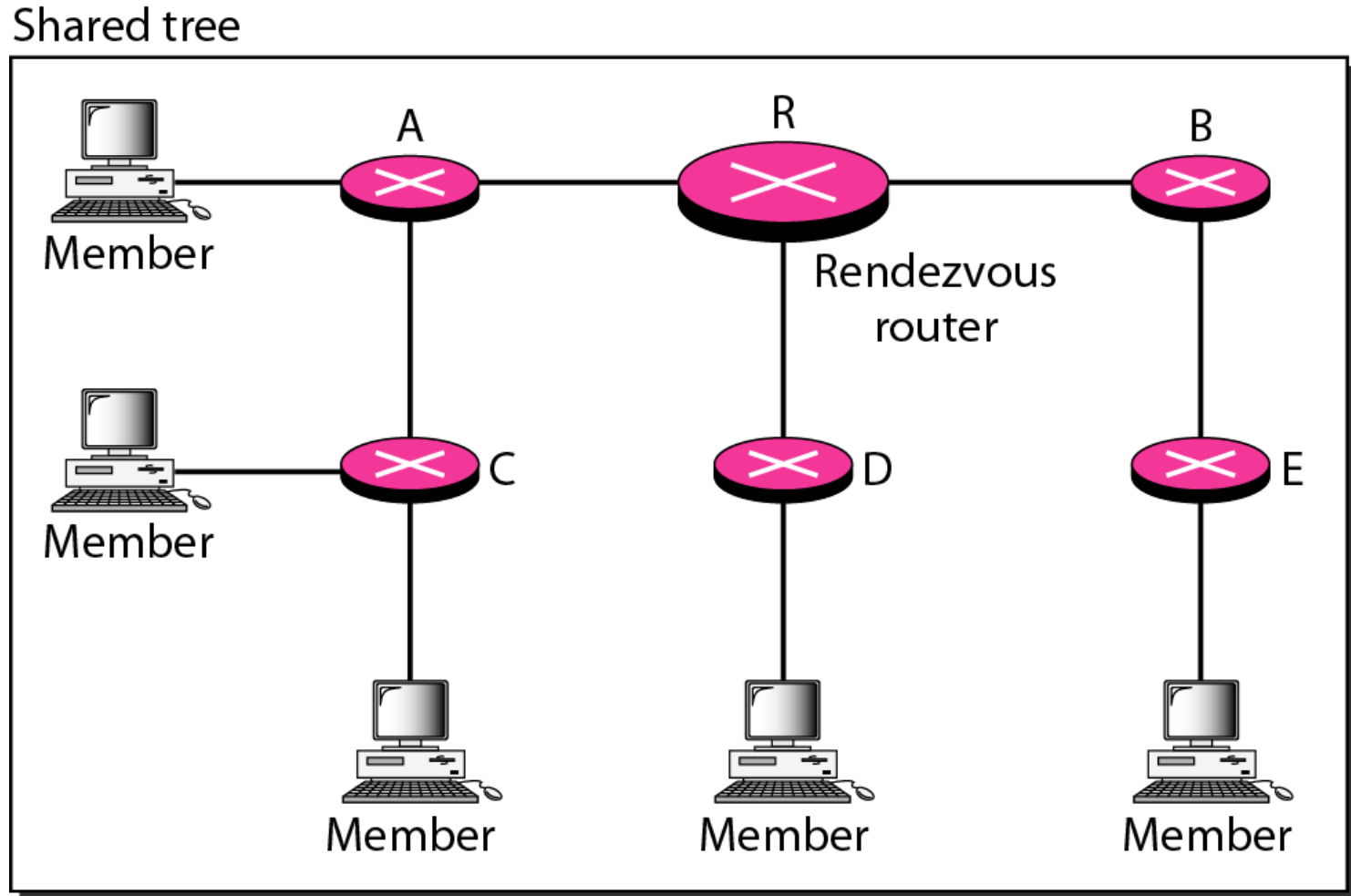# Figure 22.44 *Group-shared tree with rendezvous router*

Shared tree

A

R

B

Member

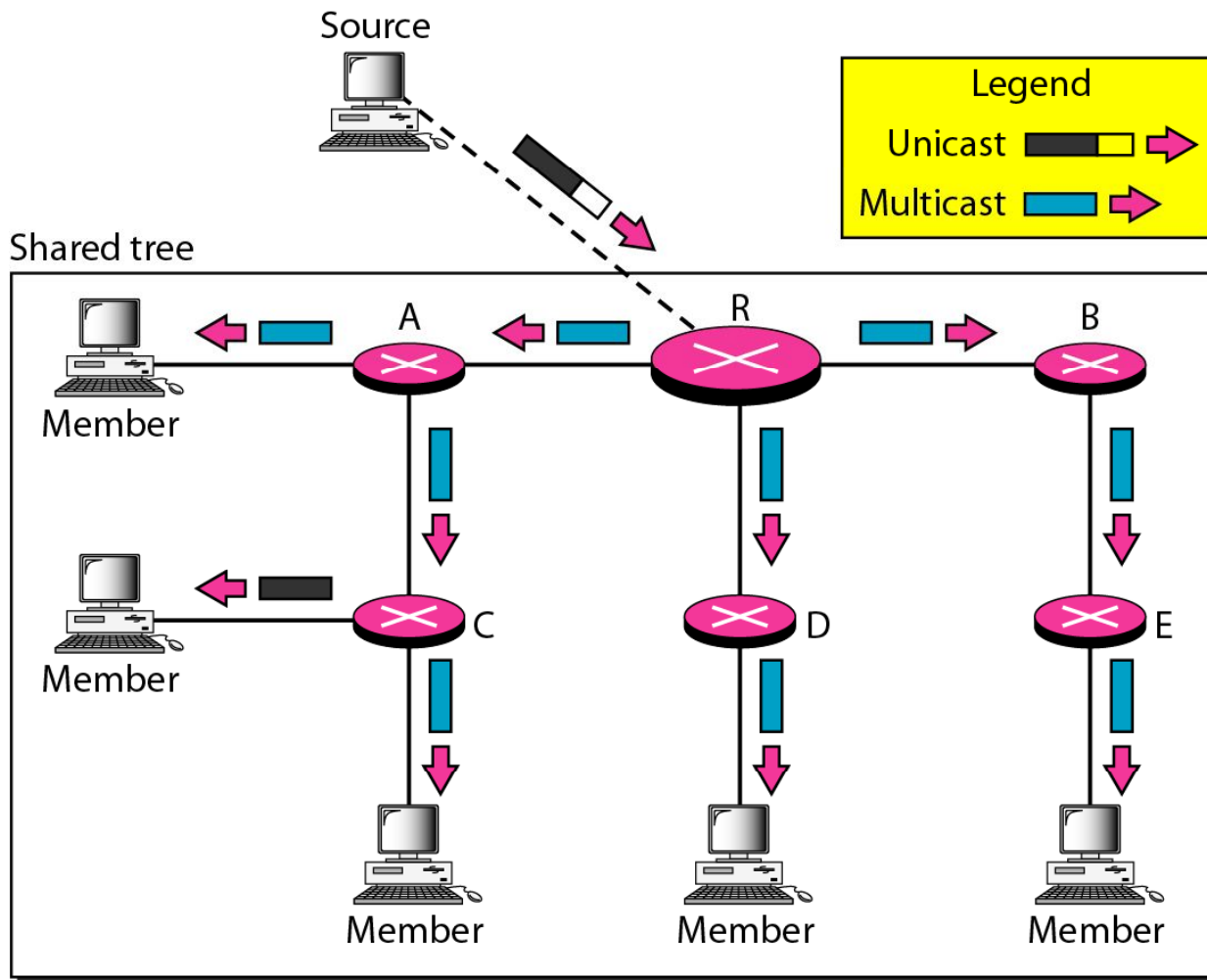Rendezvous router

C

D

E

Member

Member

Member

Member

Figure 22.45 *Sending a multicast packet to the rendezvous router*

In CBT, the source sends the multicast packet (encapsulated in a unicast packet) to the core router. The core router decapsulates the packet and forwards it to all interested interfaces.

PIM-DM is used in a dense multicast environment, such as a LAN.

PIM-DM uses RPF and pruning and grafting strategies to handle multicasting.
However, it is independent of the underlying unicast protocol.

PIM-SM is used in a sparse multicast environment such as a WAN.

PIM-SM is similar to CBT but uses a simpler procedure.
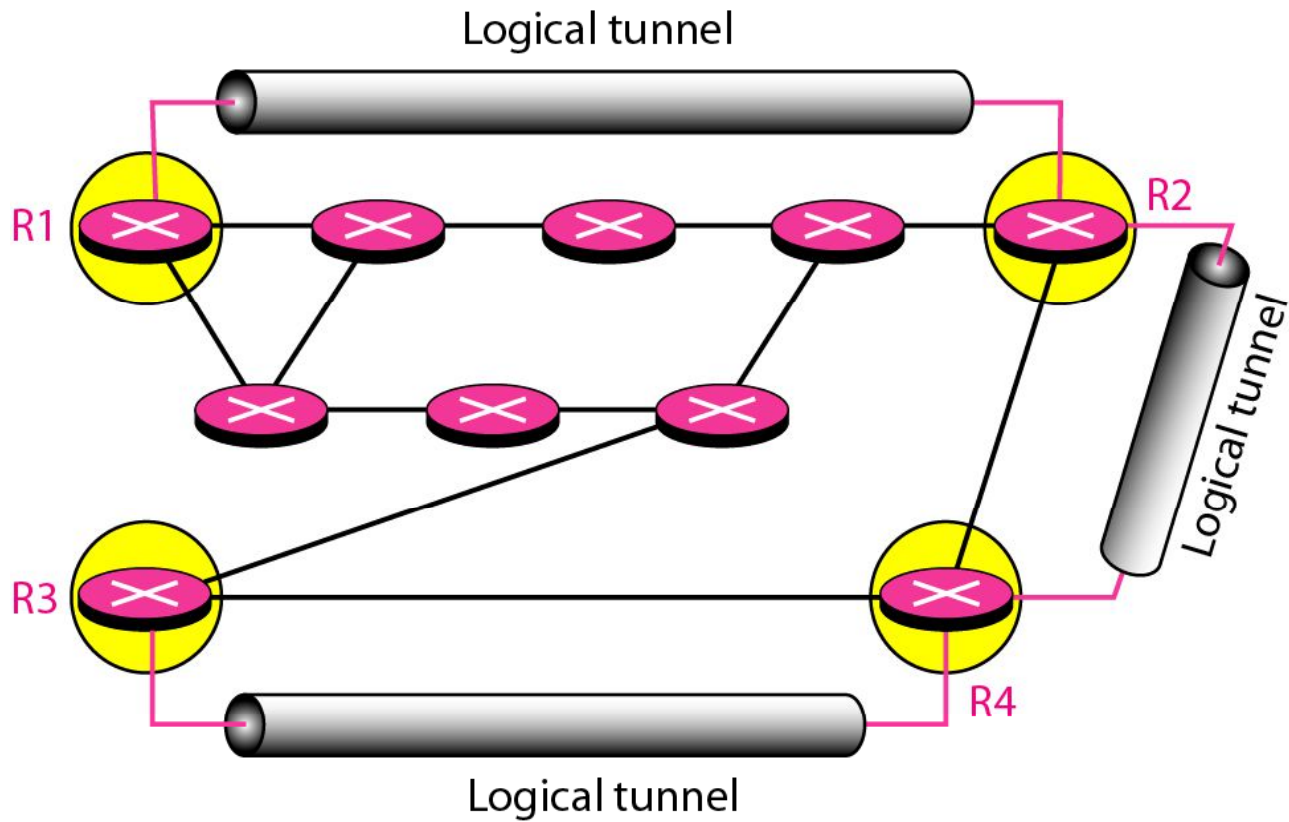
Figure 22.46 *Logical tunneling*

Figure 22.47 *MBONE*