UNIT III-NETWORK LAYER

Contents

- Network Layer Design issues
- store and forward packet switching
- connection less and connection oriented networks
- Routing algorithms
  - optimality principle
  - shortest path
  - Flooding
  - Distance Vector Routing
  - Count to Infinity Problem
  - Link State Routing
  - Path Vector Routing
  - Hierarchical Routing
  - Congestion control algorithms
- IP Addresses
- CIDR
- SubNetting
- SuperNetting
- IPv4
- Packet Fragmentation
- IPv6 protocol
- Transition from IPv4 to IPv6
- ARP
- RARP

Network Layer

Network layer works for the transmission of data from one host to the other located in different networks.

The functions of the Network layer are :

1.Routing: The network layer protocols determine which route is suitable from source to destination.

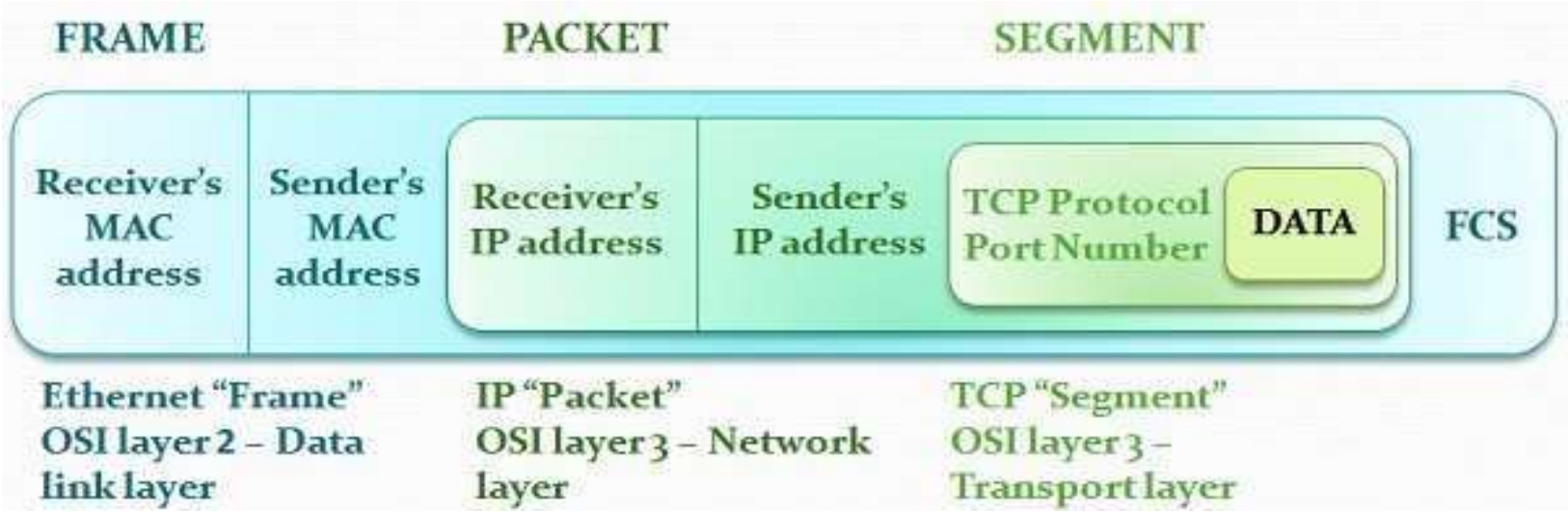
1.Logical Addressing: The sender & receiver's IP address are placed in the header by network layer. Such an address distinguishes each device uniquely and universally.

* *Segment* in Network layer is referred as **Packet**.

What is packet?

- All data sent over the Internet is broken down into smaller chunks called "packets.“

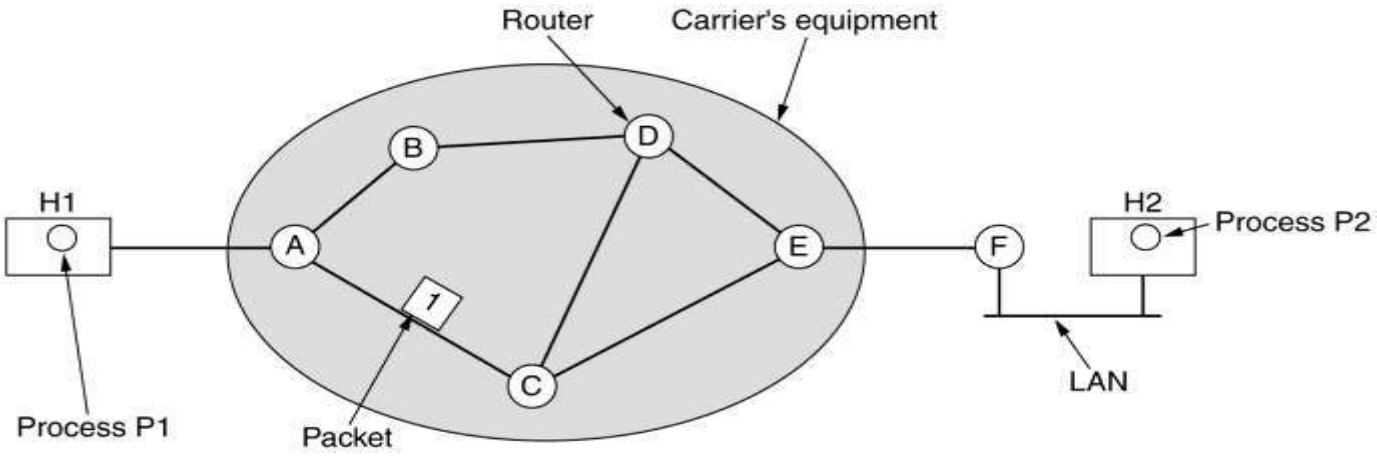
- A packet has two parts: the header, which contains senders and receivers IP address, and the body, which is the actual data being sent.



Network Layer Design Issues:

1. Store-and-Forward Packet Switching
2. Services Provided to the Transport Layer
3. Implementation of Connectionless Service
4. Implementation of Connection-Oriented Service
5. Comparison of Virtual-Circuit and Datagram Subnets

1. Store-and-Forward Packet Switching:



Network layer protocol environment

- The node which has a packet to send, delivers it to the nearest router. The packet is stored in the router until it has fully arrived and its checksum is verified for error detection.
- Once, this is done, the packet is forwarded to the next router. Since, each router needs to store the entire packet before it can forward it to the next hop, the mechanism is called store − and − forward switching.

2. Services Provided to the Transport Layer

The network layer provides services to the transport layer at the network layer/transport layer interface. The network layer services have been designed with the following goals:

1. The services should be independent of the router technology.
2. The transport layer should be shielded from the number, type, and topology of the routers present.
3. The network addresses made available to the transport layer should use a uniform numbering plan, even across LANs and WANs.

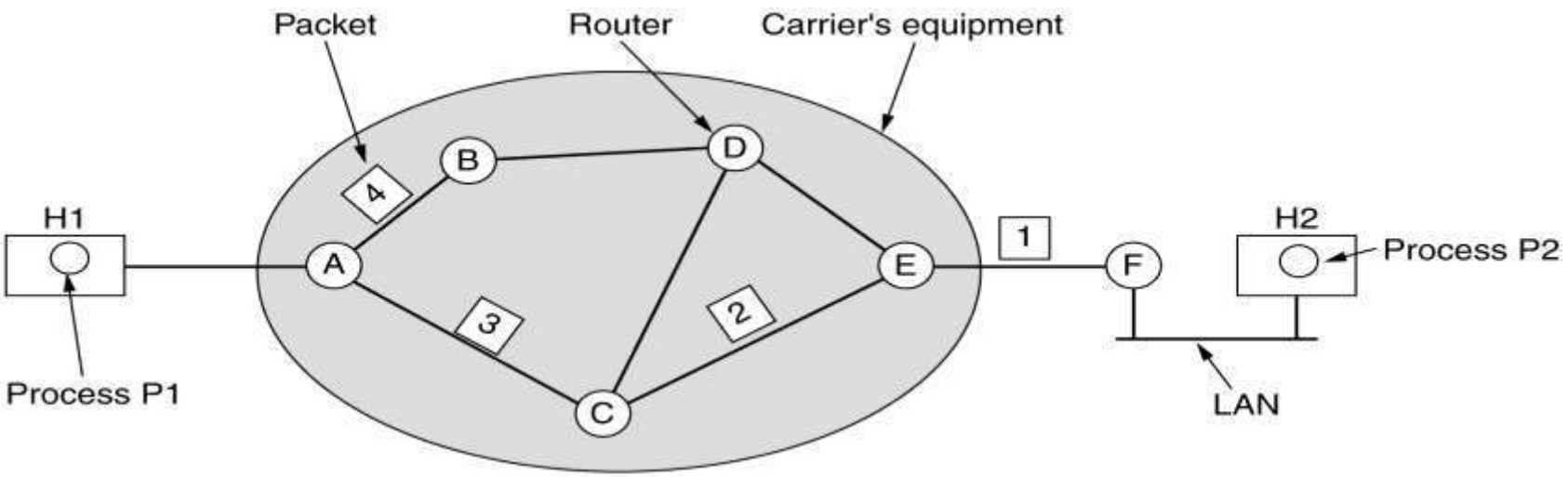
The network layer should provide connection-oriented service or connectionless service

3.Implementation of Connectionless Service

- If connectionless service is offered, packets are injected into the subnet individually and routed independently of each other.

- In this context, the packets are frequently called datagram's and the subnet is called a **datagram subnet.**

- If connection-oriented service is used, a path from the source router to the destination router must be established before any data packets can be sent. This connection is called a **VC (virtual circuit)** and the subnet is called a **virtual-circuit subnet.**



A's table

Dest.	initially	later
A	–	–
B	B	B
C	C	C
D	B	B
E	C	B
F	C	B

C's table

A	A
B	A
C	–
D	D
E	E
F	E

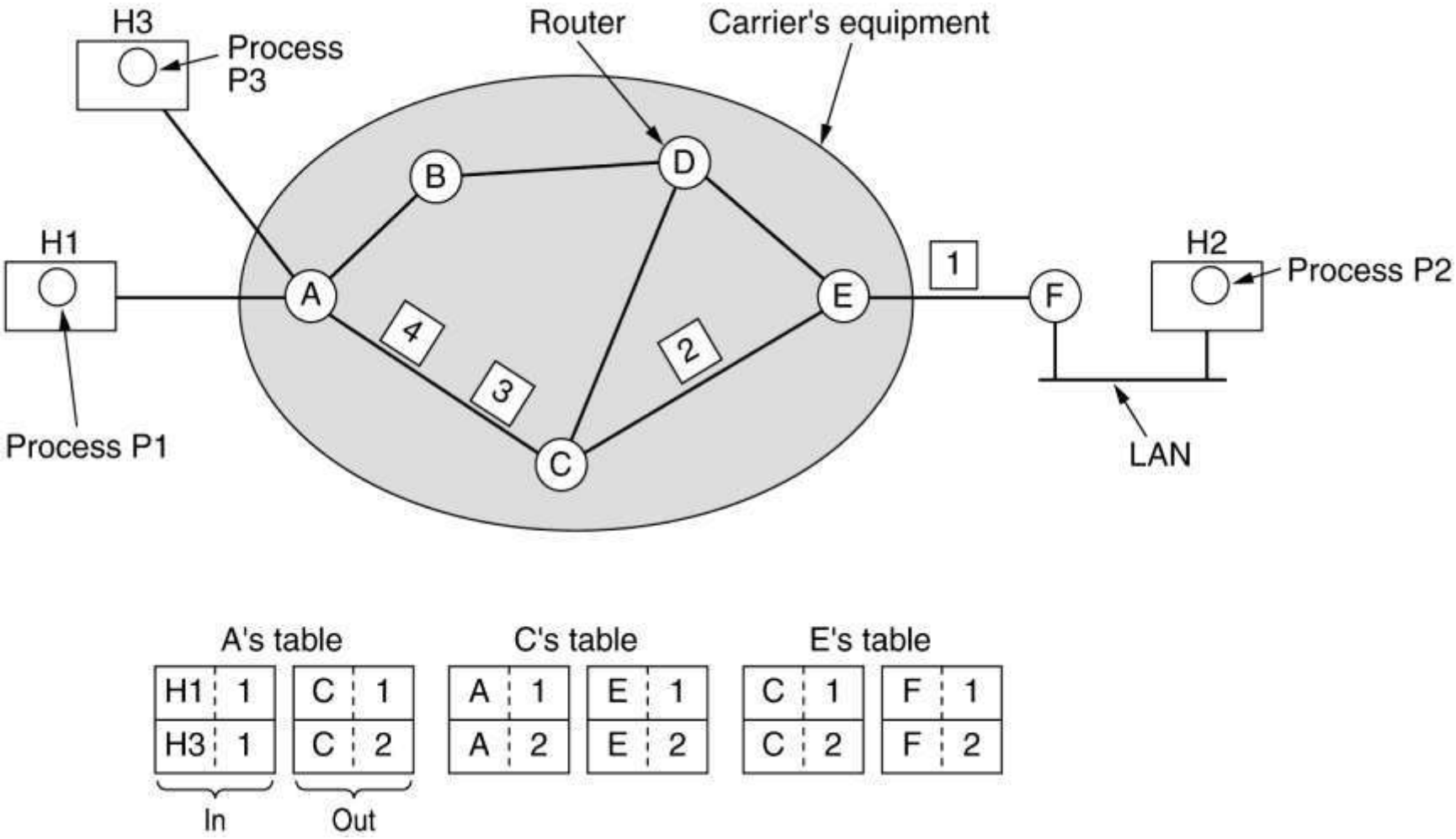
E's table

A	C
B	D
C	C
D	D
E	–
F	F

Routing within a diagram subnet.

4. Implementation of Connection-Oriented Service

- For connection-oriented service, we need a virtual-circuit subnet.
- The idea behind virtual circuits is to avoid having to choose a new route for every packet sent.
- when a connection is established, a route from the source machine to the destination machine is chosen as part of the connection setup and stored in tables inside the routers.
- When the connection is released, the virtual circuit is also terminated. With connection-oriented service, each packet carries an identifier telling which virtual circuit it belongs to.



Routing within a virtual-circuit subnet

5. Comparison of Virtual-Circuit and Datagram Subnets

Issue	Datagram subnet	Virtual-circuit subnet
Circuit setup	Not needed	Required
Addressing	Each packet contains the full source and destination address	Each packet contains a short VC number
State information	Routers do not hold state information about connections	Each VC requires router table space per connection
Routing	Each packet is routed independently	Route chosen when VC is set up; all packets follow it
Effect of router failures	None, except for packets lost during the crash	All VCs that passed through the failed router are terminated
Quality of service	Difficult	Easy if enough resources can be allocated in advance for each VC
Congestion control	Difficult	Easy if enough resources can be allocated in advance for each VC

Routing Algorithms

The main function of the network layer is routing packets from the source machine to the destination machine.

There are two processes inside router:

a)One of them handles each packet as it arrives, looking up the outgoing line to use for it in the routing table. This process is forwarding.
b)The other process is responsible for filling in and updating the routing tables. That is where the routing algorithm comes into play. This process is routing.

Regardless of whether routes are chosen independently for each packet or only when new connections are established, certain properties are desirable in a routing algorithm **correctness, simplicity, robustness, stability, fairness, optimality**

Routing algorithms can be grouped into two major classes:

1. Non-Adaptive Algorithms –

These are the algorithms which do not change their routing decisions once they have been selected. This is also known as static routing as route to be taken is computed in advance and downloaded to routers when router is booted.

2. Adaptive Algorithms -

These are the algorithms which change their routing decisions whenever network topology or traffic load changes. The changes in routing decisions are reflected in the topology as well as traffic of the network.

Different Routing Algorithms

- Optimality principle
- Shortest path algorithm
- Flooding
- Distance vector routing
- Link state routing
- Path vector routing
- Hierarchical Routing
- Congestion control algorithms

1. The Optimality Principle :

Statement of the optimality principle :

It states that if the router J is on the optimal path from router I to router K, then the optimal path from J to K also falls along the same route.

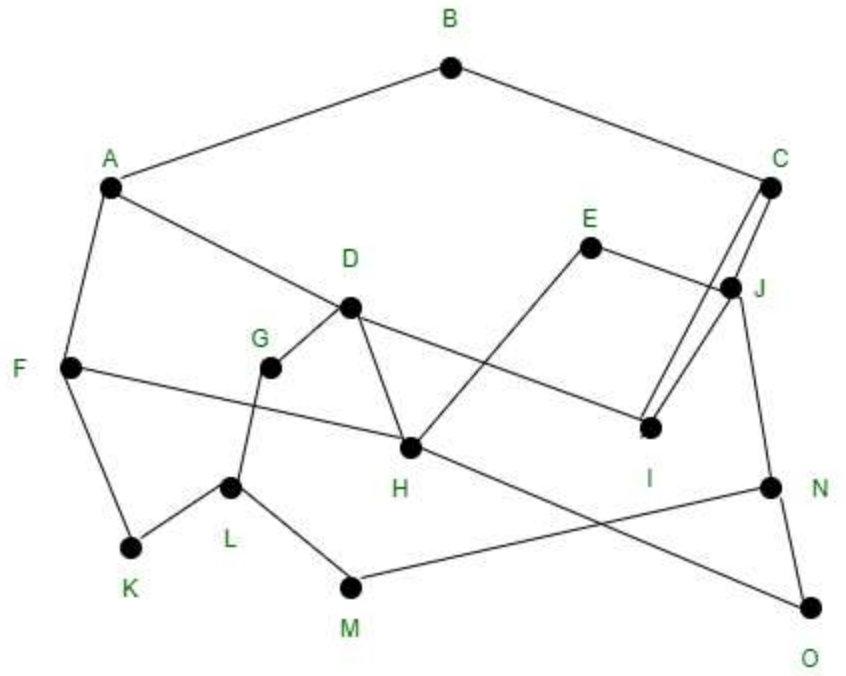
Call the route from I to J *r1* and the rest of the route *r2(J to K)*. it could be concatenated with *r1* to improve the route from I to K, contradicting our statement that *r1r2* is optimal only if a route better than r2 existed from J to K.

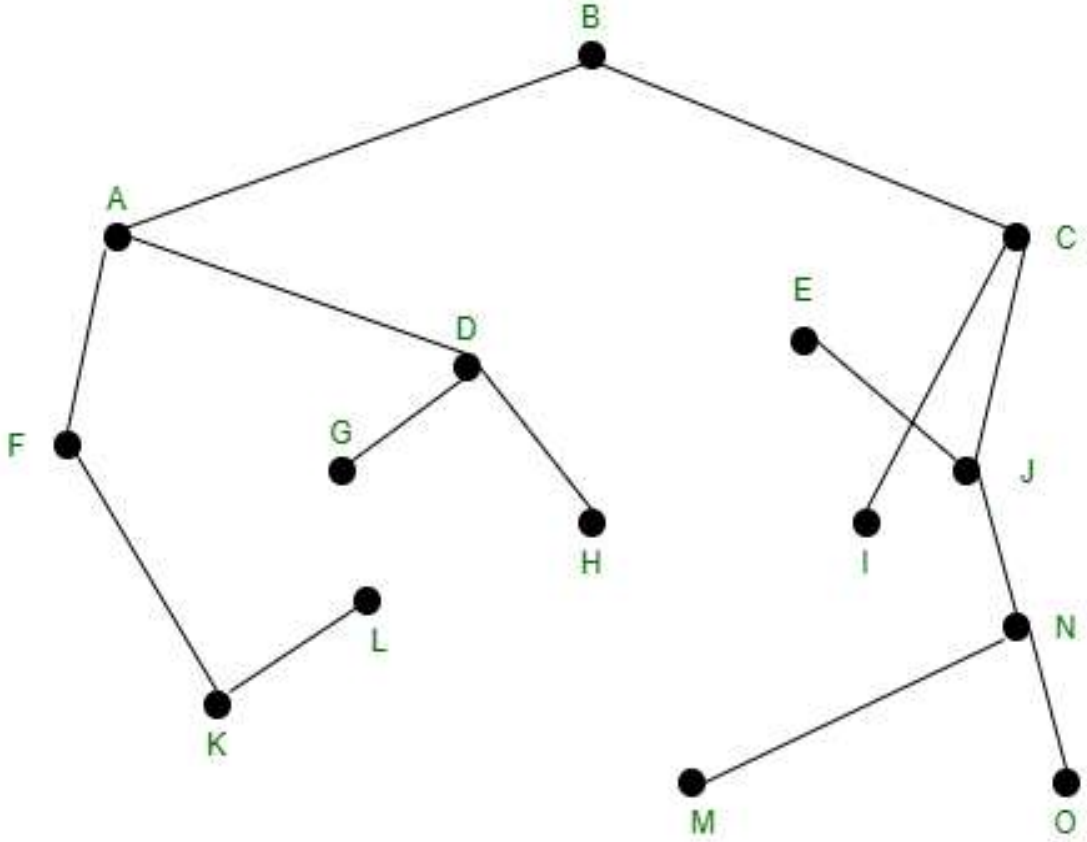
Sink Tree for routers :

We can see that the set of optimal routes from all sources to a given destination from a tree rooted at the destination as a directed consequence of the optimality principle. This tree is called a **sink tree** and is illustrated in fig(1).

Description of figure :

In the given figure the distance metric is the number of hops. Therefore, the goal of all routing algorithms is to discover and use the sink trees for all routers.




B
A
C
E
D
F
G
J
H
I
L
N
K
M
O

2. Shortest Path Routing

- In shortest path routing, the topology communications network is represented using a directed weighted **graph**.
- The nodes in the **graph** represent switching elements and the directed arcs in the **graph** represent communication links between switching elements.
- Each arc has a weight that represents the cost of sending a packet between two nodes in a particular direction.
- The objective in short path routing is to find a path between two nodes that has the smallest total cost, where the total cost of a path is the sum of the arc costs in that path

Dijkstra's algorithm:

1. Source node is initialized and can be indicated as filled circle
2. Initial path cost to neighboring nodes or link cost is computed and these nodes are relabeled considering source node
3. Examine the all adjacent nodes and finds the smallest label, make it permanent
4. The smallest label node is now working node ,then step 2,3 are repeated till the destination node reaches

Example 3.7.1 *Find the shortest path between node A and node H for the following Fig. 3.7.1 by applying Dijkstra's algorithm. Show each steps output.*

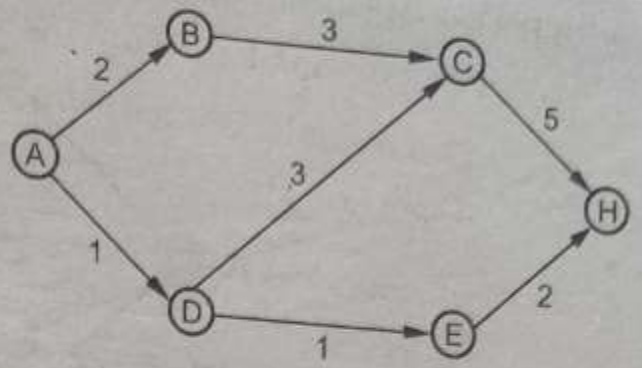


Fig. 3.7.1

Step-I : Node A is initialized as source node.

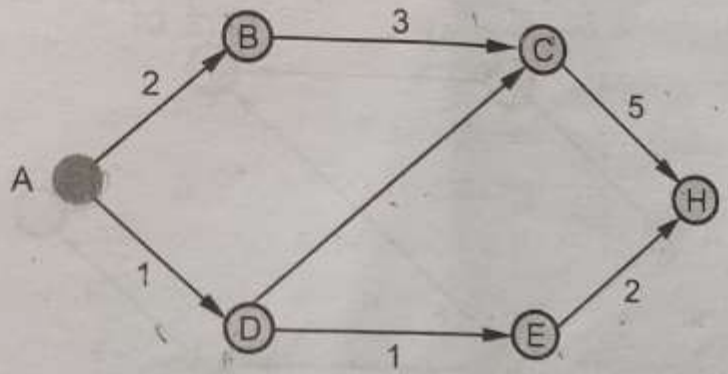


Fig. 3.7.1 (a)

Step-II : Link cost is computed for the adjacent node.

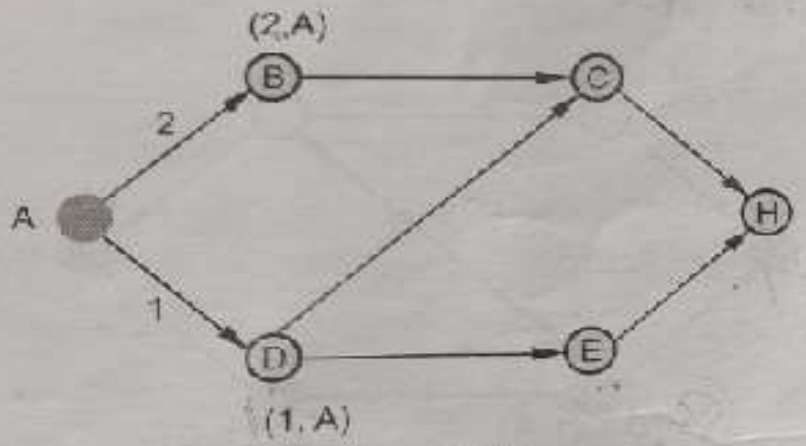
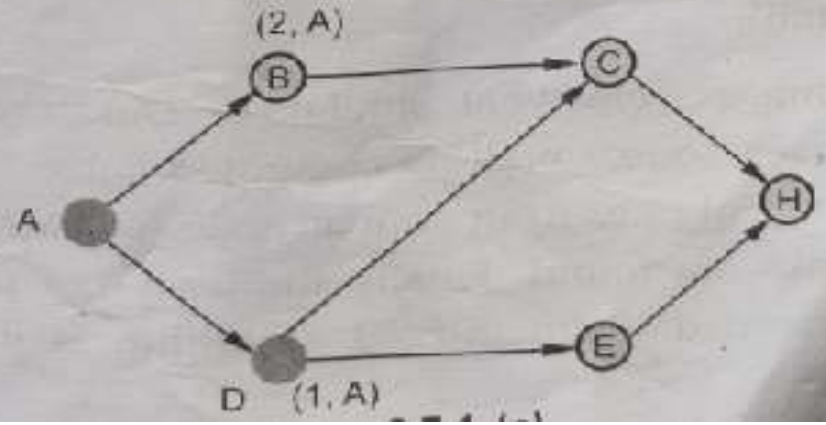


Fig. 3.7.1 (b)

Step-III : Since AD is smallest path, now D is working node.



Step-IV : Adjacent nodes to D are C and E.

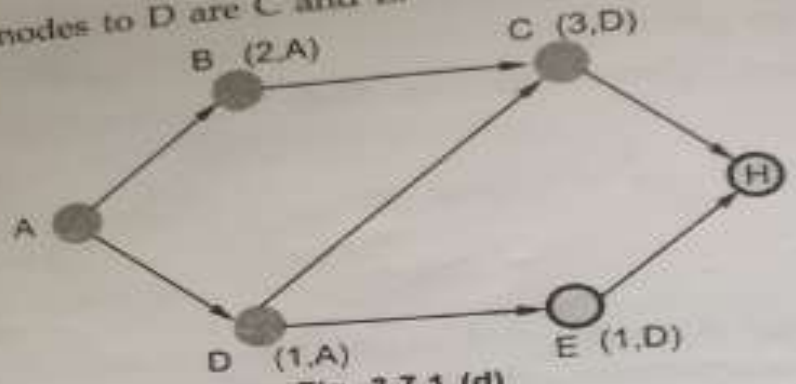


Fig. 3.7.1 (d)

Step-V : Since shortest is E, now E is working node.

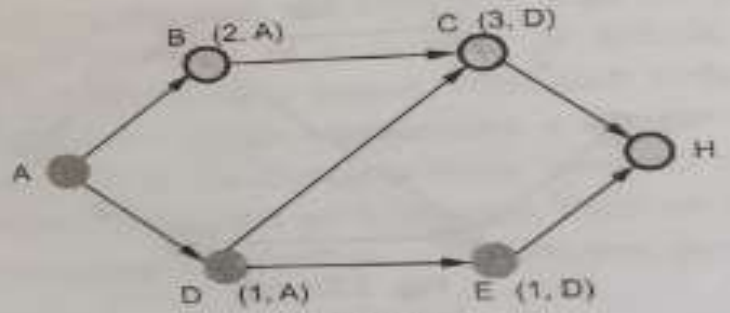


Fig. 3.7.1 (e)

Step-VI :

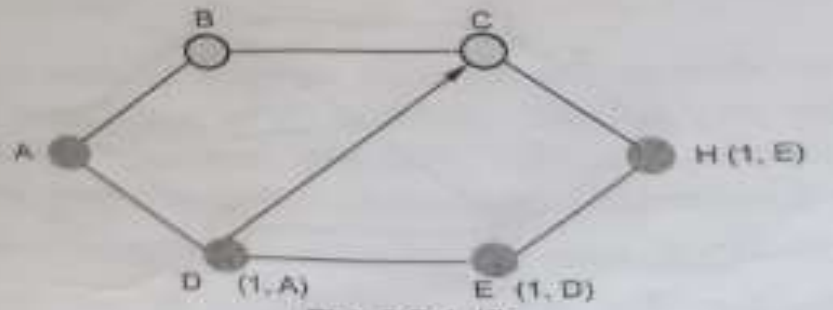


Fig. 3.7.1 (f)

Hence the shortest path between node A and node H is ADEH.

Bellman-Ford

Bellmon-Ford algorithm is some what similar to Dijkstra's algorithm but here the shortest paths from a given source node is computed subject to the constraint that the path contain at most one link i,e from source node ,at each step least cost path with maximum number of links are found. Finally the least cost path to each node and the cost of that path is computed .

Solution : **Step-1 :**

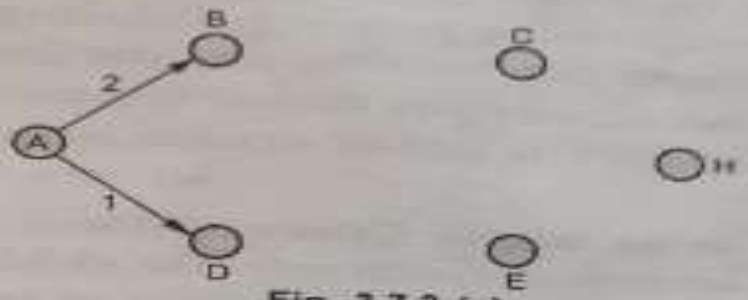


Fig. 3.7.2 (a)

Distance AD is shorter than AB. So route AD is chosen.

Step-2 :

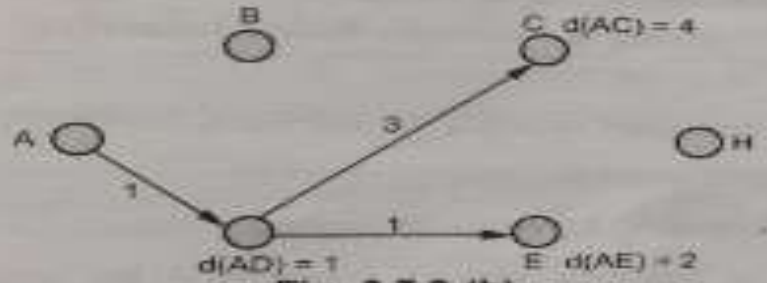


Fig. 3.7.2 (b)

$\because$ d(AE) < d(AC)

$\therefore$ d(AE) is chosen.

Step-3 :

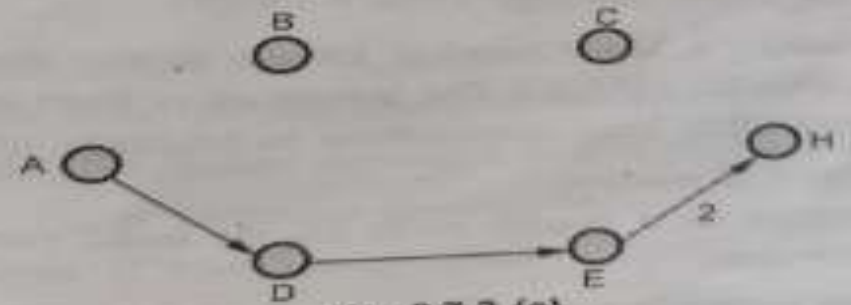


Fig. 3.7.2 (c)

So the shortest distance is ADEH, the result is same as in Dijkstra's algorithm.

3. Distance Vector Routing

- Two dynamic algorithms in particular, distance vector routing and link state routing, are the most popular.

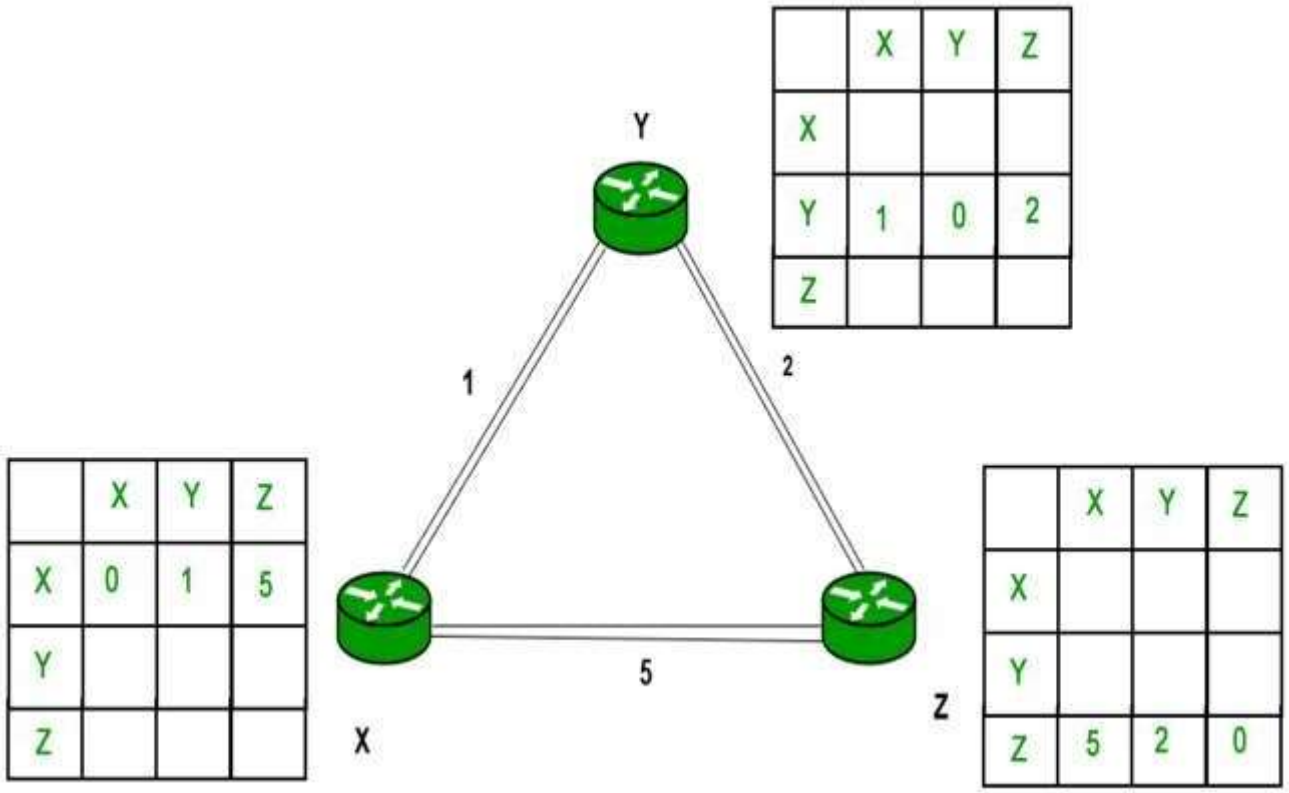
- Historically known as the old ARPANET routing algorithm (or known as Bellman-Ford algorithm).

- 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.

Distance Vector Algorithm –

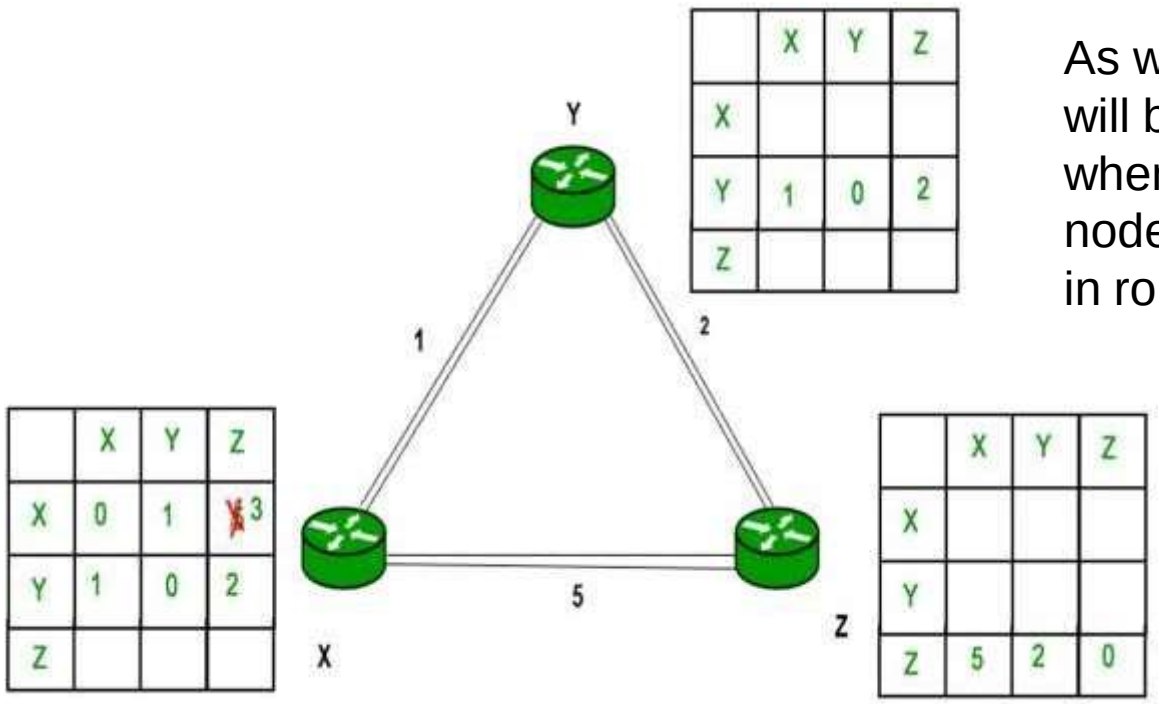
1. A router transmits its distance vector to each of its neighbors in a routing packet.
2. Each router receives and saves the most recently received distance vector from each of its neighbors.
3. A router recalculates its distance vector when:
    1. It receives a distance vector from a neighbor containing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

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.



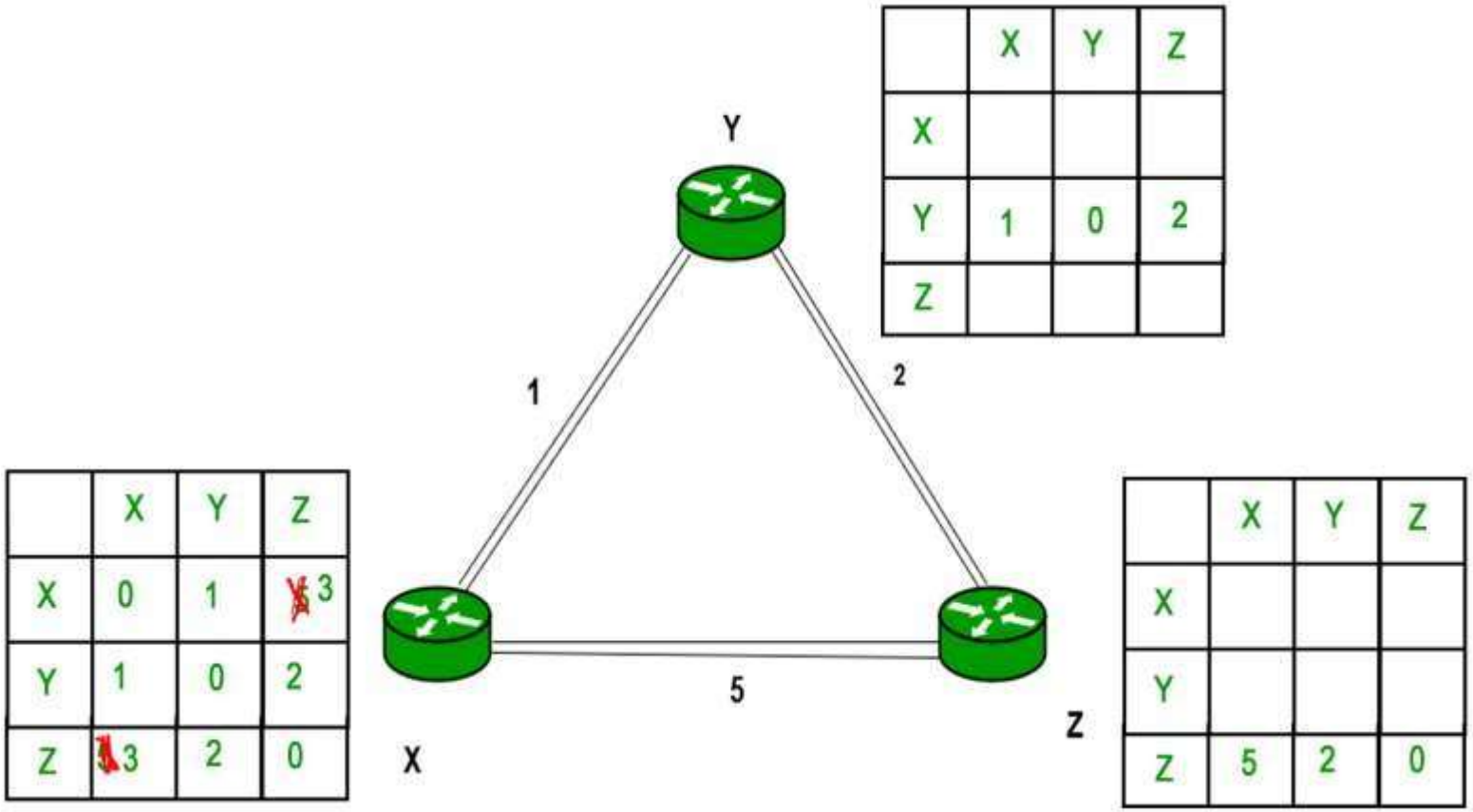
Consider router X , X will share it routing table to neighbors and neighbors will share it routing table to it to X and distance from node X to destination will be calculated using bellmen- ford equation.

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



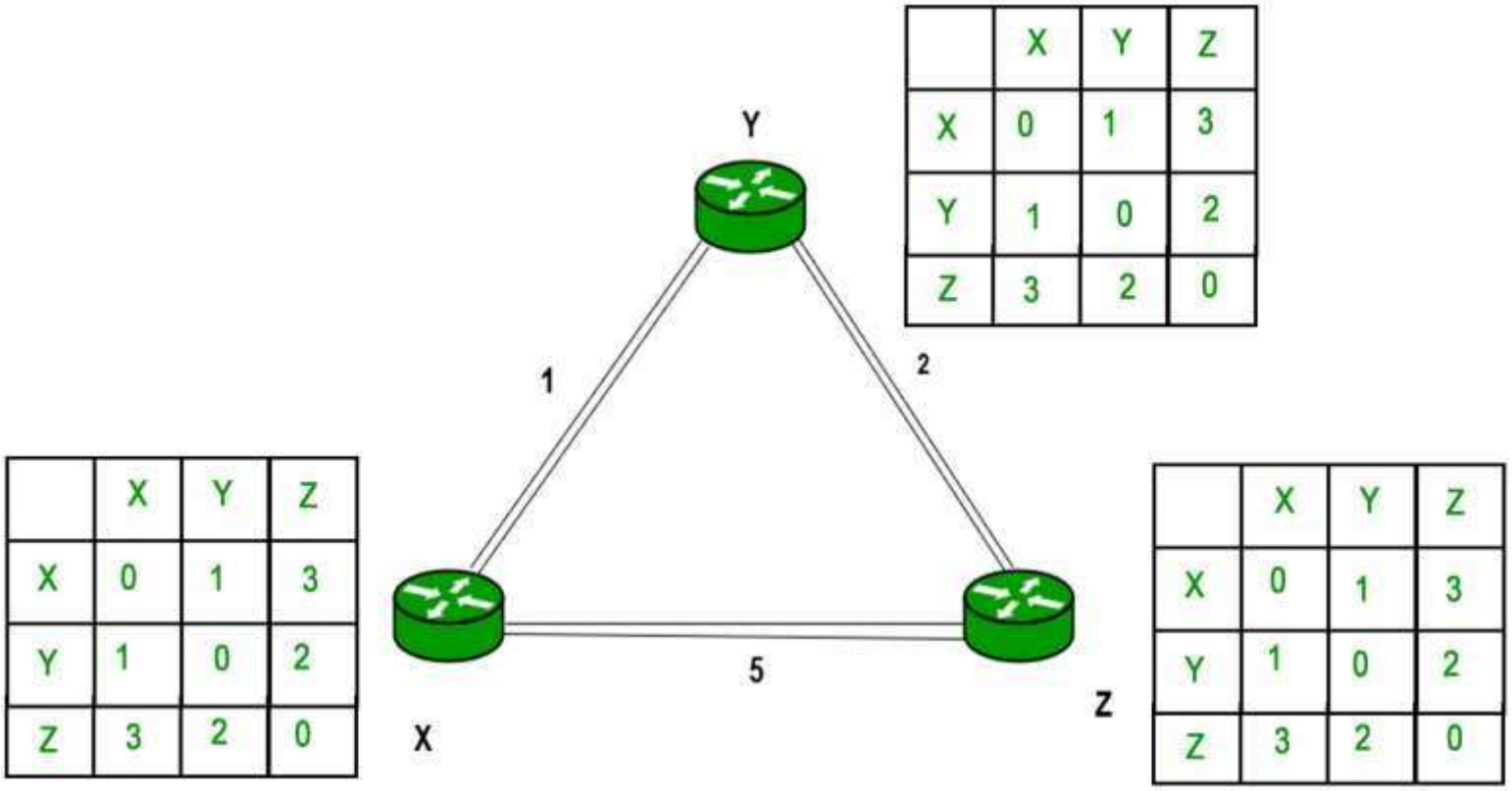
As we can see that distance will be less going from X to Z when Y is intermediate node(hop) so it will be update in routing table X.

Similarly for Z also –



Finally the routing table for all is above–

Finally the routing table for all –



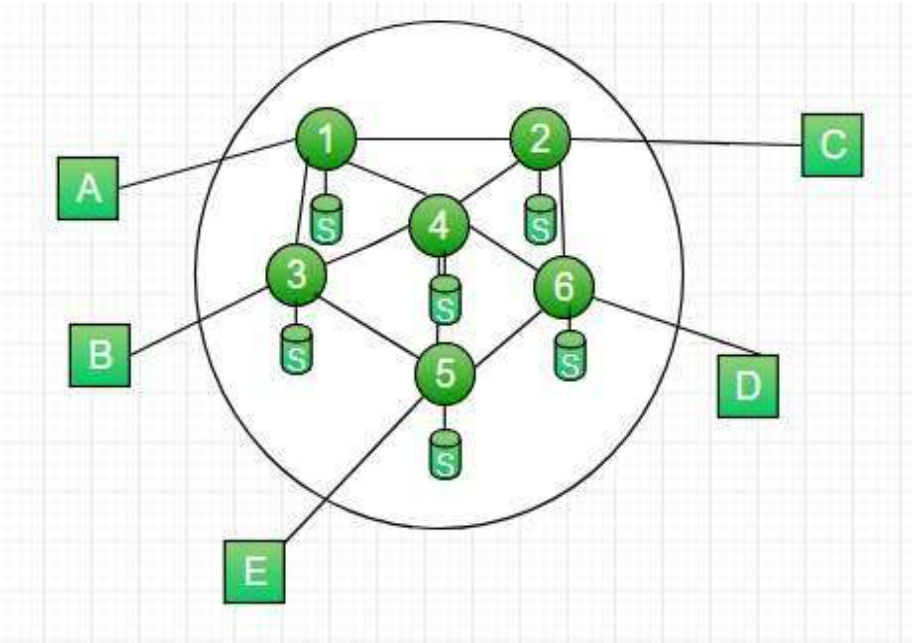
Note – Distance Vector routing uses UDP(User datagram protocol) for transportation.

4. Flooding :

- Requires no network information like topology, load condition ,cost of diff. paths

- Every incoming packet to a node is sent out on every outgoing like except the one it arrived on.

- For Example in below figure
  - A incoming packet to (1) is sent out to (2),(3)
  - from (2) is sent to (6),(4) and from (3) it is sent to (4),(5),etc



Characteristics –

• All possible routes between Source and Destination is tried. A packet will always get through if path exists

•As all routes are tried, there will be atleast one route which is the shortest. All nodes directly or indirectly connected are visited

Limitations –

•Flooding generates vast number of duplicate packets, in fact, an infinite number unless some measures are taken to damp the process.

•One such measure is to have a hop counter contained in the header of each packet, which is decremented at each hop, with the packet being discarded when the counter reaches zero. Ideally, the hop counter should be initialized to the length of the path from source to destination.

•A variation of flooding that is slightly more practical is **selective flooding.** In this algorithm the routers do not send every incoming packet out on every line, only on those lines that are going approximately in the right direction

- Flooding is not practical in most applications.

Advantages of Flooding :

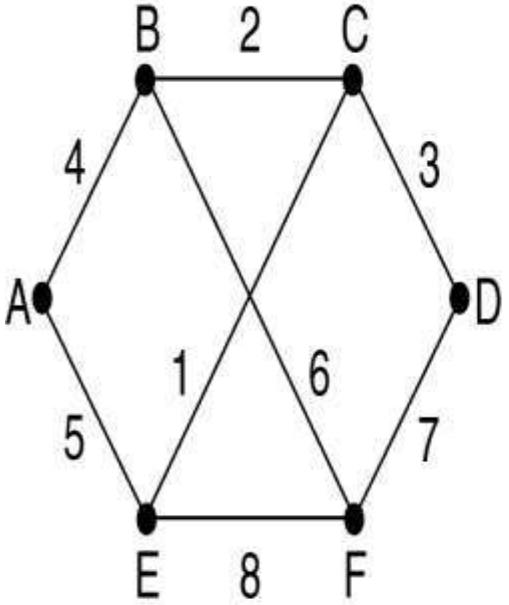
• Highly Robust, emergency or immediate messages can be sent (eg military applications)

• Broadcast messages to all the nodes

5. Link State Routing

- Distance vector routing algorithm was replaced by an entirely new algorithm, now called link state routing. Variants of link state routing are now widely used.
- The idea behind link state routing is simple and can be stated as five parts. Each router must do the following:

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



(a)

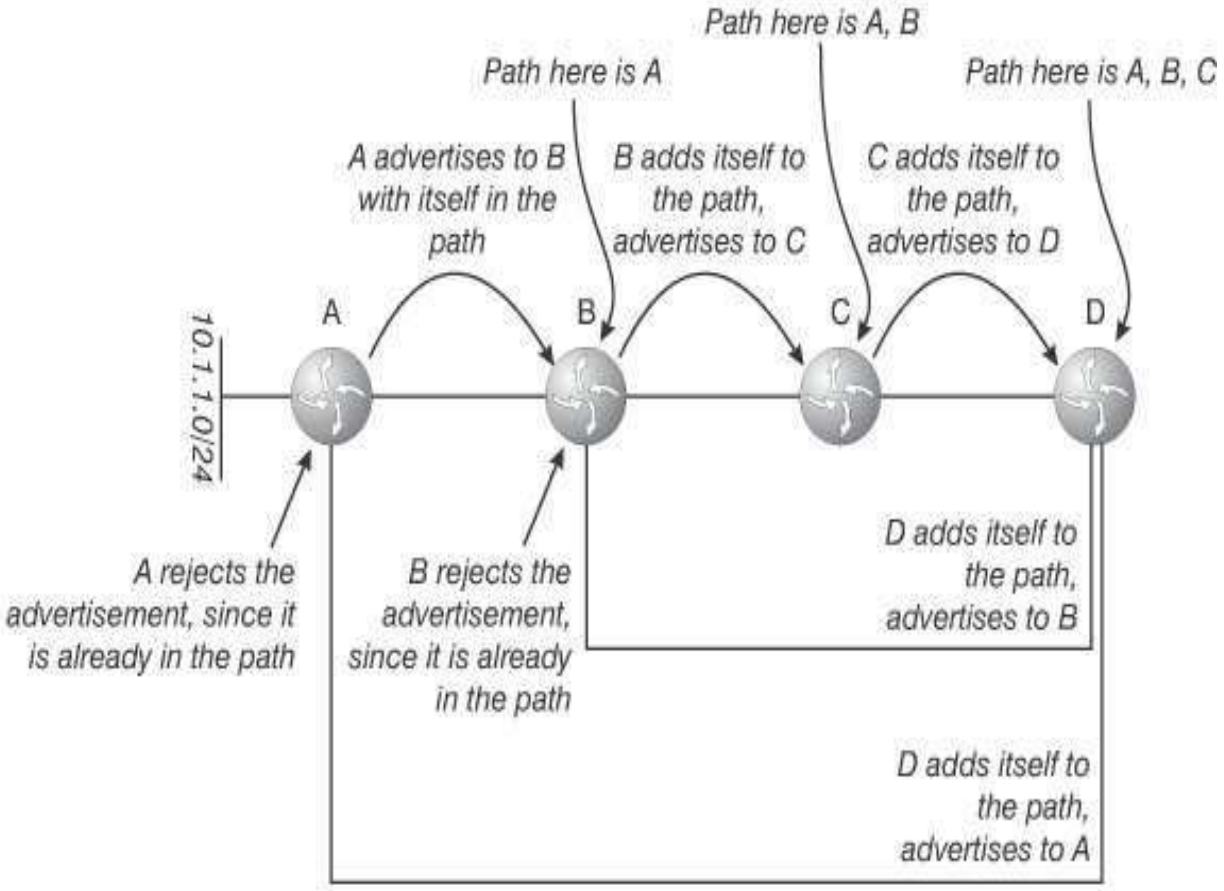
Link State Packets

A	B	C	D	E	F
Seq.	Seq.	Seq.	Seq.	Seq.	Seq.
Age	Age	Age	Age	Age	Age
B 4	A 4	B 2	C 3	A 5	B 6
E 5	C 2	D 3	F 7	C 1	D 7
	F 6	E 1		F 8	E 8

(b)

6. Path Vector Routing

- path vector protocol does not rely on the cost of reaching a given destination to determine whether each path available is loop free or not. Instead, path vector protocols rely on analysis of the path to reach the destination to learn if it is loop free or not.
- A path vector protocol guarantees loop free paths through the network by recording each hop the routing advertisement traverses through the network.
- In this case, router A advertises reachability to the 10.1.1.0/24 network to router B. When router B receives this information, it adds itself to the path, and advertises it to router C. Router C adds itself to the path, and advertises to router D that the 10.1.1.0/24 network is reachable in this direction.


Path here is A, B
Path here is A
Path here is A, B, C
A advertises to B with itself in the path
B adds itself to the path, advertises to C
C adds itself to the path, advertises to D
A
B
C
D
10.1.1.0/24
A rejects the advertisement, since it is already in the path
B rejects the advertisement, since it is already in the path
D adds itself to the path, advertises to B
D adds itself to the path, advertises to A

7. Hierarchical Routing:

- As the number of routers becomes large, the overhead involved in computing, storing, and communicating the routing table information (e.g., link state updates or least cost path changes) becomes prohibitive.

- Also an organization should be able to run and administer its network as it wishes(e.g., to run whatever routing algorithm it chooses), while still being able to connect its network to other "outside" networks.

- Clearly, something must be done to reduce the complexity of route computation in networks as large as the public Internet.

- Both of these problems can be solved by aggregating routers into **"regions"** or **"autonomous systems" (ASs).** Routers within the same AS run the same routing algorithm (e.g., a LS or DV algorithm) and have full information about each other.

- The routing algorithm running within an autonomous system is called an **intra-autonomous** system routing protocol.

- Routers in an AS that have the responsibility of routing packets to destinations outside the AS are called **gateway routers.**

- The routing algorithm that gateways use to route among the various ASs is known as an **inter-autonomous** system routing protocol.

- Here, there are three routing ASs, A, B and C. Autonomous system A has four routers, A.a, A.b, A.c and A.d, which run the intra-AS routing protocol used within autonomous system A.

- These four routers have complete information about routing paths within autonomous system A. Similarly, autonomous systems B and C have three and two routers, respectively.

- Note that the intra-AS routing protocols running in A, B and C need not be the same.

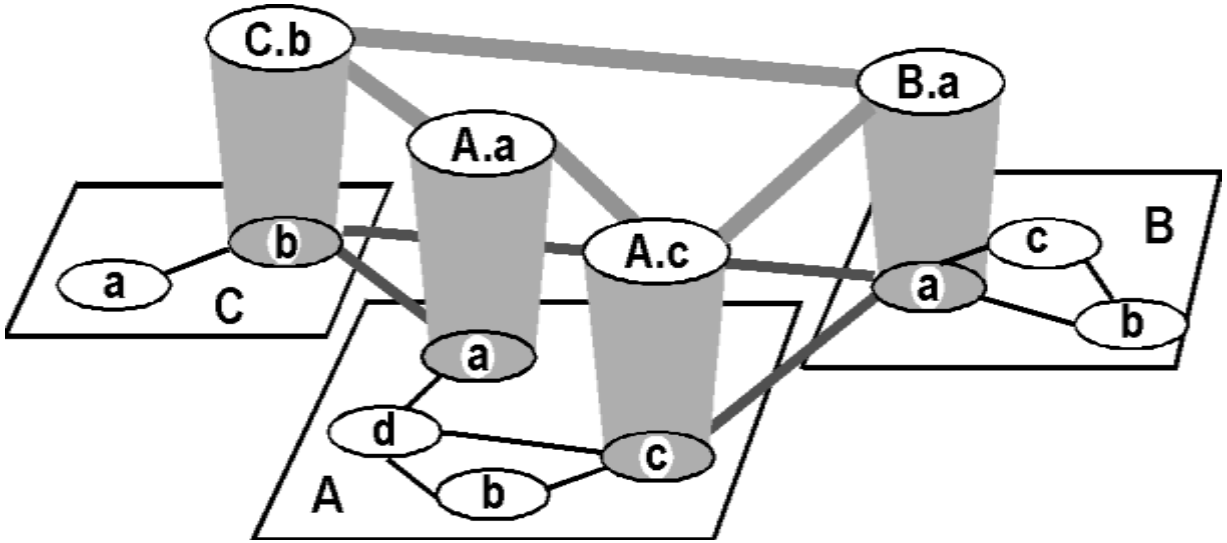
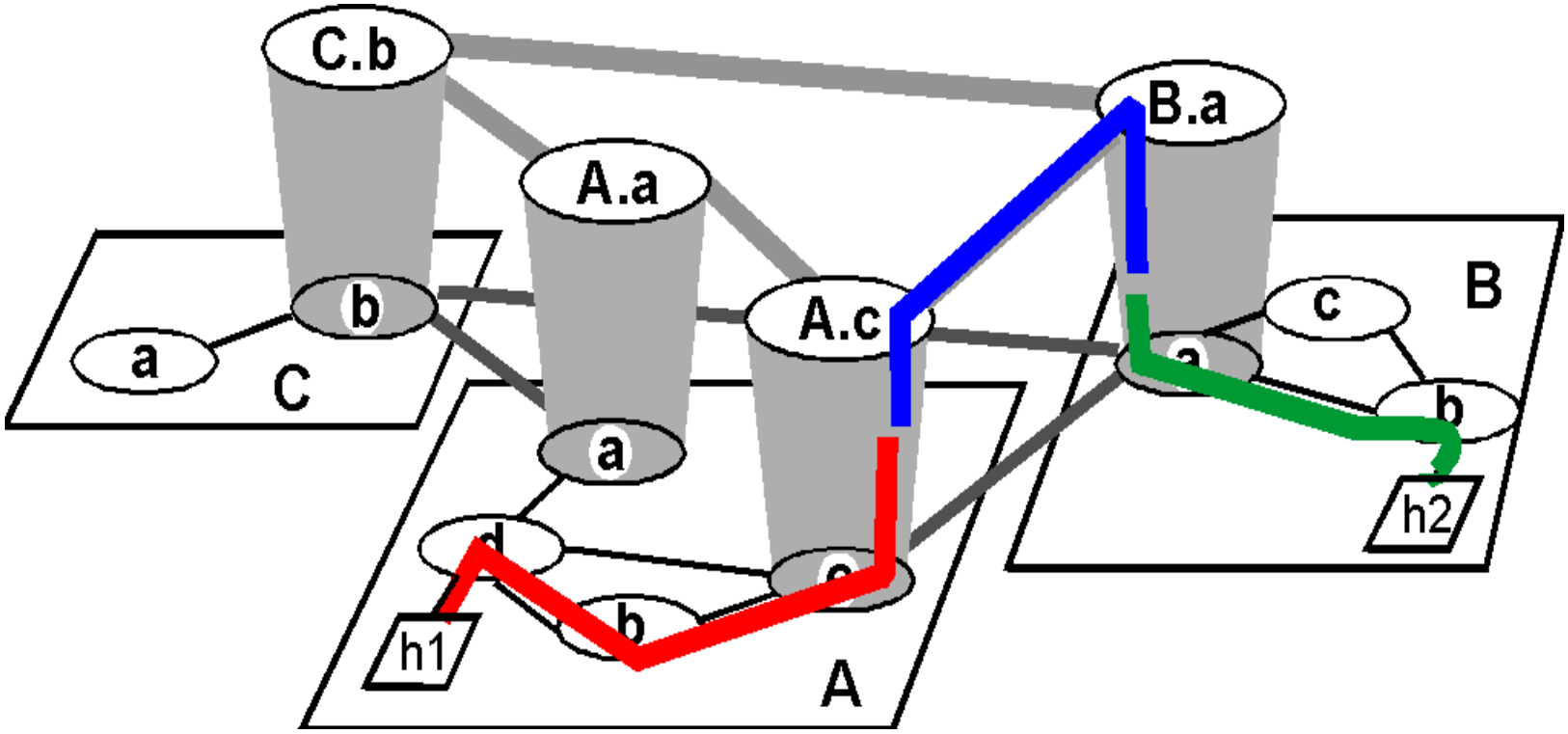


Fig : Intra-AS and Inter-AS routing.

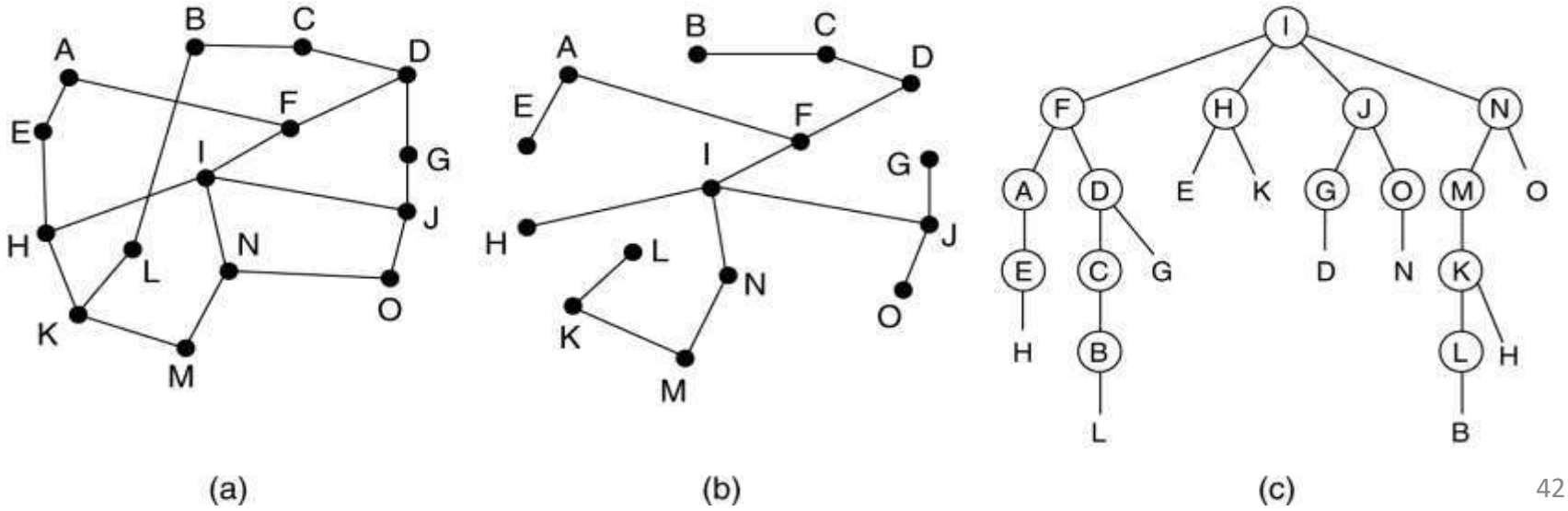
Suppose now that a host h1 attached to router A.d needs to route a packet to destination h2 in autonomous system B, as shown in below Figure



The route from A.d to B.b : intra-AS and inter-AS path segments.

Broadcast Routing

- In some applications, hosts need to send messages to many or all other hosts. Sending a packet to all destinations simultaneously is called broadcasting.
- One broadcasting method that requires no special features from the subnet is for the source to simply send a distinct packet to each destination.



8. Congestion Control Algorithms

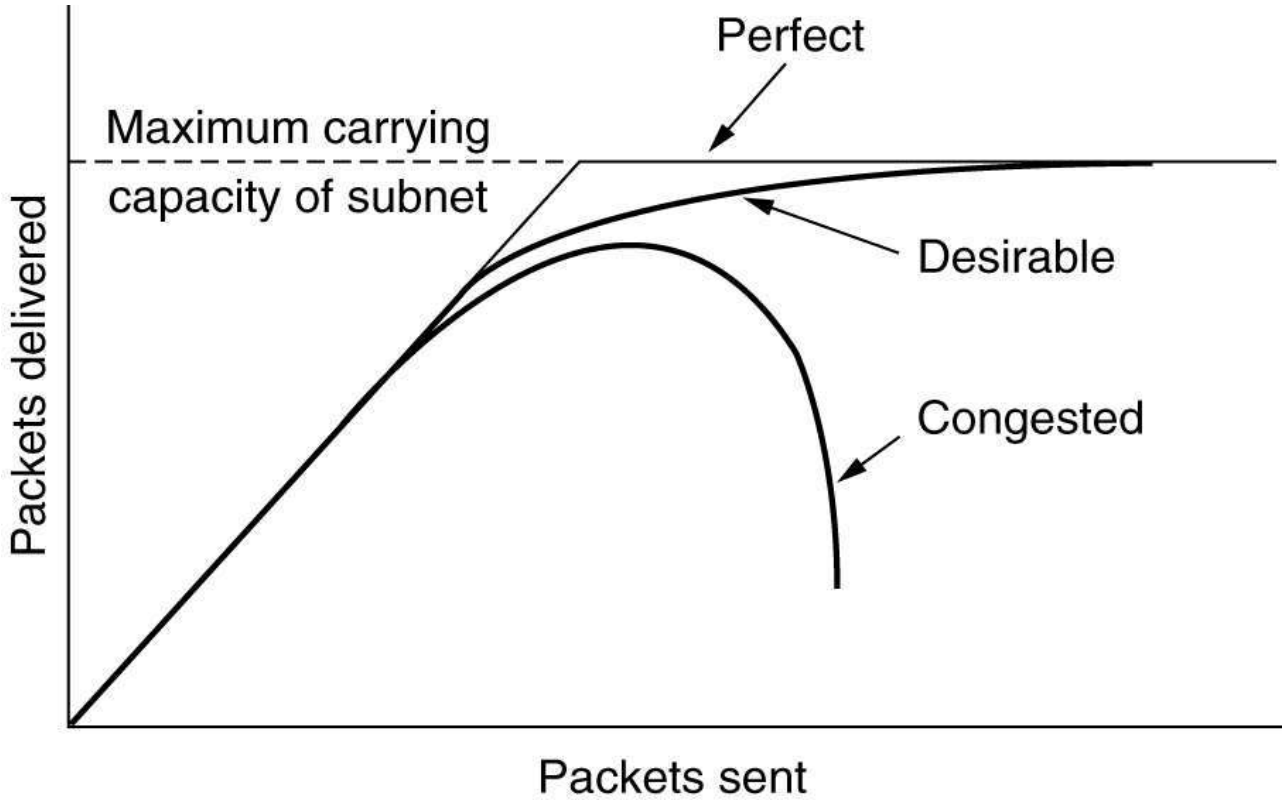
- When too many packets are present in (a part of) the subnet, performance degrades. This situation is called congestion.

- The network and transport layers share the responsibility for handling congestion. Since congestion occurs within the network, it is the network layer that directly experiences it and must ultimately determine what to do with the excess packets.

- When the number of packets dumped into the subnet by the hosts is within its carrying capacity, they are all delivered and the number delivered is proportional to the number sent.

Algorithms :

- General Principles of Congestion Control
- Congestion Prevention Policies
- Congestion Control in Virtual-Circuit Subnets
- Congestion Control in Datagram Subnets



General Principles of Congestion Control

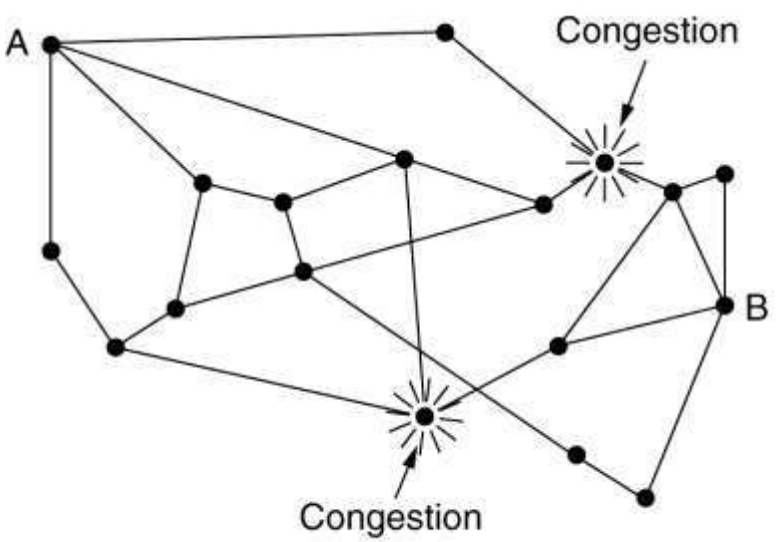
- Monitor the system to detect when and where congestion occurs.
- Pass this information to places where action can be taken.
- Adjust system operation to correct the problem.

Congestion Prevention Policies

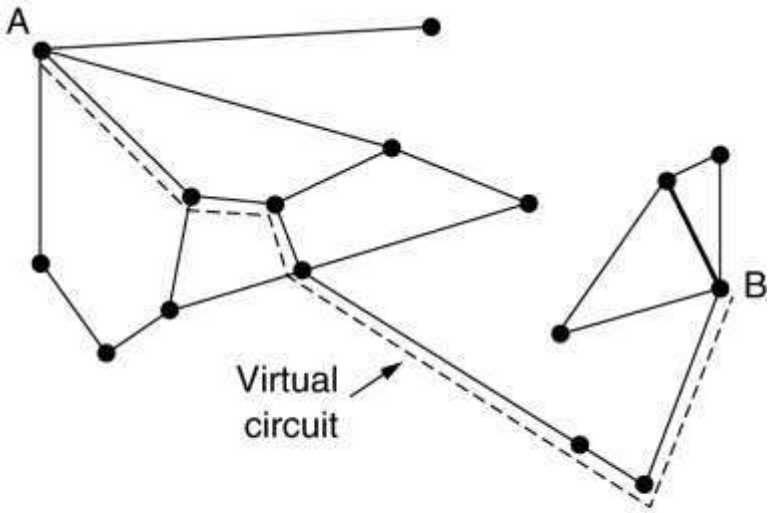
Layer	Policies
Transport	• Retransmission policy • Out-of-order caching policy • Acknowledgement policy • Flow control policy • Timeout determination
Network	• Virtual circuits versus datagram inside the subnet • Packet queueing and service policy • Packet discard policy • Routing algorithm • Packet lifetime management
Data link	• Retransmission policy • Out-of-order caching policy • Acknowledgement policy • Flow control policy

Congestion Control in Virtual-Circuit Subnets

- In virtual-circuit subnets you can control congestion dynamically.
- One technique that is widely used is admission control.
- The idea is simple: once congestion has been signaled, no more virtual circuits are set up until the problem has gone away.
- An alternative approach is to allow new virtual circuits but carefully route all new virtual circuits around problem areas. For example, consider the subnet of Fig



(a)



(b)

Congestion Control in Datagram Subnets

- Each router can easily monitor the utilization of its output lines and other resources.
- Whenever u moves above the threshold, the output line enters a "warning" state.
- Each newly-arriving packet is checked to see if its output line is in warning state. Following are several alternatives for taking action for warning state.

These include:

1. Warning bit
2. Choke packets
3. Load shedding
4. Random early discard
5. Traffic shaping

- The first 3 deal with congestion detection and recovery. The last 2 deal with congestion avoidance

Warning Bit

1. A special bit in the packet header is set by the router to warn the source when congestion is detected.

2. The bit is copied and piggy-backed on the ACK and sent to the sender.

3. The sender monitors the number of ACK packets it receives with the warning bit set and adjusts its transmission rate accordingly.

Choke Packets

1. A choke packet is a control packet generated at a congested node and transmitted to restrict traffic flow.
2. The source, on receiving the choke packet must reduce its transmission rate by a certain percentage.
3. An example of a choke packet is the ICMP Source Quench Packet.

Hop-by-Hop Choke Packets

1. Over long distances or at high speeds choke packets are not very effective.
2. A more efficient method is to send to choke packets hop-by-hop.
3. This requires each hop to reduce its transmission even before the choke packet arrive at the source

Load Shedding

1. When buffers become full, routers simply discard packets.
2. Which packet is chosen to be the victim depends on the application and on the error strategy used in the data link layer.
3. For a file transfer, for, e.g. cannot discard older packets since this will cause a gap in the received data.
4. For real-time voice or video it is probably better to throw away old data and keep new packets.
5. Get the application to mark packets with discard priority.

Random Early Discard (RED)

1. This is a proactive approach in which the router discards one or more packets before the buffer becomes completely full.
2. Each time a packet arrives, the RED algorithm computes the average queue length, avg.
3. If avg is lower than some lower threshold, congestion is assumed to be minimal or non- existent and the packet is queued.
4. If avg is greater than some upper threshold, congestion is assumed to be serious and the packet is discarded.
5. If avg is between the two thresholds, this might indicate the onset of congestion. The probability of congestion is then calculated.

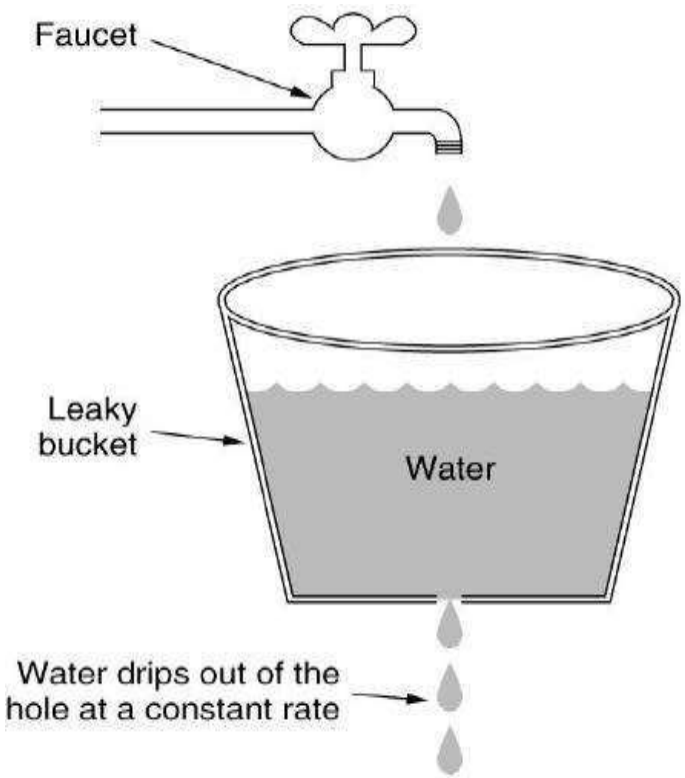
Traffic Shaping

1. Another method of congestion control is to “shape” the traffic before it enters the network.
2. Traffic shaping controls the rate at which packets are sent (not just how many). Used in ATM and Integrated Services networks.
3. At connection set-up time, the sender and carrier negotiate a traffic pattern (shape).

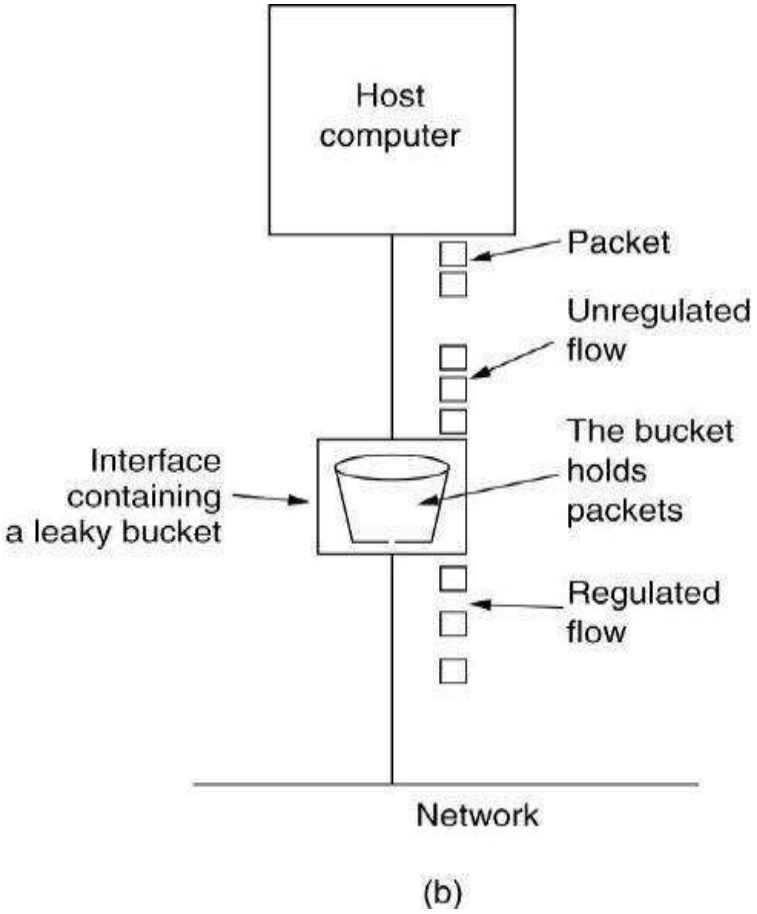
Two traffic shaping algorithms are:

1. Leaky Bucket Algorithm
2. Token Bucket Algorithm

The **Leaky Bucket Algorithm** used to control rate in a network. It is implemented as a single- server queue with constant service time. If the bucket (buffer) overflows then packets are discarded.



(a) A leaky bucket with water.



(b) a leaky bucket with packets.

1. The leaky bucket enforces a constant output rate (average rate) regardless of the burstiness of the input. Does nothing when input is idle.

2. The host injects one packet per clock tick onto the network. This results in a uniform flow of packets, smoothing out bursts and reducing congestion.

3. When packets are the same size (as in ATM cells), the one packet per tick is okay. For variable length packets though, it is better to allow a fixed number of bytes per tick.

4. E.g. 1024 bytes per tick will allow one 1024-byte packet or two 512-byte packets or four 256- byte packets on 1 tick

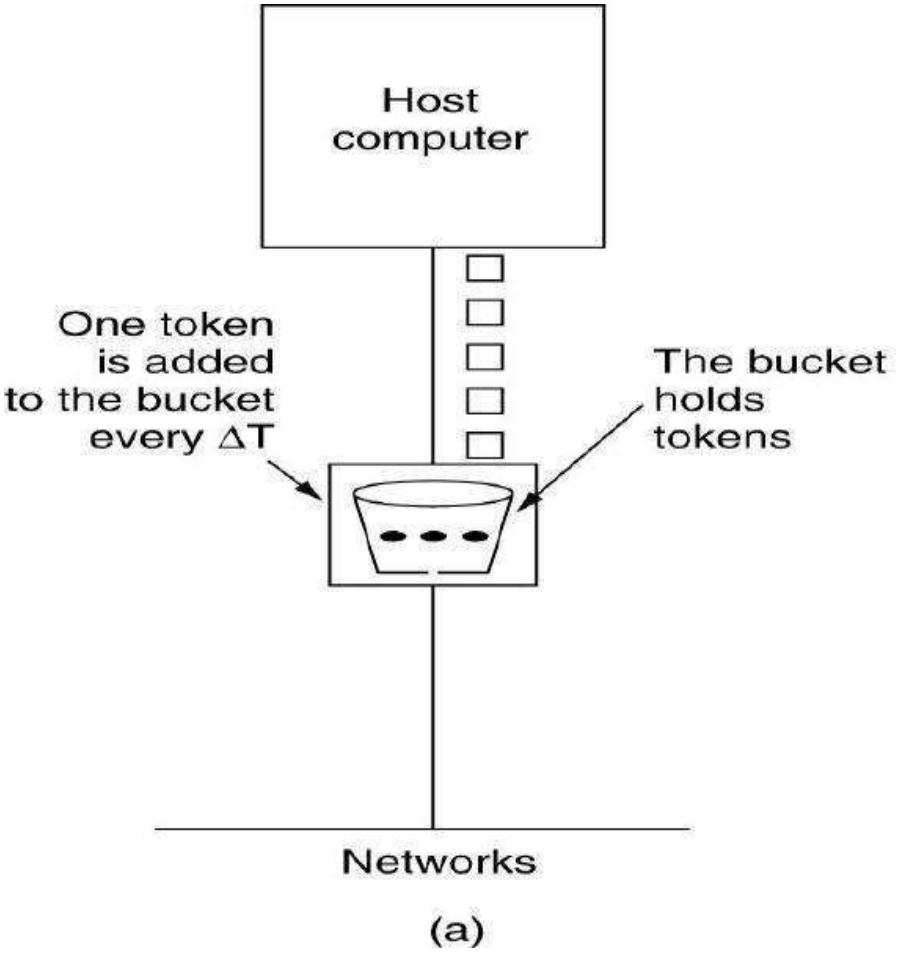
Token Bucket Algorithm

1. In contrast to the LB, the Token Bucket Algorithm, allows the output rate to vary, depending on the size of the burst.

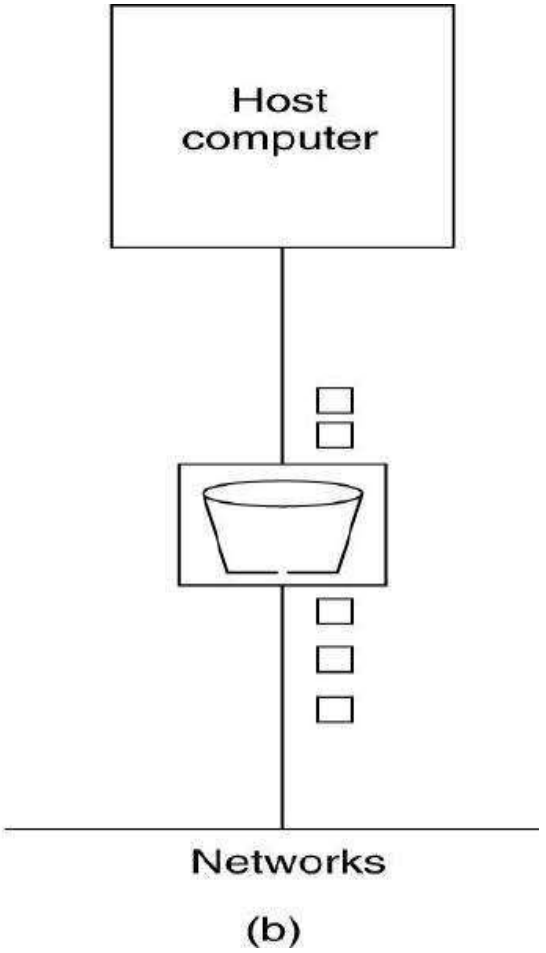
2. In the TB algorithm, the bucket holds tokens. To transmit a packet, the host must capture and destroy one token.

3. Tokens are generated by a clock at the rate of one token every t sec.

4.Idle hosts can capture and save up tokens (up to the max. size of the bucket) in order to send larger bursts later.



(a) Before.



(b) After.

Leaky Bucket vs. Token Bucket

1. LB discards packets; TB does not. TB discards tokens.

2. With TB, a packet can only be transmitted if there are enough tokens to cover its length in bytes.

3. LB sends packets at an average rate. TB allows for large bursts to be sent faster by speeding up the output.

4. TB allows saving up tokens (permissions) to send large bursts. LB does not allow saving.

UNIT III
NETWORK LAYER – Part 2

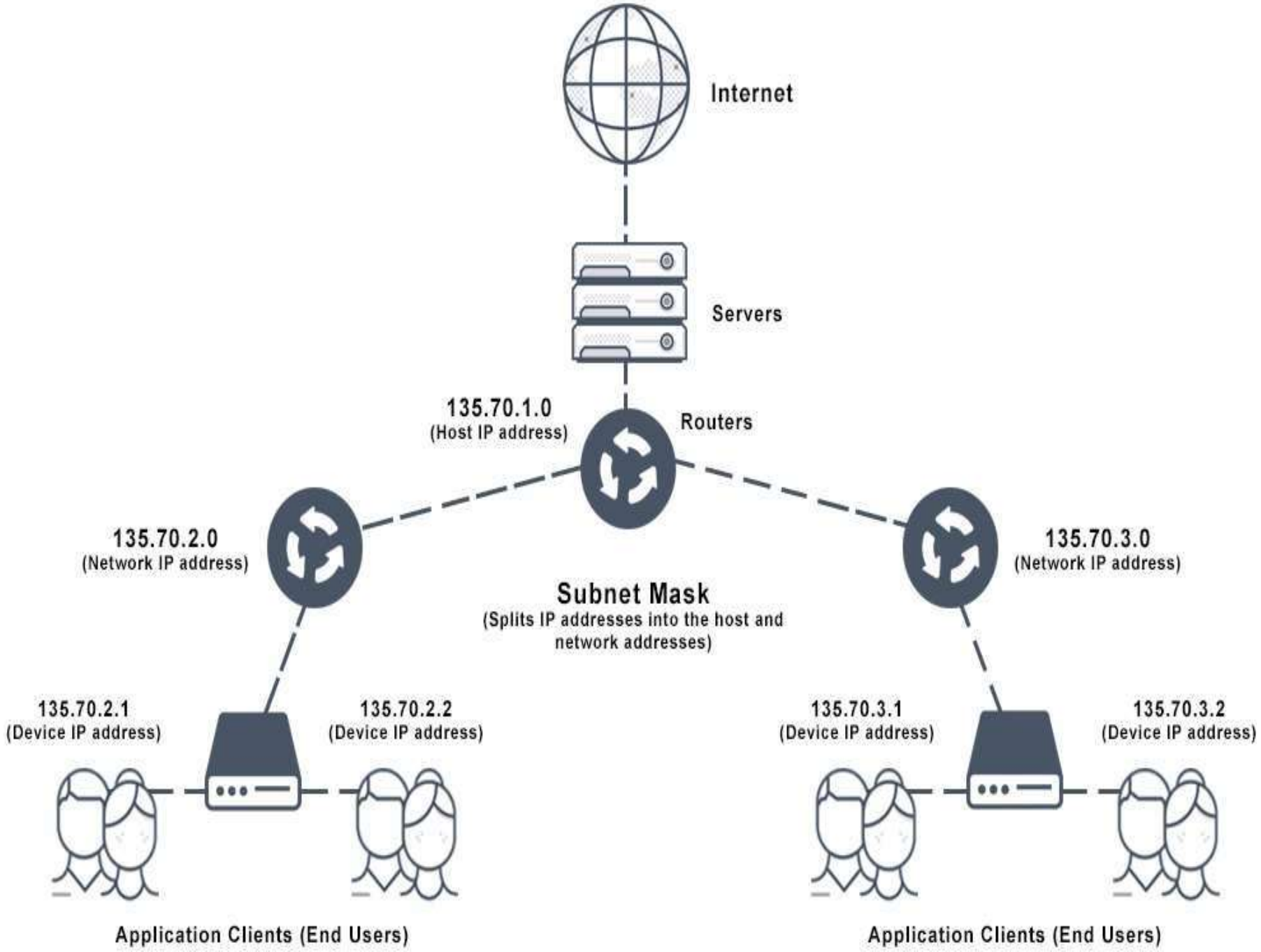
IP addresses,
CIDR,
Subnetting,
SuperNetting,
IPv4, Packet Fragmentation,
IPv6 Protocol,
Transition from IPv4 to IPv6,
ARP, RARP.

- Every device connected to the Internet needs to have an identifier. Internet Protocol (IP) addresses are the numerical addresses used to identify a particular piece of hardware connected to the Internet.

- The two most common versions of IP in use today are Internet Protocol version 4 (IPv4) and Internet Protocol version 6 (IPv6).

- For IPv4, this pool is 32-bits in size and contains 4,294,967,296 IPv4 addresses.

- The IPv6 address space is 128-bits(2128) in size, containing 340,282,366,920,938,463,463,374,607,431,768,211,456 IPv6 addresses.

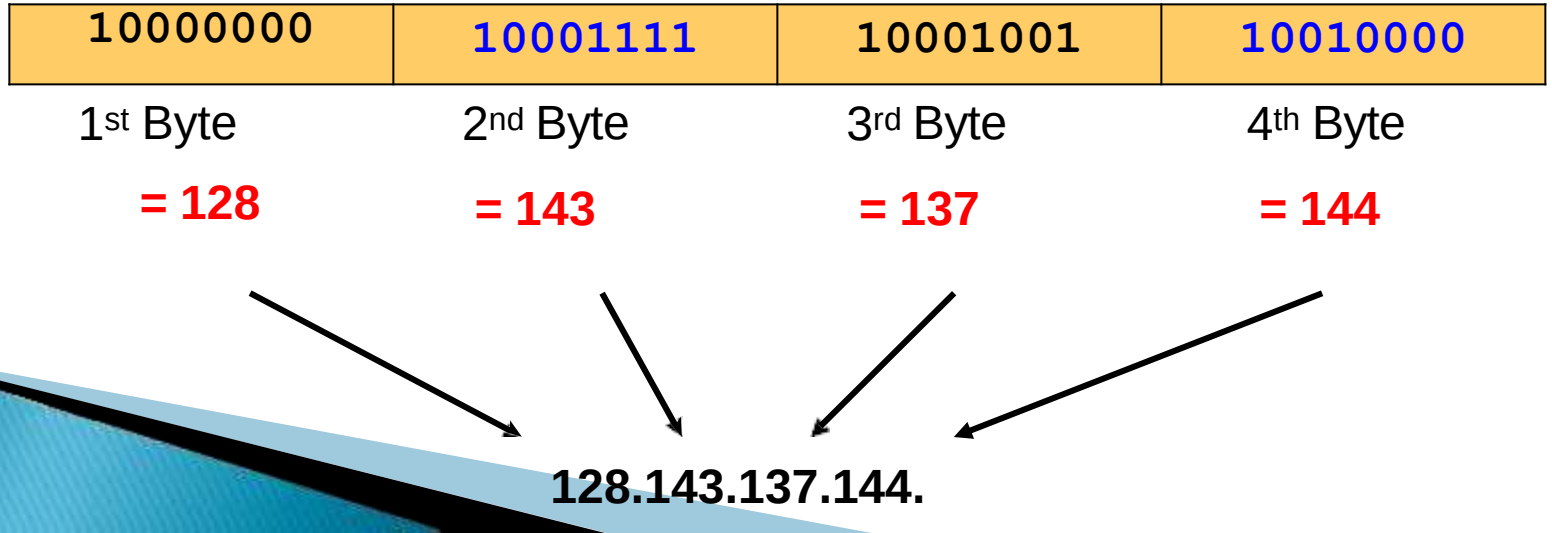

Internet
Servers
135.70.1.0
(Host IP address)
Routers
135.70.2.0
(Network IP address)
135.70.3.0
(Network IP address)
Subnet Mask
(Splits IP addresses into the host and network addresses)
135.70.2.1
(Device IP address)
135.70.2.2
(Device IP address)
135.70.3.1
(Device IP address)
135.70.3.2
(Device IP address)
Application Clients (End Users)
Application Clients (End Users)

IP Address

- What is an IP address…?
  - An IP address is a unique global address for a network interface

– Is a 32 bit long identifier(IPV4)

–Encodes a network number (network prefix) and a host number



IP Addressing

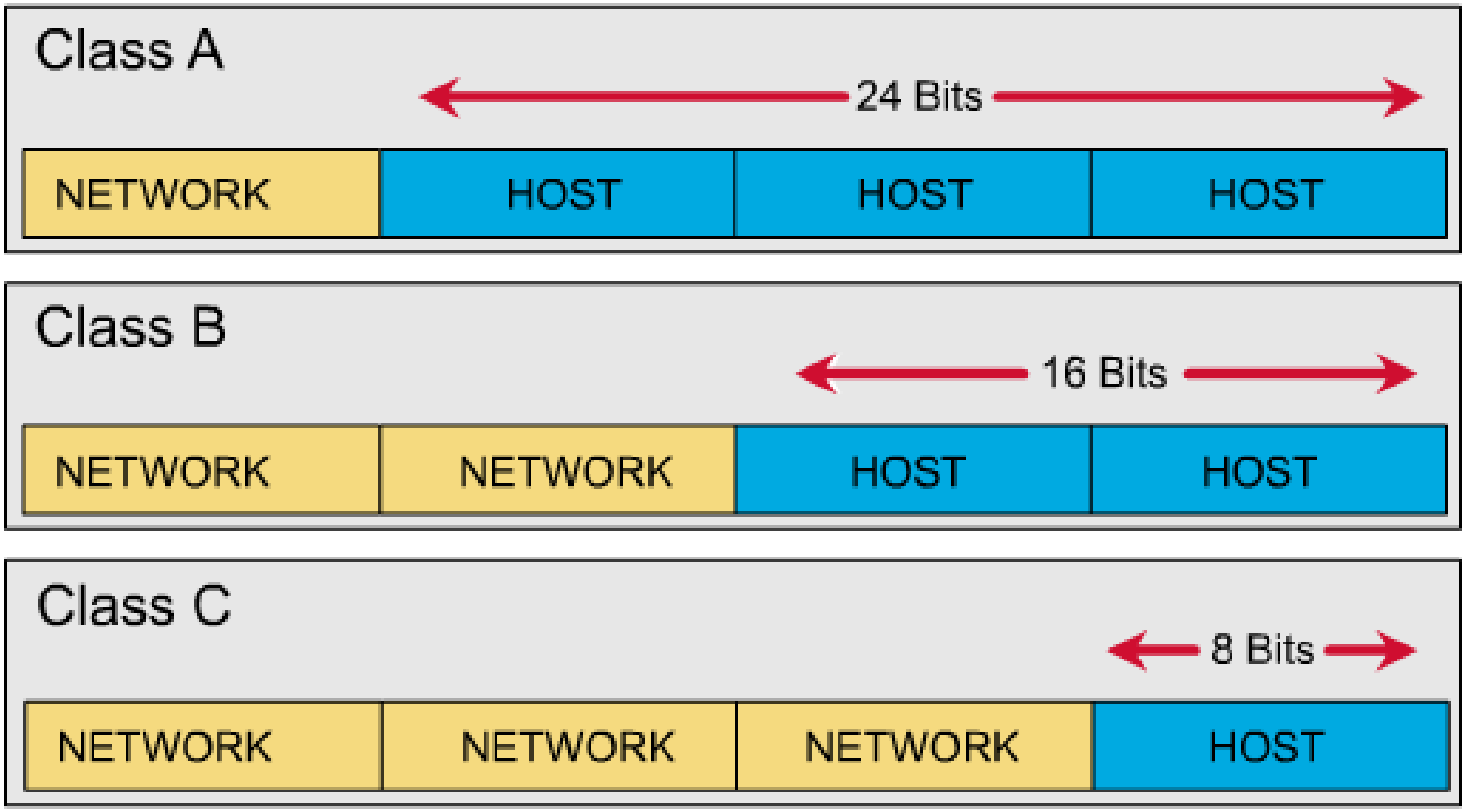
- There are two IP addressing scheme:
  - 1. Class-full
  - 2. Classless
- In classful addressing the address space is
  - divided into 5 classes:
    - A, B, C, D, and E

Class Ranges of Addresses

IPv4 Address Structure

- Example Class A network address: 114.56.20.33, 255.0.0.0
  - Network information = 114.
  - Host information = 56.204.33
- Example Class B network address: 147.12.38.81, 255.255.0.0
  - Network information = 147.12.
  - Host information = 38.81
- Example Class C network address: 214.51.42.7, 255.255.255.0
  - Network information = 214.57.42.
  - Host information = 7

IPv4 Address Classes



Class A

- Class A addresses are assigned to networks with a very large number of hosts
- Reserved for governments and large corporations throughout the world.
- Each class A address supports 16,777,214 hosts.
- The high-order bit(MSB) in a class A address is always set to zero.
- The next seven bits(completing the first octet) complete the Network ID
- The remaining 24 bits represent the host ID

Class B

- Class B addresses are assigned to large- and medium-sized companies
- Each Class B address supports 65,534 hosts
- The two high-order bits in a class B address are always set to binary 1 0.
- The next 14 bits complete the Network ID
- The remaining 16 bits represent the host ID

Class C

- Class C addresses are used for small networks.
  - Addresses are assigned to groups that meet the qualifications to obtain Class A and B addresses
  - Supports 254 hosts
- The three high-order bits(MSB) in a class C address are always set to 1 10
- The next 21 bits complete the Network ID
- The remaining 8 bits represent the host ID

Class D & E

- Class D addresses are reserved for IP multicast addresses. Also known as multicast addresses.
- Multicasting is the sending of a stream of data (usually audio and video) to multiple computers simultaneously
  - ✓ The four high-order bits in a class D address are always set to binary 1 1 1 0.
  - ✓ The remaining bits are for the address that interested hosts recognize.
- Class E addresses are reserved for research, testing.
  - ✓ The high-order bits in a class E address are set to 1111.
  - ✓ The Class E range starts where Class D leaves off

Summary of Usable Addresses

Characteristics of the IP Address Classes						
Class	Address Range	Identify Bits (binary value)	Bits in Network ID	Number of Networks	Bits in Host ID	Number of Hosts/ Network
A	0 ~ 127	1 (0)	7	126	24	16,777,214
B	128~191	2 (10)	14	16,382	16	5,534
C	192~223	3 (110)	21	2,097,150	8	254

Address Class	First Network ID	Last Network ID
Class A	1.0.0.0	126.0.0.0
Class B	128.0.0.0	191.255.0.0
Class C	192.0.0.0	223.255.255.0

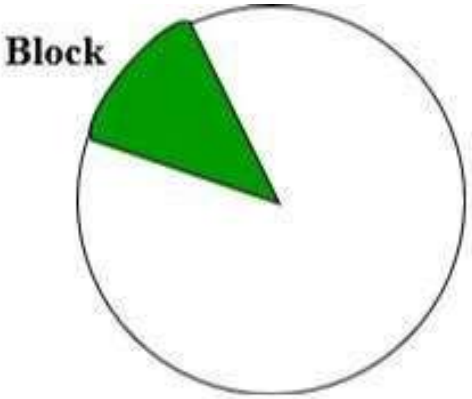
Classless Inter Domain Routing (CIDR)

- In the Classful addressing the number of Hosts within a network always remains the same depending upon the class of the Network.

  - ✓ Class A network contains 2^{24} Hosts,
  - ✓ Class B network contains 2^{16} Hosts,
  - ✓ Class C network contains 2^{8} Hosts

- Now, let's suppose an Organization requires 2^{14} hosts, then it must have to purchase a Class B network. In this case, 49152 Hosts will be wasted. This is the major drawback of Classful Addressing.

- In order to reduce the wastage of IP addresses a new concept of **Classless Inter-Domain Routing** is introduced. Now a days *IANA* is using this technique to provide the IP addresses. Whenever any user asks for IP addresses, IANA is going to assign that many IP addresses to the User.



- **Representation:** It is as also a 32-bit address, which includes a special number which represents the number of bits that are present in the Block Id.

a . b . c . d / n

Rules for forming CIDR Blocks:

1. All IP addresses must be contiguous.
2. Block size must be the power of 2 (2n).

   If the size of the block is the power of 2, then it will be easy to divide the Network. Finding out the Block Id is very easy if the block size is of the power of 2.

Example:

- If the Block size is 2^5 then, Host Id will contain 5 bits and Network will contain 32 – 5 = 27 bits.

1.First IP address of the Block must be evenly divisible by the size of the block. in simple words, the least significant part should always start with zeroes in Host Id. Since all the least significant bits of Host Id is zero, then we can use it as Block Id part.

Example:

Check whether 100.1.2.32 to 100.1.2.47 is a valid IP address block or not?

1. All the IP addresses are contiguous.
2. Total number of IP addresses in the Block = 16 = 2^4.
3. 1st IP address: 100.1.2.00100000

Since, Host Id will contains last 4 bits and all the least significant 4 bits are zero. Hence, first IP address is evenly divisible by the size of the block.

All the three rules are followed by this Block. Hence, it is a valid IP address block.

What is Subnetting?

- Subnetting is a process of dividing a single large network in multiple smaller networks.
- A single large network is just like a town without any sector and street address. In such a town, a postman may take 3 to 4 days in finding a single address. While if town is divided in sectors and streets, he can easily find any address in less than one hour.

Finding in whole town

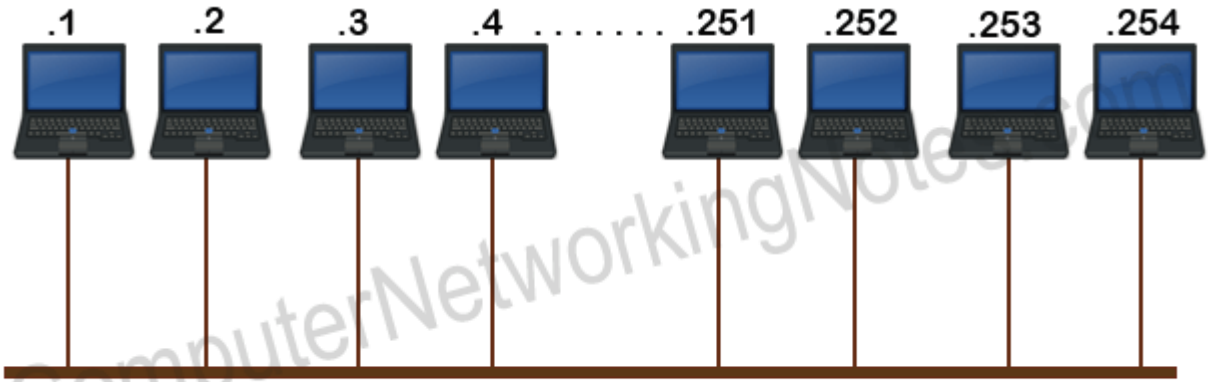
Finding in a specific street

Subnetting

- Subnetting is used to divide a large IP network in smaller IP networks known as subnets.

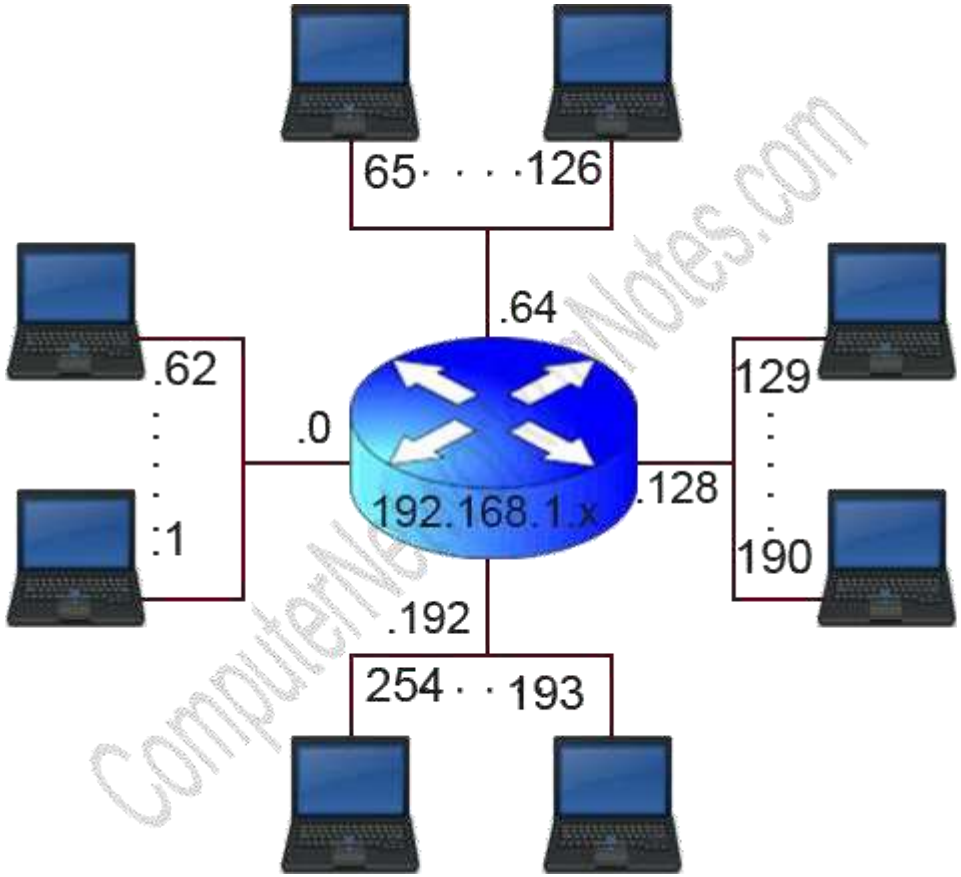
- A default class A, B and C network provides 16777214, 65534, 254 hosts respectively. Having so many hosts in a single network always creates several issues such as broadcast, collision, congestion, etc.

- Let’s take a simple example. In a company there are four departments; sales, production, development and management. In each department there are 50 users. Company used a private class C IP network. Without any Subnetting, all computers will work in a single large network.



A single large class C IP Network

- Since company has four departments, it can divide its network in four subnets.
- Following figure shows same network after Subnetting.



Subnetting table

Description	Network 1	Network 2	Network 3	Network 4
Network address	192.168.1.0	192.168.1.64	192.168.1.128	192.168.1.192
valid hosts	192.168.1.1 to 192.168.1.62	192.168.1.65 to 192.168.1.126	192.168.1.129 to 192.168.1.190	192.168.1.193 to 192.168.1.254
Broadcast address	192.168.1.63	192.168.1.127	192.168.1.191	192.168.1.255

Advantage of Subnetting

•Subnetting reduces network traffic by allowing only the broadcast traffic which is relevant to the subnet.

• By reducing unnecessary traffic, Subnetting improves overall performance of the network.

• By blocking a subnet' traffic in subnet, Subnetting increases security of the network.

Disadvantage of Subnetting

•Different subnets need an intermediate device known as router to communicate with each other.

•Subnetting adds complexity in network. An experienced network administrator is required to manage the subnetted network.

Supernetting

- Supernetting is the opposite of Subnetting. In subnetting, a single big network is divided into multiple smaller subnetworks.
- In Supernetting, multiple networks are combined into a bigger network termed as a Supernetwork or Supernet.
- Supernetting is mainly used in Route Summarization, where routes to multiple networks with similar network prefixes are combined into a single routing entry, with the routing entry pointing to a Super network, encompassing all the networks.
- This in turn significantly reduces the size of routing tables and also the size of routing updates exchanged by routing protocols.

There are some points which should be kept in mind while supernetting:

1. All the Networks should be contiguous.
2. The block size of every networks should be equal and must be in form of 2^n.
3. First Network id should be exactly divisible by whole size of supernet.

Example – Suppose 4 small networks of class C

200.1.0.0,
200.1.1.0,
200.1.2.0,
200.1.3.0

Build a bigger network which have a single Network Id.

First, lets check whether three condition are satisfied or not:

Contiguous: You can easily see that all network are contiguous all having size 256 hosts.

Range of first Network from 200.1.0.0 to 200.1.0.255. If you add 1 in last IP address of first network that is 200.1.0.255 + 0.0.0.1, you will get the next network id that is 200.1.1.0. Similarly, check that all network are contiguous.

Equal size of all network: As all networks are of class C, so all of the have a size of 256 which in turn equal to 2^8.

First IP address exactly divisible by total size:

- When a binary number is divided by 2^n then last n bits are the remainder.
- Hence in order to prove that first IP address is exactly divisible by while size of Supernet Network.
- In given example first IP is 200.1.0.0 .If last 10 bits of first IP address are zero then IP will be divisible.

11001000	00000001	00000000	00000000
200 .	1 .	0 .	0

Last 10 bits of first IP address are zero (highlighted by green color). So 3rd condition is also satisfied.

Therefore, you can join all these 4 networks and can make a Supernet. New Supernet Id will be 200.1.0.0.

IPv6

- The network layer protocol in the TCP/IP protocol suite is currently IPv4 (Internetworking Protocol, version 4).
- IPv4 provides the host-to-host communication between systems in the Internet.
- Although IPv4 is well designed, data communication has evolved since the inception of IPv4 in the 1970s.
- IPv4 has some deficiencies that make it unsuitable for the fast-growing Internet.

Deficiencies:

- Despite all short-term solutions, such as subnetting, classless addressing, and NAT, address depletion is still a long-term problem in the Internet.
- The Internet must accommodate real-time audio and video transmission. This type of transmission requires minimum delay strategies and reservation of resources not provided in the IPv4 design.
- The Internet must accommodate encryption and authentication of data for some applications. No encryption or authentication is provided by IPv4.

To overcome these deficiencies, IPv6 (Internetworking Protocol, version 6), also known as IPng (Internetworking Protocol, next generation), was proposed and is now a standard.

Advantages

The next-generation IP, or IPv6, has some advantages over IPv4 that can be summarized as follows:

Larger address space: An IPv6 address is 128 bits long, compared with the 32-bit address of IPv4, this is a huge (2^{96}) increase in the address space.

Better header format IPv6 uses a new header format in which options are separated from the base header and inserted, when needed, between the base header and the upper-layer data. This simplifies and speeds up the routing process because most of the options do not need to be checked by routers.

New options: IPv6 has new options to allow for additional functionalities.

Advantages

Allowance for extension: IPv6 is designed to allow the extension of the protocol if required by new technologies or applications.

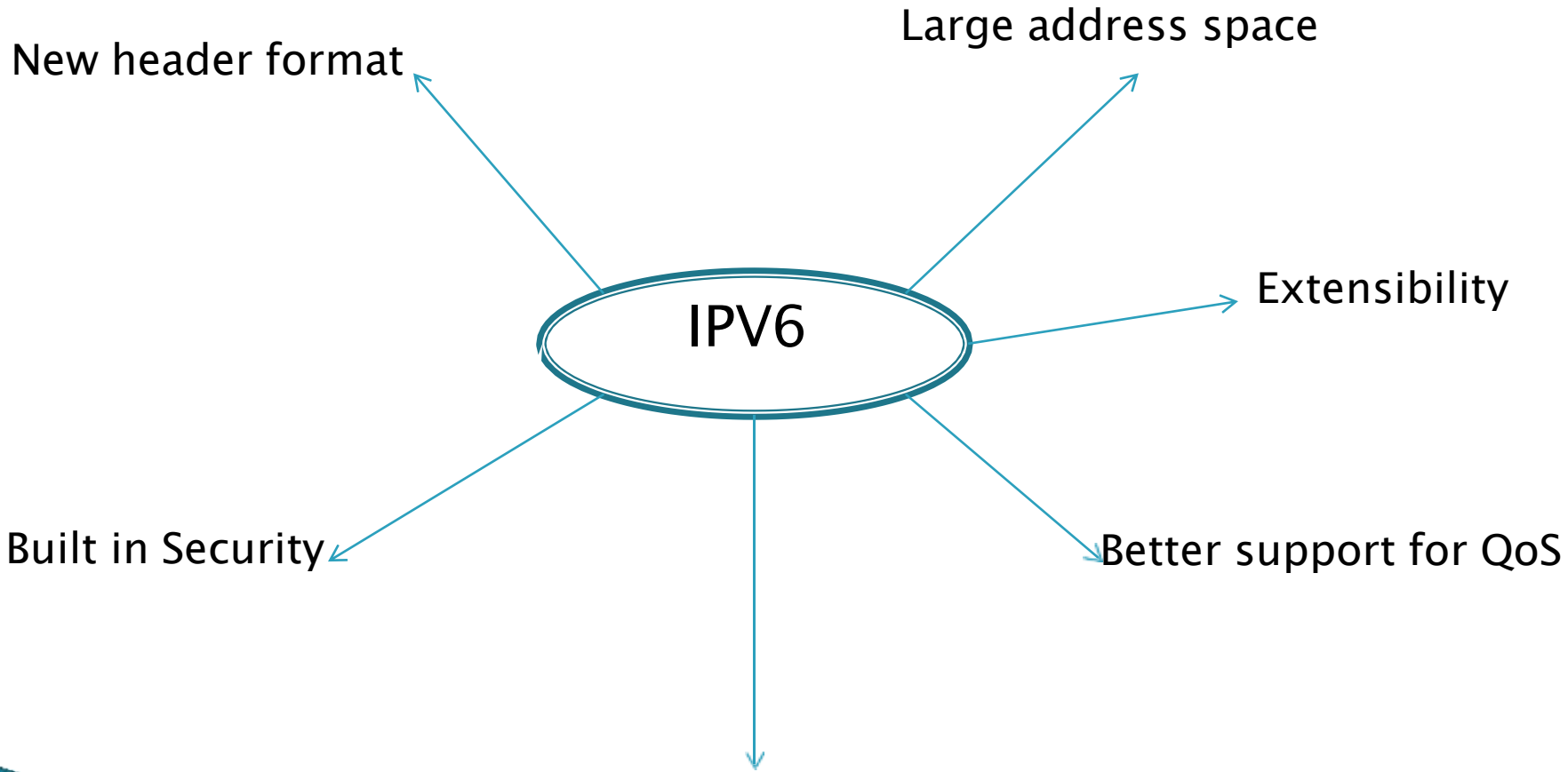
Support for resource allocation: In IPv6, the type-of-service field has been removed, but a mechanism (called flow *label) has been added to enable the source* to request special handling of the packet. This mechanism can be used to support traffic such as real-time audio and video.

Support for more security: The encryption and authentication options in IPv6 provide confidentiality and integrity of the packet.

IPv6 Address Structure

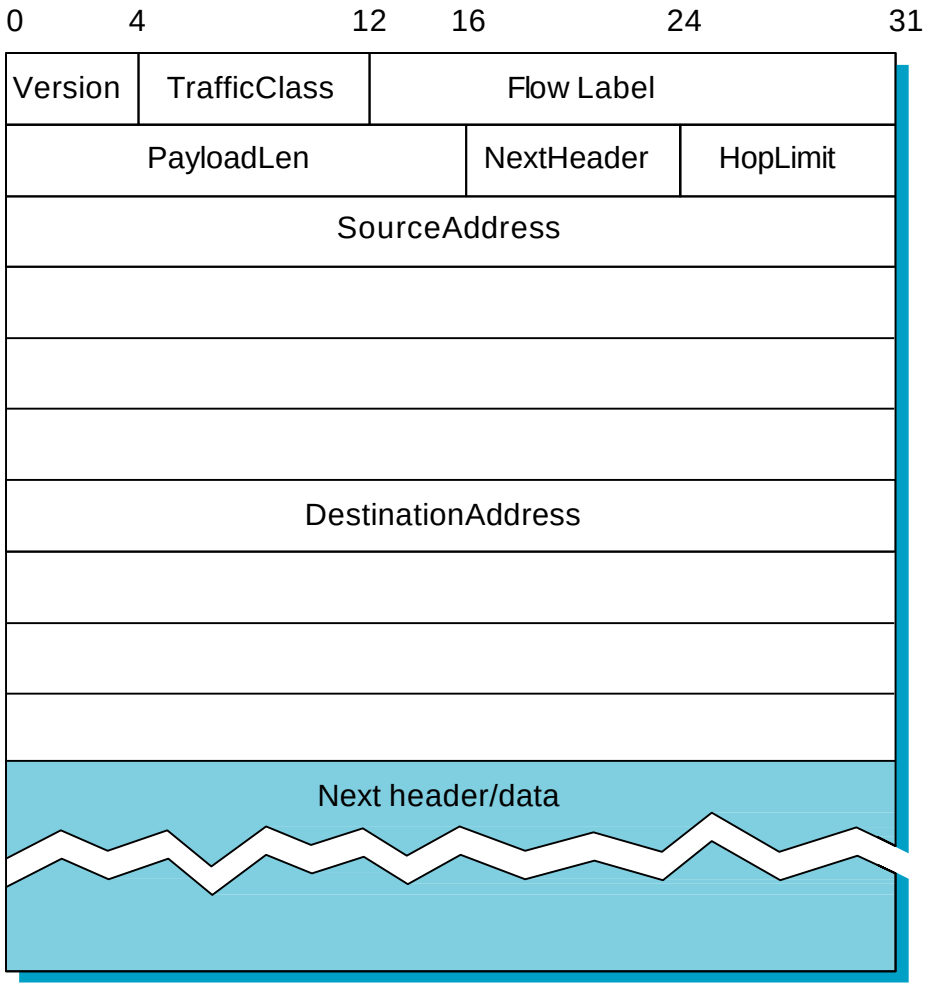
Feature	IPv6
Size of address (bits or byes per octets)	128 bits, 16 octets
Example address	0000:0000:0000:0000:0000:FFFF:FFFF:0A01:0101
Number of possible address, ignoring reserved values	2^{128}, or roughly $3.4 * 10^{38}$

Benefits of IPV6…..



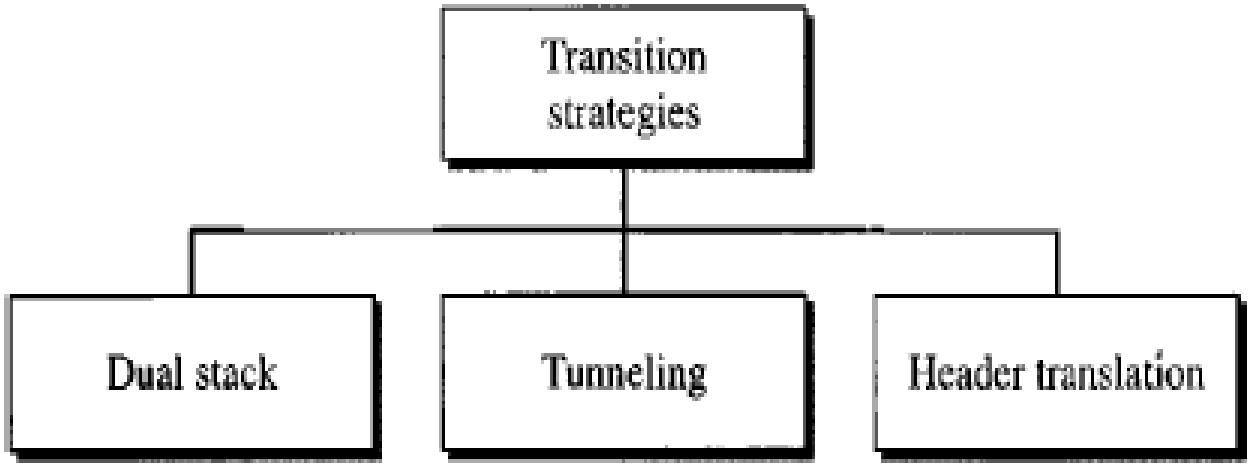
IPv6 Packet Format

- Version: 6 for IPv6
- PayloadLen: the length of the packet in terms of byte, excluding the header
- NextHeader: the upper layer protocol (e.g., TCP or UDP) or the next extended header.
- HopLimit: same as TTL
- SourceAddress and DestinationAddress



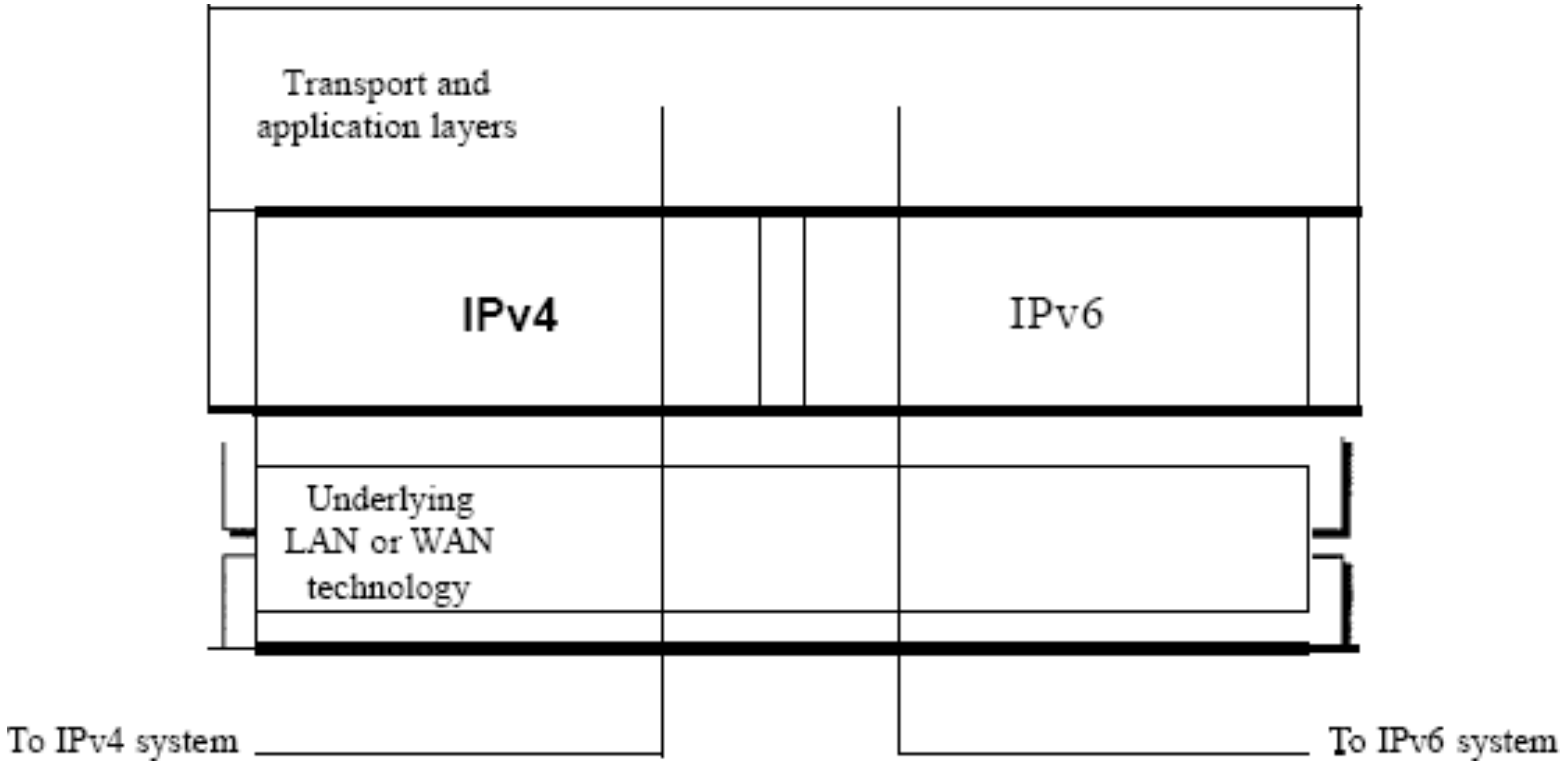
TRANSITION FROM IPv4 TO IPv6:

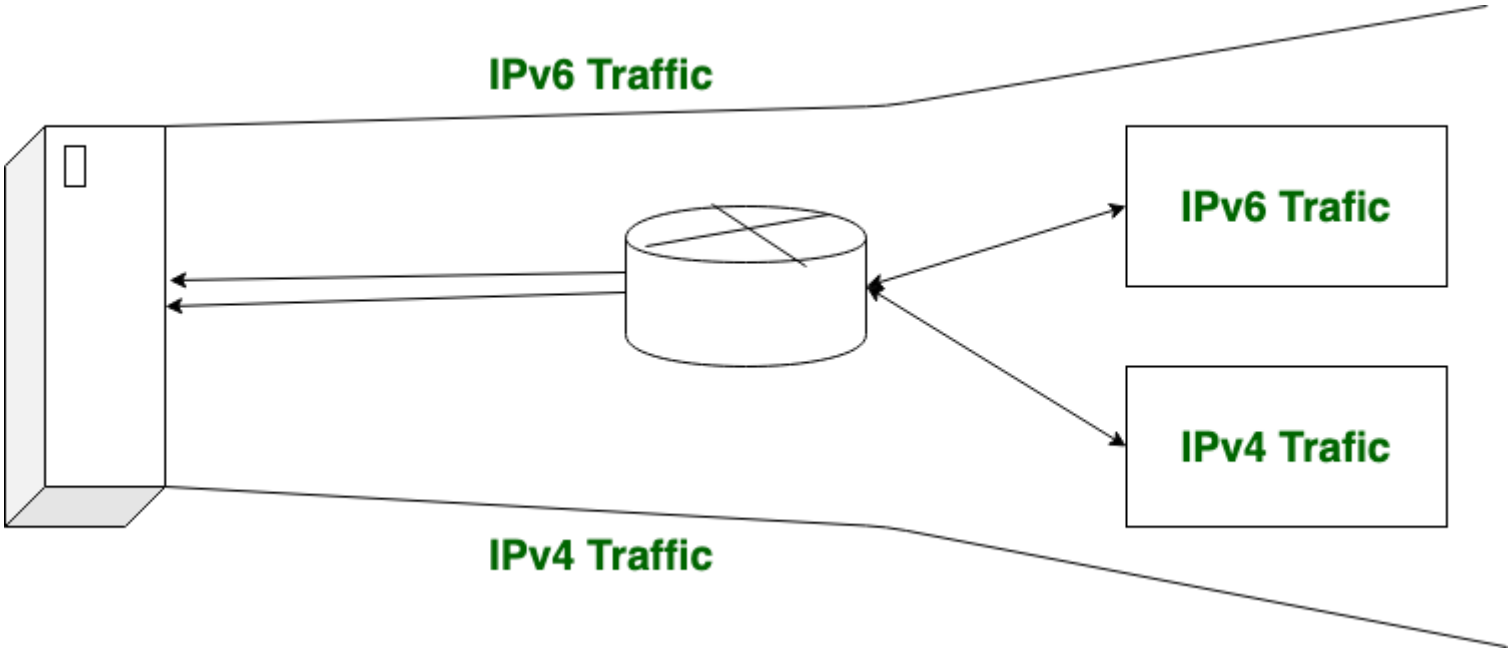
- Because of the huge number of systems on the Internet, the transition from IPv4 to IPv6 cannot happen suddenly.
- It takes a considerable amount of time before every system in the Internet can move from IPv4 to IPv6.
- The transition must be smooth to prevent any problems between IPv4 and IPv6 systems.



Dual Stack

- It is recommended that all hosts, before migrating completely to version 6, have a **dual** stack of protocols.
- In other words, a station must run IPv4 and IPv6 simultaneously until all the Internet uses IPv6.



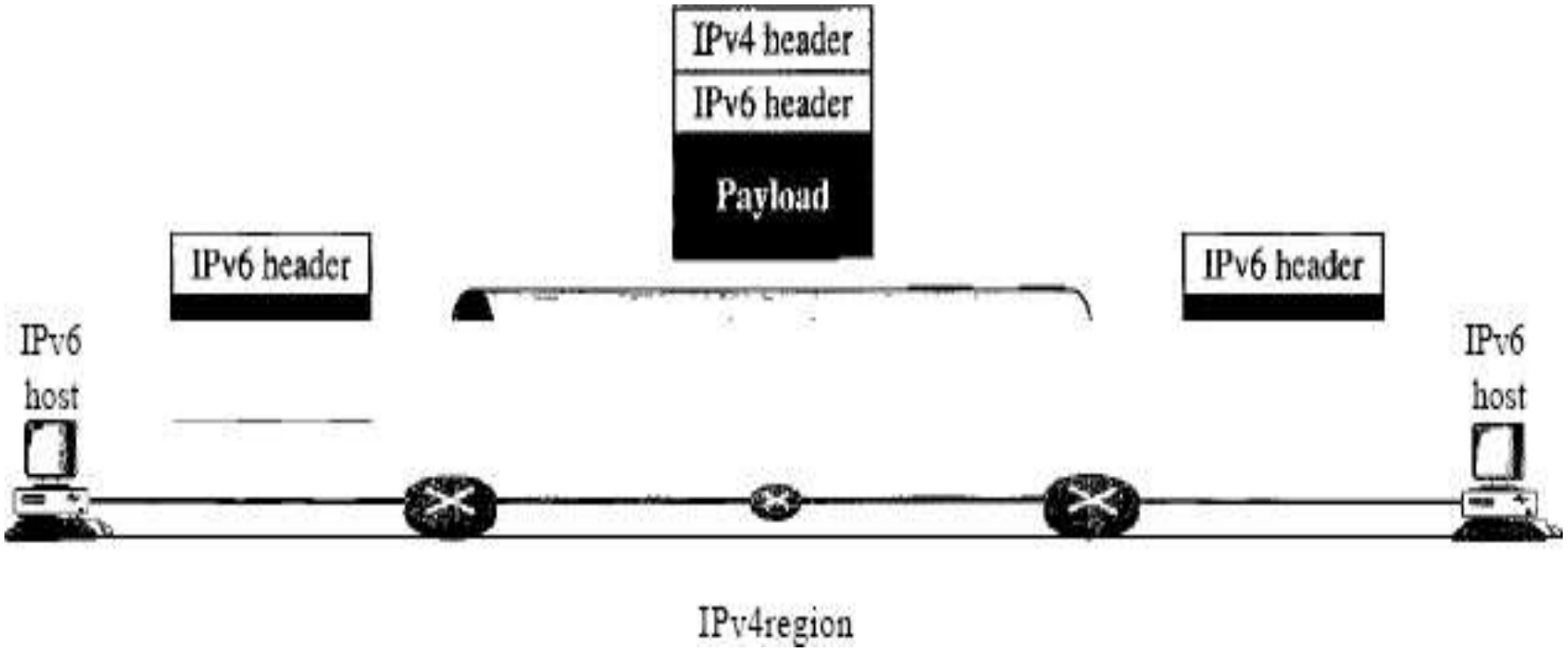

IPv6 Traffic
IPv6 Trafic
IPv4 Trafic
IPv4 Traffic

Tunneling

- Tunneling is a strategy used when two computers using IPv6 want to communicate with each other and the packet must pass through a region that uses IPv4.

- To pass through this region, the packet must have an IPv4 address. So the IPv6 packet is encapsulated in an IPv4 packet when it enters the region, and it leaves its capsule when it exits the region. It seems as if the IPv6 packet goes through a tunnel at one end and emerges at the other end. To make it clear that the IPv4 packet is carrying an IPv6 packet as data.

Tunneling


IPv4 header
IPv6 header
Payload
IPv6 header
IPv6 header
IPv6
host
IPv6
host
IPv4region

Header Translation

- Header translation is necessary when the majority of the Internet has moved to IPv6 but some systems still use IPv4.

- The sender wants to use IPv6, but the receiver does not understand IPv6. Tunneling does not work in this situation because the packet must be in the IPv4 format to be understood by the receiver.

- In this case, the header format must be totally changed through header translation. The header of the IPv6 packet is converted to an IPv4 header

Header Translation

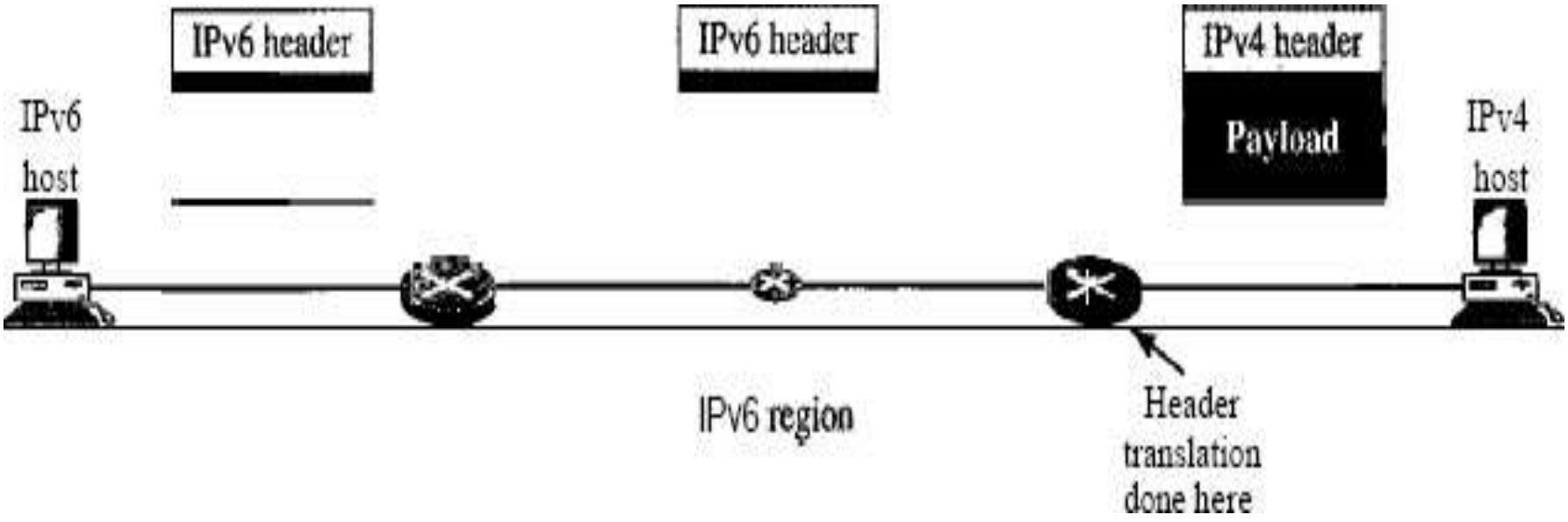
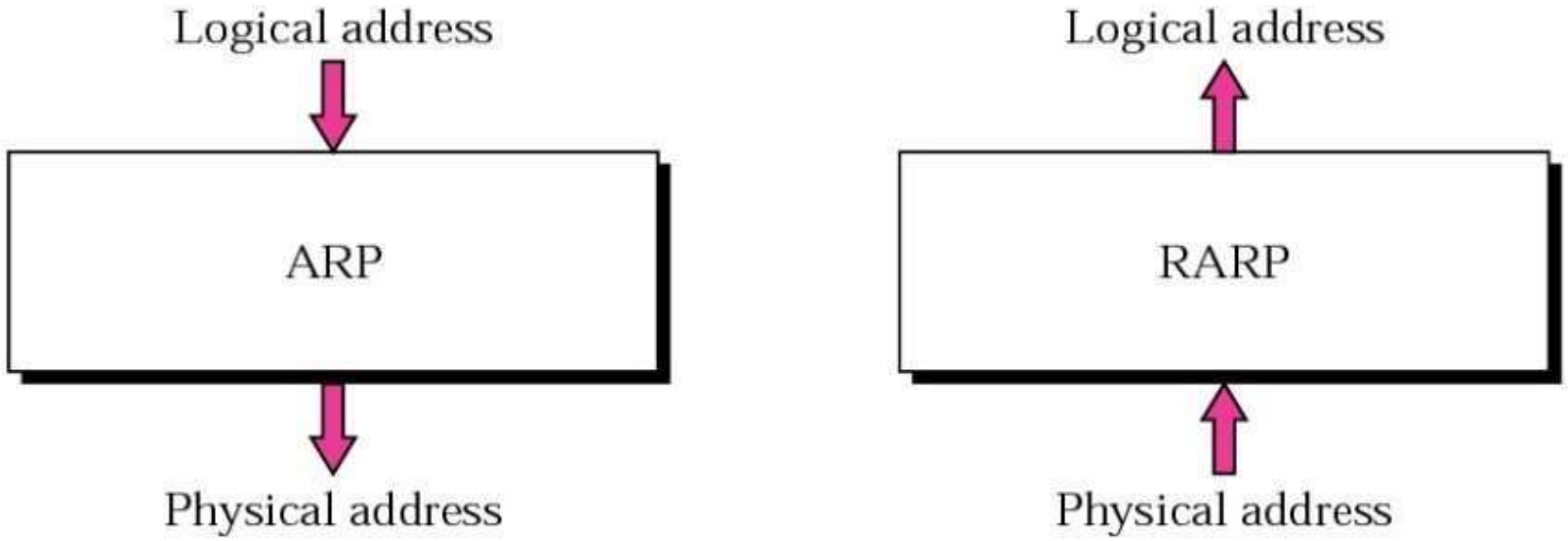


Figure 7.1 *ARP and RARP*

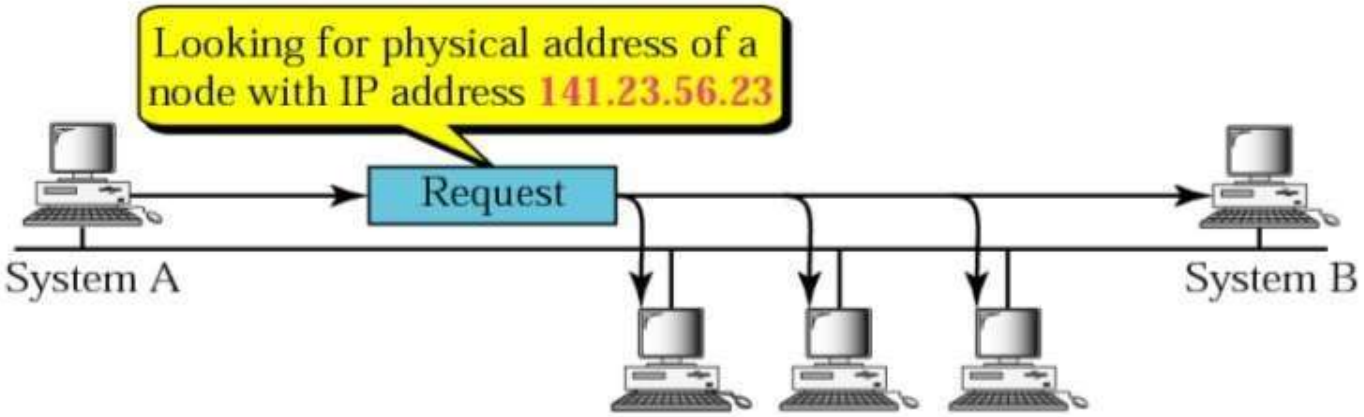


7.1 ARP

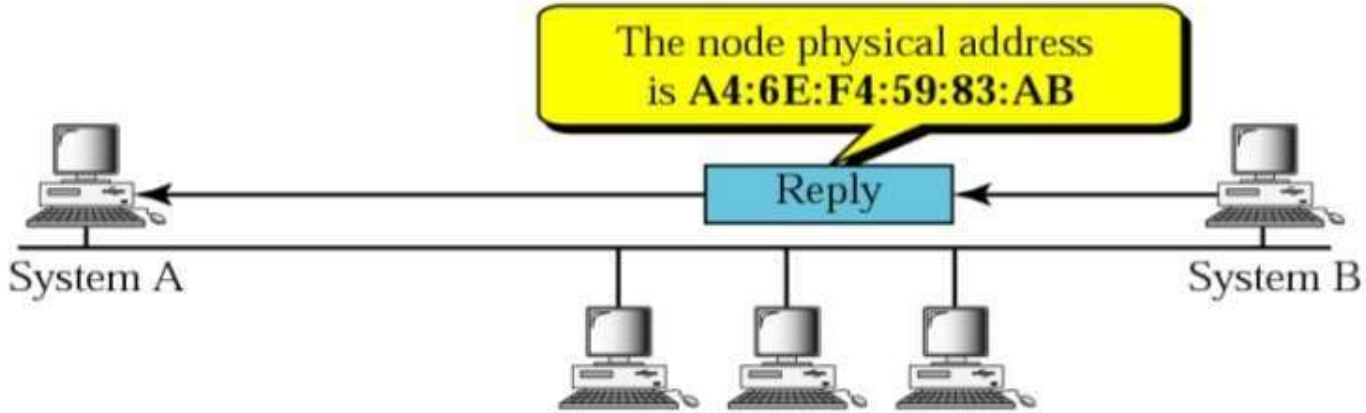
ARP associates an IP address with its physical address. On a typical physical network, such as a LAN, each device on a link is identified by a physical or station address that is usually imprinted on the NIC.

Logical address to physical address translation can be done statically (not practical) or dynamically (with ARP).

Figure 7.3 *ARP operation*

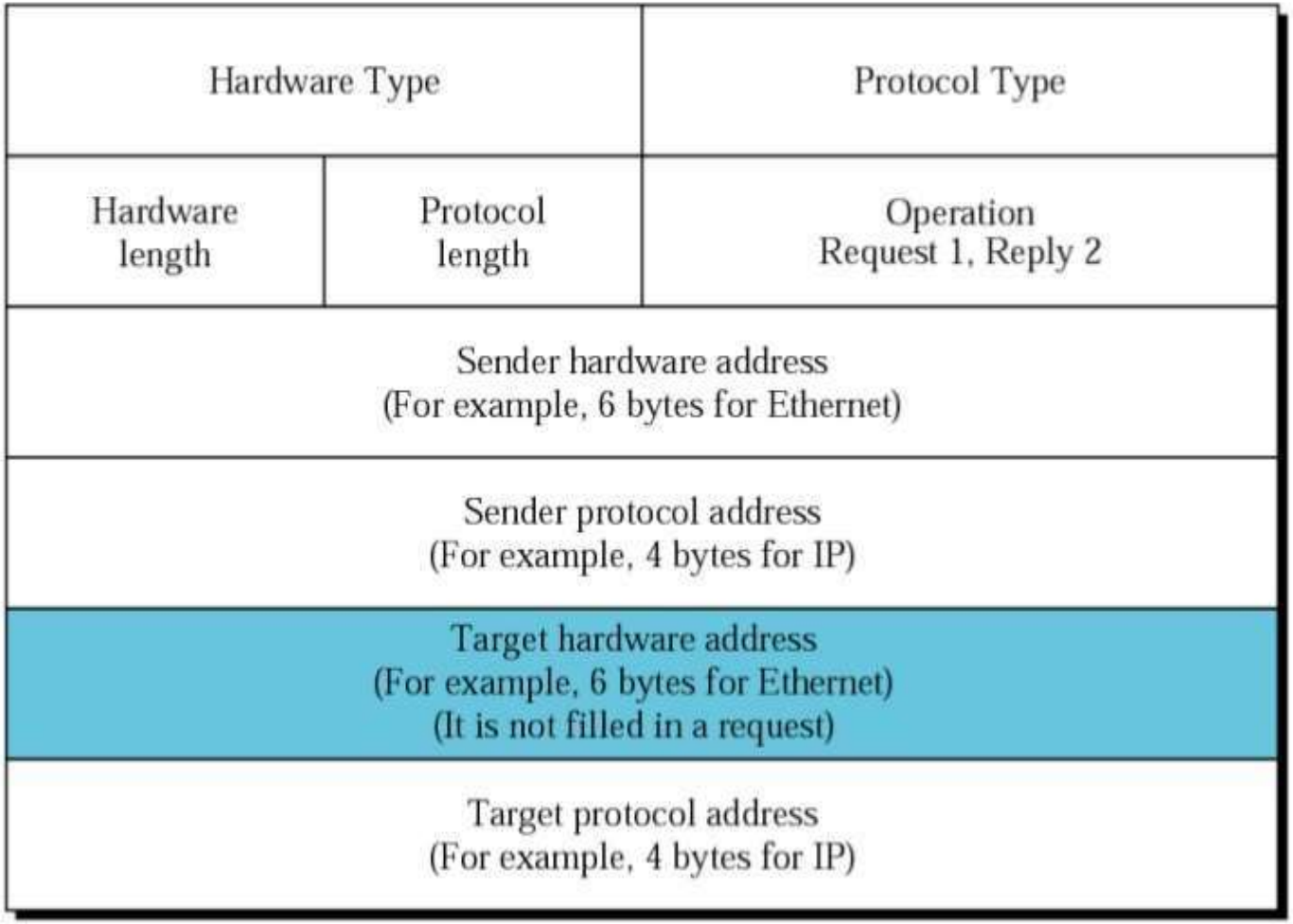


a. ARP request is broadcast



b. ARP reply is unicast

Figure 7.4 *ARP packet*



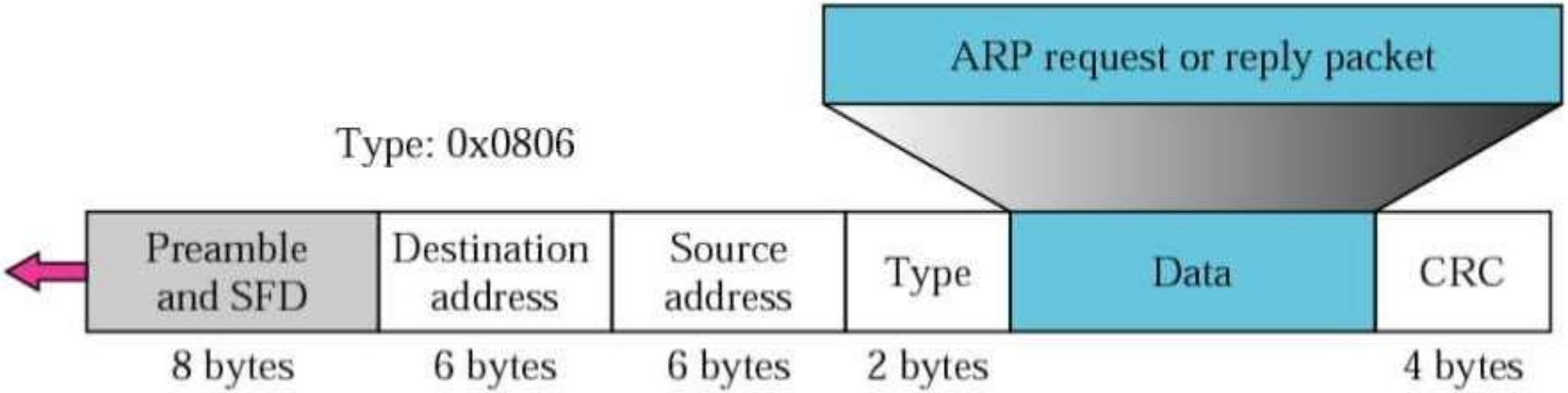
Hardware Type - Ethernet is type 1

Protocol Type- IPv4=x0800

Hardware Length:length of Ethernet Address (6)

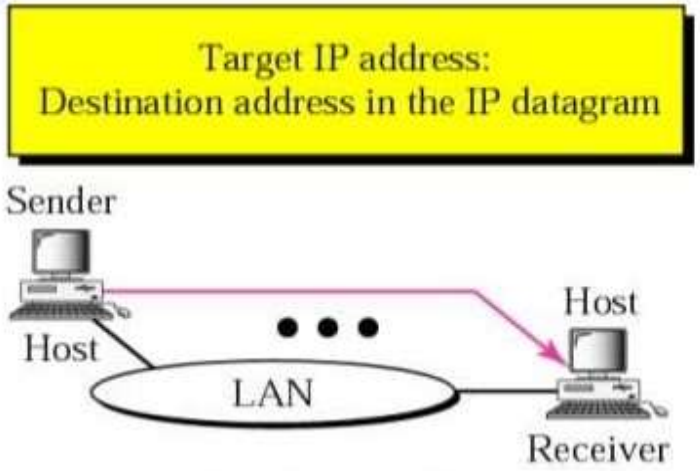
Protocol Length:length of IPv4 address (4)

Figure 7.5 *Encapsulation of ARP packet*

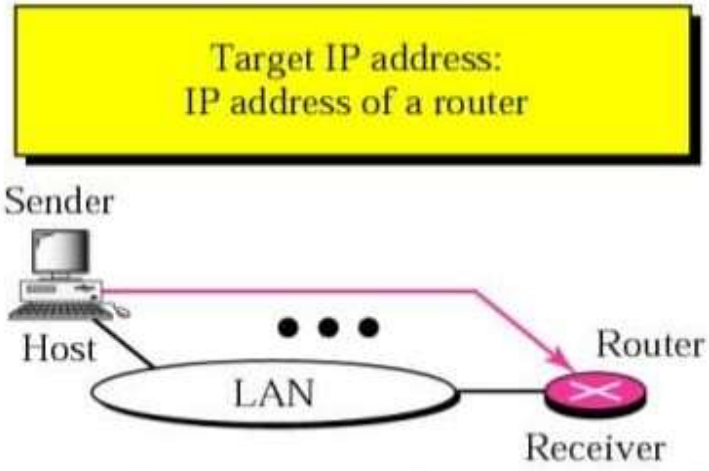


The ARP packet is encapsulated within an Ethernet packet.
Note: Type field for Ethernet is x0806

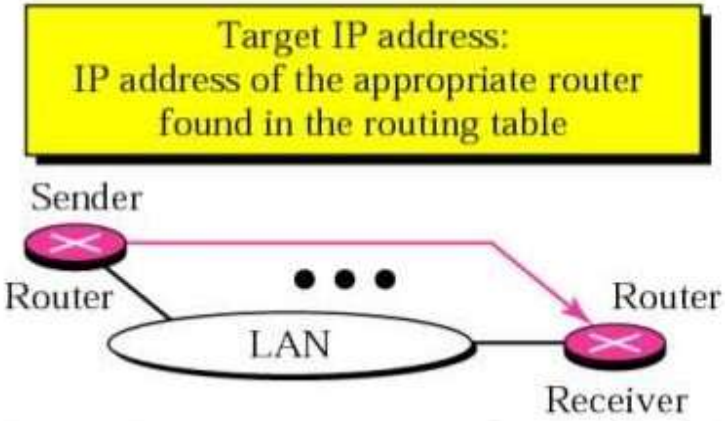
Figure 7.6 *Four cases using ARP*



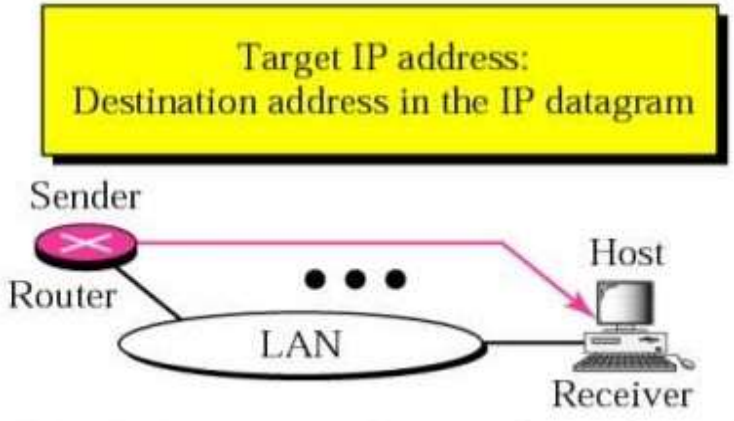
Case 1. A host has a packet to send to another host on the same network.



Case 2. A host wants to send a packet to another host on another network.
It must first be delivered to a router.



Case 3. A router receives a packet to be sent to a host on another network.
It must first be delivered to the appropriate router.



Case 4. A router receives a packet to be sent to a host on the same network.

7.3 RARP

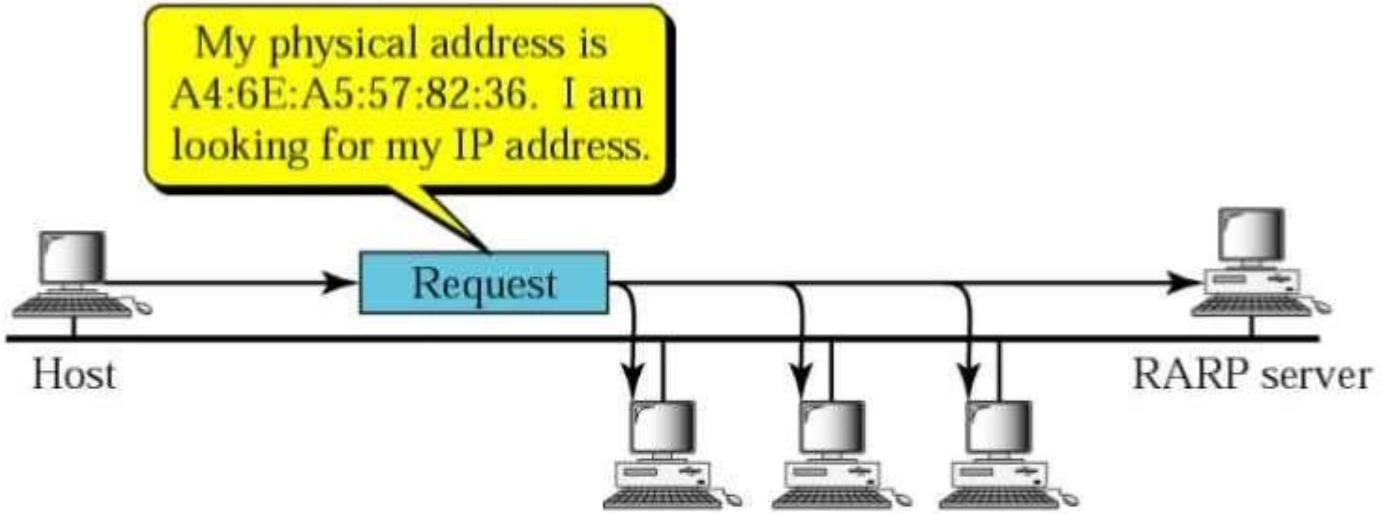
RARP finds the logical address for a machine that only knows its physical address.

This if often encountered on thin-client workstations. No disk, so when machine is booted, it needs to know its IP address (don't want to burn the IP address into the ROM).

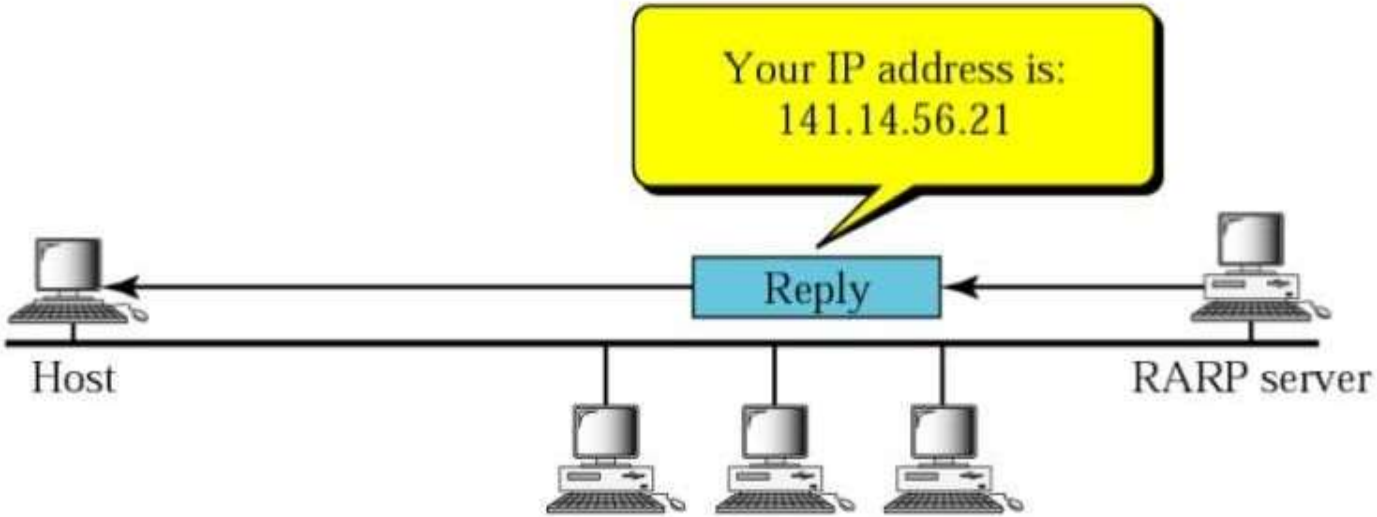
RARP requests are broadcast, RARP replies are unicast.

If a thin-client workstation needs to know its IP address, it probably also needs to know its subnet mask, router address, DNS address, etc. So we need something more than RARP. BOOTP, and now DHCP have replaced RARP.

Figure 7.10 *RARP operation*



a. RARP request is broadcast



b. RARP reply is unicast

Figure 7.11 *RARP packet*

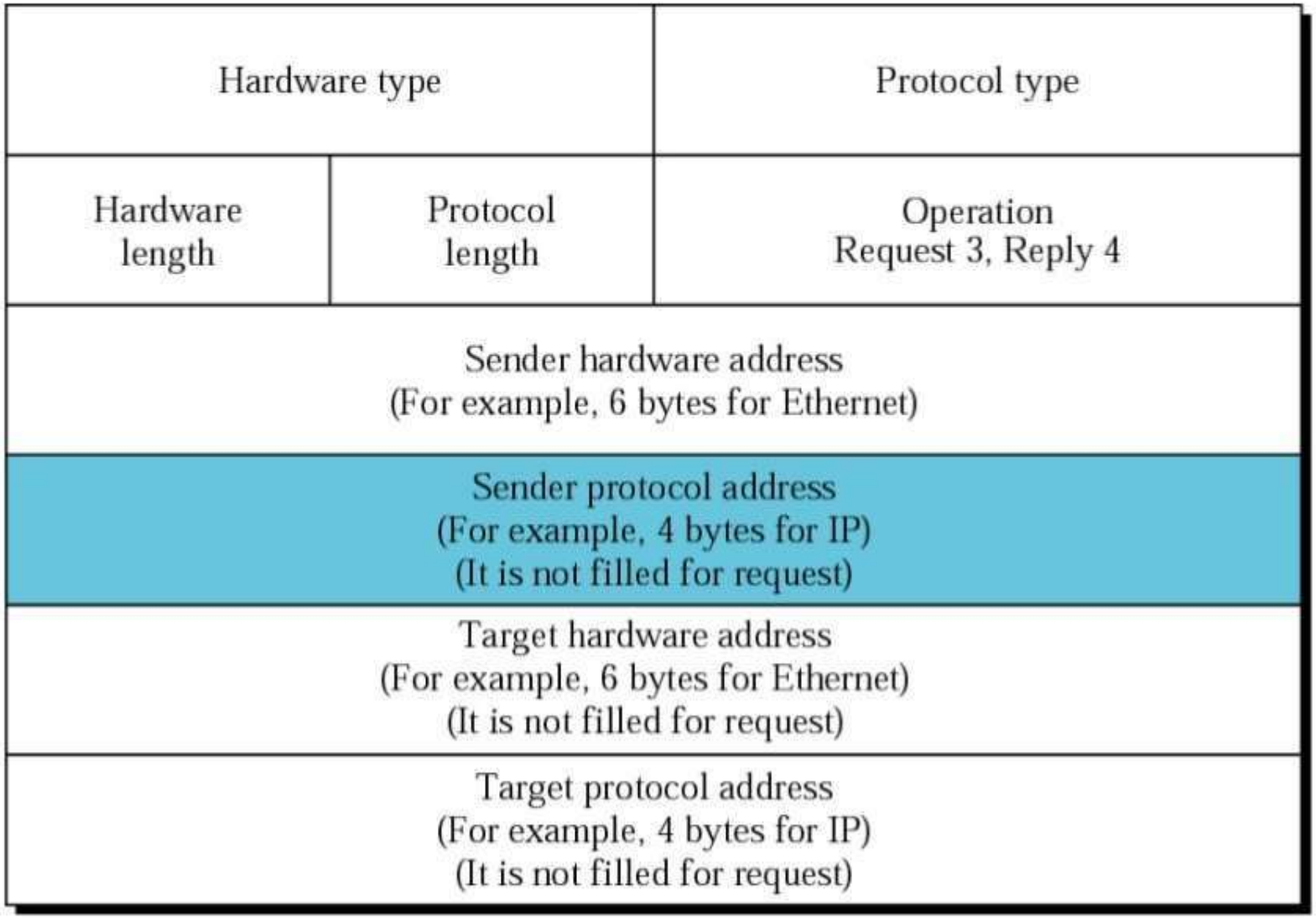
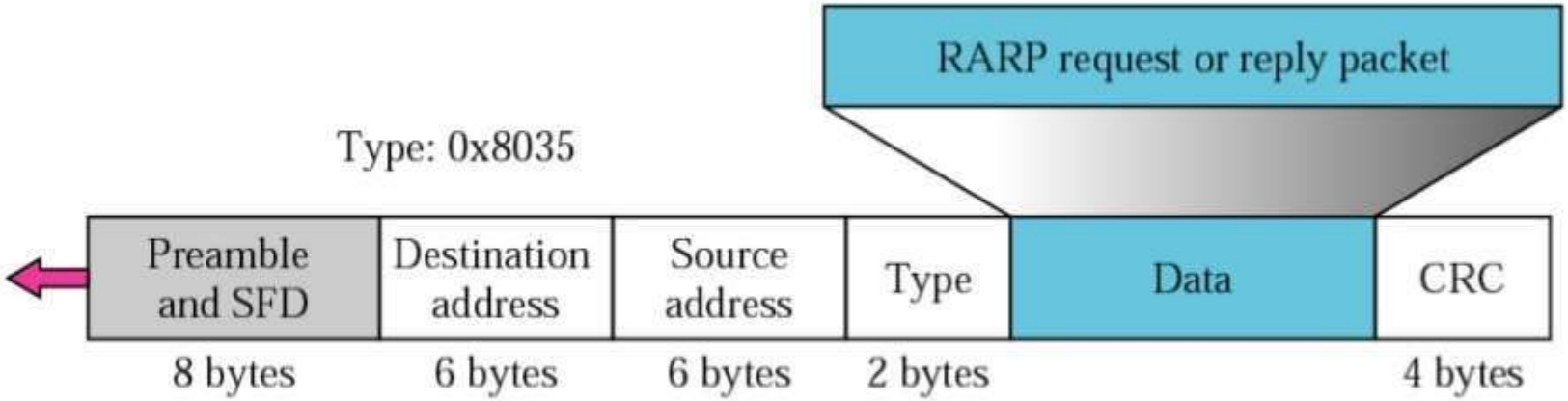


Figure 7.12 *Encapsulation of RARP packet*



FRAGMENTATION

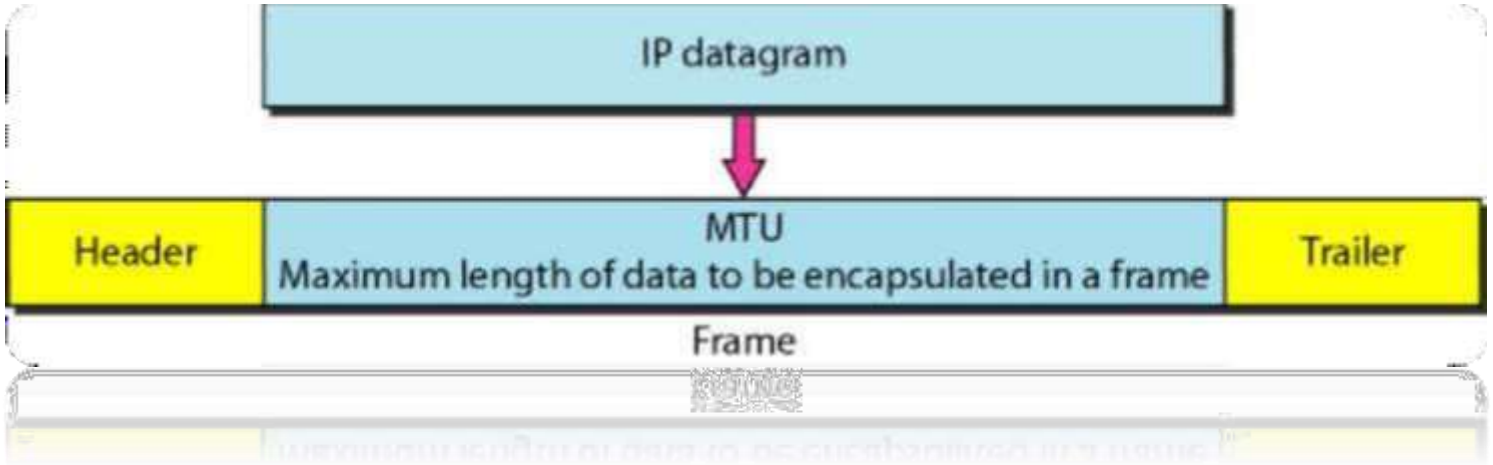
- Packet Fragmentation is a process of dividing the datagram into fragments during its transmission.
- It is done by intermediary devices such as routers at the destination host at network layer.

- Fragmentation is done by the network layer when the maximum size of datagram is greater than maximum size of data that can be held a frame i.e., its Maximum Transmission Unit (MTU). The network layer divides the datagram received from transport layer into fragments so that data flow is not disrupted.

- For example, if a router connects a LAN or WAN, its receives a frame in the LAN format and sends a frame in the WAN format

MAXIMUM TRANSFER UNIT

- Each data link layer protocol has its own frame format in most protocol.
- When a datagram is encapsulated in a frame, the total size of the datagram must be less than its maximum size which is defined by the restriction imposed by the hardware and software used in the network



- To make the IPv4 protocol independent of the physical network, the designers to make the maximum length of the IPv4 datagram equal to 65,535 bytes.