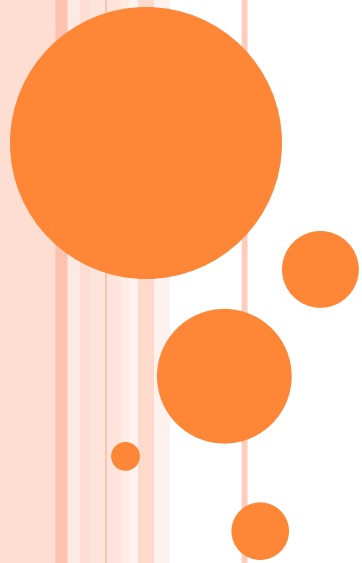


CS8591- COMPUTER NETWORKS

UNIT III

NETWORK LAYER



SYLLABUS

- Routing - Unicast Routing – Algorithms – Protocols – Multicast Routing and its basics – Overview of Intradomain and interdomain protocols – Overview of IPv6 Addressing – Transition from IPv4 to IPv6



ROUTING

- In an internet, the goal of the network layer is to deliver a datagram from its source to its destination or destinations.
- If a datagram is destined for only one destination (one-to-one delivery), we have *unicast routing*.
- If the datagram is destined for several destinations (one-to-many delivery), we have *multicast routing*.



UNICAST ROUTING

- If a datagram is destined for only one destination (one-to-one delivery)
- Unicast routing in the Internet, with a large number of routers and a huge number of hosts, can be done only by using hierarchical routing.
- Routing a packet from its source to its destination means routing the packet from a source router (the default router of the source host) to a destination router (the router connected to the destination network).
- There are several routes that a packet can travel from the source to the destination; what must be determined is which route the packet should take.



AN INTERNET AS A GRAPH

- To find the best route, an internet can be modeled as a graph.
- A graph is a set of nodes and edges (lines) that connect the nodes.
- To model an internet as a graph, each router as a node and each network between a pair of routers as an edge.
- An internet is, in fact, modeled as a weighted graph, in which each edge is associated with a cost.
- If there is no edge between the nodes, the cost is infinity.



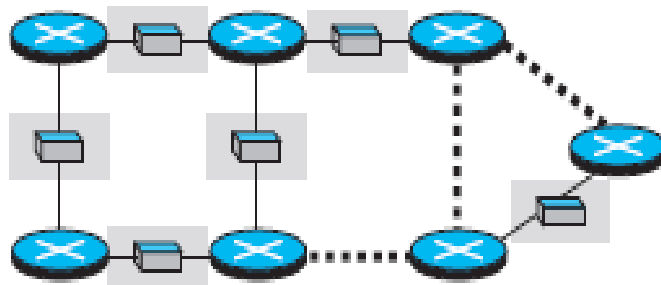
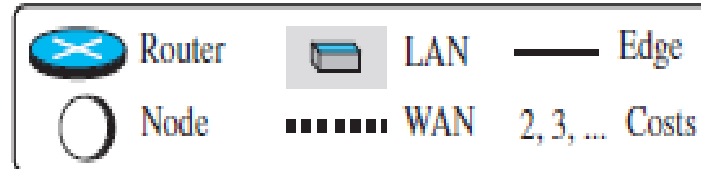
LEAST-COST ROUTING

- When an internet is modeled as a weighted graph, one of the ways to interpret the *best* route from the source router to the destination router is to find the *least cost* between the two.
- In other words, the source router chooses a route to the destination router in such a way that the total cost for the route is the least cost among all possible routes.
- In Figure , the best route between A and E is A-B-E, with the cost of 6.
- This means that each router needs to find the least-cost route between itself and all the other routers to be able to route a packet using this criteria.

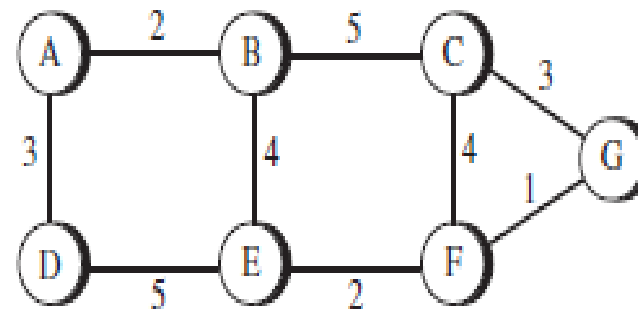


An internet and its graphical representation

Legend



a. An internet

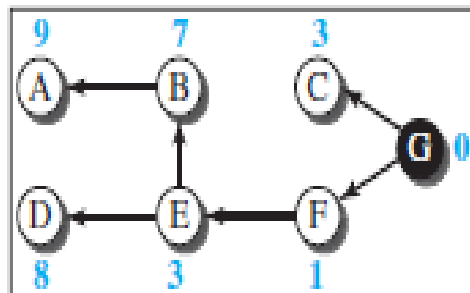
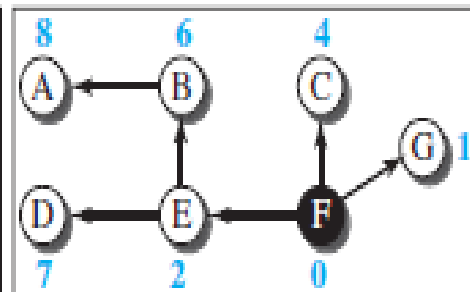
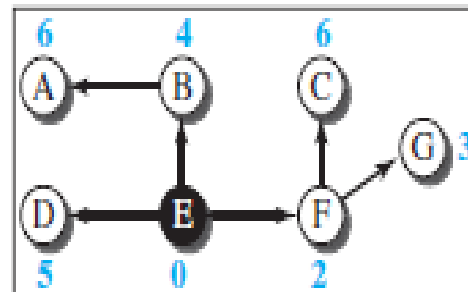
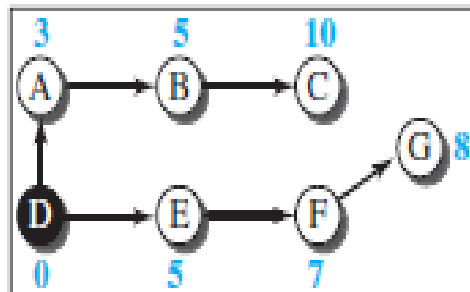
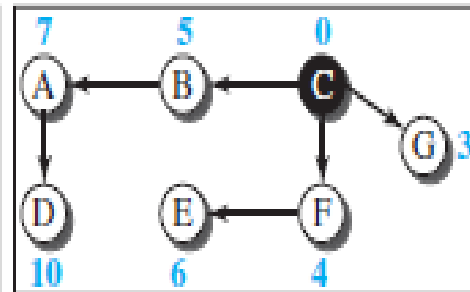
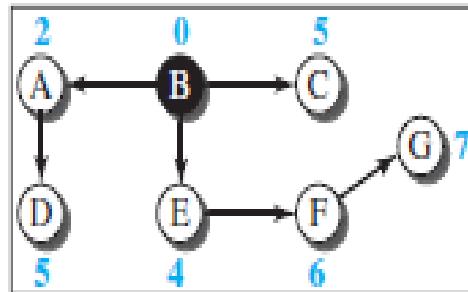
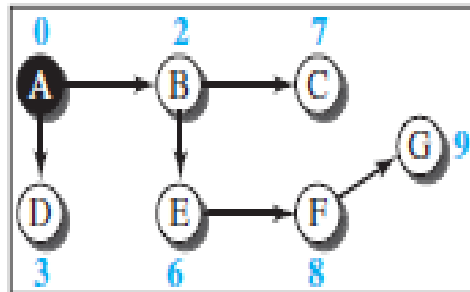


b. The weighted graph

LEAST-COST TREES

- If there are N routers in an internet, there are $(N - 1)$ least-cost paths from each router to any other router.
- This means we need $N \times (N - 1)$ least-cost paths for the whole internet.
- If we have only 10 routers in an internet, we need 90 least-cost paths.
- A better way to see all of these paths is to combine them in a **least-cost tree**.
- A least-cost tree is a tree with the source router as the root that spans the whole graph (visits all other nodes) and in which the path between the root and any other node is the shortest.
- In this way, we can have only one shortest-path tree for each node; we have N least-cost trees for the whole internet.

Least-cost trees for nodes in the internet for the above diagram



Legend



Root of the tree



Intermediate or end node

1, 2, ... Total cost from the root





UNICAST ALGORITHM

ROUTING ALGORITHMS

- The general idea behind least-cost trees and the forwarding tables that can be made from them, now we concentrate on the routing algorithms.
- Several routing algorithms have been designed as follows:
 - Distance-Vector Routing
 - Link-State Routing
 - Path-Vector Routing



DISTANCE-VECTOR ROUTING

- DVR goal is to find the best route.
- In distance-vector routing, the first thing each node creates is its own least-cost tree with the rudimentary information it has about its immediate neighbors.
- In distance-vector routing, a router continuously tells all of its neighbors what it knows about the whole internet.
- Before we show how incomplete least-cost trees can be combined to make complete ones, we need to discuss two important topics:
 - the Bellman-Ford equation and
 - The concept of distance vectors,



BELLMAN-FORD EQUATION

- The heart of distance-vector routing is the famous Bellman-Ford equation.
- This equation is used to find the least cost (shortest distance) between a source node x, and a destination node y through some intermediary nodes (a, b, c, . . .) when the costs between the Source and the intermediary nodes.
- The following shows the general case in which D_{ij} is the shortest distance and c_{ij} is the cost between nodes i and j.

$$D_{xy} = \min \{ (c_{xa} + D_{ay}), (c_{xb} + D_{by}), (c_{xc} + D_{cy}), \dots \}$$

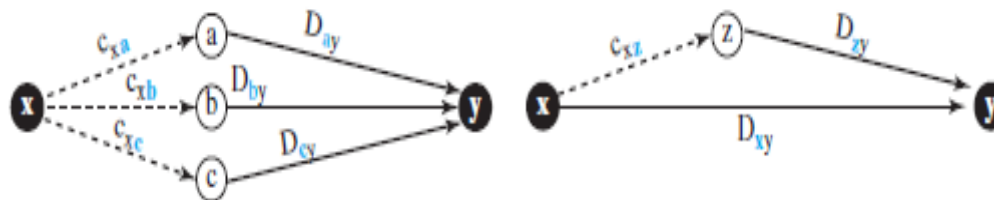


CONT..

- In distance-vector routing, normally we want to update an existing least cost with a least cost through an intermediary node, such as z, if the latter is shorter. In this case, the equation becomes simpler, as shown below:

$$D_{xy} = \min \{ D_{xy}, (c_{xz} + D_{zy}) \}$$

Graphical idea behind Bellman-Ford equation



a. General case with three intermediate nodes

b. Updating a path with a new route

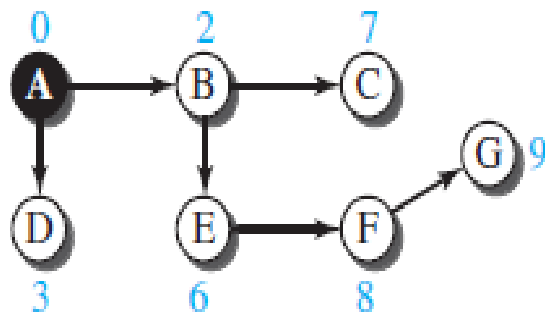


DISTANCE VECTORS:

- The concept of a distance vector is the rationale for the name distance-vector routing.
- A least-cost tree is a combination of least-cost paths from the root of the tree to all destinations.
- These paths are graphically glued together to form the tree.
- Distance-vector routing unglues these paths and creates a distance vector, a one-dimensional array to represent the tree.



The distance vector corresponding to a tree

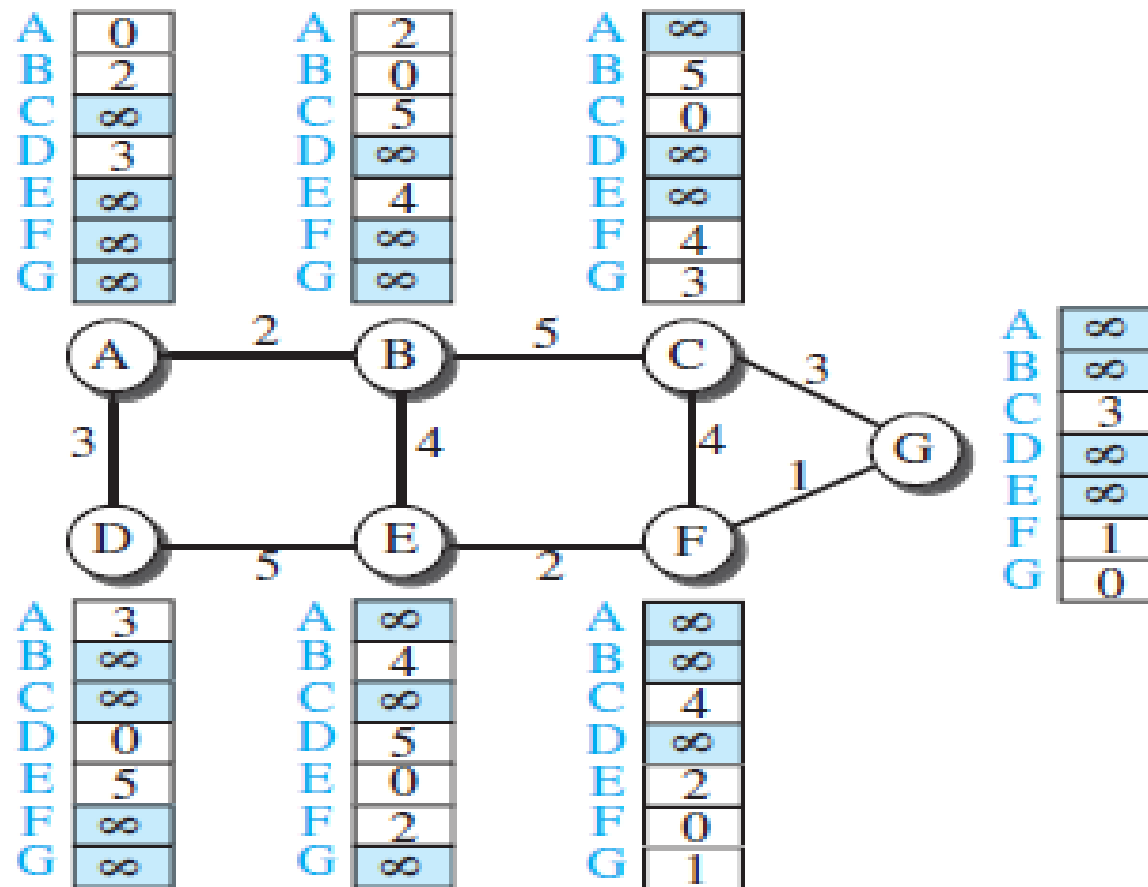


a. Tree for node A

A	
A	0
B	2
C	7
D	3
E	6
F	8
G	9

b. Distance vector for node A

The first distance vector for an internet



CONT..

- After a node receives a distance vector from a neighbor, it updates its distance vector using the Bellman-Ford equation (second case).
- In the first event, node A has sent its vector to node B. Node B updates its vector using the cost $c_{BA} = 2$.
- After the first event, node B has one improvement in its vector: its least cost to node D has changed from infinity to 5 (via node A).
- In the second event, node E has sent its vector to node B. Node B updates its vector using the cost $c_{EA} = 4$.
- After the second event, node B has one more improvement in its vector; its least cost to node F has changed from infinity to 6 (via node E).



Updating distance vectors

New B		Old B		A	
A	2	A	2	A	0
B	0	B	0	B	2
C	5	C	5	C	∞
D	5	D	∞	D	3
E	4	E	4	E	∞
F	∞	F	∞	F	∞
G	∞	G	∞	G	∞

$B[j] = \min(B[j], 2 + A[j])$

a. First event: B receives a copy of A's vector.

New B		Old B		E	
A	2	A	2	A	∞
B	0	B	0	B	4
C	5	C	5	C	∞
D	5	D	5	D	5
E	4	E	4	E	0
F	6	F	∞	F	2
G	∞	G	∞	G	∞

$B[j] = \min(B[j], 4 + E[j])$

b. Second event: B receives a copy of E's vector.

Note:

$X[j]$: the whole vector



COUNT TO INFINITY

- A problem with distance-vector routing is that any decrease in cost (good news) propagates quickly, but any increase in cost (bad news) will propagate slowly.
- The problem is referred to as count to infinity. It sometimes takes several updates before the cost for a broken link is recorded as infinity by all routers.



DISTANCE-VECTOR ROUTING ALGORITHM

Table 20.1 Distance-Vector Routing Algorithm for a Node

```
1 Distance_Vector_Routing ( )
2 {
3     // Initialize (create initial vectors for the node)
4     D[myself] = 0
```

Table 20.1 Distance-Vector Routing Algorithm for a Node (continued)

```
5     for (y = 1 to N)
6     {
7         if (y is a neighbor)
8             D[y] = c[myself][y]
9         else
10            D[y] =  $\infty$ 
11    }
12    send vector {D[1], D[2], ..., D[N]} to all neighbors
13    // Update (improve the vector with the vector received from a neighbor)
14    repeat (forever)
15    {
16        wait (for a vector  $D_w$  from a neighbor  $w$  or any change in the link)
17        for (y = 1 to N)
18        {
19            D[y] = min [D[y], (c[myself][w] +  $D_w$ [y])]    // Bellman-Ford equation
20        }
21        if (any change in the vector)
22            send vector {D[1], D[2], ..., D[N]} to all neighbors
23    }
24 } // End of Distance Vector
```

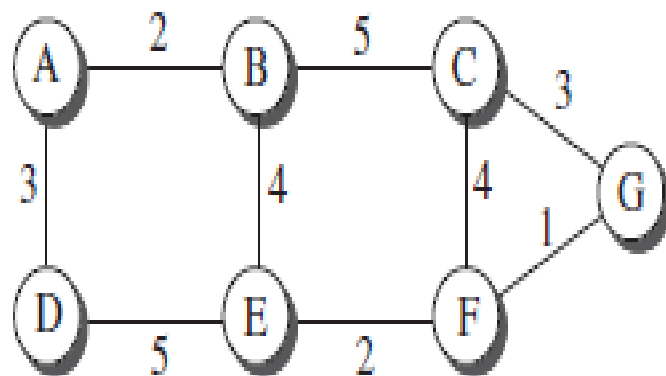


LINK-STATE ROUTING

- A routing algorithm that directly follows our discussion for creating least-cost trees and forwarding tables is **link-state (LS) routing**.
- This method uses the term *link-state* to define the **characteristic of a link** (an edge) that represents a network in the internet.
- In this algorithm the cost associated with an edge defines the state of the link.
- Links with lower costs are preferred to links with higher costs;
- if the cost of a link is infinity, it means that the link does not exist or has been broken.



e 20.8 *Example of a link-state database*



a. The weighted graph

	A	B	C	D	E	F	G
A	0	2	∞	3	∞	∞	∞
B	2	0	5	∞	4	∞	∞
C	∞	5	0	∞	∞	4	3
D	3	∞	∞	0	5	∞	∞
E	∞	4	∞	5	0	2	∞
F	∞	∞	4	∞	2	0	1
G	∞	∞	3	∞	∞	1	0

b. Link state database

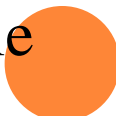


LINK-STATE DATABASE (LSDB)

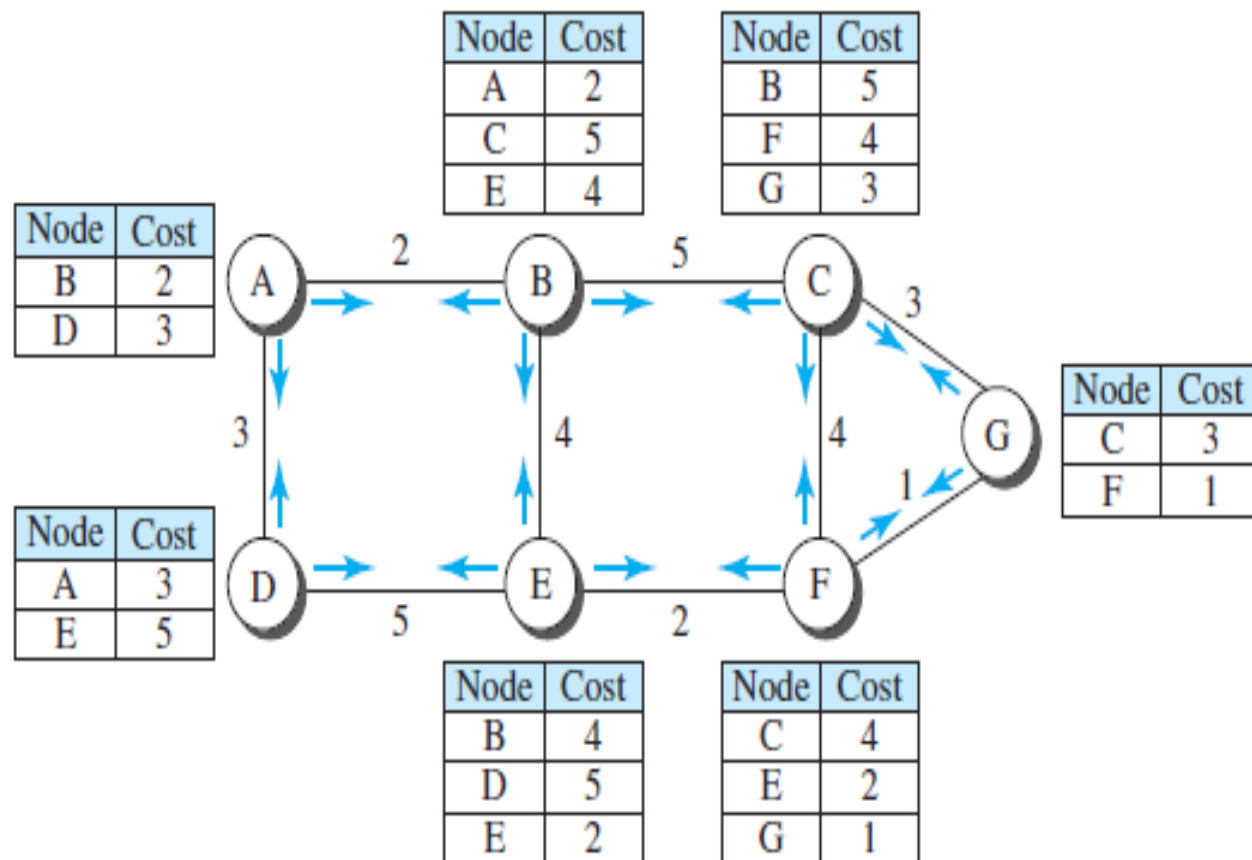
- The collection of states for all links is called the link-state database (LSDB).
- There is only one LSDB for the whole internet.
- Each node needs to have a duplicate of it to be able to create the least-cost tree.
- The LSDB can be represented as a two-dimensional array (matrix) in which the value of each cell defines the cost of the corresponding link.
- **Flooding** is the process in which each node can send some greeting messages to all its immediate neighbors (those nodes to which it is connected directly) to collect two pieces of information for each neighboring node.
 - The identity of the node
 - The cost of the link.




CONT..

- The combination of these two pieces of information is called the **LS packet** (LSP). The LSP is sent out of each interface.
 - When a node receives an LSP from one of its interfaces, it compares the LSP with the copy it may already have. If the newly arrived LSP is older than the one it has it discards the LSP.
 - If it is newer or the first one received, the node discards the old LSP (if there is one) and keeps the received one.
 - We can compare the link-state routing algorithm with the distance-vector routing algorithm.
 - In the distance-vector routing algorithm, each router tells its neighbors what it knows about the whole internet;
 - in the link-state routing algorithm, each router tells the whole internet what it knows about its neighbors.
- 

LSPs created and sent out by each node to build LSDB

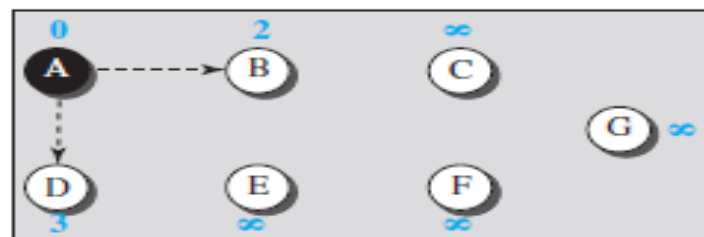


FORMATION OF LEAST-COST TREES

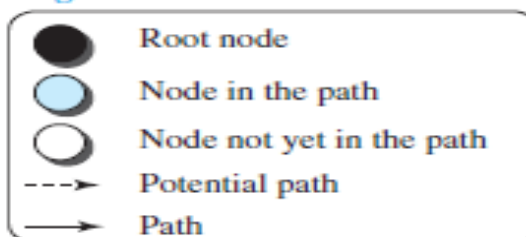
- To create a least-cost tree for itself, using the shared LSDB, each node needs to run the famous **Dijkstra Algorithm**. This iterative algorithm uses the following steps:
 - 1. The node chooses itself as the root of the tree, creating a tree with a single node, and sets the total cost of each node based on the information in the LSDB.
 - 2. The node selects one node, among all nodes not in the tree, which is closest to the root, and adds this to the tree. After this node is added to the tree, the cost of all other nodes not in the tree needs to be updated because the paths may have been changed.
 - 3. The node repeats step 2 until all nodes are added to the tree. We need to convince ourselves that the above three steps finally create the least-cost.
- 

Least-cost tree

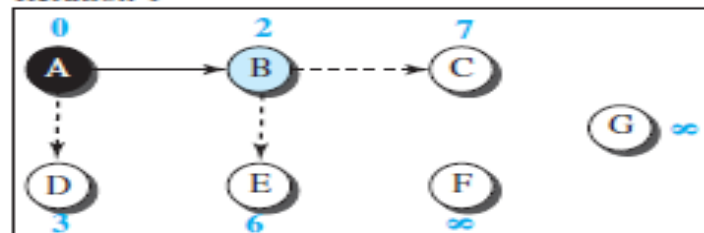
Initialization



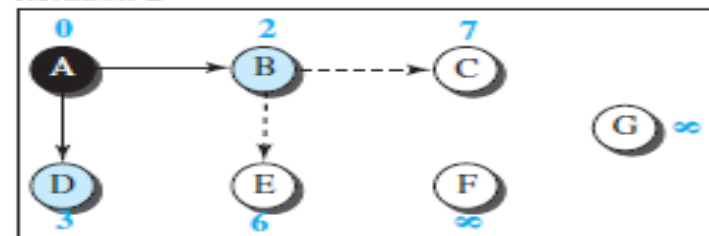
Legend



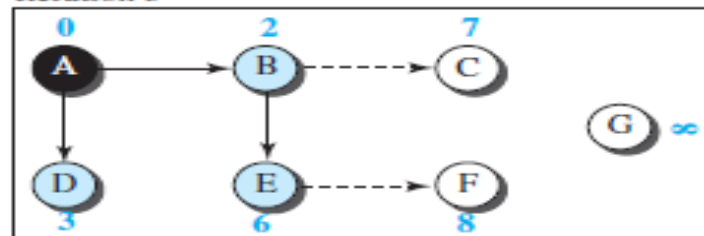
Iteration 1



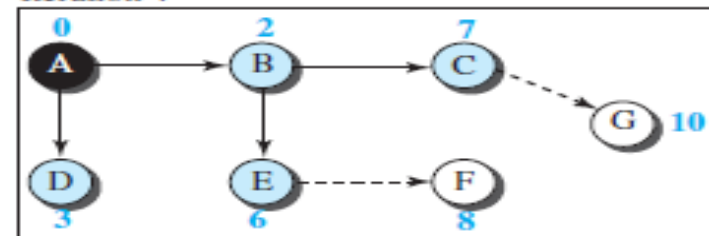
Iteration 2



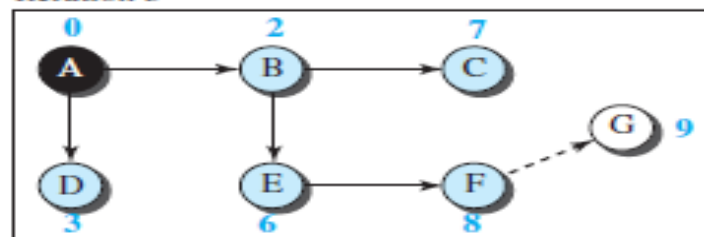
Iteration 3



Iteration 4



Iteration 5



Iteration 6

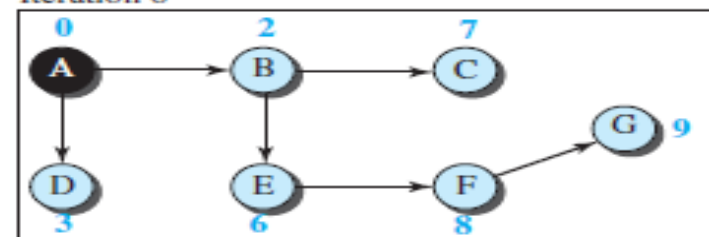


Table 20.2 *Dijkstra's Algorithm*

```

1  Dijkstra's Algorithm ( )
2  {
3      // Initialization
4      Tree = {root}           // Tree is made only of the root
5      for (y = 1 to N)       // N is the number of nodes
6      {
7          if (y is the root)
8              D[y] = 0        // D[y] is shortest distance from root to node y
9          else if (y is a neighbor)
10             D[y] = c[root][y] // c[x][y] is cost between nodes x and y in LSDB
11          else
12             D[y] =  $\infty$ 
13      }
14      // Calculation
15      repeat
16      {
17          find a node w, with D[w] minimum among all nodes not in the Tree
18          Tree = Tree  $\cup$  {w}    // Add w to tree
19          // Update distances for all neighbors of w
20          for (every node x, which is a neighbor of w and not in the Tree)
21          {
22              D[x] = min {D[x], (D[w] + c[w][x])}
23          }
24      } until (all nodes included in the Tree)
25  } // End of Dijkstra

```



PATH-VECTOR ROUTING

- Both link-state and distance-vector routing are based on the least-cost goal.
- The best route is determined by the source using the policy it imposes on the route.
- In other words, the source can control the path.
- Although path-vector routing is not actually used in an internet, and is mostly designed to **route a packet between ISPs**.
- we discuss the principle of this method in this section as though applied to an internet.



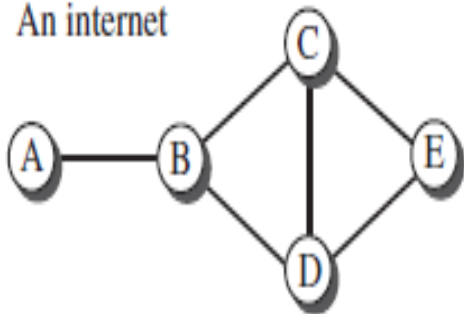
SPANNING TREES:

- In path-vector routing, the path from a source to all destinations is also determined by the best spanning tree.
- The best spanning tree, however, is not the least-cost tree; it is the tree determined by the source when it **imposes its own policy**.
- If there is more than one route to a destination, the source can choose the route that meets its **policy best**.
- A source may apply several policies at the same time.

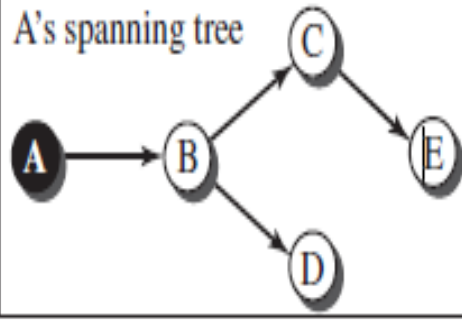


Spanning trees in path-vector routing

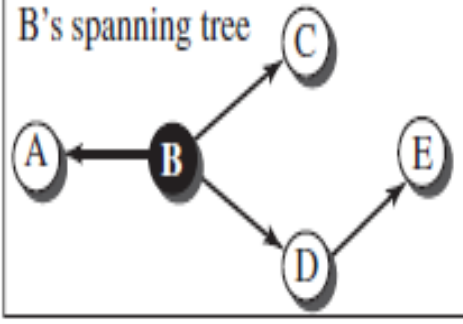
An internet



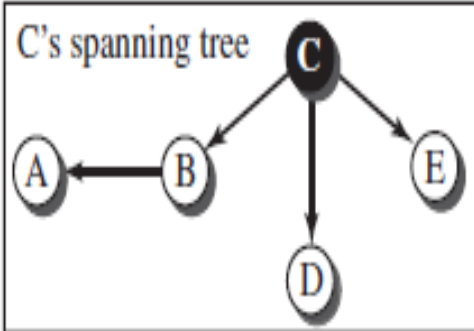
A's spanning tree



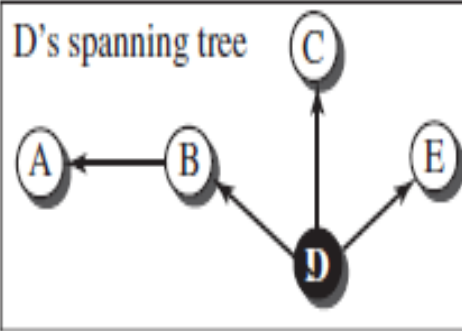
B's spanning tree



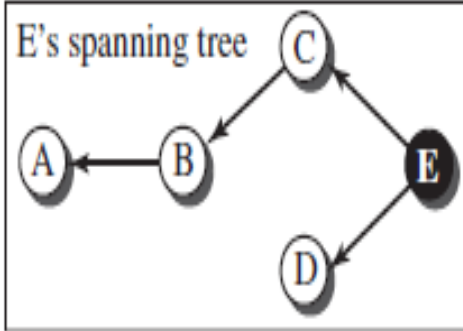
C's spanning tree



D's spanning tree



E's spanning tree



CREATION OF SPANNING TREES:

- Path-vector routing is an asynchronous and distributed routing algorithm.
- The spanning trees are made, gradually and asynchronously, by each node.
- When a node is booted, it creates a path vector based on the information it can obtain about its immediate neighbor.
- A node sends greeting messages to its immediate neighbors to collect these pieces of information.
- Each node, after the creation of the initial path vector, sends it to all its immediate neighbors.



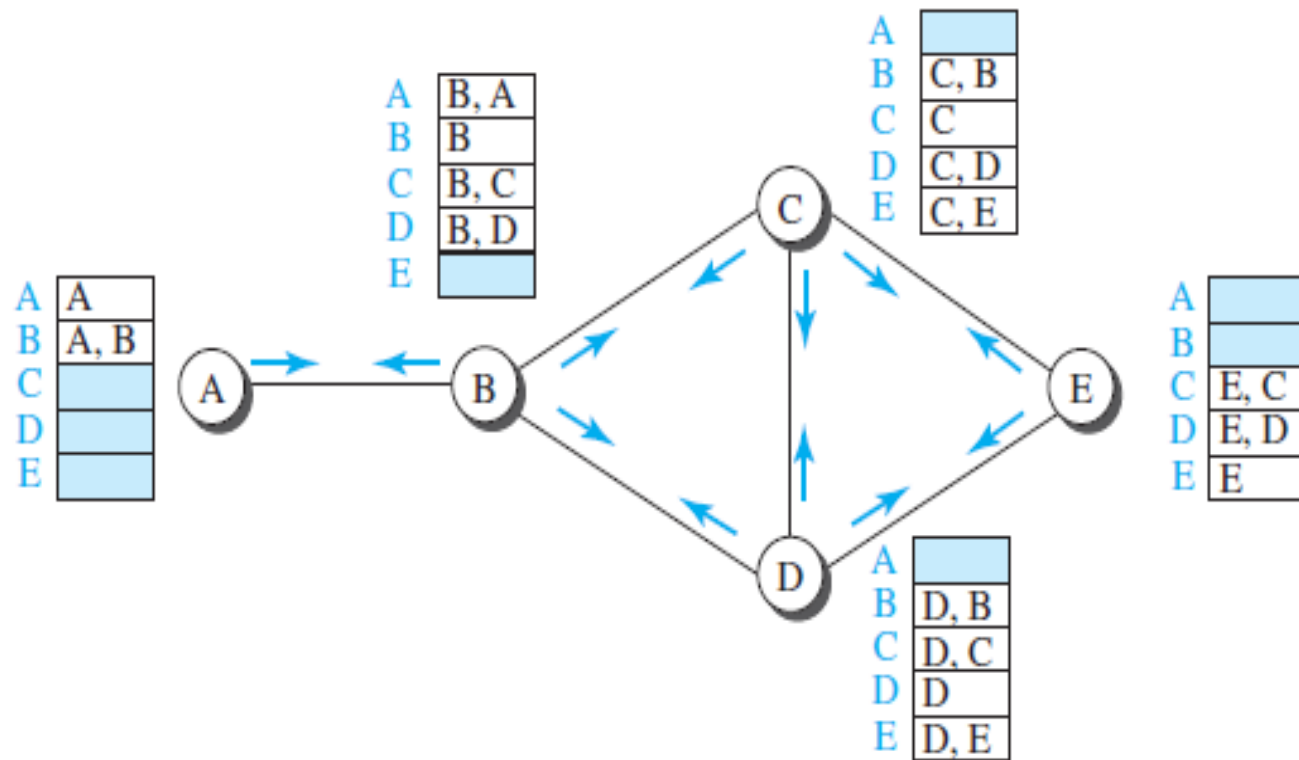
CONT..

- Each node, when it receives a path vector from a neighbor, updates its path vector using an equation similar to the Bellman-Ford, but applying its own policy instead of looking for the least cost.
- We can define this equation as

$$\text{Path}(x, y) = \text{best} \{ \text{Path}(x, y), [(x + \text{Path}(v, y))] \} \quad \text{for all } v\text{'s in the internet.}$$



Path vectors made at booting time



Updating path vectors

Note:

$X[]$: vector X

Y: node Y

	New C		Old C		B
A	C, B, A	A		A	B, A
B	C, B	B	C, B	B	B
C	C	C	C	C	B, C
D	C, D	D	C, D	D	B, D
E	C, E	E	C, E	E	

$C[] = \text{best}(C[], C + B[])$

Event 1: C receives a copy of B's vector

	New C		Old C		D
A	C, B, A	A	C, B, A	A	
B	C, B	B	C, B	B	D, B
C	C	C	C	C	D, C
D	C, D	D	C, D	D	D
E	C, E	E	C, E	E	D, E

$C[] = \text{best}(C[], C + D[])$

Event 2: C receives a copy of D's vector



CONT..

- In the first event, node C receives a copy of B's vector, which improves its vector: now it knows how to reach node A.
- In the second event, node C receives a copy of D's vector, which does not change its vector. As a matter of fact the vector for node C after the first event is stabilized and serves as its forwarding table.

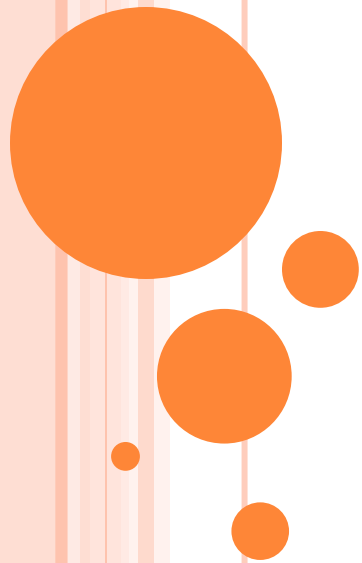


PATH-VECTOR ALGORITHM:

```
1 Path_Vector_Routing ()
2 {
3     // Initialization
4     for (y = 1 to N)
5     {
6         if (y is myself)
7             Path[y] = myself
8         else if (y is a neighbor)
9             Path[y] = myself + neighbor node
10        else
11            Path[y] = empty
12    }
13    Send vector {Path[1], Path[2], ..., Path[y]} to all neighbors
14    // Update
15    repeat (forever)
16    {
17        wait (for a vector Pathw from a neighbor w)
18        for (y = 1 to N)
19        {
20            if (Pathw includes myself)
21                discard the path // Avoid any loop
22            else
23                Path[y] = best {Path[y], (myself + Pathw[y])}
24        }
25        If (there is a change in the vector)
26            Send vector {Path[1], Path[2], ..., Path[y]} to all neighbors
27    }
28 } // End of Path Vector
```



UNICAST PROTOCOLS



UNICAST ROUTING PROTOCOLS

- A protocol is more than an algorithm.
- A protocol needs to define its domain of operation, the messages exchanged, communication between routers, and interaction with protocols in other domains.
- Three common protocols used in the Internet as follows.
- Routing Information Protocol (RIP), based on the Distance-Vector Algorithm
- Open Shortest Path First (OSPF), based on the Link-State Algorithm
- Border Gateway Protocol (BGP), based on the Path-Vector Algorithm



HIERARCHICAL ROUTING:

- Hierarchical routing means considering each ISP as an autonomous system (AS).
- Each AS can run a routing protocol that meets its needs, but the global Internet runs a global protocol to glue all ASs together.
- The routing protocol run in each AS is referred to as intra-AS routing protocol, intra domain routing protocol, or **interior gateway protocol (IGP)**.
- The global routing protocol is referred to as inter-AS routing protocol, inter domain routing protocol, or **exterior gateway protocol (EGP)**.
- The two common intra domain routing protocols are **RIP and OSPF**.
- The only inter domain routing protocol is **BGP**.



AUTONOMOUS SYSTEMS:

- Each ISP is an autonomous system when it comes to managing networks and routers under its control.
- Each AS is given an autonomous number (ASN) by the ICANN.
- Each ASN is a 16-bit unsigned integer that uniquely defines an AS.
- The autonomous systems, however, are not categorized according to their size; they are categorized according to the way they are connected to other ASs.



CONT..

- **Stub AS**: A stub AS has only one connection to another AS.
- **Multi homed AS**: A multi homed AS can have more than one connection to other ASs, but it does not allow
- **Transient AS**. A transient AS is connected to more than one other AS and also allows the traffic to pass through. data traffic to pass through it.



ROUTING INFORMATION PROTOCOL (RIP)

- The Routing Information Protocol (RIP) is one of the most widely used intra domain routing protocols based on the distance-vector routing algorithm.
- **Hop Count**
- First a router in an AS needs to know how to forward a packet to different networks (subnets) in an AS, RIP routers advertise the cost of reaching different networks instead of reaching other nodes in a theoretical graph.
- Second, to make the implementation of the cost simpler ,the cost is defined as the number of hops, which means the number of networks (subnets) a packet needs to travel through from the source router to the final destination host.

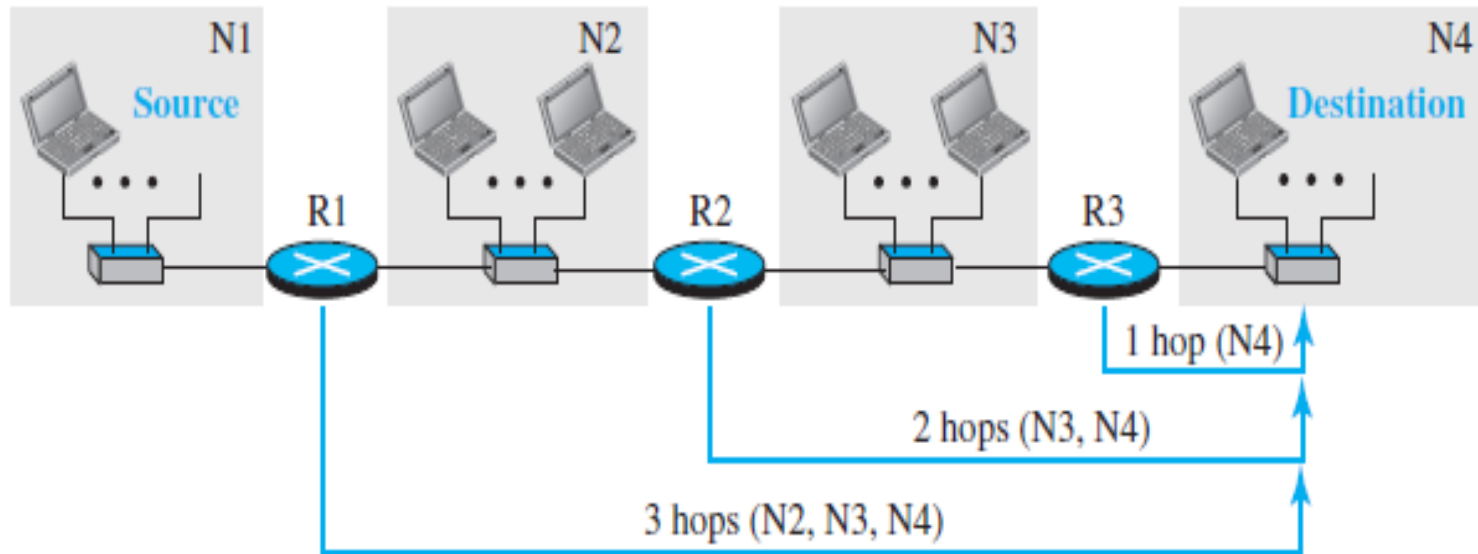


CONT..

- Note that the network in which the source host is connected is not counted in this calculation because the source host does not use a forwarding table.
- In RIP, the maximum cost of a path can be 15, which means 16 is considered as infinity (no connection).
- For this reason, RIP can be used only in autonomous systems in which the diameter of the AS is not more than 15 hops.



Hop counts in RIP



FORWARDING TABLES

- A forwarding table in RIP is a **three-column table**.
- The first column is the address of the **destination network**.
- The second column is the address of the **next router** to which the packet should be forwarded.
- The third column is the **cost** (the number of hops) to reach the destination network.
- The first and the third columns together convey the same information as does a distance vector, but the cost shows the number of hops to the destination networks.



Forwarding tables for the above network

Forwarding table for R1

Destination network	Next router	Cost in hops
N1	—	1
N2	—	1
N3	R2	2
N4	R2	3

Forwarding table for R2

Destination network	Next router	Cost in hops
N1	R1	2
N2	—	1
N3	—	1
N4	R3	2

Forwarding table for R3

Destination network	Next router	Cost in hops
N1	R2	3
N2	R2	2
N3	—	1
N4	—	1



RIP IMPLEMENTATION:

- RIP is implemented as a process that uses the service of UDP on the well-known **port number 520**.
- RIP is a **daemon process** (a process running in the background), named routed (abbreviation for route daemon and pronounced route-dee).
- This means that, although RIP is a routing protocol to help IP route its datagrams through the AS, the RIP messages are encapsulated inside UDP user datagrams, which in turn are encapsulated inside IP datagrams.
- In other words, RIP runs at the application layer, but creates forwarding tables for IP at the network layer.
- RIP has gone through two versions: RIP-1 and RIP-2.
- The second version is backward compatible with the first section;

RIP MESSAGES:

- **Two RIP processes**, a client and a server, like any other processes, need to exchange messages.
- Part of the message, which we call entry, can be repeated as needed in a message.
- Each entry carries the information related to one line in the forwarding table of the router that sends the message.
- RIP has two types of messages: **request and response**.
- A request message is sent by a router that has just come up or by a router that has some time-out entries.
- A request message can ask about specific entries or all entries.

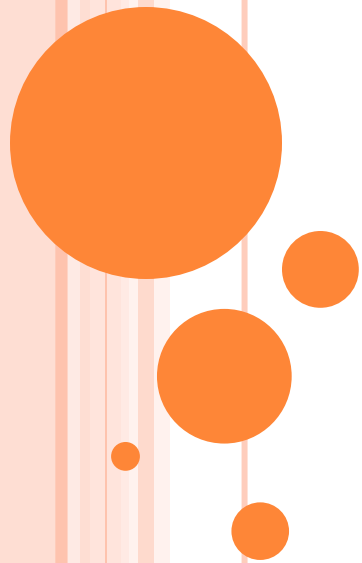


CONT..

- A response (or update) message can be either **solicited or unsolicited**.
- A solicited response message is sent only in answer to a request message.
- It contains information about the destination specified in the corresponding request message.
- An unsolicited response message, on the other hand, is sent periodically, every 30 seconds or when there is a change in the forwarding table.



MULTICAST ROUTING PROTOCOLS

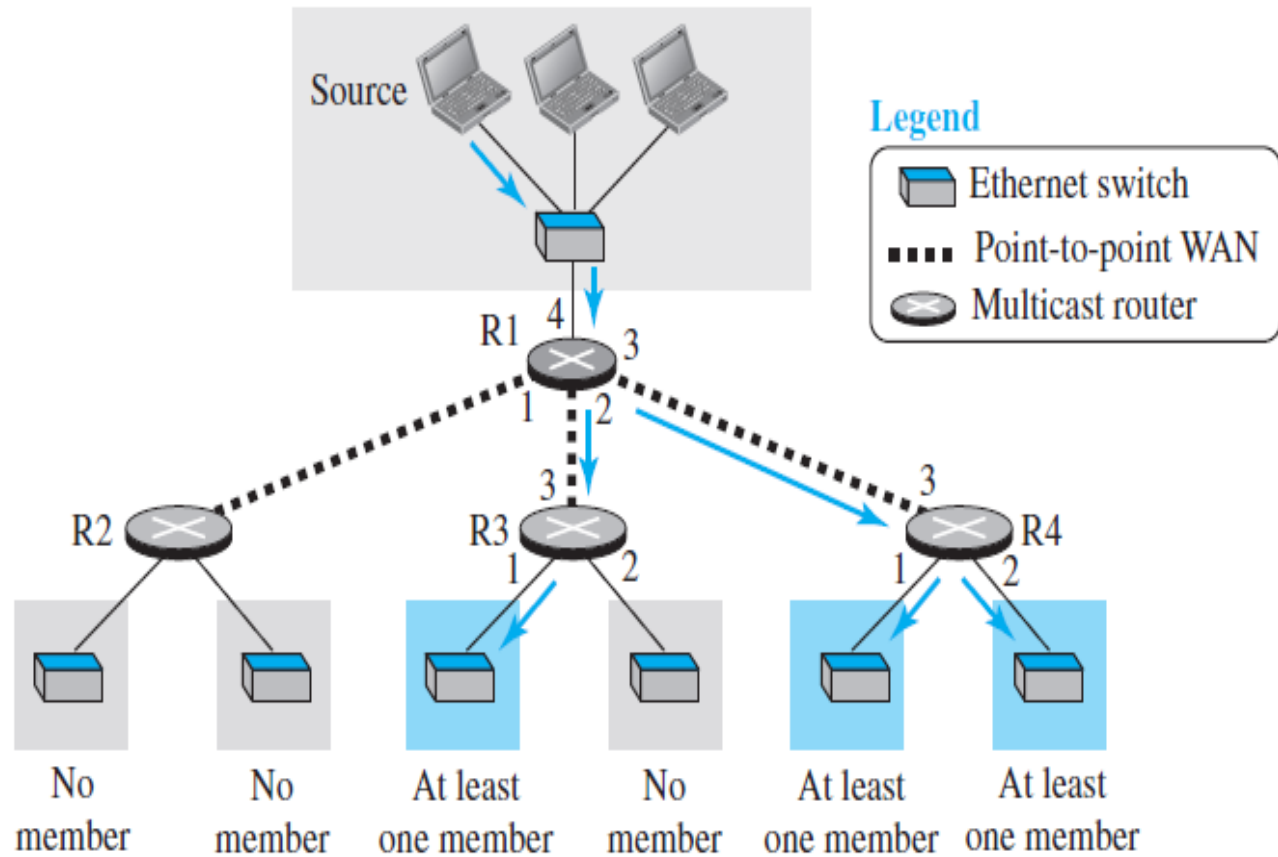


MULTICAST

- In multicasting, there is one source and a group of destinations.
- The relationship is one to many.
- In this type of communication, the source address is a unicast address, but the destination address is a group address, a group of one or more destination networks in which there is at least one member of the group that is interested in receiving the multicast datagram.
- The group address defines the members of the group



Multicasting



MULTICAST APPLICATIONS

- Multicasting has many applications today, such as access to distributed databases, information dissemination, teleconferencing, and distance learning.
- *Access to Distributed Databases*
- *Information Dissemination*
- *Teleconferencing*
- *Distance Learning*



BROADCASTING

- Broadcasting means one-to-all communication: a host sends a packet to all hosts in an internet.
- Broadcasting in this sense is not provided at the Internet level for the obvious reason that it may create a huge volume of traffic and use a huge amount of bandwidth.
- Partial broadcasting, however, is done in the Internet.



MULTICASTING BASICS

- we need to discuss some multicasting basics:
- multicast addressing,
- collecting information about multicast groups,
- multicast optimal trees.



MULTICAST ADDRESSES

- A multicast address defines a group of recipients, not a single one.
- In other words, a multicast address is an identifier for a group.
- If a new group is formed with some active members, an authority can assign an unused multicast address to this group to uniquely define it.
- This means that the **source address** of a packet in multicast communication can be a **unicast address** that uniquely defines the sender, but the **destination address** can be the **multicast address** that defines a group.

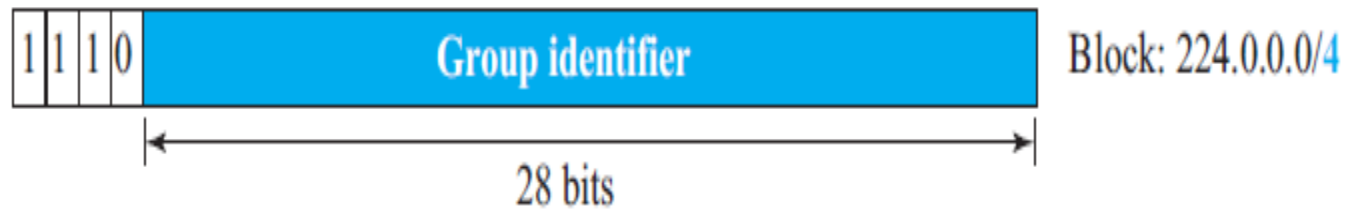


MULTICAST ADDRESSES IN IPv4

- A router or a destination host needs to distinguish between a unicast and a multicast datagram.
- IPv4 and IPv6 each assign a block of addresses for this purpose.
- In this section, we discuss only IPv4 multicast addresses.
- Multicast addresses in IPv4 belong to a large block of addresses that are specially designed for this purpose.
- In **classful addressing**, all of class D was composed of these addresses; **classless addressing** used the same block, but it was referred to as the block 224.0.0.0/4 (from 224.0.0.0 to 239.255.255.255).



A multicast address in binary



CONT..

- The number of addresses in the multicast block is huge (228).
- We definitely cannot have that many individual groups.
- However, the block is divided into several subblocks, and each subblock is used in a particular multicast application.
- The following gives some of the common subblocks:



- ***Local Network Control Block.***
- The subblock 224.0.0.0/24 is assigned to a multicast routing protocol to be used inside a network,
- ***Internetwork Control Block.***
- The subblock 224.0.1.0/24 is assigned to a multicast routing protocol to be used in the whole Internet.
- ***Source-Specific Multicast (SSM) Block.***
- The block 232.0.0.0/8 is used for sourcespecific multicast routing.
- ***GLOP Block.*** The block 233.0.0.0/8 is called the GLOP block.
- ***Administratively Scoped Block.***
- The block 239.0.0.0/8 is called the Administratively Scoped Block.



SELECTING MULTICAST ADDRESS

- To select a multicast address to be assigned to a group is not an easy task. The selection of address depends on the type of application.
- ***Limited Group***
- The administrator can use the AS number (x.y)256 and choose an address between 239.x.y.0 and 239.x.y.255 that is not used by any other group, as the multicast address for that particular group.
- ***Larger Group***
- If the group is spread beyond an AS territory, the previous solution does not work.
- The group needs to choose an address from the SSM block (232.0.0.8).



DELIVERY AT DATA-LINK LAYER

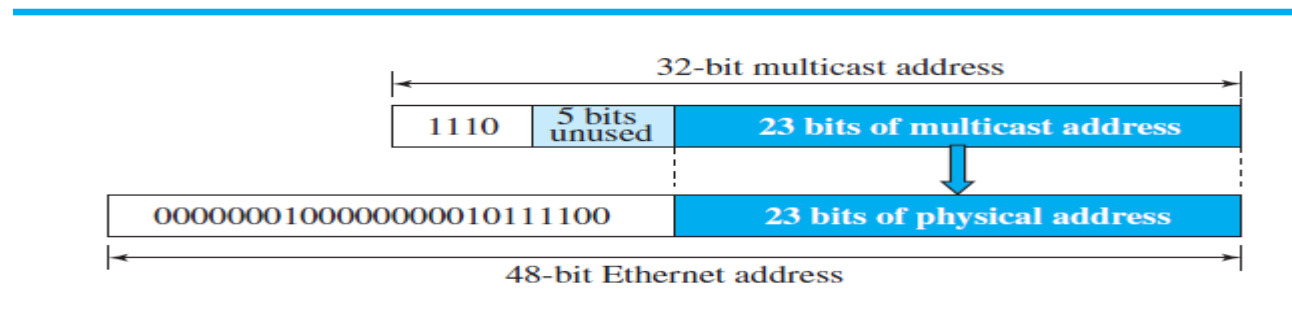
- In multicasting, the delivery at the Internet level is done using network-layer multicast addresses (IP addresses allocated for multicasting).
- However, data-link layer multicast addresses are also needed to deliver a multicast packet encapsulated in a frame.



NETWORK WITH MULTICAST SUPPORT

- Most LANs support physical multicast addressing.
- Ethernet is one of them.
- An Ethernet physical address (MAC address) is six octets (48 bits) long.
- If the first 25 bits in an Ethernet address are 000000001 00000000 01011110 0, this identifies a physical multicast address for the TCP/IP protocol.
- The remaining 23 bits can be used to define a group.

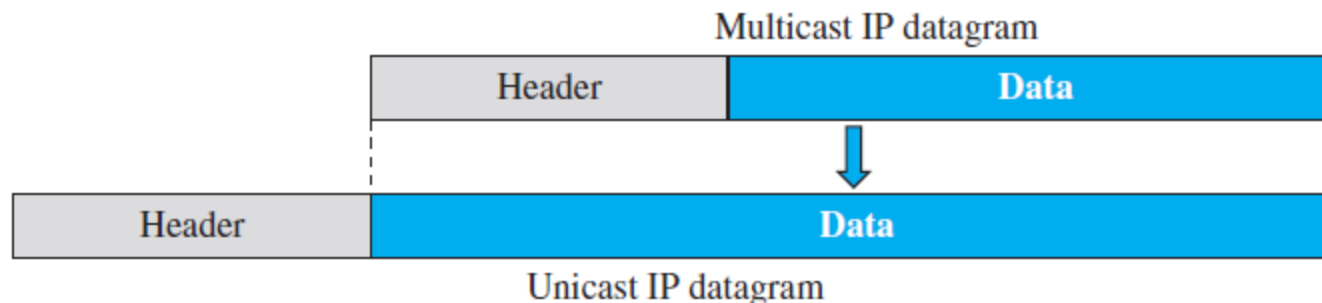
Mapping class D to Ethernet physical address



NETWORK WITH NO MULTICAST SUPPORT

- Most WANs do not support physical multicast addressing.
- To send a multicast packet through these networks, a process called *tunneling* is used.
- In **tunneling**, the multicast packet is encapsulated in a unicast packet and sent through the network, where it emerges from the other side as a multicast packet

Tunneling



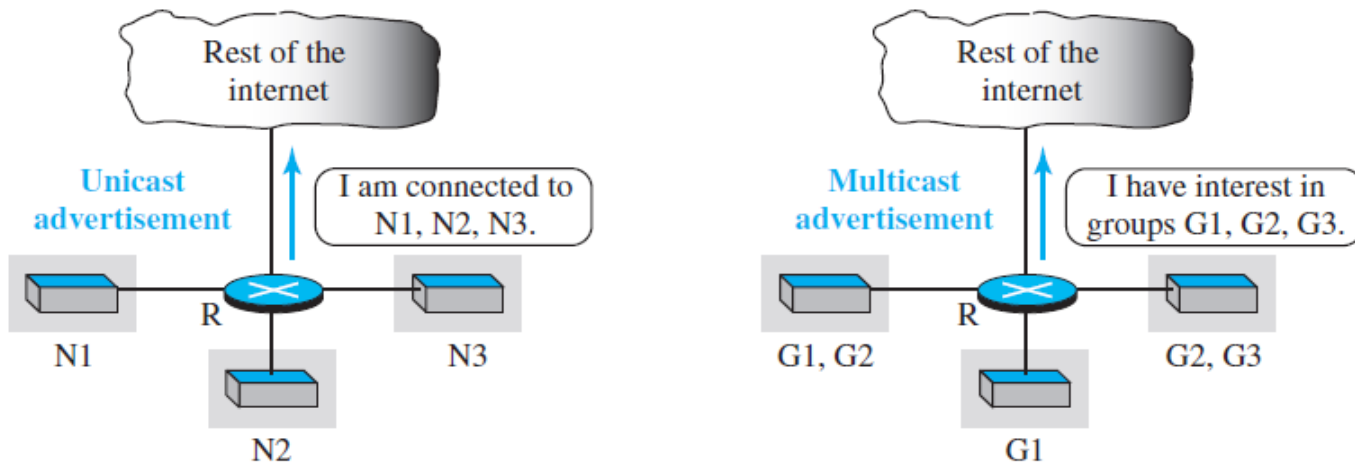
COLLECTING INFORMATION ABOUT GROUPS

- Creation of forwarding tables in both unicast and multicast routing involves two steps:
- 1. *A router needs to know to which destinations it is connected.*
- 2. *Each router needs to propagate information obtained in the first step to all other routers so that each router knows to which destination each other router is connected.*
- In unicast routing, the collection of the information in the first step is automatic;
- Each router knows to which network it is connected, and the prefix of the network (in CIDR) is what a router needs.



- In multicast routing, the collection of information in the first step is not automatic for two reasons.
- First, a router does not know which host in the attached network is a member of a particular group; membership in the group does not have any relation to the prefix associated with the network.

Unicast versus multicast advertisement



CONT..

- In the case of unicasting, router R knows hosts with prefixes N1, N2, and N3 are connected to its interfaces; it propagates this information to the rest of the internet.
- In the case of multicasting, router R needs to know that there are hosts with at least one loyal member in groups G1, G2, and G3 in networks connected to its interfaces.
in multicasting we need two protocols:
- one to collect these pieces of information and the second to propagate them.
- Collecting pieces of information is done by the Internet Group Management Protocol (IGMP),



MULTICAST FORWARDING

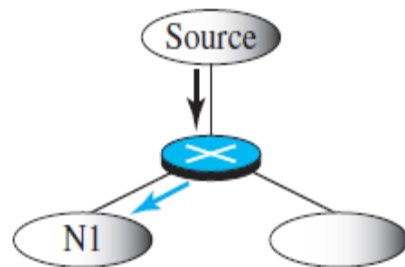
- Forwarding in unicast and multicast communication is different in two aspects:
- i) In unicast communication, the destination address of the packet defines one single destination.
- The packet needs to be sent only out of one of the interfaces, the interface which is the branch in the shortest-path tree reaching the destination with the minimum cost.
- In multicast communication, the destination of the packet defines
- one group, but that group may have more than one member in the internet.



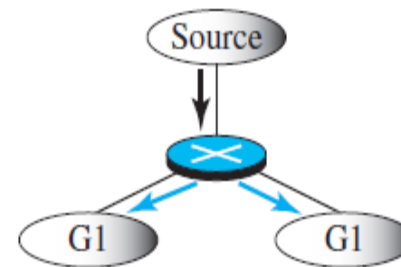
CONT..

- ii). Forwarding decisions in unicast communication depend only on the destination address of the packet.
- Forwarding decisions in multicast communication depend on both the destination and the source address of the packet.

Destination in unicasting and multicasting



a. Destination in unicasting is one



b. Destination in multicasting is more than one



TWO APPROACHES TO MULTICASTING

- In multicast routing, as in unicast routing, we need to create routing trees to optimally route the packets from their source to their destination.
- Two different approaches in multicast routing have been developed:
 - Routing using source-based trees
 - Routing using group-shared trees



ROUTING USING SOURCE-BASED TREES

- In the **source-based tree** approach to multicasting, each router needs to create a separate tree for each source-group combination.
- In other words, if there are m groups and n sources in the internet, a router needs to create $(m \times n)$ routing trees.
- In each tree, the corresponding source is the root, the members of the group are the leaves, and the router itself is somewhere on the tree.
- Although it may appear that each router needs to create and store a huge amount of information about all of these trees.



GROUP-SHARED TREE APPROACH

- In the **group-shared tree** approach, we designate a router to act as the phony source for each group.
- The designated router, which is called the *core* router or the *rendezvouspoint* router, acts as the representative for the group.
- Any source that has a packet to send to a member of that group sends it to the core center (unicast communication) and the core center is responsible for multicasting.
- The core center creates one single routing tree with itself as the root and any routers with active members in the group as the leaves.



- In this approach, there are m core routers (one for each group) and each core router has a routing tree, for the total of m trees.
- This means that the number of routing trees is reduced from $(m \times n)$ in the source-based tree approach to m in this approach.



INTRADOMAIN MULTICAST PROTOCOLS

INTRODUCTION

- During the last few decades, several intradomain multicast routing protocols have emerged.
- In this section, we discuss three of these protocols.
- Two are extensions of unicast routing protocols (RIP and OSPF), using the source-based tree approach;
- The third is an independent protocol which is becoming more and more popular.
- It can be used in two modes, employing either the source-based tree approach or the shared-group tree approach.

Multicast Distance Vector (DVMRP)

- The **Distance Vector Multicast Routing Protocol (DVMRP)** is the extension of the Routing Information Protocol (RIP) which is used in unicast routing.
- It uses the source based tree approach to multicasting.
- It is worth mentioning that each router in this protocol that receives a multicast packet to be forwarded implicitly creates a source-based multicast tree in three steps:

Cont..

- **1.** The router uses an algorithm called *reverse path forwarding* (RPF) to simulate creating part of the optimal source-based tree between the source and itself.
- **2.** The router uses an algorithm called *reverse path broadcasting* (RPB) to create a broadcast (spanning) tree whose root is the router itself and whose leaves are all networks in the internet.
- **3.** The router uses an algorithm called *reverse path multicasting* (RPM) to create a multicast tree by cutting some branches of the tree that end in networks with no member in the group

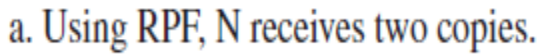
Reverse Path Forwarding (RPF)

- The first algorithm, **reverse path forwarding (RPF)**, forces the router to forward a multicast packet from one specific interface:
- The one which has come through the shortest path from the source to the router.

Reverse Path Broadcasting (RPB)

- The RPF algorithm helps a router to forward only one copy received from a source and drop the rest.
- However, when we think about broadcasting in the second step, we need to remember that destinations are all the networks (LANs) in the internet.
- To be efficient, we need to prevent each network from receiving more than one copy of the packet.

RPF versus RPB

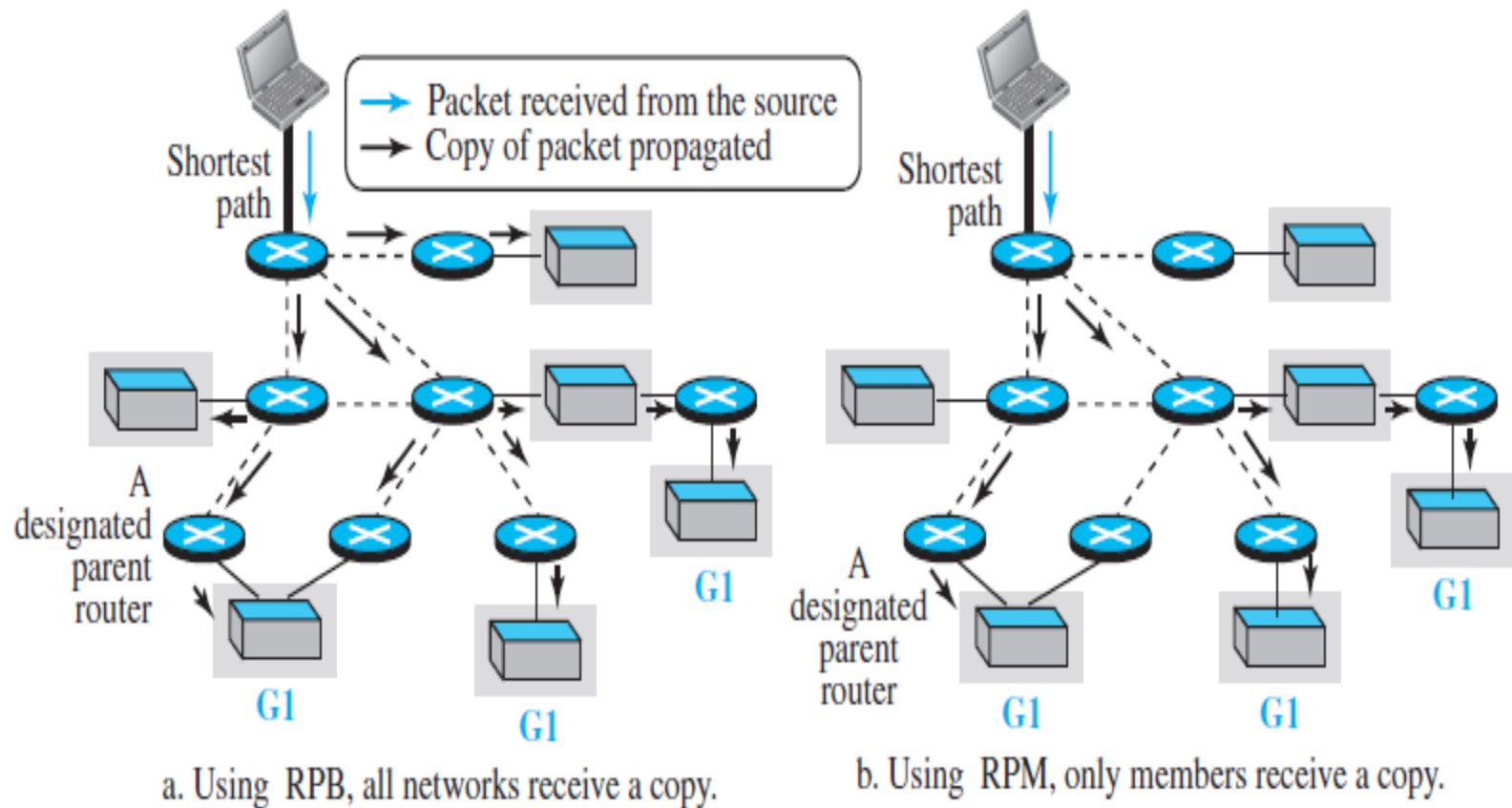


b. Using RPB, N receives only one copy.

Reverse Path Multicasting (RPM)

- As you may have noticed, RPB does not multicast the packet, it broadcasts it.
- This is not efficient.
- To increase efficiency, the multicast packet must reach only those networks that have active members for that particular group.
- This is called ***reverse path multicasting (RPM)***.
- To change the broadcast shortest-path tree to a multicast shortest-path tree, each router needs to prune (make inactive) the interfaces that do not reach a network with active members corresponding to a particular source-group combination.

Figure 21.12 *RPB versus RPM*



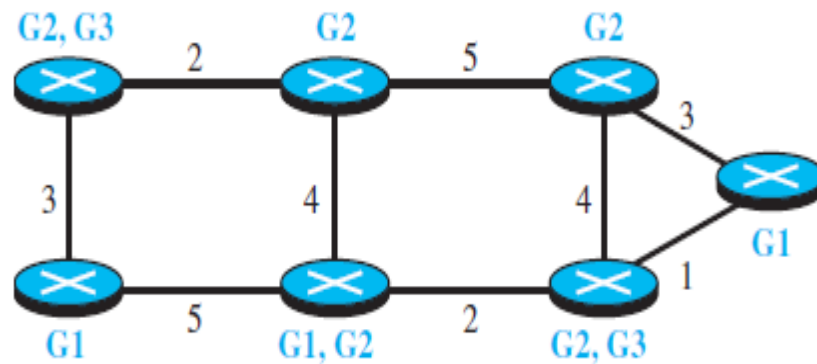
Multicast Link State (MOSPF)

- **Multicast Open Shortest Path First (MOSPF)** is the extension of the Open Shortest Path First (OSPF) protocol, which is used in unicast routing.
- It also uses the source based tree approach to multicasting.
- If the internet is running a unicast link-state routing algorithm, the idea can be extended to provide a multicast link-state routing algorithm.
- Recall that in unicast link-state routing, each router in the internet has a link-state database (LSDB) that can be used to create a shortest-path tree.

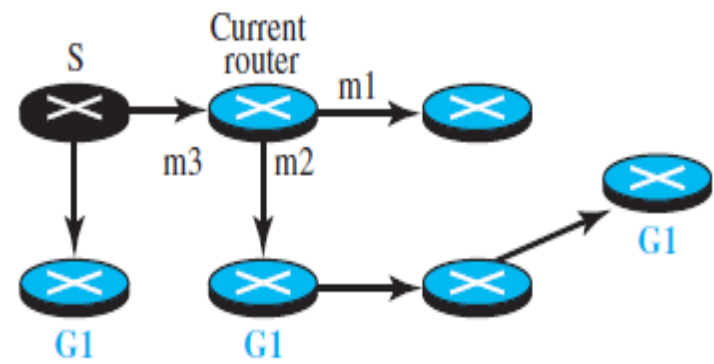
STEPS

- The router uses the Dijkstra algorithm to create a shortest-path tree with S as the root and all destinations in the internet as the leaves.
- The router finds itself in the shortest-path tree created in the first step.
- The shortest-path subtree is actually a broadcast subtree with the router as the root and all networks as the leaves.
- The router can now forward the received packet out of only those interfaces that correspond to the branches of the multicast tree.

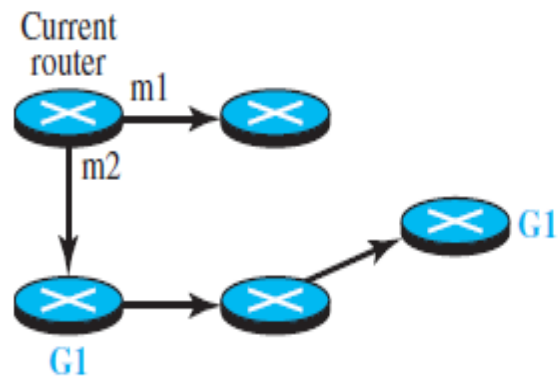
Figure 21.13 *Example of tree formation in MOSPF*



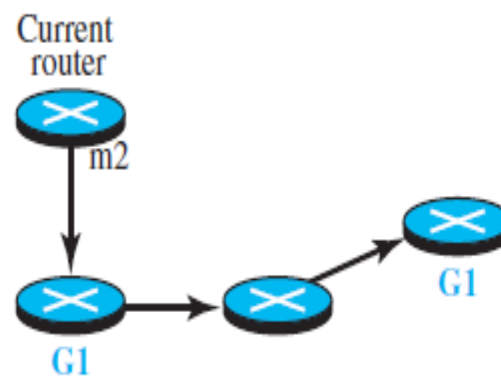
a. An internet with some active groups



b. S-G1 shortest-path tree



c. S-G1 subtree seen by current router



d. S-G1 pruned subtree

Forwarding table
for current router

Group-Source	Interface
S, G1	m2
...	...

Protocol Independent Multicast (PIM)

- **Protocol Independent Multicast (PIM)** is the name given to a common protocol that needs a unicast routing protocol for its operation, but the unicast protocol can be either a distance-vector protocol or a link-state protocol.
- In other words, PIM needs to use the forwarding table of a unicast routing protocol to find the next router in a path to the destination, but it does not matter how the forwarding table is created.
- PIM has another interesting feature: it can work in two different modes: **dense and sparse**.

Cont..

- *DENSE*
- The term *dense* here means that the number of active members of a group in the internet is large;
- the probability that a router has a member in a group is high.
- *SPARSE*
- The term *sparse*, on the other hand, means that only a few routers in the internet have active members in the group;
- the probability that a router has a member of the group is low.

- When the protocol is working in the dense mode, it is referred to as **PIM-DM**;
- when it is working in the sparse mode, it is referred to as **PIM-SM**.

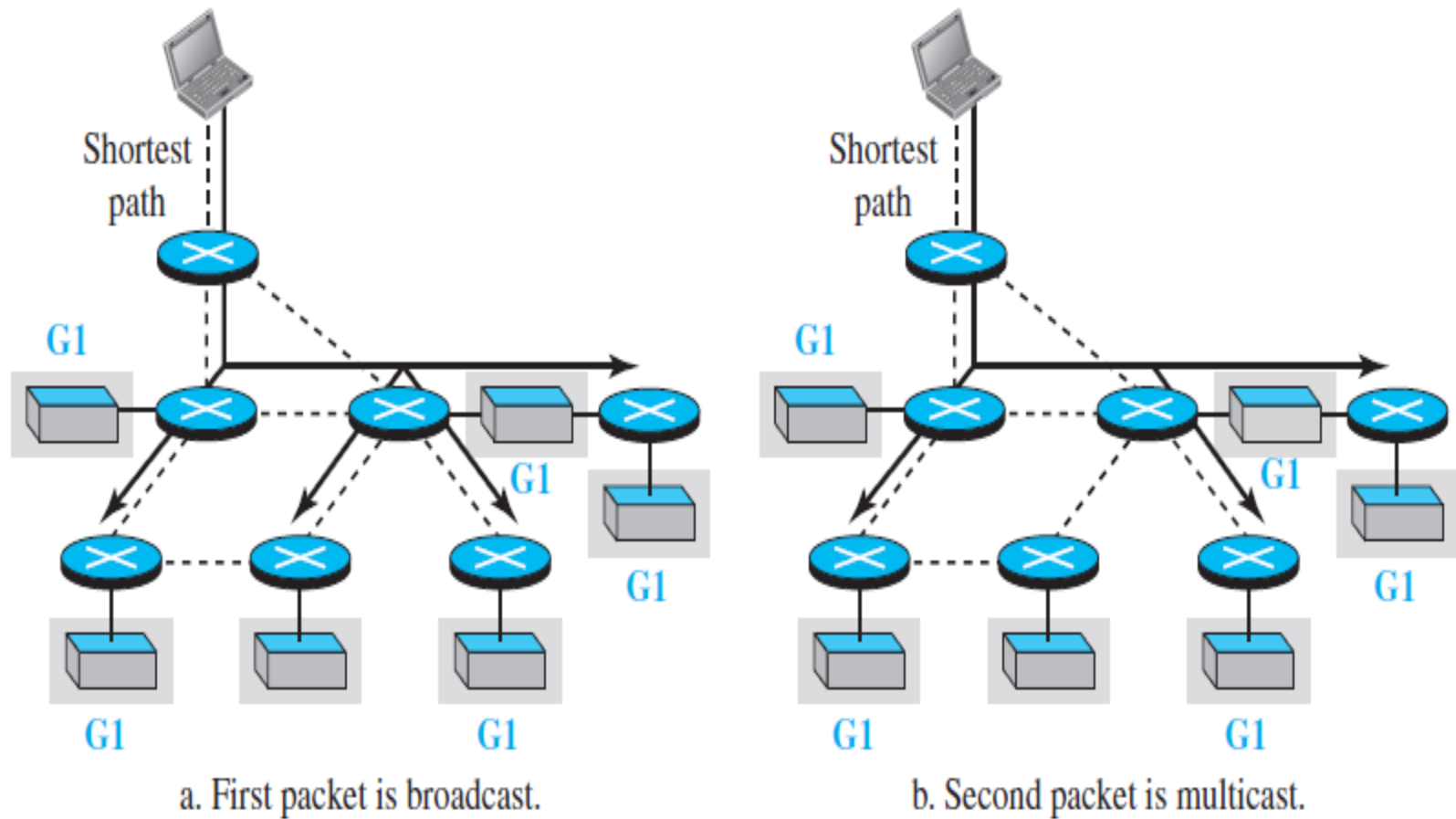
Protocol Independent Multicast-Dense Mode (PIM-DM)

- When the number of routers with attached members is large relative to the number of routers in the internet, PIM works in the dense mode and is called ***PIM-DM***.
- In this mode, the protocol uses a source-based tree approach and is similar to DVMRP, but simpler.
- PIM-DM uses only two strategies described in DVMRP: **RPF and RPM**

Protocol Independent Multicast-Sparse Mode (PIM-SM)

- When the number of routers with attached members is small relative to the number of routers in the internet, PIM works in the sparse mode and is called ***PIM-SM***.
- In this environment, the use of a protocol that broadcasts the packets until the tree is pruned is not justified;
- PIM-SM uses a group-shared tree approach to multicasting.
- The core router in PIM-SM is called the *rendezvous point* (RP).

Figure 21.14 *Idea behind PIM-DM*



INTERDOMAIN MULTICAST PROTOCOLS

INTRODUCTION

- The three protocols we discussed for multicast routing, DVMRP, MOSPF, and PIM, are designed to provide multicast communication inside an autonomous system.
- When the members of the groups are spread among different domains (ASs), we need an interdomain multicast routing protocol.
- One common protocol for interdomain multicast routing is called *Multicast Border Gateway Protocol* (MBGP)

CONT

- MBGP provides two paths between ASs: **one for unicasting and one for multicasting.**
- Information about multicasting is exchanged between border routers in different ASs.
- MBGP is a shared-group multicast routing protocol in which one router in each AS is chosen as the rendezvous point (RP).

CONT..

- The problem with MBGP protocol is that it is difficult to inform an RP about the sources of groups in other ASs.
- The Multicast Source Discovery Protocol (MSDP) is a new suggested protocol that assigns a source representative router in each AS to inform all RPs about the existence of sources in that AS.

CONT..

- Another protocol that is thought of as a possible replacement for the MBGP is **Border Gateway Multicast Protocol (BGMP)**, which allows construction of shared group trees with a single root in one of the ASs.
- In other words, for each group, there is only one shared tree, with leaves in different ASs, but the root is located in one of the ASs.

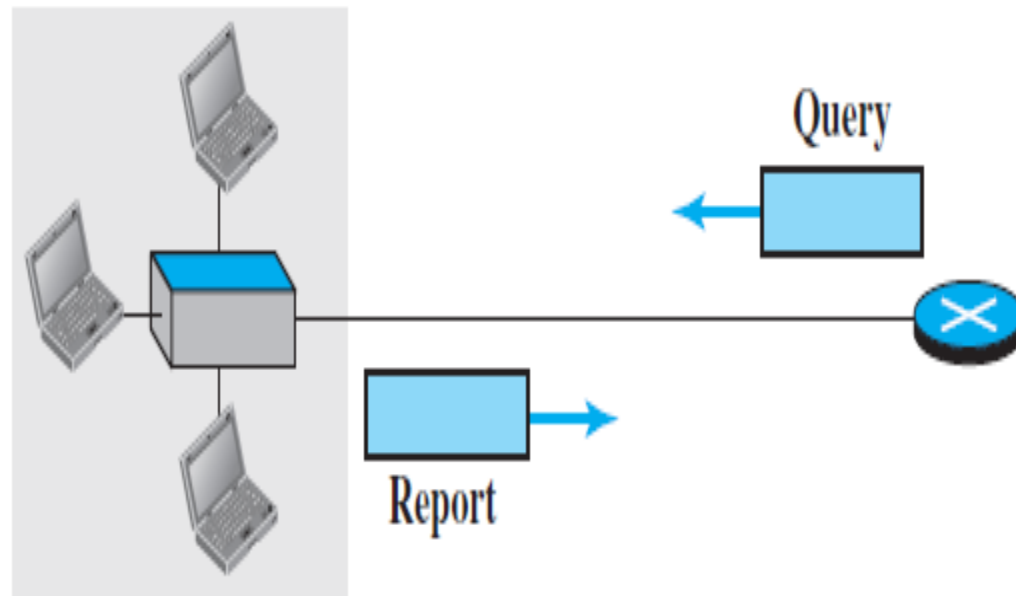
IGMP

- The protocol that is used today for collecting information about group membership is the **Internet Group Management Protocol (IGMP)**.
- IGMP is a protocol defined at the network layer; it is one of the auxiliary protocols, like ICMP, which is considered part of the IP.
- IGMP messages, like ICMP messages, are encapsulated in an IP datagram.

Messages

- There are only two types of messages in IGMP version 3, **query and report messages**.
- A query message is periodically sent by a router to all hosts attached to it to ask them to report their interests about membership in groups.
- A report message is sent by a host as a response to a query message.

IGMP operation



Query Message

- The query message is sent by a router to all hosts in each interface to collect information about their membership.
- There are three versions of query messages, as described below:
- a. A *general query message* is sent about membership in any group.
- It is encapsulated in a datagram with the destination address 224.0.0.1

CONT..

- **b.** A *group-specific* query message is sent from a router to ask about the related to a specific group.
- **c.** A *source-and-group-specific* query message is sent from a router to ask about the membership related to a specific group when the message comes from a specific source or sources.

Report Message

- A report message is sent by a host as a response to a query message.
- The message contains a list of records in which each record gives the identifier of the corresponding group (multicast address) and the addresses of all sources that the host is interested in receiving messages from (inclusion).
- The record can also mention the source addresses from which the host does not desire to receive a group message.
- The message is encapsulated in a datagram with the multicast address 224.0.0.22.

Propagation of Membership Information

- After a router has collected membership information from the hosts and other routers at its own level in the tree, it can propagate it to the router located in a higher level of the tree.
- Finally, the router at the tree root can get the membership information to build the multicast tree.
- The process, however, is more complex than what we can explain in one paragraph.
- Interested readers can check the book website for the complete description of this protocol.

Encapsulation

- The IGMP message is encapsulated in an IP datagram with the value of the protocol field set to 2 and the TTL field set to 1.
- The destination IP address of the datagram.

Table 21.1 *Destination IP Addresses*

<i>Message Type</i>	<i>IP Address</i>
General Query	224.0.0.1
Other Queries	Group address
Report	224.0.0.22