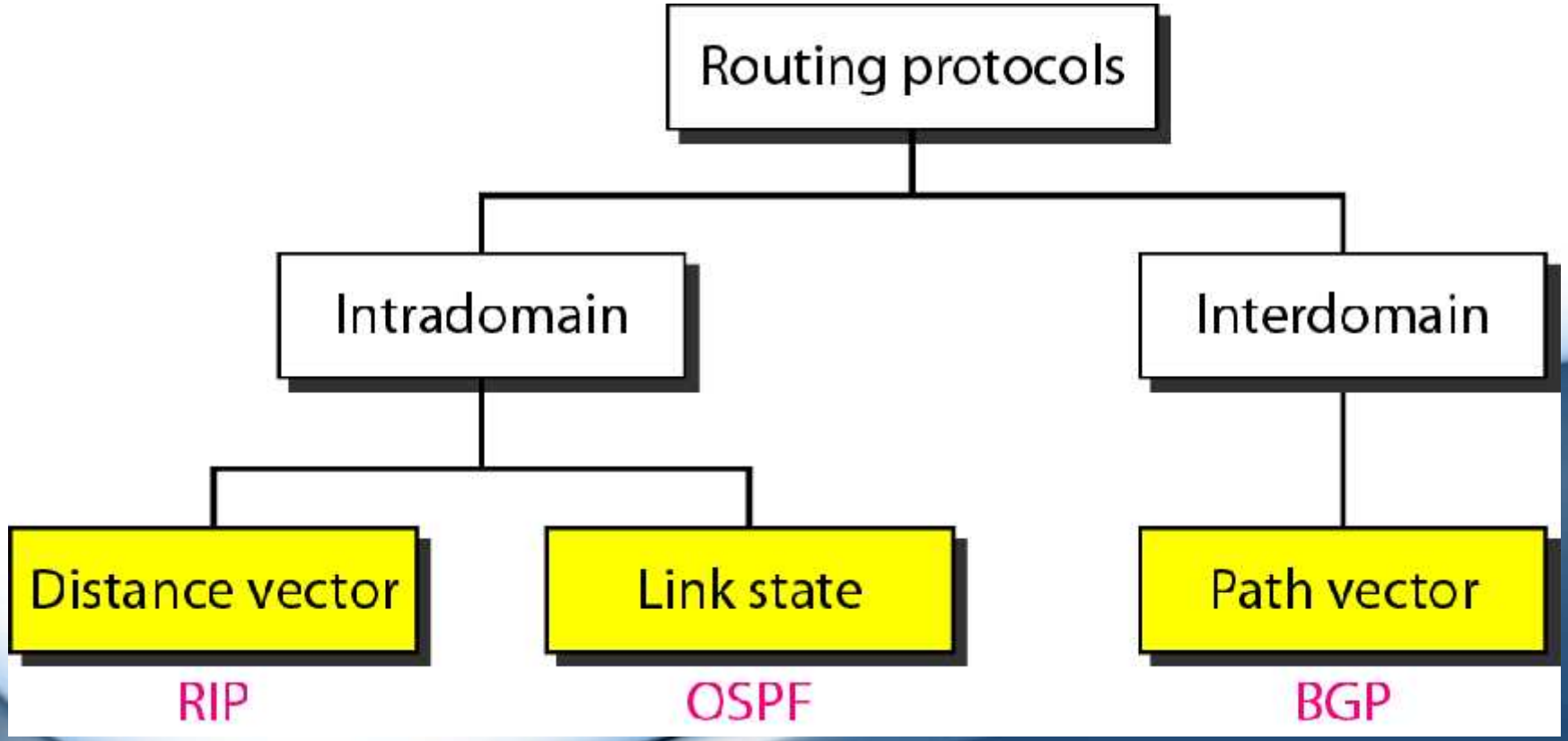
Unicast Routing Protocols

Unicast Routing Protocols

Topics Covered:

1. Routing Protocols
2. Intra- and Interdomain Routing
3. Distance Vector Routing
4. RIP

Routing Protocols



Intra- and Interdomain Routing :

- Today, an internet can be so large that one routing protocol cannot handle the task of updating the routing tables of all routers.
- For this reason, an internet is divided into autonomous systems.
- An autonomous system (AS) is a group of networks and routers under the authority of a single administration.
  - Routing inside an autonomous system is referred to as intradomain routing.
  - Routing between autonomous systems is referred to as interdomain routing.
- Each autonomous system can choose one or more intradomainrouting protocols to handle routing inside the autonomous system.
- However, only one interdomain routing protocol handles routing between autonomous systems.

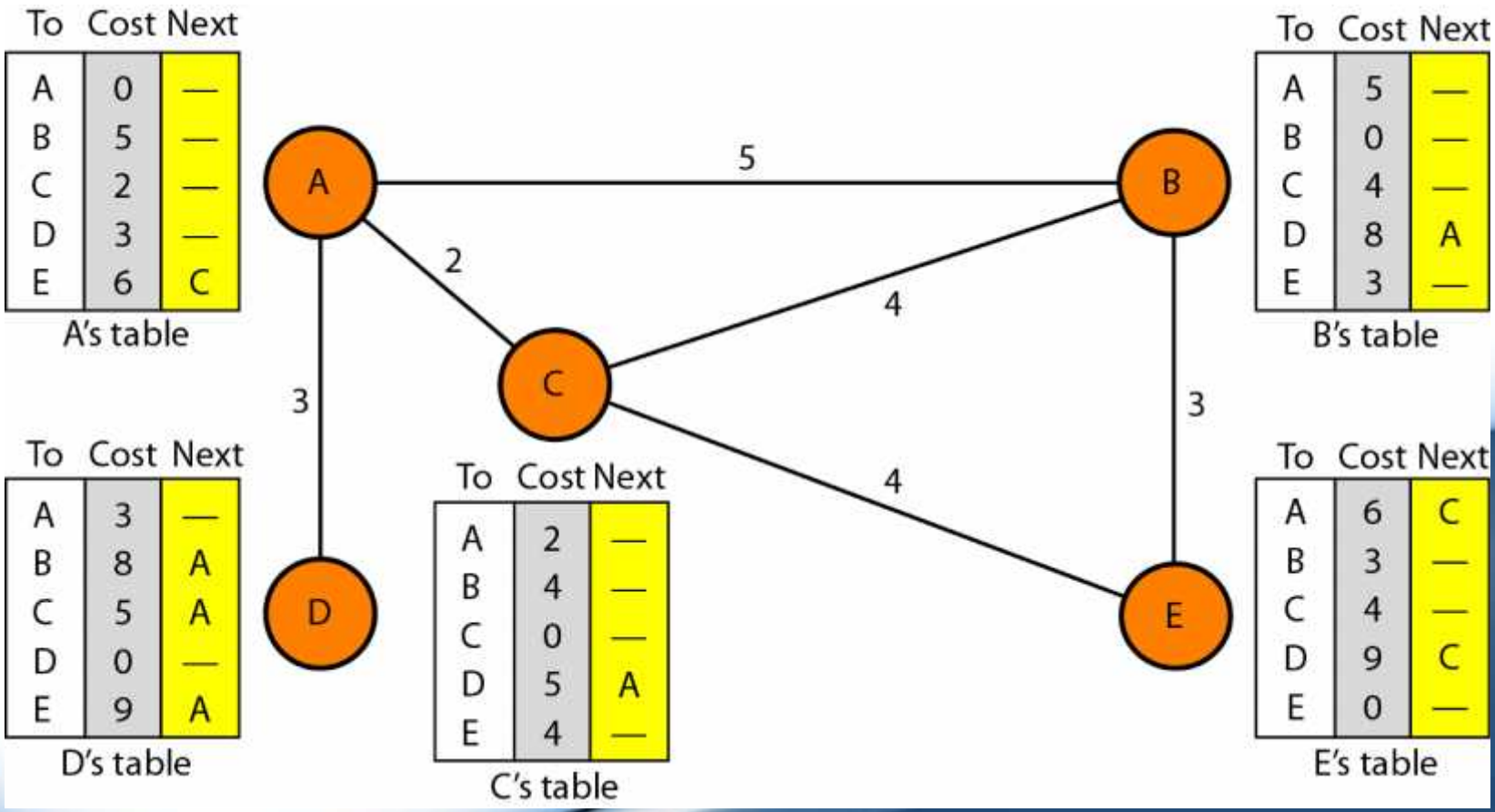
Distance Vector Routing :

- In distance vector routing, the least-cost route between any two nodes is the route with minimum distance.

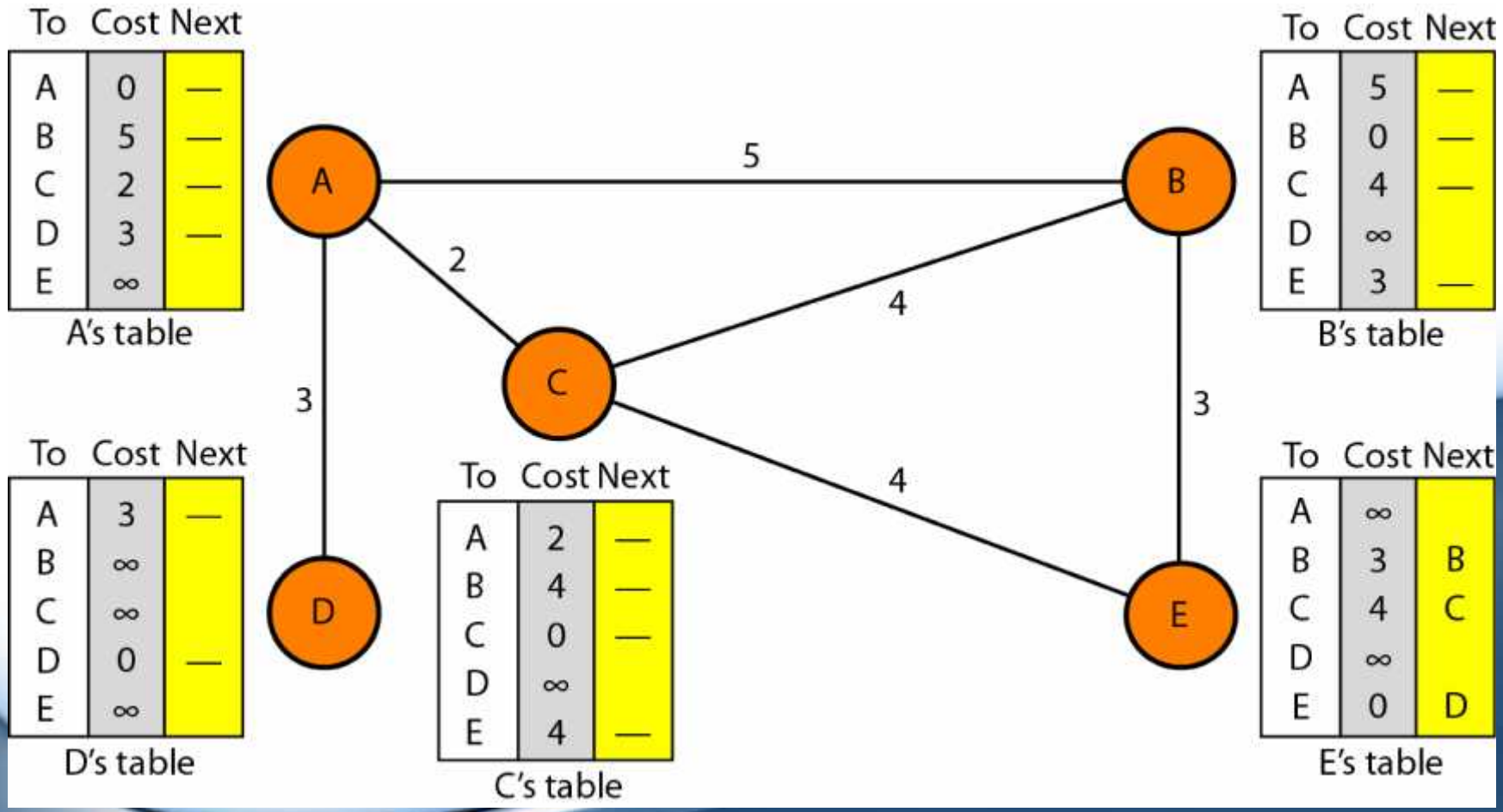
- In this protocol, as the name implies, each node maintains a vector (table) of minimum distances to every node. The table at each node also guides the packets to the desired node by showing the next stop in the route (next-hop routing).

- We can think of nodes as the cities in an area and the lines as the roads connecting them. A table can show a tourist the minimum distance between cities. In Figure 22.14, we show a system of five nodes with their corresponding tables

Distance Vector Routing tables:



Initialization



Sharing :

- The whole idea of distance vector routing is the sharing of information between neighbors. Although node A does not know about node E, node C does. So if node C shares its routing table with A, node A can also know how to reach node E.
- On the other hand, node C does not know how to reach node D, but node A does. If node A shares its routing table with node C, node C also knows how to reach node D. In other words, nodes A and C, as immediate neighbors, can improve their routing tables if they help each other.
- There is only one problem. How much of the table must be shared with each neighbor? A node is not aware of a neighbor's table.
- The best solution for each node is to send its entire table to the neighbor and let the neighbor decide what part to use and what part to discard. However, the third column of a table (next stop) is not useful for the neighbor.

Sharing :

- When the neighbor receives a table, this column needs to be replaced with the sender's name. If any of the rows can be used, the next node is the sender of the table.

- A node therefore can send only the first two columns of its table to any neighbor. In other words, sharing here means sharing only the first two columns. In distance vector routing, each node shares its routing table with its immediate neighbors periodically and when there is a change.

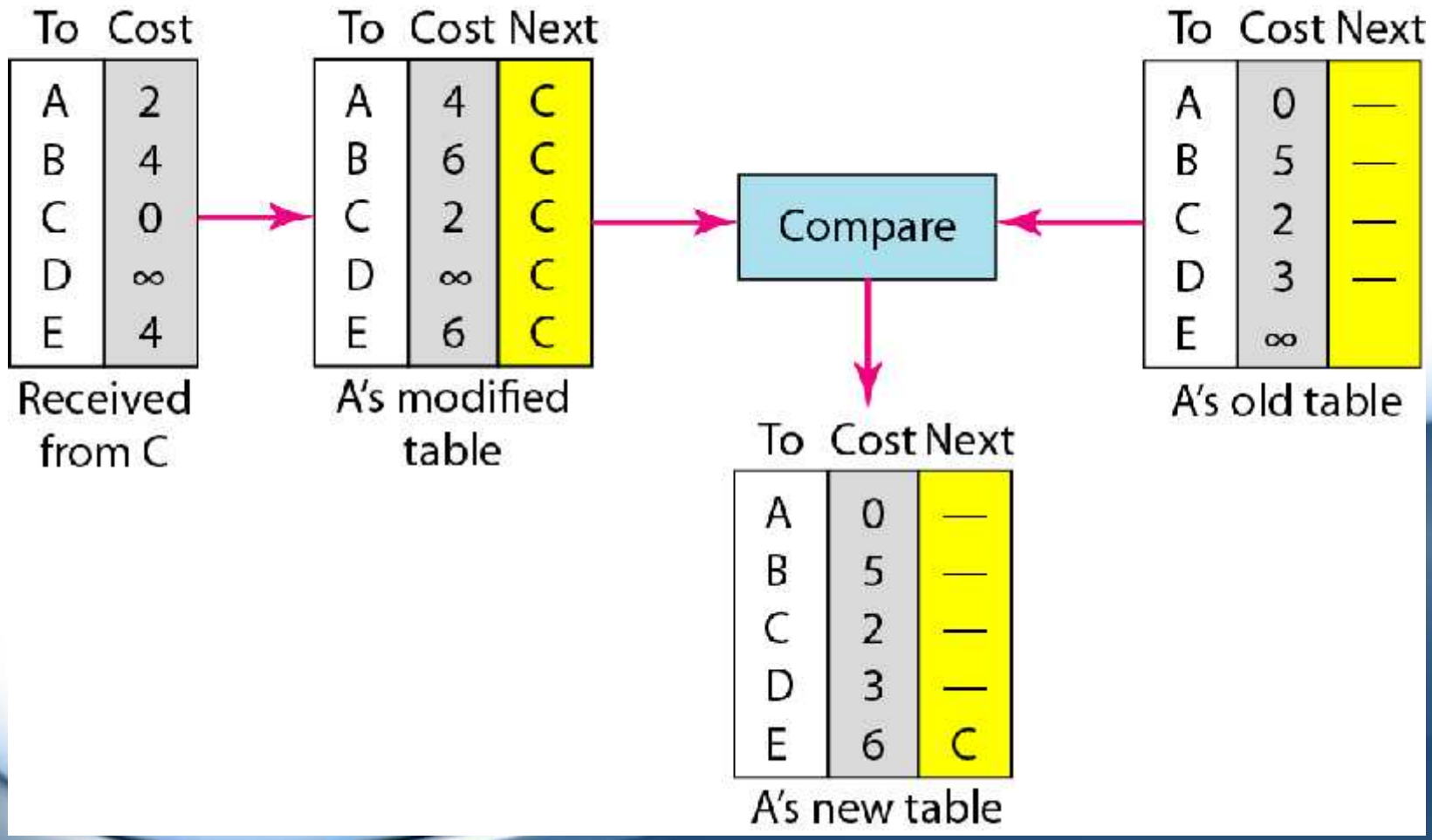
Updating :

- When a node receives a two-column table from a neighbor, it needs to update its routing table. Updating takes three steps:
- The receiving node needs to add the cost between itself and the sending node to each value in the second column. The logic is clear. If node C claims that its distance to a destination is *x* mi, and the distance between A and C is *y* mi, then the distance between A and that destination, via C, is *x* + *y* mi.
- The receiving node needs to add the name of the sending node to each row as the third column if the receiving node uses information from any row. The sending node is the next node in the route.
- The receiving node needs to compare each row of its old table with the corresponding row of the modified version of the received table.
  - If the next-node entry is different, the receiving node chooses the row with the smaller cost. If there is a tie, the old one is kept.
  - If the next-node entry is the same, the receiving node chooses the new row. For example, suppose node C has previously advertised a route to node X with distance

Updating :

- Suppose that now there is no path between C and X; node C now advertises this route with a distance of infinity.
- Node A must not ignore this value even though its old entry is smaller.
- The old route does not exist any more.
- The new route has a distance of infinity.

Updating


To Cost
A 2
B 4
C 0
D ∞
E 4
Received from C
To Cost Next
A 4 C
B 6 C
C 2 C
D ∞ C
E 6 C
A's modified table
Compare
To Cost Next
A 0 —
B 5 —
C 2 —
D 3 —
E ∞
A's old table
To Cost Next
A 0 —
B 5 —
C 2 —
D 3 —
E 6 C
A's new table

Important Points

- There are several points we need to emphasize here.
  - First, as we know from mathematics, when we add any number to infinity, the result is still infinity.
  - Second, the modified table shows how to reach A from A via C. If A needs to reach itself via C, it needs to go to C and come back, a distance of 4.
  - Third, the only benefit from this updating of node A is the last entry, how to reach E. Previously, node A did not know how to reach E (distance of infinity); now it knows that the cost is 6 via C.

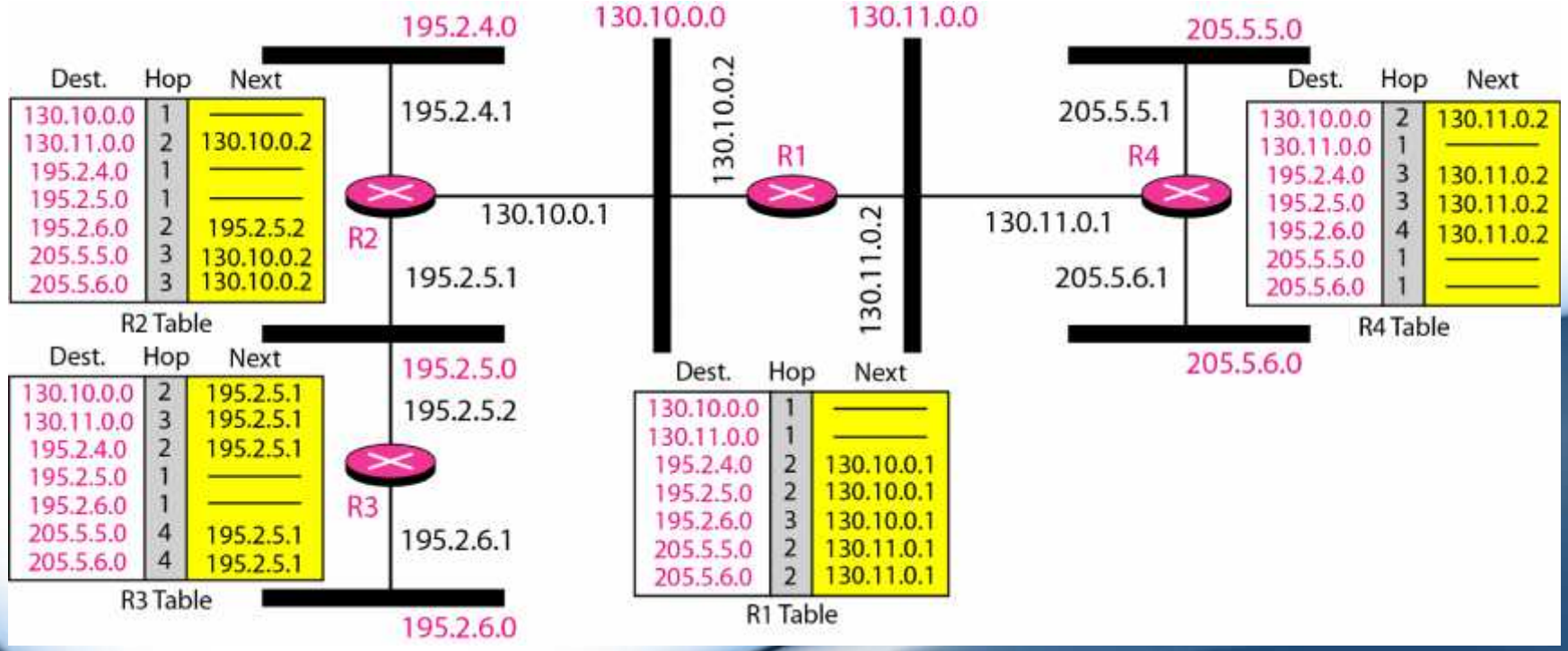
When to Share :

- When does a node send its partial routing table (only two columns) to all its immediate neighbors? The table is sent both periodically and when there is a change in the table.
  - *Periodic Update* A node sends its routing table, normally every 30 s, in a periodic update. The period depends on the protocol that is using distance vector routing.
  - *Triggered Update* A node sends its two-column routing table to its neighbors anytime there is a change in its routing table. This is called a triggered update. The change can result from the following.
    - A node receives a table from a neighbor, resulting in changes in its own table after updating.
    - A node detects some failure in the neighboring links which results in a distance change to infinity.

RIP :

- The Routing Information Protocol (RIP) is an intradomain routing protocol used inside an autonomous system. It is a very simple protocol based on distance vector routing. RIP implements distance vector routing directly with some considerations:
  - In an autonomous system, we are dealing with routers and networks (links). The routers have routing tables; networks do not.
  - The destination in a routing table is a network, which means the first column defines a network address.
  - The metric used by RIP is very simple; the distance is defined as the number of links (networks) to reach the destination. For this reason, the metric in RIP is called a hop count.
  - Infinity is defined as 16, which means that any route in an autonomous system using RIP cannot have more than 15 hops.
  - The next-node column defines the address of the router to which the packet is to be sent to reach its destination.

A Sample of RIP



Explanation of Sample

- The table of each router is also shown. Let us look at the routing table for RI.
- The table has seven entries to show how to reach each network in the autonomous system.
- Router RI is directly connected to networks 130.10.0.0 and 130.11.0.0, which means that there are no next-hop entries for these two networks.
- To send a packet to one of the three networks at the far left, router RI needs to deliver the packet to R2.
- The next-node entry for these three networks is the interface of router R2 with IP address 130.10.0.1.
- To send a packet to the two networks at the far right, router RI needs to send the packet to the interface of router R4 with IP address 130.11.0.1.

Thank You