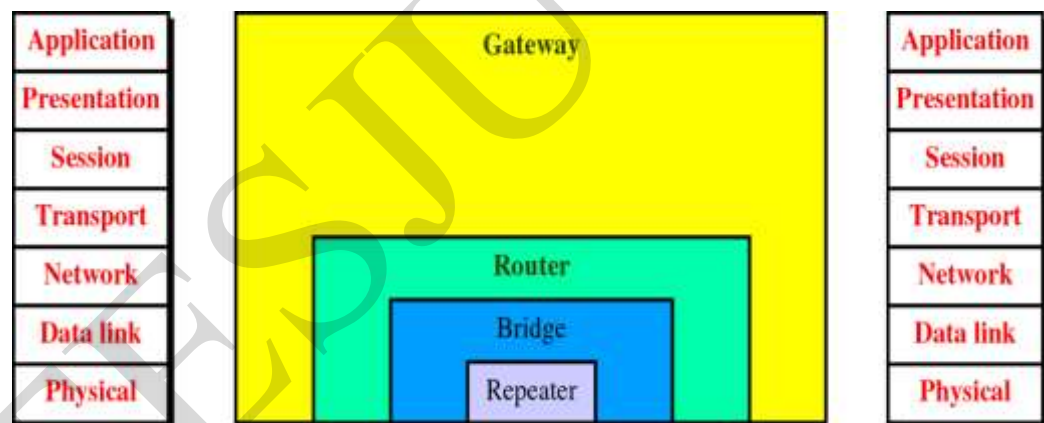
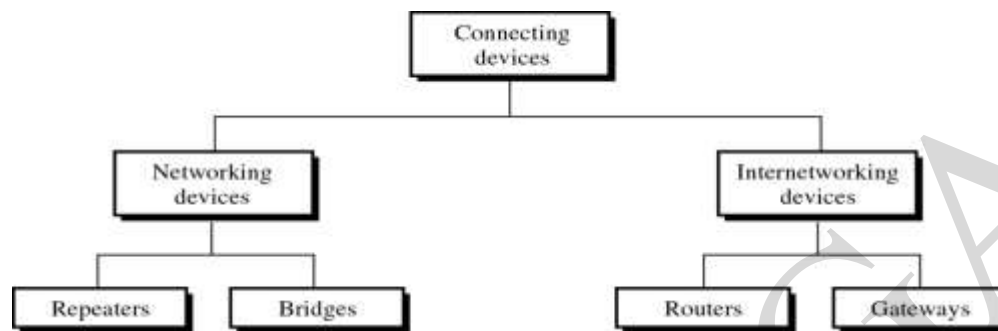


UNIT - III

Networking and Internetworking Devices



1) Network Interface Cards

- Network interface cards, commonly referred to as NICs, are used to connect a PC to a network.
- The NIC provides a physical connection between the networking cable and the computer's internal bus.
- NICs come in three basic varieties: 8-bit, 16-bit, and 32-bit. The larger the number of bits that can be

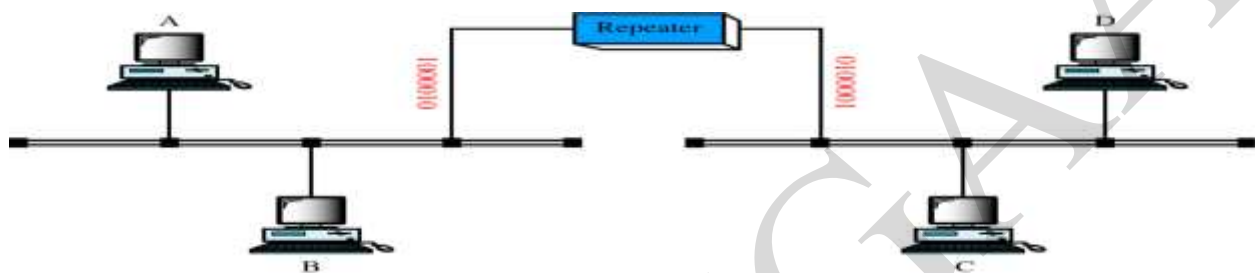
transferred to the NIC, the faster the NIC can transfer data to the network cable.



Repeaters: Since a signal loses strength as it passes along a cable, it is often necessary to boost the signal with a device called a repeater. A repeater is an electronic device that receives a signal, cleans it of unnecessary noise, regenerates it, and retransmits it at a higher power level so that the signal can cover longer distances without degradation.

This device can be used to convert between different Physical Layers (e.g., Twisted Pair to Fiber Optic).

Repeaters attempt to preserve signal integrity and extend the distance over which data can safely travel.

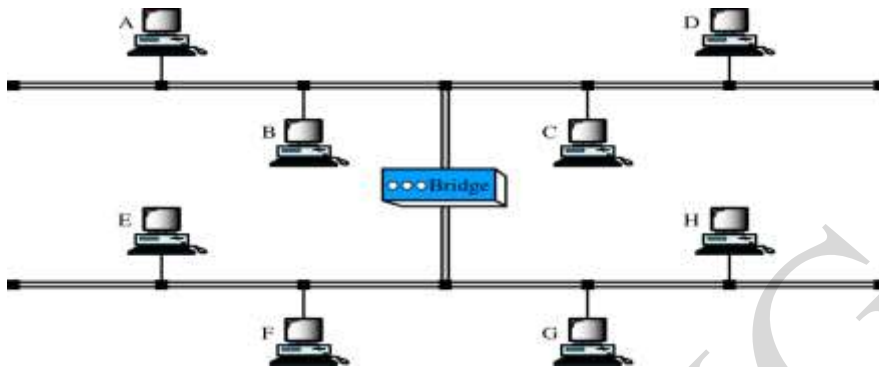


Bridges: A Bridge is a device performing network interconnection at Data Link Layer. Bridges use the Data Link Layer address on every packet to determine how to forward it and monitor traffic continuously to learn which Data Link Layer addresses reside on which ports of the bridge.

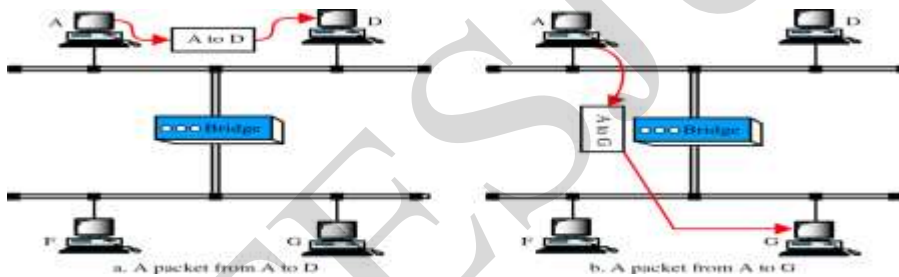
A bridge is a device that connects a local area network (LAN) to another local area network that uses the same protocol (for example, Ethernet or token ring).

- The function of a bridge is to connect separate networks together.

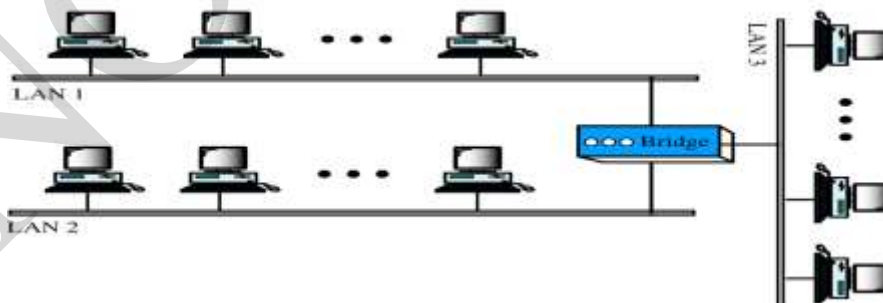
- When a packet is received by the bridge, the bridge determines the destination and source segments.



Function of a Bridge



Multiport Bridge



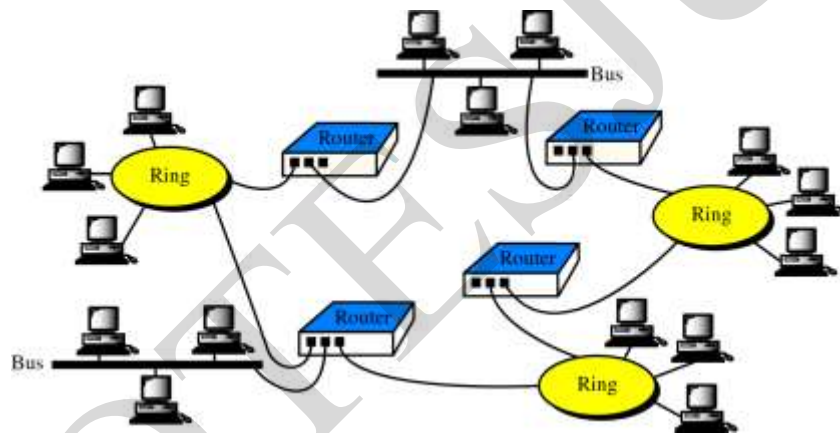


Routers: A Router is a device that performs network interconnection at network layer. Like bridges, routers maintain a table.

Routers use "routing protocols" to communicate with other routers in the network and maintain the information in these routing tables. Unlike bridges, routers do not flood broadcast or multicast traffic. Instead, they use the Layer 3 information in the packets to make selective forwarding decisions for this type of traffic and can therefore scale the available bandwidth of the network for all traffic types. And because routers have greater awareness of the network topology (learned via routing protocols), they can support multiple active paths between communicating hosts .

This device interconnects SIMILAR networks, e.g. similar protocols and workstations and servers.

- A **router** is an electronic device that interconnects two or more computer networks, and selectively interchanges packets of data between them.
- Each data packet contains address information that a router can use to determine if the source and destination are on the same network, or if the data packet must be transferred from one network to another.



Gateways: A Gateway is a device that performs interconnection at Presentation Layer. (Note, the term Gateway is also sometimes applied to Routers) Gateways are generally built to enable communication between two different applications with a similar purpose: a good

example is an e-mail gateway converting between two different email protocols.

Gateways are always very application -specific; there is no general-purpose interconnection device operating above Network Layer. Gateways are used to interconnect two different networks having different protocols.

- Networks using different protocols use different addressing formats.
- A gateway is a network point that acts as an entrance to another network.
- Gateways are also called **protocol converters**.



What is the difference?

- Bridge: device to interconnect two LANs that use the SAME logical link control protocol but may use different medium access control protocols.
- Router: device to interconnect SIMILAR networks, e.g .similar protocols and workstations and servers.

- Gateway: device to interconnect DISSIMILAR protocols and servers, and Macintosh and IBM LANs and equipment

Functions of Network Layer

- **Addressing** – Maintains both the source and destination addresses at the frame header. The network layer performs addressing to find out the specific devices on the network.
- **Packetizing** – The network layer works on the conversion of packets those received from its upper layer. This feature is accomplished by Internet Protocol (IP).
- **Routing** – Being considered as the major functionality, the network layer chooses the best path for data transmission from a source point to the destination.
- **Internetworking** – Internetworking works to deliver a logical connection across multiple devices

Functions of Router

- **Forwarding Function** It is a **function** that allows selection of the appropriate route based on IP header information and sends packets through this route.
- **Filtering Function** It is a **function** that allows dumping of invalid packets for a specific network instead of **forwarding**.
-

The router has to function Forwarding Function and Filtering Function .

1. **Forwarding Function:-** This function allows you choose the router to reserve the IP header information and send their packets over to the router .In forwarding function the most important is to resolve the route, process and forward the IP

address to the next router. Forwarding function deals with packet to send on the same network by executing its functions and work as an integral part of routing.

The Routing has been done on a same network, which is also called a **Local Routing**. When the address is a host on the same IP address then it becomes easy to routing in the same subnet and if it is not on the same host then it becomes little difficult as, the packet first sent to the boundary router. This boundary router indicates that the routing table at the next hop (IP address of next router). The routing table is fully depends upon the routing method as it is fully composed of a table. The routing table involves with router and their destination address and the IP address.

- **Unicast pattern**

This pattern is considered to be the simplest one whereby the forwarding of the packet data is quite straight forward; the needed packet is usually relayed from one chain to another on a path that is described as from the source to the required destination. It is common in most of the network technologies like the internet.

- **Multicast pattern**

It is usually common in the PIM system, the packet duplicated into several copies that are later on delivered to the required recipients presented as sets.

- **Broadcast pattern**

This type requires the packets to be duplicated after which the copies are sent to multiple links and ensuring that every device on the network has received a copy. The idea is to only transfer copies on network devices that are within a certain broadcast domain. It is commonly found in the bridged Ethernet where there are different broadcast domains.

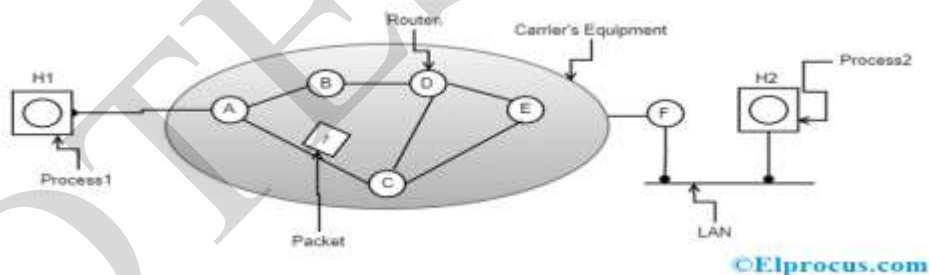
2. Filtering Function:-Filtering Function is used as a dumping (disposition)for a specific network. It is used to filter the packet and pass through a router.This function is different from forwarding function as it work to filter if two networks connect through a router than it communicate using UDP packets and filter as it pass the signals to one another by filtering process.

Network Layer Design Issues

Network layer comes up with certain design issues and they can be described as below:

1). Store-and-Forward Packet Switching

Here, the foremost elements are the carrier's equipment (the connection between routers through transmission lines) and the customer's equipment.



and-forward packet switching

- H1 has a direct connection with carrier router 'A', while H2 is connected to carrier router 'F' on a LAN connection.

- One of the carrier router 'F', is pointed outside the carrier's equipment as it does not come under the carrier, whereas considered as protocols, software, and construction.
- This switching network performs as Transmission of data happens when the host (H1) with a packet transfers it to the nearby router through LAN (or) point-to-point connection to the carrier. The carrier stores the packet until it completely arrives thus confirms the checksum.
- Then after, the packet is transmitted over the path until H2 is reached.

2). Services Provided to the Transport Layer

Through the network/transport layer interface, the network layer delivers its services to the transport layer.

Services offered by the network layer are outlined considering few objectives. Those are:

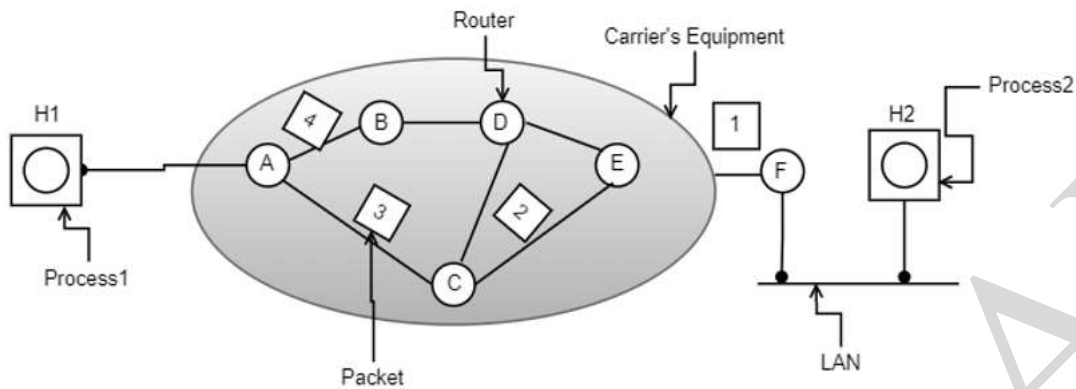
- Offering services must not depend on router technology
- The transport layer needs to be protected from type, number and the topology of the available routers.

Connectionless – Here, routing and insertion of packets into subnet is accomplished individually. No additional setup is necessary.

Connection-Oriented – Subnet must offer reliable service and all the packets are transmitted over a single route.

3). Implementation of Connectionless Service

In this scenario, packets are termed as datagrams and the corresponding subnet is termed as datagram subnet. Routing in datagram subnet is as follows:



When the message size that has to be transmitted is 4 times the size of the packet, then the network layer divides into 4 packets and then transmits each packet to router 'A' through a few protocols. Each router is provided with a routing table where it decides the destination points. In the above figure, it is clear that packets from 'A' need to be transmitted either to B or C even when the destination is 'F'. The routing table of 'A' is clearly outlined above.

Whereas in the case of packet 4, the packet from 'A' is routed to 'B', even the destination node is 'F'. Packet 'A' chooses to transmit packet 4 through a different path than the initial three paths. This might happen because of traffic congestion along the path ACE. So, the

4). Implementation of Connection-Oriented Service

Here, the functionality of connection-oriented service works on the virtual subnet. A virtual subnet performs the operation of avoiding a new path for each packet transmission. As a substitute for this,

when there forms a connection, a route from a source node to a destination node is selected and maintained in tables.

Network Layer Services

The network layer provides services that permit end devices for information exchange across the network. To achieve this, it makes use of four processes where those are of

- Addressing end devices
- Encapsulation
- Routing
- De-encapsulation

With all the routing protocols, types, services, and other frameworks, the network layer stands as a great support for the OSI model. The functionality of the network layer contains in every router.

DELIVERY, FORWARDING & ROUTING

- **Delivery refers to the way packet is handled by the network.**
- **Forwarding refers to the way a packet is delivered to the next station.**
- **Routing refers to the way routing tables are created to help in forwarding.**

Delivery is of 2 ways : Direct and Indirect

Direct delivery occurs when the source and destination of the packet are located on the same network or when the delivery is between last router and the destination host.

In indirect delivery the packet goes from router to router until it reaches the router connected on the destination network.

Routing

Routing is the process of forwarding of a packet in a network so that it reaches its intended destination. The main goals of routing are:

1. **Correctness:** The routing should be done properly and correctly so that the packets may reach their proper destination.
2. **Simplicity:** The routing should be done in a simple manner so that the overhead is as low as possible. With increasing complexity of the routing algorithms the overhead also increases.
3. **Robustness:** Once a major network becomes operative, it may be expected to run continuously for years without any failures. The algorithms designed for routing should be robust enough to handle hardware and software failures and should be able to cope with changes in the topology and traffic without requiring all jobs in all hosts to be aborted and the

network rebooted every time some router goes down.

4. **Stability:** The routing algorithms should be stable under all possible circumstances.
5. **Fairness:** Every node connected to the network should get a fair chance of transmitting their packets. This is generally done on a first come first serve basis.
6. **Optimality:** The routing algorithms should be optimal in terms of throughput and minimizing mean packet delays. Here there is a trade-off and one has to choose depending on his suitability.

ROUTING TABLE:

A router has a routing table with an entry for each destination to route packets.

- Routing tables can be static or dynamic.
- Static Routing tables contains information entered manually. The administrator enters the route for each destination in the table and updated for changes manually by the administrator.
- Dynamic Routing table is updated periodically by the routing protocols such as RIP, OSPF or BGP. Whenever there is a change in the Internet, such as shutdown of router or breaking of link, the dynamic routing protocols update all tables in the routers.

 **Common fields in a routing table**

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

5 Flags are as follows:

U(up): This flag indicates the router is up & running. If this flag is not present, it means that the router is down. The packet cannot be forwarded & is discarded.

G(gateway): This means that the destination is in another network. When this flag is missing, it means the destination is in this network(direct delivery).

H(host-specific): this flag indicates that the entry in the network address field is a host specific address. When it is missing, it means that the address is only the network address of the destination.

D(added by direction): This flag indicates that routing information for this destination has been added to the host routing table by a redirection message from ICMP.

M(modified by redirection): The M flag indicates that the routing information for this destination has been modified by a redirection message from ICMP.

Interface: This field shows the name of the interface.

Reference Count: This field gives the number of the users of the route at the moment.

Use: This field shows the number of packets transmitted through the router for the corresponding destination.

Classification of Routing Algorithms(vimp)

The routing algorithms may be classified as follows:

1. Adaptive Routing Algorithm: These algorithms change their routing decisions to reflect changes in the topology and in traffic as well. These get their routing information from adjacent routers or from all routers. The optimization parameters are the distance, number of hops etc.

2. Non-Adaptive Routing Algorithm: These algorithms do not base their routing decisions on measurements and estimates of the current traffic and topology. Instead the route to be taken in going from one node to the other is computed in advance, off-line, and downloaded to the routers when the network is booted. This is also known as static routing.

(a) Shortest Path Routing:

The basic idea of this technique is to build a graph of the subnet, with each node of the graph representing a router and each arc of the graph representing a communication line i.e., link. For finding a route between a given pair of routers, the algorithm just finds the shortest path between them on the graph. The length of a path can be measured in a number of ways like number of hops, or on the basis of geographic distance etc.

Strategy is given below:

1. Each node is labelled with the name of the source node and its distance from the current node.
2. At the start of the algorithm all nodes are labeled tentatively.
3. As the algorithm progresses, the labels may change.
4. At any stage, when it becomes clear that the current label represents the smallest distance / shortest path

between a node and the source node, former's label is marked as a permanent label.

5. As the algorithm progresses, more and more nodes acquire permanent labels.

6. The algorithm terminates when the destination node gets a permanent label.

(b) Flooding

Every incoming packet to a node is sent out on every outgoing line except the one it arrived on. All possible routes between source and destination are tried.


A packet will always get through if a path exists. As all routes are tried, at least one packet will pass through the shortest route. All nodes, directly or indirectly connected, are visited. Main limitation of flooding is that it generates vast number of duplicate packets. A variation, which is slightly more practical, is ***selective flooding***.

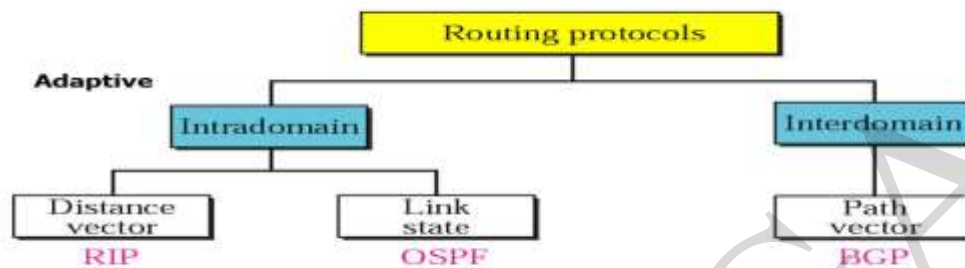
The routers do not send every incoming packet out on every line, only on those lines that go in approximately in the direction of destination.

| Routing | Flooding |
|--------------------------------|-----------------------------------|
| --> Routing table is required. | --> No routing table is required. |
| --> May give shortest path. | --> Always gives shortest path. |
| --> Less reliable. | --> More reliable. |
| --> Traffic is less. | --> Traffic is high. |
| --> No duplicate packets. | --> Duplicate packets are present |

Dynamic Routing Algorithms

Nowadays, computer networks generally use dynamic routing algorithms rather than the static ones described above because; static algorithms do not take the current network load into account.

 **Figure 14.2** Popular routing protocols



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 *intra-domain routing*.
- Routing between autonomous systems is referred to as *inter-domain routing*.

| Distance Vector Routing | Link State Routing |
|--|---|
| --> Bandwidth required is less due to local sharing, small packets and no flooding. | --> Bandwidth required is more due to flooding and sending of large link state packets. |
| --> Based on local knowledge since it updates table based on information from neighbors. | --> Based on global knowledge i.e. it have knowledge about entire network. |
| --> Make use of Bellman Ford algo | --> Make use of Dijkstra's algo |
| --> Traffic is less | --> Traffic is more |
| --> Converges slowly i.e. good news spread fast and bad news spread slowly. | --> Converges faster. |
| --> Count to infinity problem. | --> No count to infinity problem. |
| --> Persistent looping problem i.e. loop will there forever. | --> No persistent loops, only transient loops. |
| --> Practical implementation is RIP and IGRP. | --> Practical implementation is OSPF and ISIS. |

Distance vector routing and link state routing are two main dynamic algorithms.

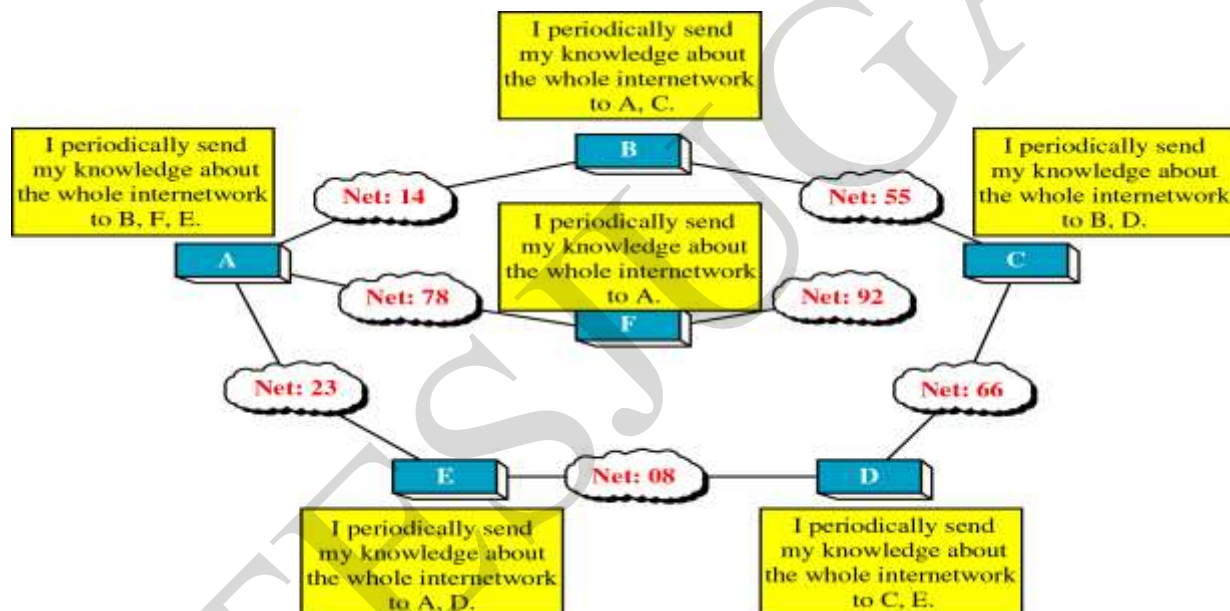
Distance vector routing algorithm is also known as **Bellman-Ford routing algorithm.**

It is the original Dynamic Routing Algorithm used in the ARPANET.

In brief, this scheme may be expressed as:

- Each router shares its knowledge about the network to its neighbors.
- Each router periodically sends its knowledge about the network to those routers to which it has direct links.
- This method assumes a cost one unit for every link, the cost is based on hop count.

First step : Initialization



Concept of Distance Vector Routing

Second Step: Sharing

A router sends its knowledge to its neighbors. The neighbors add this knowledge to their own knowledge and send the whole table to their neighbors. In this way after some time, every router will know about every other router on the network.

Third Step: Updation

Update the routing tables, if required.

Distance Vector Routing (DVR) Protocol (with Example)

Bellman Ford Basics – Each router maintains a Distance Vector table containing the distance between itself and ALL possible destination nodes. Distances, based on a chosen metric, are computed using information from the neighbors' distance vectors.

Information kept by DV router -

- Each router has an ID
- Associated with each link connected to a router, there is a link cost.
- Intermediate hops

Distance Vector Table Initialization -

- Distance to itself = 0
- Distance to ALL other routers = infinity number.

Distance Vector Algorithm –

1. A router transmits its distance vector to each of its neighbors in a routing packet.
2. Each router receives and saves the most recently received distance vector from each of its neighbors.
3. A router recalculates its distance vector when:
 - It receives a distance vector from a neighbor containing different information than before.
 - It discovers that a link to a neighbor has gone down.

The DV calculation is based on minimizing the cost to each destination

$D_x(y)$ = Estimate of least cost from x to y

$C(x,v)$ = Node x knows cost to each neighbor v

D_x = $[D_x(y): y \in N]$ = Node x maintains distance vector

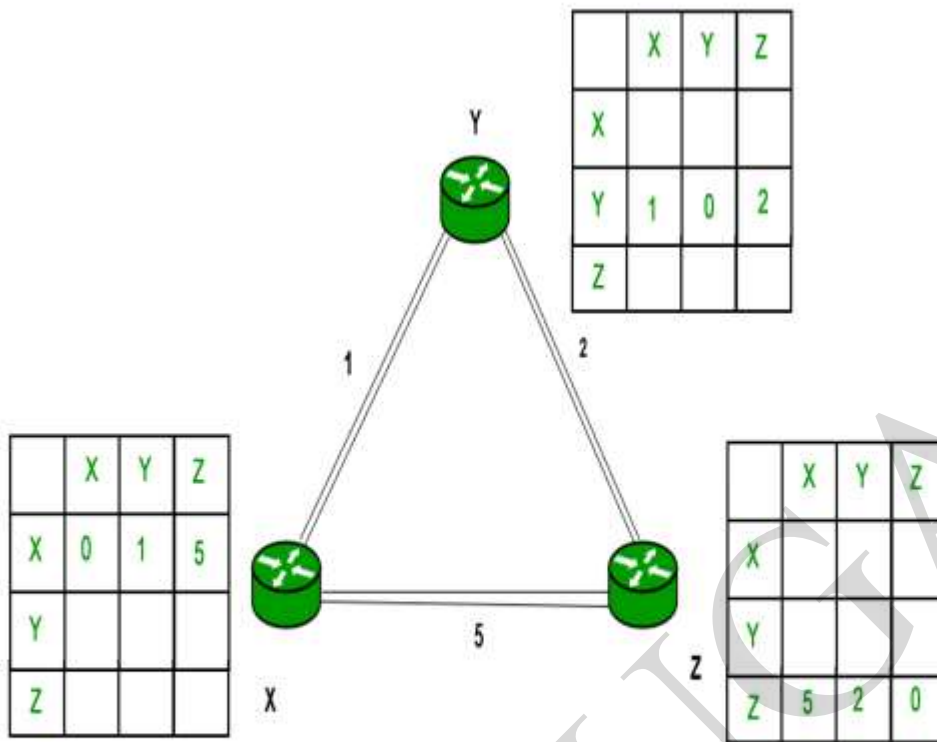
Node x also maintains its neighbors' distance vectors

– For each neighbor v , x maintains $D_v = [D_v(y): y \in N]$

Note –

- From time-to-time, each node sends its own distance vector estimate to neighbors.
- When a node x receives new DV estimate from any neighbor v , it saves v 's distance vector and it updates its own DV using B-F equation:
- $D_x(y) = \min \{ C(x,v) + D_v(y), D_x(y) \}$ for each node $y \in N$

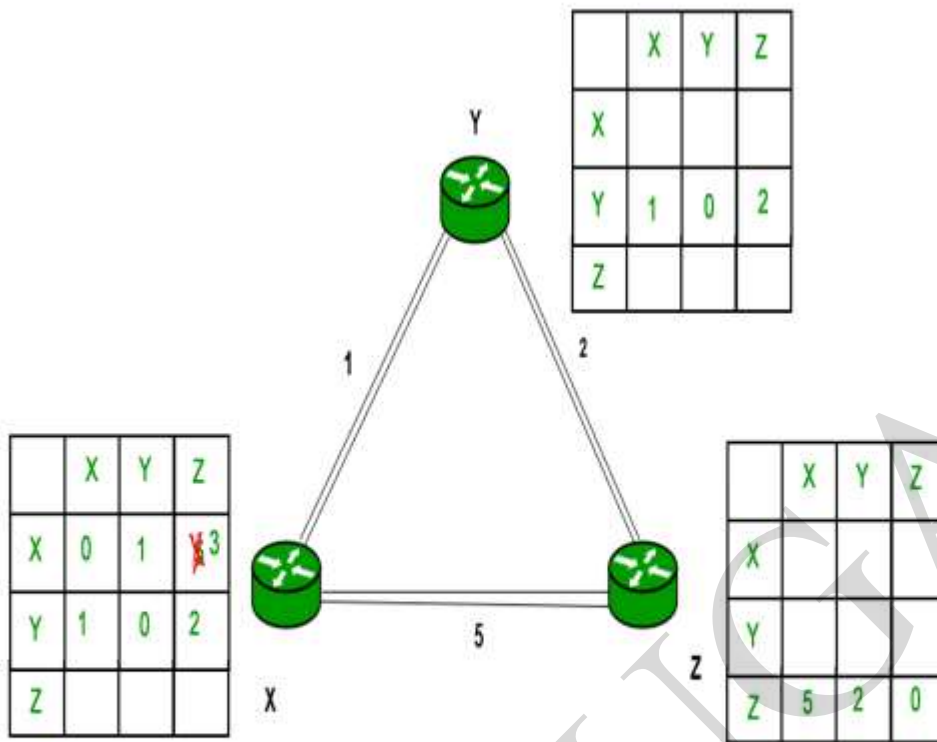
Example – Consider 3-routers X , Y and Z as shown in figure. Each router have their routing table. Every routing table will contain distance to the destination nodes.



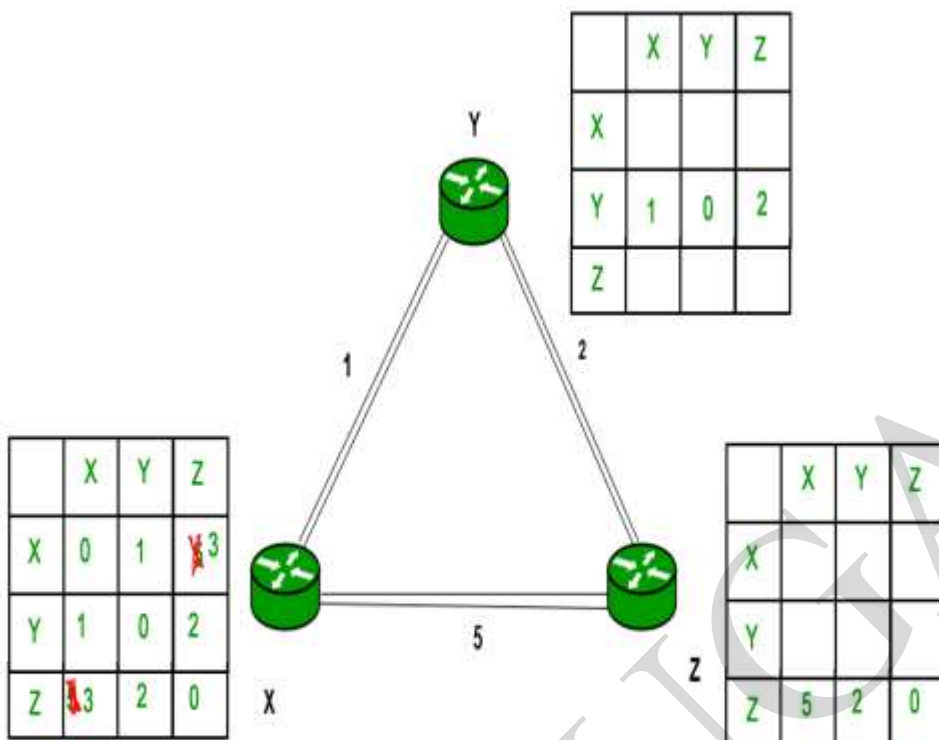
Consider router X , X will share it routing table to neighbors and neighbors will share it routing table to it to X and distance from node X to destination will be calculated using bellmen- ford equation.

$$D_x(y) = \min \{ C(x,v) + D_v(y) \} \text{ for each node } y \in N$$

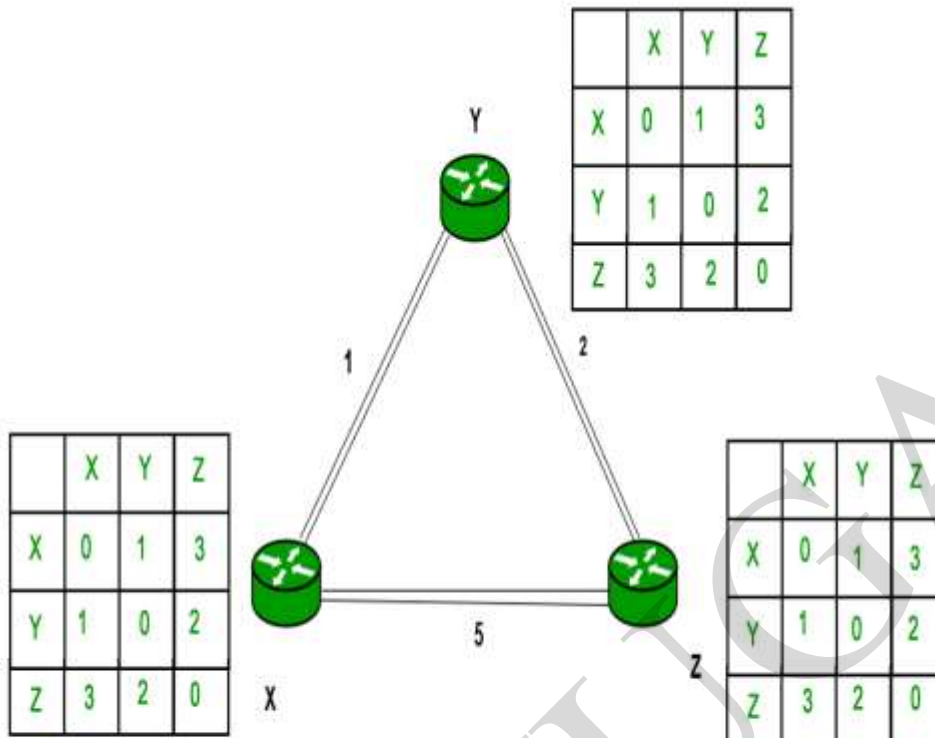
As we can see that distance will be less going from X to Z when Y is intermediate node(hop) so it will be update in routing table X.



Similarly for Z also –



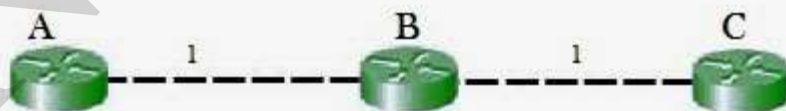
Finally the routing table for all –



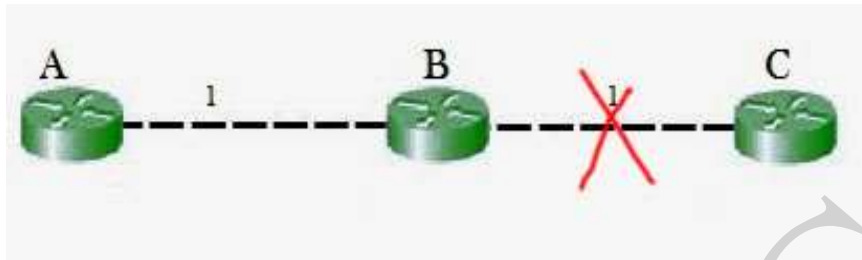
Problem in Distance vector Routing: Count to infinity problem:

1. One of the important issue in Distance Vector Routing is County of Infinity Problem.
2. Counting to infinity is just another name for a routing loop.
3. In distance vector routing, routing loops usually occur when an interface goes down.
4. It can also occur when two routers send updates to each other at the same time.

Counting to infinity problem:



So in this example, the Bellman-Ford algorithm will converge for each router, they will have entries for each other. B will know that it can get to C at a cost of 1, and A will know that it can get to C via B at a cost of 2.



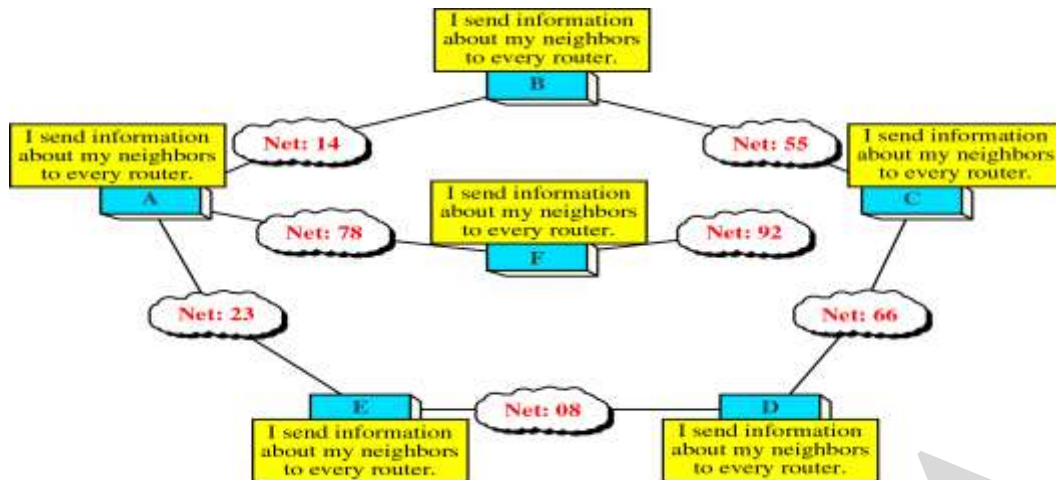
If the link between B and C is disconnected, then B will know that it can no longer get to C via that link and will remove it from its table. Before it can send any updates it's possible that it will receive an update from A which will be advertising that it can get to C at a cost of 2. B can get to A at a cost of 1, so it will update a route to C via A at a cost of 3. A will then receive updates from B later and update its cost to 4. They will then go on feeding each other bad information toward infinity which is called as **Count to Infinity problem**.

LINK STATE ROUTING

In Link State Routing, each router shares the knowledge of its neighbourhood with every other router in the internet, ie. Instead of sending the entire routing table, a router sends information about its neighbours only.

Each router sends this information to every router by means of a process called flooding. Each router sends out information about the neighbours when there is a change.

Link-state is also known as **shortest path first** algorithm,



Concept of Link State Routing

Link State Packet

When a router floods the network with information about its neighbourhood, it is said to be advertising. The basis of this advertising is a short packet called a link state packet (**LSP**).

An LSP usually contains four fields: the ID of the advertiser, the ID of the destination network, the cost, and the ID of the neighbour router.

The structure of a LSP is shown in Table.

| Advertiser | Network | Cost | Neighbor |
|------------|---------|-------|----------|
| | | | |
| | | | |
| | | | |

How Link State Routing Operates

The idea behind link state routing is simple and can be stated in five parts. Each router must do the following:

1) Neighbour discovery

The Router has to discover its neighbors and learn their network addresses. As a router is booted, its first task is to learn who its neighbors are. The Router does this by sending a special HELLO packet on each point-to-point line. The router on the other end is expected to send a reply disclosing its identity.

2) Measure Cost

Another job that a router needs to perform is to measure the cost to each of its neighbors. The most common way to determine this is to send over the line a special ECHO packet that the other side is required to send back immediately. By measuring the round-trip time and dividing it by two, the sending router can get a reasonable estimate of the cost.

3) Building link state packets

After collecting the information needed for the exchange, the next step for each router is to build a link state packet containing all the data. This packet starts with the identity of the sender, followed by a sequence number and age, and a list of neighbours. For each neighbour, the cost to that neighbour is given.

4) Distribute the packets

Next step in algorithm is distributing the link state packet. The fundamental concept here is flooding to distribute the packets. But to keep the number of packets flowing in the subnet under control, each packet contains a sequence number that is incremented for each new packet delivered. When a new link state packet arrives, it is checked

against the list of packets already saved by a router. It is discarded in case the packet is old; otherwise it is forwarded on all lines except the one it arrived on. A router discards an obsolete packet (i.e., with a lower sequence) in case it has seen the packet with a highest sequence number.

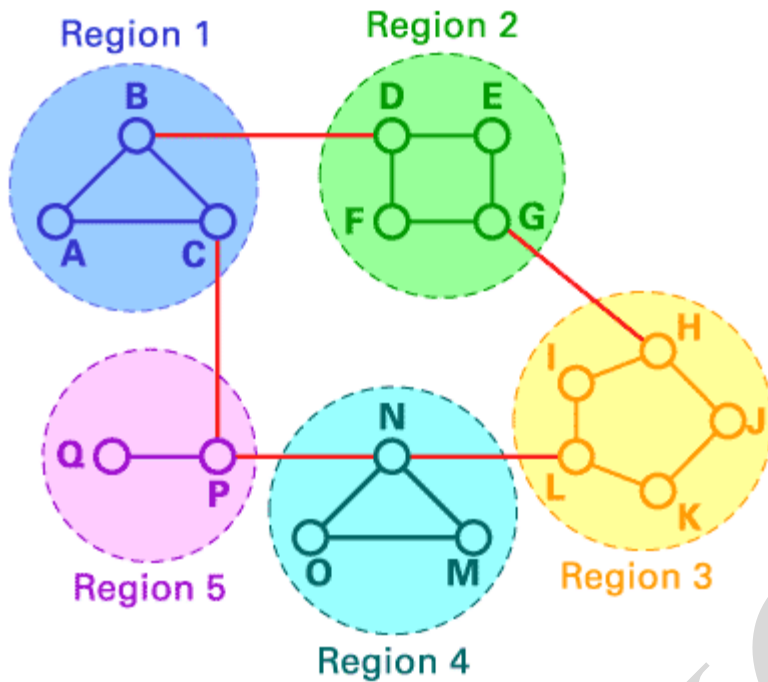
And, every router receives the LSP and puts the information into a Link State Database.

5) Compute shortest path

After accumulating all link state packets, a router constructs the entire subnet graph because every link is represented. Now, an algorithm like Dijkstra's algorithm is run locally to construct the shortest path to all possible destinations. The results of this algorithm is installed in the routing tables, and normal operation resumed.

HIERARCHICAL ROUTING

In both link state and distance vector algorithms, every router has to save some information about other routers. When the network size grows, the number of routers in the network increases. Consequently, the size of routing tables increases, as well, and routers cannot handle network traffic as efficiently. We use hierarchical routing to overcome this problem. In hierarchical routing, routers are classified in groups known as **regions**. Each router has only the information about the routers in its own region and has no information about routers in other regions. So routers just save one record in their table for every other region.



| Destination | Line | Weight |
|-------------|------|--------|
| A | - | - |
| B | B | 1 |
| C | C | 1 |
| Region 2 | | |
| Region3 | | |
| Region 4 | | |
| Region5 | | |

MULTICAST ROUTING

In many cases, you need to send same data to multiple clients at the same time. In such cases i.e., for sending a message to a group of users

(clients), we use another technique known as multicasting. The routing algorithm used for multicasting, is called multicast routing.

Centralized vs. Distributed Routing



- **Centralized Routing**

- All routes determined by a central node
- All state information sent to central node
- Problems adapting to frequent topology changes
- Does not scale

- **Distributed Routing**

- Routes determined by routers using distributed algorithm
- State information exchanged by routers
- Adapts to topology and other changes
- Scales

Centralized vs. Distributed Routing Algorithms

Centralized:

- A centralized route server collects routing information and network topology, makes route selection decisions, then distributes them to routers

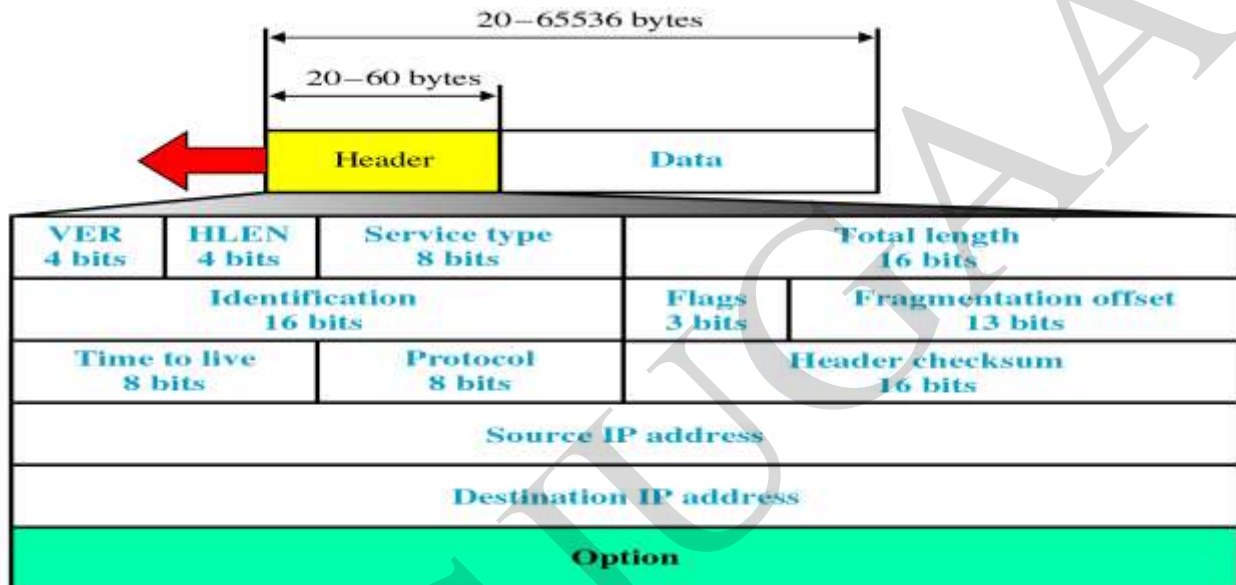
Distributed:

- Routers **cooperate** using a distributed protocol
 - to create **mutually consistent** routing tables
- Two standard **distributed** routing algorithms
 - Link State (LS) routing
 - Distance Vector (DV) routing

IP Datagram

IP transports data in packets called datagrams. A datagram is a variable length packet consisting of 2 parts: header and data.

The header can be from 20 to 60 bytes and contains information essential for routing and delivery.

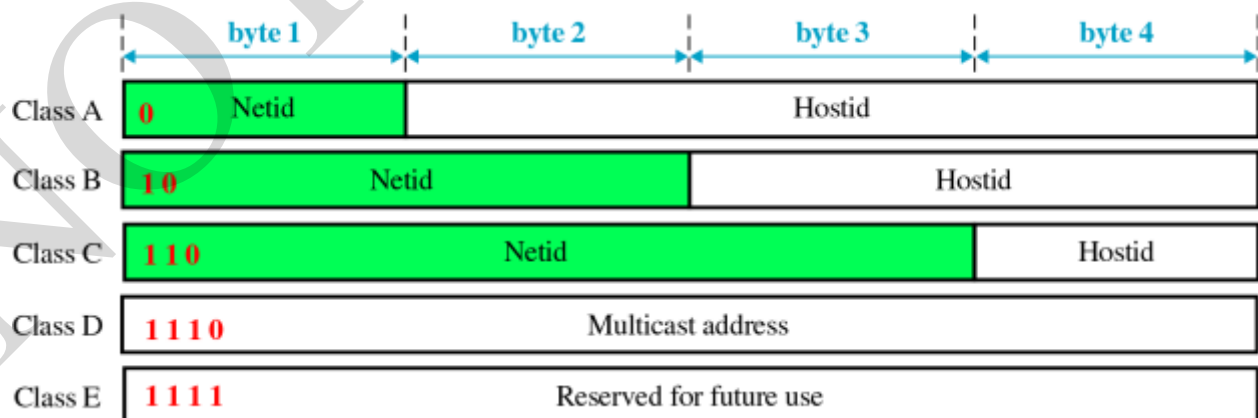


The description of the fields are as follows:

1. **Version (VER)**: The first field defines the version number of the IP.
2. **Header Length (HLEN)**: It defines the length of the header.
3. **Service Type**: This field defines how the datagram should be handled. It includes bits that specify the priority of the datagram.
4. **Total Length**: This field defines the total length of the IP datagram.
5. **Identification**: This field is used in fragmentation. A datagram when passes through diff networks can be divided into fragments, and if done, it is identified with a sequence number in this field.
6. **Flags**: The bits in this field deal with fragmentation.

7. **Fragmentation offset:** This field is a pointer that shows the offset of the data in the original datagram.
8. **Time to Live:** It defines the number of hops a datagram can travel before it is discarded.
9. **Protocol:** It defines which upper layer protocol are encapsulated in the datagram
10. **Header checksum:** This field defines the checksum bits for the header.
11. **Source address:** defines the IP address of the sender.
12. **Destination address:** defines the IP address of the destination.
13. **Options:** This field is optional, carry fields that control routing, timing, management.

Internet Addressing Classes



| | From | To |
|----------------|------------------------------------|--|
| Class A | 0 .0.0.0 Netid Hostid | 127 .255.255.255 Netid Hostid |
| Class B | 128 .0.0.0 Netid Hostid | 191 .255.255.255 Netid Hostid |
| Class C | 192 .0.0.0 Netid Hostid | 223 .255.255.255 Netid Hostid |
| Class D | 224 .0.0.0 Group address | 239 .255.255.255 Group address |
| Class E | 240 .0.0.0 Undefined | 255 .255.255.255 Undefined |

Class Ranges of Internet Addresses