# MODULE 4

# Wired LANs: ETHERNET

- LAN is used for a limited geographical area.
- To solve the issues related to media sharing was handles using CSMA/CD approach.
- Number of technologies other than Ethernet came into existence. (Eg: ATM LAN)
- Only Ethernet survived.
- It could evolve with the demand for higher transmission rates.

# Wired LANs: ETHERNET

IEEE Project 802

- In 1985
- The computer society of IEEE.
- Goal was to set standards to enable intercommunication among equipment from a variety of manufacturers.

# Wired LANs: ETHERNET

IEEE Project 802

- IEEE has subdivided the DLL into 2 sub-layers.
- ***Logical Link Control and Media Access Control.***

# Wired LANs: ETHERNET

**Logical Link Control (LLC)**

- In IEEE project 802, flow control, error control, and part of the framing duties are collected into one sub-layer called ***Logical Link Control.***
- ***Framing is handled in both LLC and MAC.***
- The LLC provides a single link layer control protocol for all IEEE LANs.
- LLC protocol can provide interconnectivity between different LANs because it makes the MAC sub-layer transparent.

# Wired LANs: ETHERNET

**Media Access Control (MAC)**

- It was a specific access method for LAN.
- It defines CSMA/CD, and defines Token Passing method.
- Framing function is also handled.

# Wired LANs: ETHERNET

**Four Generations**

1. Standard Ethernet(10 Mbps)
2. Fast Ethernet (100 Mbps)
3. Gigabit Ethernet (1 Gbps)
4. 10 Gigabit Ethernet (10 Gbps)

# STANDARD ETHERNET

***Data rate of 10 Mbps.***

*Characteristics*

- Connectionless and unreliable service
  - It provides a connectionless service.
  - No connection establishment and termination phase.
  - Sender sends a frame whenever it has it.
  - The receiver may or may not be ready for it.

# STANDARD ETHERNET

***Data rate of 10 Mbps.***

*Characteristics*

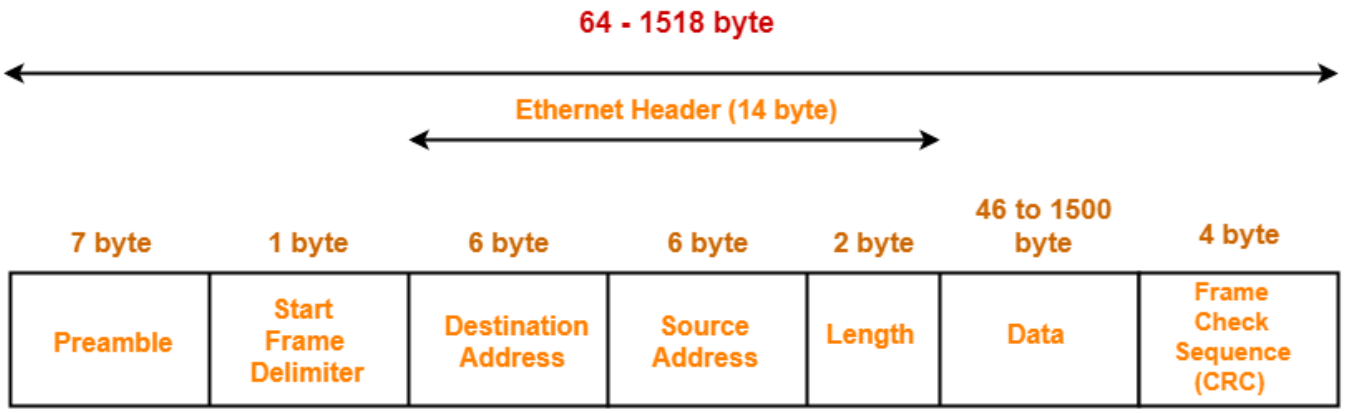
- Connectionless and unreliable service
  - Ethernet is unreliable like IP and UDP.
  - If a frame is corrupted during transmission and the receiver finds out about the corruption, which has a high probability of happening because of the CRC 32, the receiver drops the frame silently.
  - It is the duty of the high level protocols to find out about it.

# STANDARD ETHERNET

***Data rate of 10 Mbps.***

*Characteristics*

- **Frame Format**
  - The Ethernet contains 7 fields.




IEEE 802.3 Ethernet Frame Format

# STANDARD ETHERNET

***Frame format***

- *Preamble*
  - Field contains 7 bytes (56 bits) of alternating 0s and 1s.
  - Actually added at the physical layer and is not part of the frame.
  - **It alerts the receiving system to the coming frame.**
  - **Enables the frame to synchronize its clock if its out of synchronization.**

# STANDARD ETHERNET

***Frame format***

- *Start Frame Delimiter (SFD)*
  - This field signals the beginning of the frame.
  - 1 byte (10101011)
  - The SFD warns the stations, that this is the last chance for synchronization.
  - The last 2 bits alerts the receiver that the next field is the destination address.
  - Also added at the physical layer.

# STANDARD ETHERNET

***Frame format***

- *Destination Address (DA)*
  - This field is 6 bytes (48 bits)
  - Contains the link layer address of the destination.

# STANDARD ETHERNET

***Frame format***

- *Source Address (SA)*
  - This field is having 6 bytes.
  - Link layer address of the sender of the packet.

# STANDARD ETHERNET

***Frame format***

- *Type*
  - Defines the upper layer protocol whose packet is encapsulated in the frame.
  - The protocol can be IP, ARP and so on.

# STANDARD ETHERNET

***Frame format***

- *Data*
  - This field carries data encapsulated from the upper layer protocols.
  - Min: 46 bytes, Max: 1500 bytes.
  - If the data coming is more than 1500 bytes, it should be fragmented and encapsulated in more than one frame.
  - If it is less than 46 bytes, it needs to be padded with extra 0s.
  - Padding will be removed by the upper layer.
  - The upper layer protocol needs to know the length of the data.

# STANDARD ETHERNET

***Frame format***

- *CRC*
  - Contains error correction detection information.
  - If the receiver calculates the CRC and finds that it is not zero, it discards the frame.

# STANDARD ETHERNET

***Frame Length***

- There are restrictions on both minimum and maximum length.
- The minimum is required for the correct operation of CSMA/CD.
- Minimum length: 512 bits or 64 bytes.
- Part of this length is the header and trailer.

# STANDARD ETHERNET

***Frame Length***

- If we count 18 bytes of header and trailer, then the minimum length of data from the upper layer is 64 – 18 = 46 bytes.
- If the upper layer packet is less than 46 bytes, padding is added to make up the difference.
- The standard defines the maximum length of the frame as 1518 bytes.
- If we subtract the 18 bytes of header and trailer, the maximum length of the payload is 1500 bytes.

# STANDARD ETHERNET

## Addressing

- Each station on an Ethernet network has its own network interface card (NIC).
- NIC is fixed inside the station and provides a link-layer address.
- The Ethernet address is 6 bytes (48 bits).
- Normally written in hexadecimal notation.

***4A:30:10:21:10:1A***

# STANDARD ETHERNET

## Addressing

*Transmission of address bits:*

- The transmission is left to right, byte by byte.
- The LSB is sent first, and the MSB sent last.

*Unicast, Multicast and Broadcast addresses:*

- A source address is always a *unicast address*. Because the frame comes from only one station.

# STANDARD ETHERNET

## Addressing

*Unicast, Multicast and Broadcast addresses:*

- The destination address however can be, unicast, multicast or broadcast.
- If the LSB of the first byte in a destination address is 0, the address is unicast. Otherwise it is multicast.
- For broadcast, it will be forty-eight 1s.

# STANDARD ETHERNET

## Addressing

- *Distinguish between Unicast, Multicast and Broadcast Transmission:*
  - Standard Ethernet uses a coaxial cable or a set of twisted pair cables with a HUB.
  - The transmission in Ethernet is always broadcast, no matter if the intention is unicast, multicast or broadcast.

# STANDARD ETHERNET

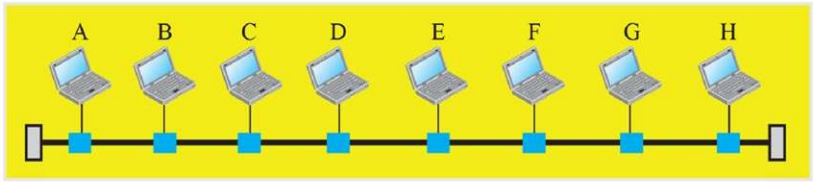
## Addressing

- *Distinguish between Unicast, Multicast and Broadcast Transmission:*
  - In a unicast transmission, all stations will receive the frame, the intended recipient keeps and handles the frame; the rest discard it.
  - In a multicast transmission, all stations will receive the frame, the stations that are members of the group keep and handle it; the rest discard it.
  - In a broadcast transmission, all stations (except the sender) will receive the frame and all stations (except the sender) keep and handle it.

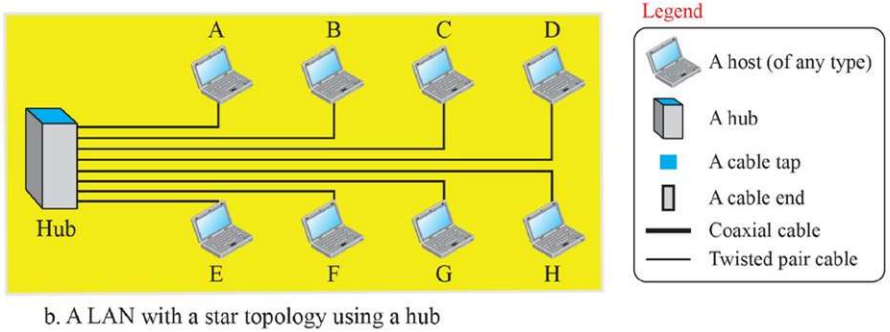
# STANDARD ETHERNET

## Access Method

- The standard Ethernet chooses CSMA/CD with 1-persistent method.



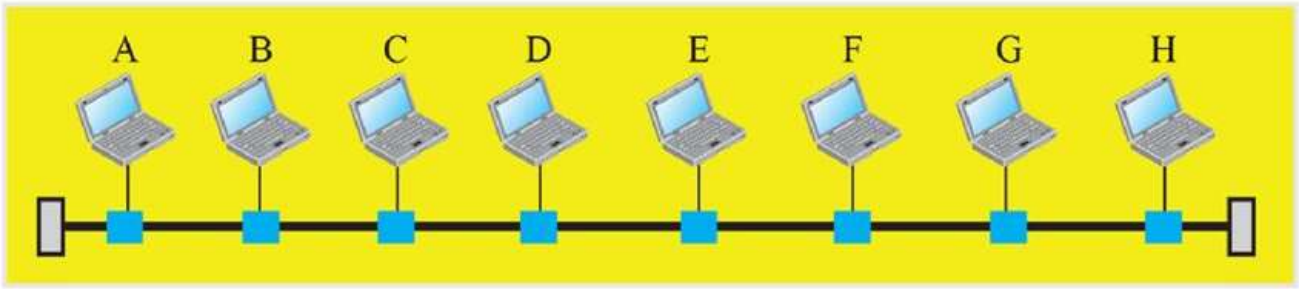

a. A LAN with a bus topology using a coaxial cable



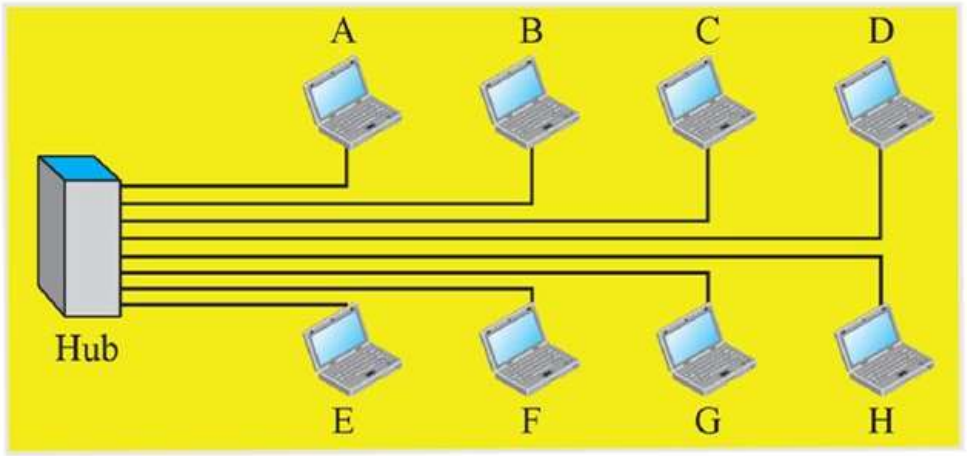

b. A LAN with a star topology using a hub

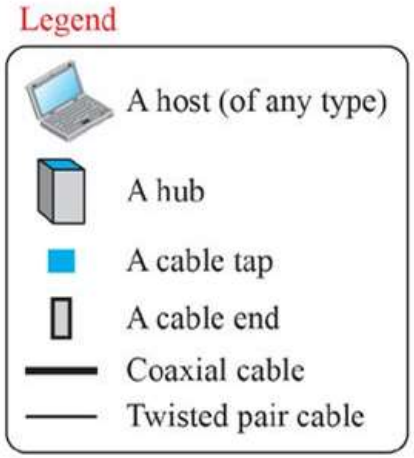
# STANDARD ETHERNET

## Access Method




a. A LAN with a bus topology using a coaxial cable




b. A LAN with a star topology using a hub

# UNICAST ROUTING

- If a datagram is destined for only one destination ***(one-to-one delivery)*** , we have unicast routing.
- Routing Tables.
- A packet is routed, hop by hop, from its source to destination by the help of forwarding tables.

# UNICAST ROUTING

- To find the best route, an internet can be modelled as a graph.
- A set of nodes and edges that connect the nodes.
- Router – As a node
- Each network between two nodes – As an edge.

# UNICAST ROUTING

## Least cost routing

Total cost for the route is the least cost among all possible routes.

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.

# UNICAST ROUTING

## Least-Cost Trees

There are ***N*** routers in an internet.

There are ***(N – 1)*** least cost paths from each router to any other router.

Means we need ***N x (N – 1)*** least cost paths for the whole internet.

Better way to find the least cost path is, ***Least-Cost Trees.***

# UNICAST ROUTING

## Least-Cost Trees

A tree with source router as the root that spans the whole graph (visits all other nodes) and in which the path between the route and any other node is the shortest.

# ROUTING ALGORITHMS

***Distance Vector Routing***

A **distance vector routing algorithm operates by having each router maintain** a table (i.e., a vector) giving the best known distance to each destination and which link to use to get there.

These tables are updated by exchanging information with the neighbours.

Eventually, every router knows the best link to reach each destination.

# ROUTING ALGORITHMS

***Bellman-Ford Equation***

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, etc.).

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

Where $D_{ij}$ is the shortest distance;

$C_{ij}$ is the cost between the nodes i and j

# ROUTING ALGORITHMS

***Bellman-Ford Equation***

In distance vector routing, each router maintains a routing table indexed by, and containing one entry for each router in the network.

This entry has two parts:

1. The preferred outgoing line to use for that destination.
2. An estimate of the distance to that destination.

The distance might be measured as the number of hops or using another metric, as we discussed for computing shortest paths.

# ROUTING ALGORITHMS

- The router is assumed to know the ‘‘distance’’ to each of its neighbours.
- If the metric is hops, the distance is just one hop.
- If the metric is propagation delay, the router can measure it directly with special ECHO packets that the receiver just timestamps and sends back as fast as it can.

# ROUTING ALGORITHMS

***Bellman-Ford Equation***

Information kept by DV router

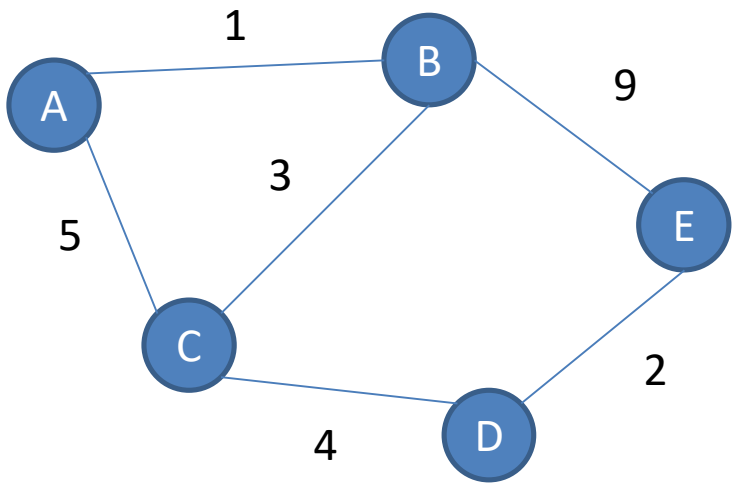
- Each router has an ID
- Associated with each link connected to a router, there is a link cost (static or dynamic).
- Intermediate hops

Distance Vector Table Initialization

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

# ROUTING ALGORITHMS

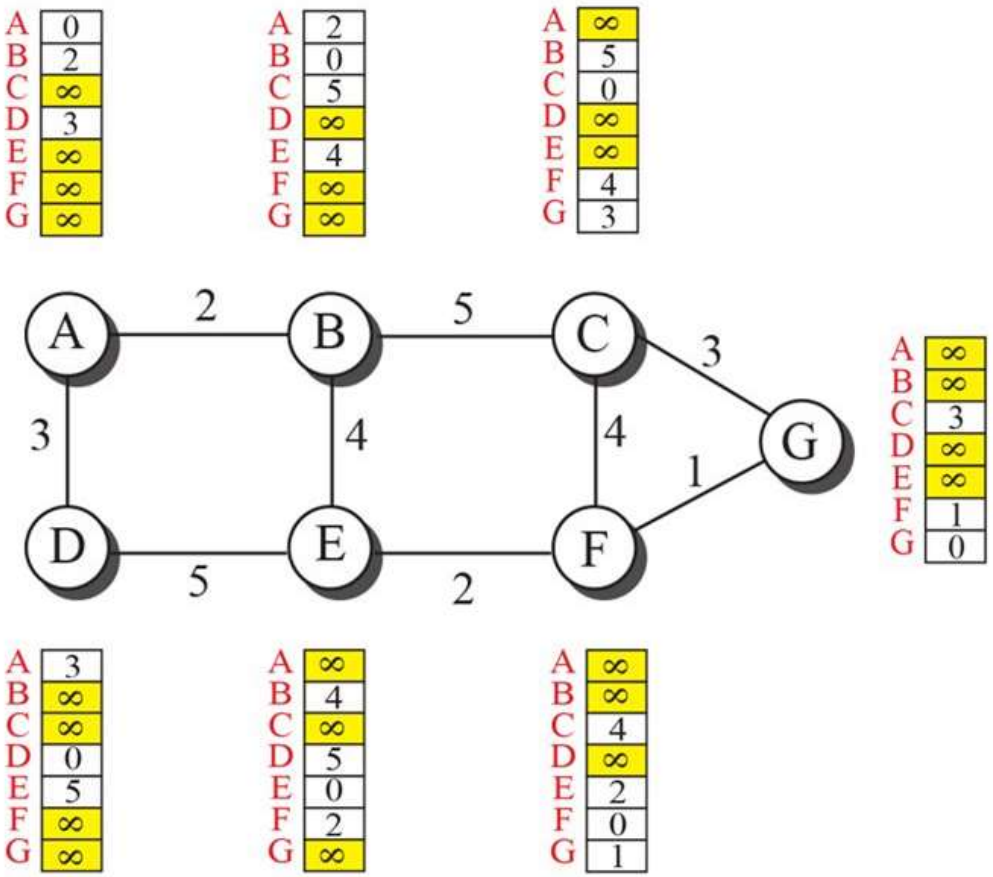
## ***Routing Table***



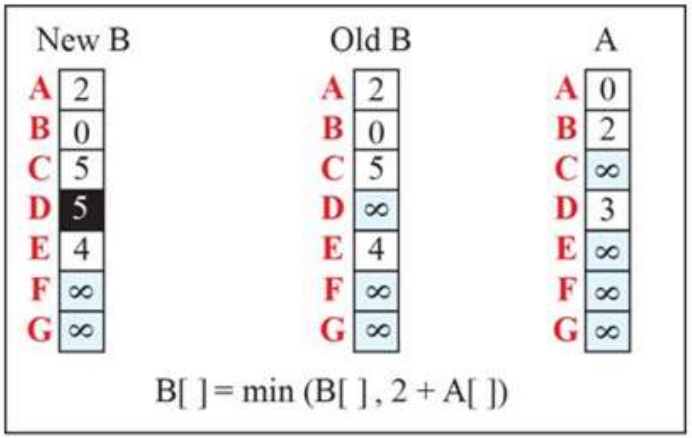

| Destination | Cost | Next Hop |
|---|---|---|
| A | 1 | A |
| C | 3 | C |
| E | 9 | E |

**Initial Routing Table of B**

# ROUTING ALGORITHMS

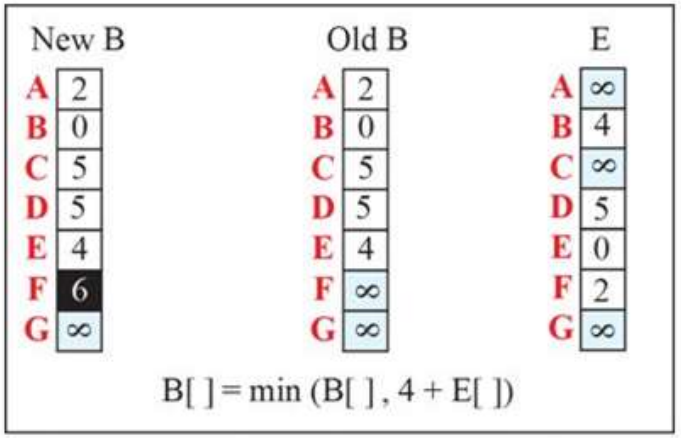


# ROUTING ALGORITHMS




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

**Note:**
X[ ]: the whole vector




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

# LINK-STATE ROUTING

Distance vector routing was used in the ARPANET until 1979, when it was replaced by link state routing.

- The primary problem that caused its demise was that the algorithm often took too long to converge after the network topology changed.
- Variants of link state routing called IS-IS and OSPF are the routing algorithms that are most widely used inside large networks and the Internet today.

# LINK-STATE ROUTING

The idea behind link state routing is fairly simple and can be stated as five parts.

Each router must do the following things to make it work:

1. Discover its neighbors and learn their network addresses.
2. Set the distance or cost metric to each of its neighbors.
3. Construct a packet telling all it has just learned.
4. Send this packet to and receive packets from all other routers.
5. Compute the shortest path to every other router.

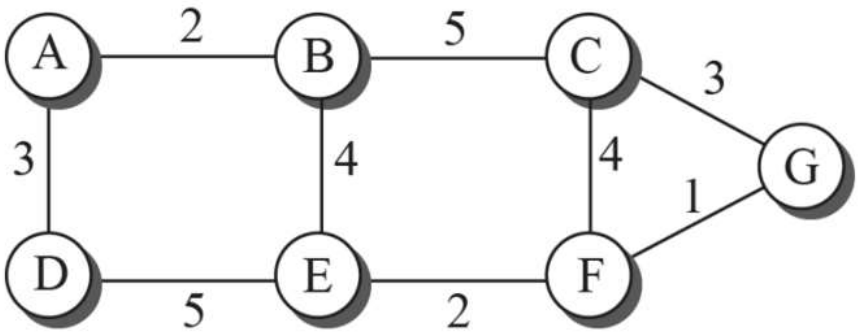
# LINK-STATE ROUTING

In effect, the complete topology is distributed to every router.

Then Dijkstra's algorithm can be run at each router to find the shortest path to every other router.

# LINK-STATE ROUTING

## LSDB (Least Cost DataBase)



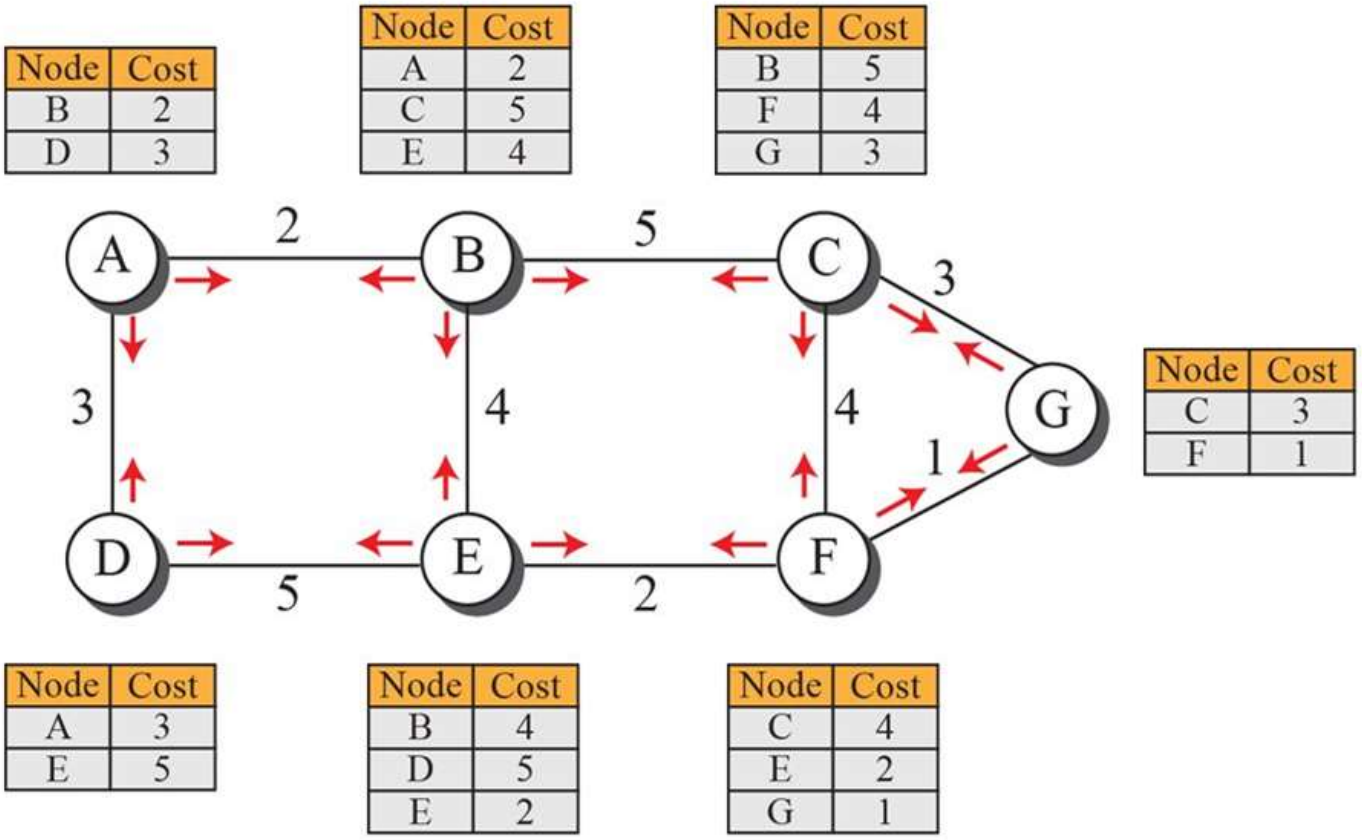

a. The weighted graph

| | 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 ROUTING

## LSDB (Least Cost DataBase)



# LINK-STATE ROUTING

**Least-Cost Trees**

Created using Dijkstra's Algorithm.

Steps:

1. The node chooses itself as the root node of the tree.
2. The node selects one node, among all nodes 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.

# PATH VECTOR ROUTING

Both previous protocols were based on least cost method.

To overcome the disadvantages of both LS and DV, Path Vector routing came in to existence.

Here, the best route is selected by the source using the policy it imposes on the route.

That means, the source can control the path.

# PATH VECTOR ROUTING

***Spanning Trees***

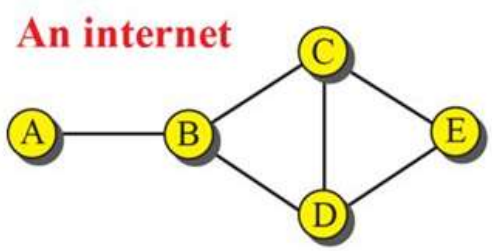
The path is determined by the best spanning tree.

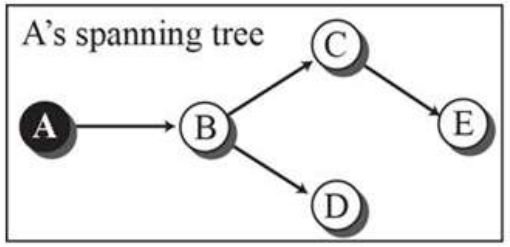
It is not the least cost tree.

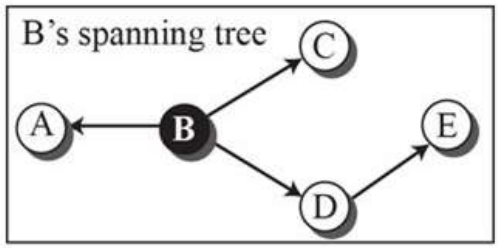
It is the tree determined by the source when it imposes its own policy.

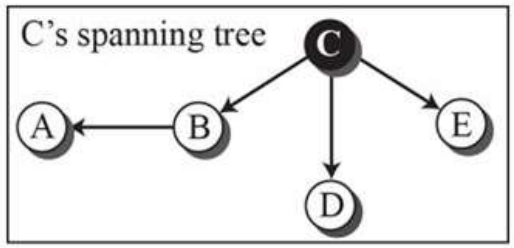
# PATH VECTOR ROUTING

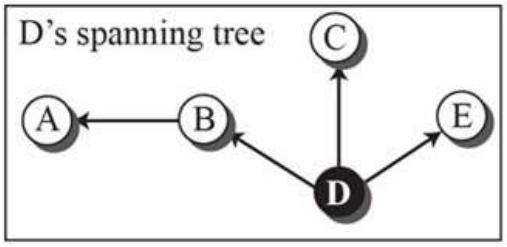
## ***Spanning Trees***

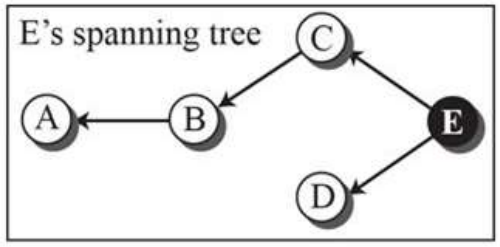


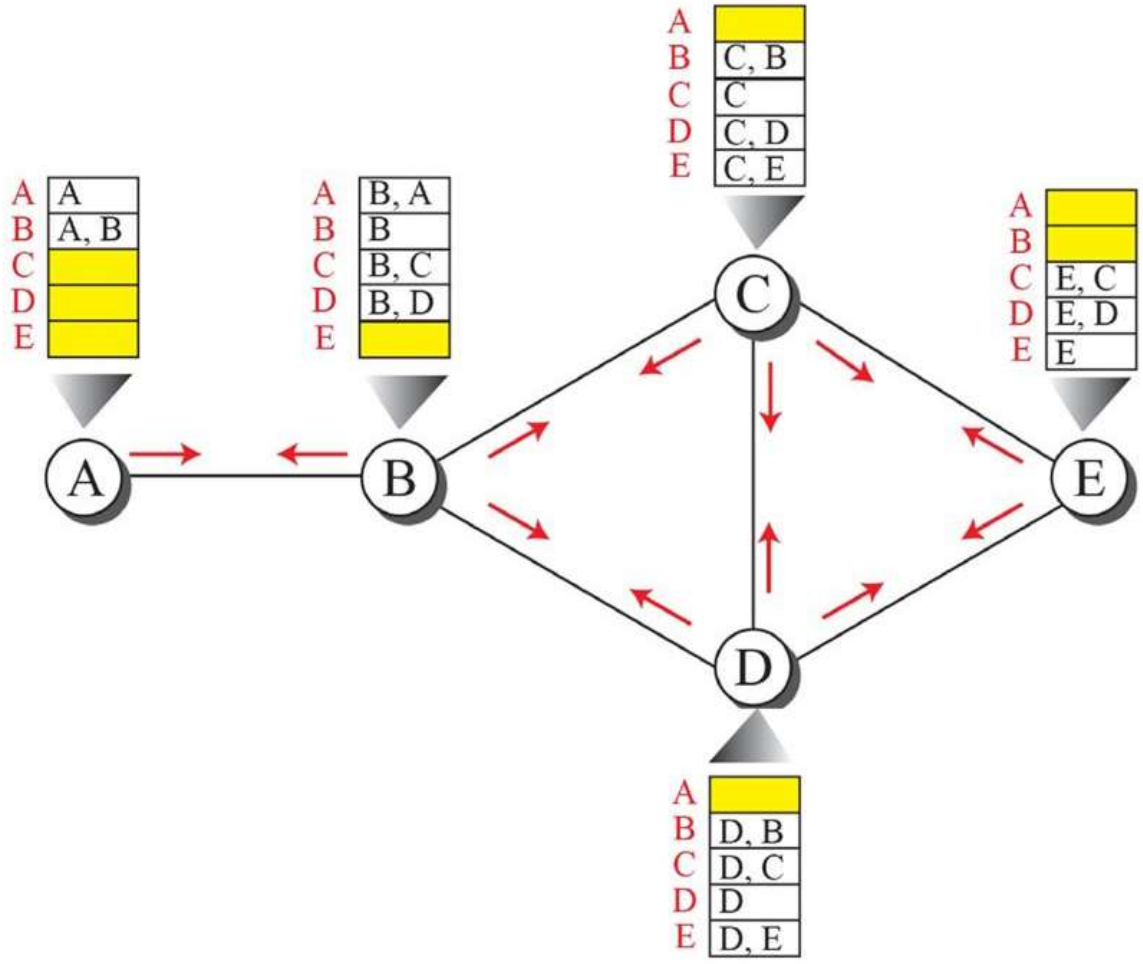



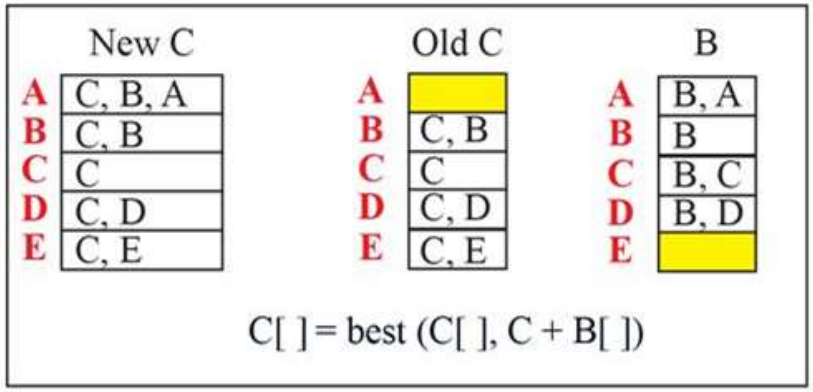



# PATH VECTOR ROUTING

## ***Spanning Trees***

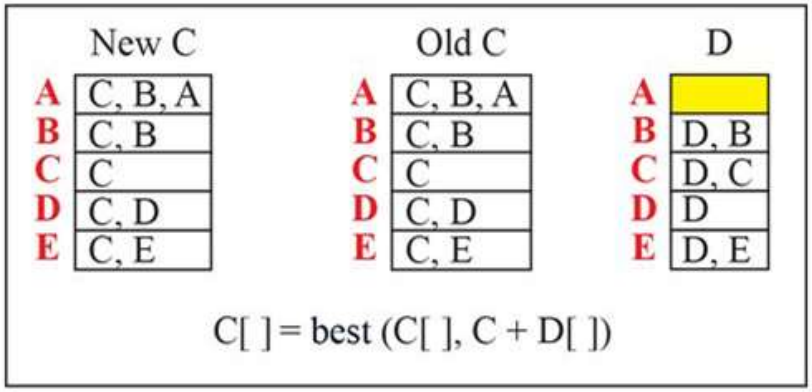


# PATH VECTOR ROUTING




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

**Note:**
**X [ ]: vector X**
**Y: node Y**




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

# IPV4 ADDRESSES

- **IP Address:** It is an identifier, which is used to identify the connection of each device to the internet.
- It's a 32 bit address that uniquely and universally defines the connection of a host or a router to the internet.
- It is the address of the connection, not the host or the router.
- If the device is moved to another network, the IP address may be changed.

# IPV4 ADDRESSES

- **IPV4 address** are unique in the sense, that each address defines one and only one, connection to the internet.
- If a device has two connections to the internet, via two networks, it has two IPV4 addresses.
- IPV4 addresses are universal in the sense that the addressing system must be accepted by any host that wants to be connected to the internet.

# IPV4 ADDRESSES

**Address Space**

A protocol like IPV4 that defines addresses has an address space.

An address space is the total number of addresses used by the protocol.

If a protocol uses ***b*** bits to define an address, the address space is $2^b$ because each bit can have two different values (0 or 1).

IPV4 uses 32 bit addresses, which means that the address space is $2^{32}$ ***or 4,294,967,296*** (more than four billion).

# IPV4 ADDRESSES

**Notation**

There are three common notations to show an IPV4 address:

- Binary notation( base 2 )
- Dotted-decimal notation (base 256)
- Hexadecimal notation (base 16)

# IPV4 ADDRESSES

**Notation**

In binary notation – it is displayed as 32 bits.

Each octet is referred to as a byte.

To make the IPV4 more compact and easier to read, it is usually written in decimal form with a decimal point separating the bytes.

This format is known as dotted-decimal notation.

# IPV4 ADDRESSES

**Notation**

Note that because each byte is only 8 bits (octet), each number in the dotted-decimal notation is between 0 and 255.

We sometimes see an IPV4 in hexadecimal notation.

Each hexadecimal digit is having 4 bits.

This means that a 32 bit address has 8 hexadecimal digits.

This notation is often used in network programming.

# IPV4 ADDRESSES

**Notation**

Note that because each byte is only 8 bits (octet), each number in the dotted-decimal notation is between 0 and 255.

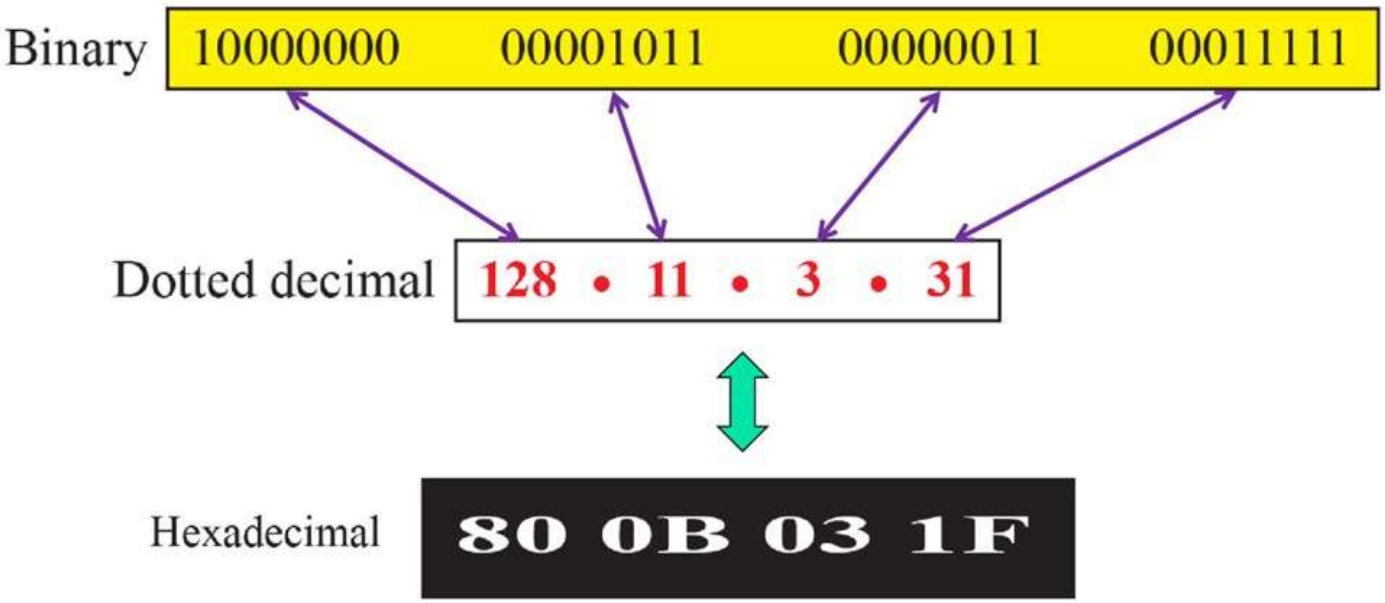
We sometimes see an IPV4 in hexadecimal notation.

Each hexadecimal digit is having 4 bits.

This means that a 32 bit address has 8 hexadecimal digits.

This notation is often used in network programming.

# IPV4 ADDRESSES



# IPV4 ADDRESSES

**Hierarchy in addressing**

The addressing system is hierarchical.

A 32 bit IPV4 address is divided into two parts:

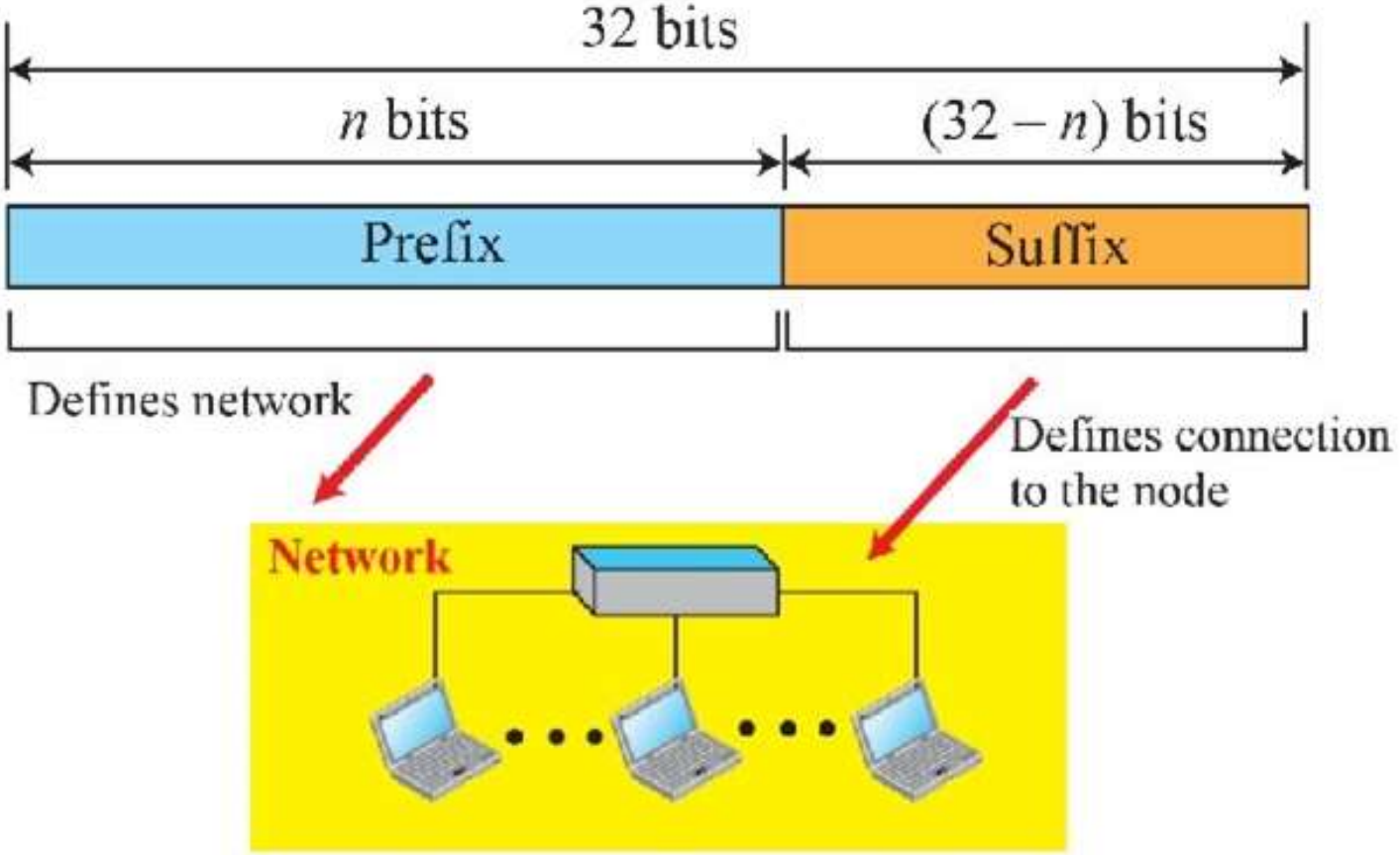
***Prefix and suffix***

Prefix defines the network and suffix defines the node.

The network identifier in the IPV4 was first designed as a fixed-length prefix, and is referred to as classful addressing.

The new scheme, which is referred to as classless addressing, uses a variable length network prefix.

# IPV4 ADDRESSES



# IPV4 ADDRESSES

## CLASSFUL ADDRESSING

When internet started, the IPV4 address was designed with a fixed length prefix.

But to accommodate both small and large networks, three fixed-length prefixes were designed instead of one. (n = 8, n = 16, and n = 24).

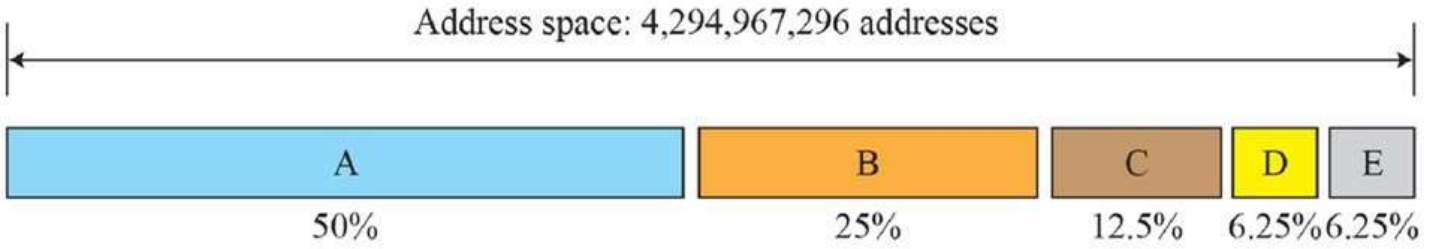
The whole address space was divided into five classes.

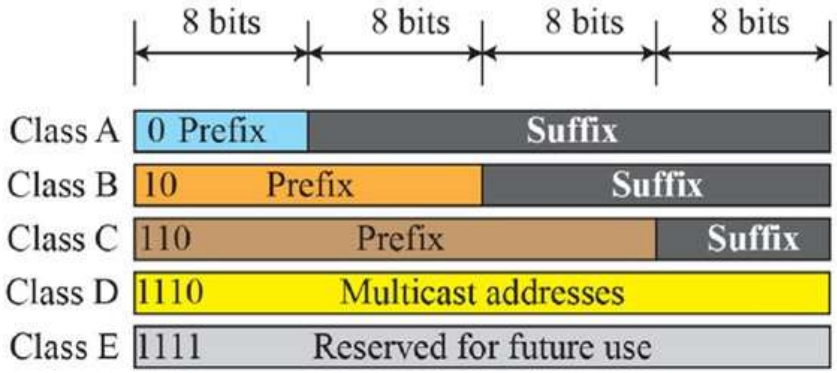
CLASS A, CLASS B, CLASS C, CLASS D and CLASS E

This scheme is referred to as classful addressing.

# IPV4 ADDRESSES

## CLASSFUL ADDRESSING




| Class | Prefixes | First byte |
|---|---|---|
| A | $n = 8$ bits | 0 to 127 |
| B | $n = 16$ bits | 128 to 191 |
| C | $n = 24$ bits | 192 to 223 |
| D | Not applicable | 224 to 239 |
| E | Not applicable | 240 to 255 |

# IPV4 ADDRESSES

## CLASSLESS ADDRESSING

Subnetting and supernetting in classful addressing did not really solve the address depletion problem.

With the growth of the internet, it was clear that a larger address space was needed as a long term solution.

The larger address space, however, requires that the length of IP addresses also be increased.

It means that the format of the Ip Packets needs to be changed.

# IPV4 ADDRESSES

## CLASSLESS ADDRESSING

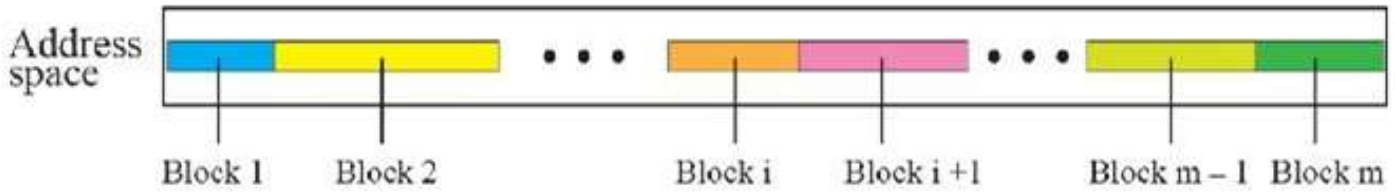
The short term solution for solving this problem was called as classless addressing.

In other words, the class privilege was removed from the distribution to compensate for the address depletion.

In classless addressing variable length blocks are used that belong to no classes.

# IPV4 ADDRESSES

## CLASSLESS ADDRESSING



# INTERNET PROTOCOL (IP)

The network layer in version 4 can be thought of as one main protocol and three auxiliary ones.

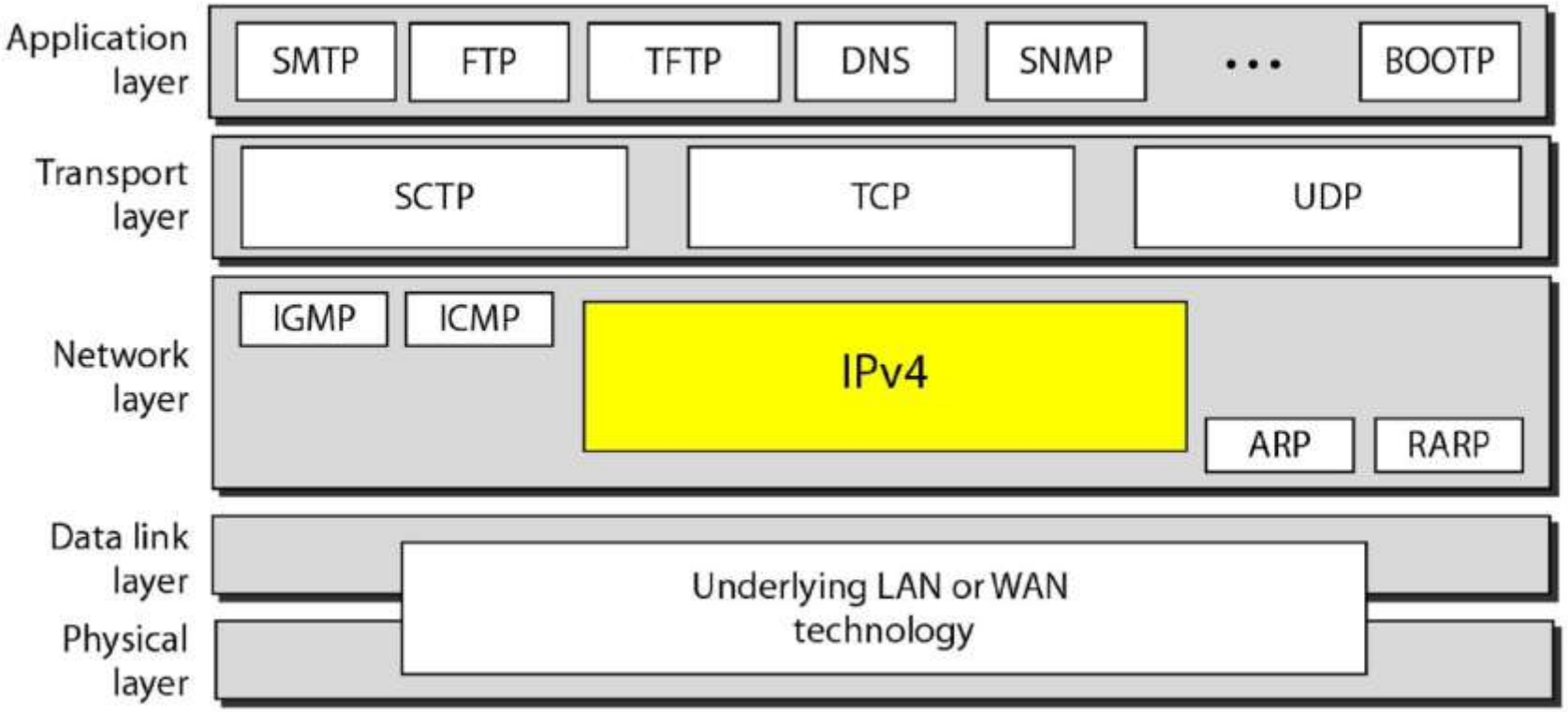
the main protocol, internet protocol version 4 (IPV4), is responsible for packetizing, forwarding and delivery of a packet at the network layer.

The ICMPv4 helps IPv4 to handle some errors that may occur in the network layer delivery.

The IGMP is used to help IPv4 in multicasting.

The ARP is used to glue the network and data-link layers in mapping network layer addresses to link-layer addresses.

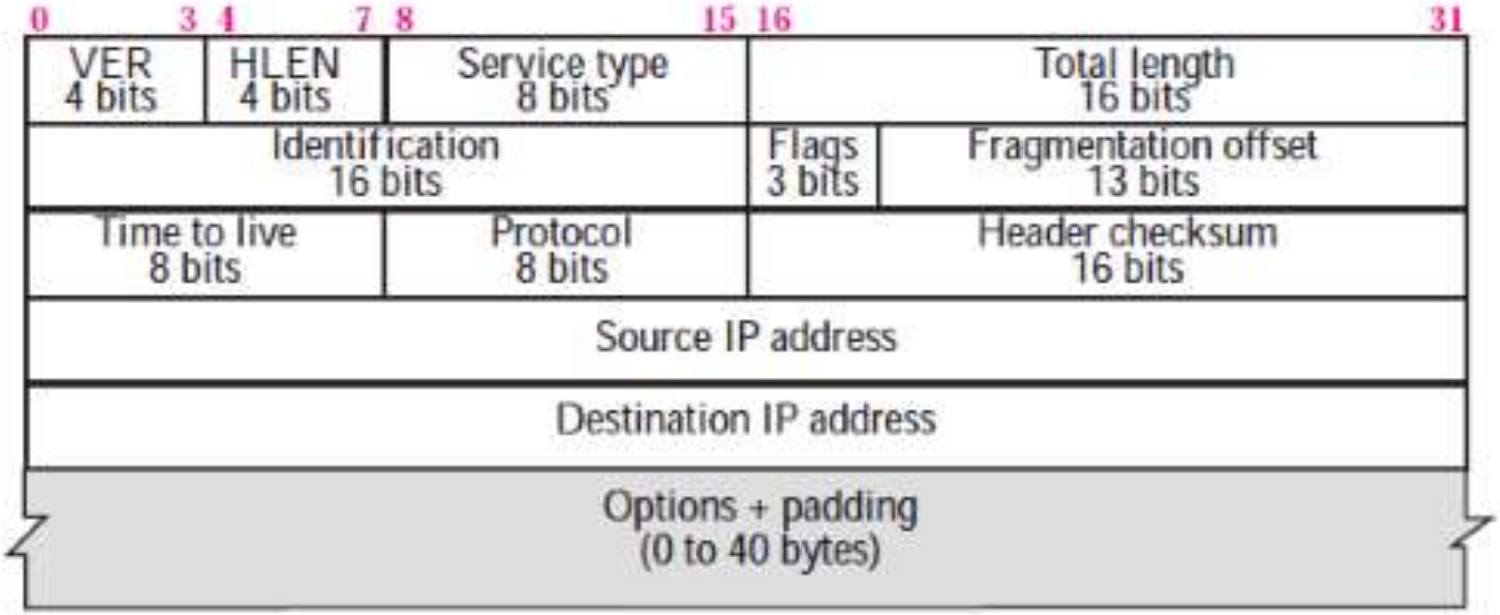
# INTERNET PROTOCOL (IP)



# INTERNET PROTOCOL (IP)

## DATAGRAM FORMAT


a. IP datagram




b. Header format

# INTERNET PROTOCOL (IP)

## SUBNETTING

A sub-network or **subnet** is a logical subdivision of an **IP** network.

In Subnetting we create multiple small manageable networks from a single large IP network.

The practice of dividing a network into two or more networks is called **subnetting.**

# INTERNET PROTOCOL (IP)

## SUBNETTING

To best utilize available addresses if we put more than 16000000 hosts in a single network, due to broadcast and collision, that network will never work.

If we put less hosts then remaining addresses will be wasted. Subnetting provides a better way to deal with this situation.

Subnetting allows us to create smaller networks from a single large network which not only fulfill our hosts' requirement but also offer several other networking benefits.

# INTERNET PROTOCOL (IP)

## SUBNETTING