Course Plan			
Module	**Contents**	**Hours**	**End Sem. Exam Marks**
IV	Congestion control algorithms – QoS. Internetworking – Network layer in internet. IPv4 - IP Addressing – Classless and Classfull Addressing. Sub-netting.	**07**	**15%**

Congestion Control Algorithms

Congestion : Too many packets present in the subnet causes congestion. It will result in the degradation of performance.

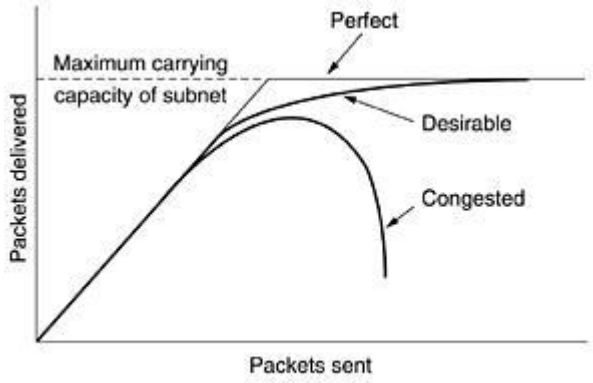


Figure: When too much traffic is offered, congestion sets in and performance degrades sharply

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

However as traffic increases too far, the routers are no longer able to cope and they begin to lose packets. This tends to make matters worse. At very high traffic, performance collapses completely and almost no packets are delivered.

Causes of the congestion are as follows:

- Mismatch in the speed at which the packets are dumped into the router and the speed at which they are delivered.
- Packets may arrive at four or more different lines are send out in single line which may build up congestion. Here the solution is resources may be increased which can handle the huge amount of data or the load can be deceased to avoid congestion. It can also be handled by splitting the output into many lines.
- Insufficient memory can also lead to congestion where the receiver router may not be able to hold the incoming packets. But if we increase the memory then a queue might build up

for the packets and they have to wait for long before they get processed. Therefore the packet delivery speed will be reduced.

- If there occurs a serious congestion spare routers may be kept to handle the congestion.
- Slow processor is another reason for congestion build up.
- Low bandwidth can also cause congestion.

Even though the term flow control and congestion control look same they are different. Congestion control make sure the subnet is able to carry the offered traffic. Flow control make sure that a fast sender cannot actually transmit data faster than the receivers ability to absorb it.

General principles of congestion control

Congestion occurs as

There are two approaches for congestion control

- OPEN LOOP: control before the congestion occurs. Different methods are
  - Retransmission policy
    - Retransmits the data and timers kept to optimize efficiency.
  - Window policy
    - Selective repeat is used here than GoBackN
  - Acknowledgement policy
    - Does not acknowledge every packet
  - Discarding policy
    - The packets are discarded and they are not informed back to the sender to avoid congestion. When the sender does not receive any acknowledgement the same packets are send again.
  - Admission policy
    - Check the packets for the resource requirement of a flow before admitting to the network.
- CLOSED LOOP:it detects the congestion,pass the information to where the action can be taken and adjust the system to correct the problem(detect ,feedback,correct). Different methods are
  - Back pressure
    - Inform the previous router to reduce the rate of outgoing packets
  - Choke packet
    - Packet send back to the sender to inform about the congestion.
  - Implicit signaling
    - Sender slows down the sending rate by detecting implicit signal concerning congestion.
  - Explicit signaling
    - Backward or forward signaling done by packets send from congestion to warn the source.

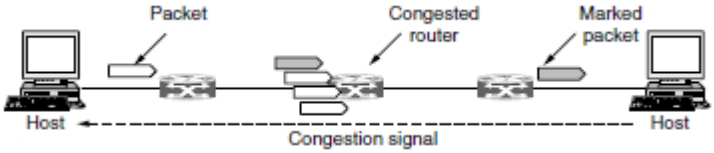


Figure: Explicit Congestion signaling

Congestion Prevention Policies in different layers

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

Figure: Policies that affect Congestion

- Transport Layer
  - Retransmission policy
  - Out of order caching policy
  - Acknowledgement policy
  - Flow control policy
  - Timeout determination
    - If the time out interval is too short extra packets will be sent unnecessarily. If it is too long, congestion will be reduced .
- Network layer
  - Virtual circuits versus datagrams in subnet
    - Congestion control algorithms work only with virtual circuit subnets
  - Packet queuing and service policy
    - It relates to whether the routers have one queue per input line one queue per output line or both. It also relates ot the order in which the packets are processed.(e.g round robin or priority based.)
  - Packet discard policy
    - It tells which packet is dropped . a good policy can help alleviate congestion and a bad one can make it worse.
  - Routing algorithm

        - Good routing algorithm can avoid congestion by spreading the traffic all over the line and abad one can send too much traffic over already congested lines.
    - Packet lifetime management
        - Deals with how long a packet may live before being discarded. If life time is too long lost packets may clog the network, but if it is too short, packets may time out before reaching destination thus inducing retransmission.
- Data link Layer
    - Retransmission policy
        - Concerned with how fast a sender times out and what it transmits upon timeout. Selective repeat or GoBackN can be used for retransmission.
    - Out of order caching policy
    - Acknowledgement policy
        - Here the acknowledgements are saved for a group of packets and piggybacked onto reversed traffic.
    - Flow control policy
        - Tight flow control may be implemented by using small window may help flow control.

Congestion Control in virtual circuit subnet

- Closed loop is designed for virtual circuit subnet.
- ADMISSION CONTROL: keeps congestion from getting worse. It tries not to build up virtual circuits until the congestion is solved. Otherwise it will carefully enroute all virtual circuits.

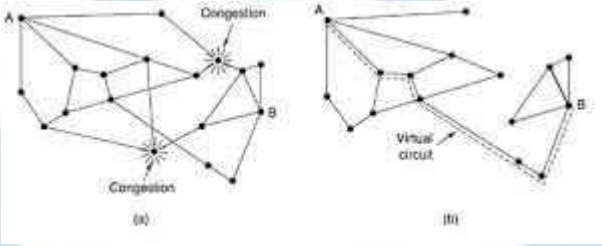


Figure: a) A congested subnet b)a redrawn subnet that eliminates the congestion. A virtual circuit from A to B is also shown

- Another strategies used are volume control; Quality of Service is ensured and other parameters.
- It might keep the resources (tables, buffer, and bandwidth) reserved. This might lead to wastage of resources.

Congestion control in datagram network

- Each router monitors the utilization of the output lines and resources using the formula

U_{new}= aU_{old}+(1-a)f where a is constant,u is utilization value between 0 and 1.0,f instant line utilization either 0 or 1.

- When U reaches above threshold warning state is reached.

Warning Bits

- It is the bit set to control the traffic flow
- Traffic is low until the router ifs free from the heavy traffic

Choke Packets

- To slow down the sending rate of packets by the source the choke packets are used.
- When choke packets are received the source traffic is reduced by X%.
- When the choke packets are received the sender will stop sending the packets for a while. After that it will restart the transmission. After that if the choke packet is again received the host will again cut short the packets. If no choke packet arrives the host may increase rate of transmission.
- Problem with the choke packet is that for high speed WANs return path for choke packet may be so long that too many packets have already been send by the source before the source notes congestion and takes no action.
- Solution for this is use PUSH-BACK or hop-by-hop choke packets.

HOP-BY-HOP packet

- It works as follows
  - The router sends choke packets to the source host
  - The immediate neighbor of the router get information about the traffic congestion
  - It sends back the choke packet after updating its data rate by reducing the data rate.
  - All the routers getting choke packets reduce their data rates immediately without waiting for the original source to reduce the data rate.

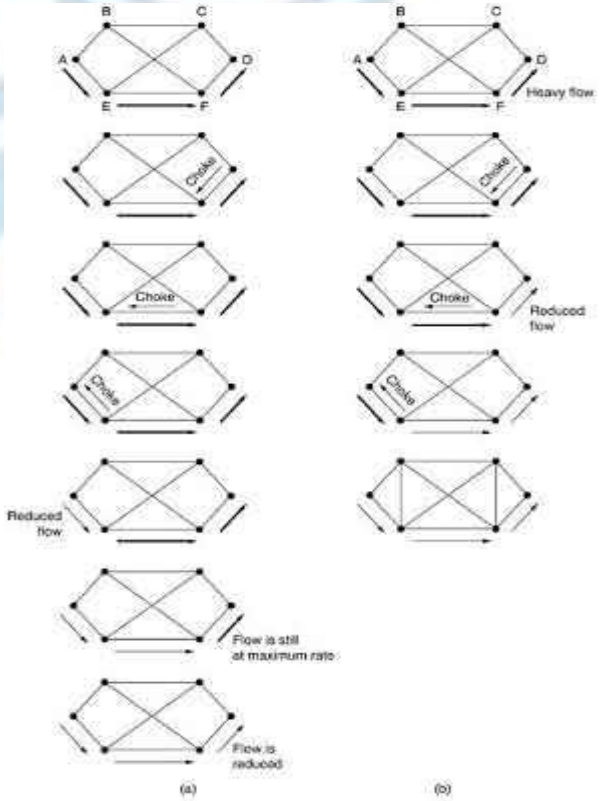


Figure: a) A choke packet that affects only the source. B) A choke packet that affects each hop it passes through

Load Shedding

- Here the router will drop the packets but which one to drop will be decided by the router.
- Which packet to discard depend on the application running
    - Wine policy: for file transfer old packet is worth than a new one. Here dropping old packet may cause more packets to be retransmitted which may cause congestion.
    - Milk Policy: for multimedia data a new packet is more important than old one. Thus fresher is better.

Random Early Detection

- This is an approach where the router discards one or more packets before the buffer becomes completely full.
- Each time a packet arrives , the RED algorithm computes the average queue length. When the average queue length on some lines exceeds threshold the line is said to be congested and action is taken.
- If average queue length is lower than threshold, the congestion is assumed to be minimal or non existant and the packet is queued.
- If average queue length is higher than threshold, congestion is assumed to be serious and the packet is discarded.
- If the average queue length is between the two thresholds, this might indicate the oneset of congestion. The probability of congestion is then calculated.

Jitter Control

- Jitter: it is the variation in the packet arrival time.
- Jitter may cause unevenness . For example if the data received in audio or video gets disturbed it is due to jitter.
- To avoid this a time delay between 24.5ms-25.5 ms delay is kept
- In each hop it is checked whether the packet is ahead of schedule or it is long enough to get it back on schedule. If it is behind the schedule the router tries to get it out of the door quickly.

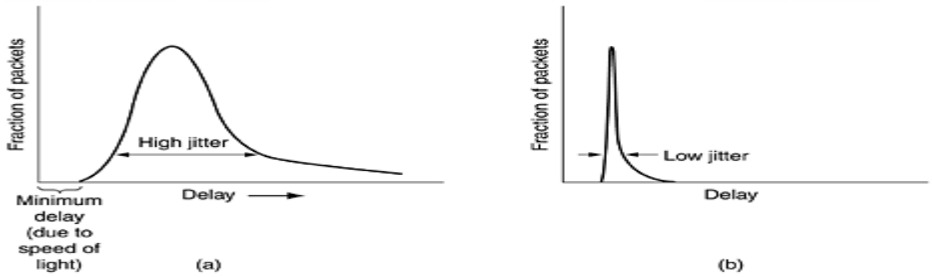


Figure: a) High Jitter b) Low Jitter

Quality of Service(QoS)

- QoS is defined as something a **flow** seeks to attain.

- A stream of packets from a source to destination is called a flow. In a connection oriented network all packets belonging to a flow, follow the same route; in a connection less service they may follow different routes.
- The needs of each flow can be characterized by primary parameters namely reliability, delay, jitter and bandwidth. Together these determine the QoS the flow requires.

Flow Characteristics

1. Reliability: lack of reliability means losing a packet or acknowledgement, which entails retransmission.
2. Delay: application can tolerate delay in different degrees. In this case, telephony, audio conferencing and remote login need minimum delay, while delay in file transfer or email is less important.
3. Jitter: it is the variation in delay for packets belonging to the same flow. High jitter means difference between delay is large; low jitter means variation is small.
4. Bandwidth: different applications need different bandwidth. In video conferencing one need to send millions of bits per second to refresh a color.

Techniques for achieving good QoS

1. Over provisioning
    a. Gives more router capacity, buffer space and bandwidth so that packets fly through easily. But it's expensive
2. Buffering
    a. Data is buffered here. For example in video play the data can be initially fully buffered and the video can be played. This will avoid the jitter related with it.

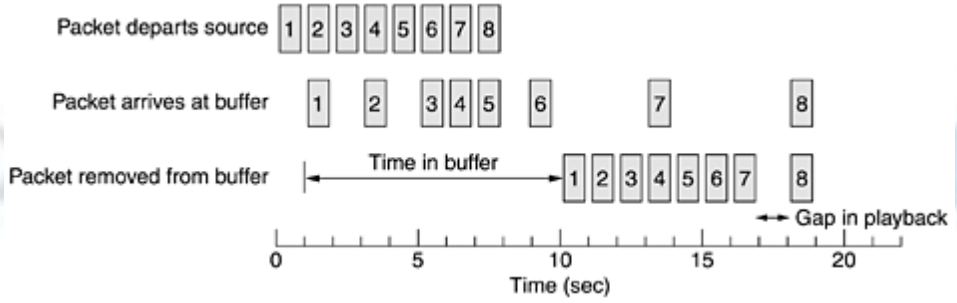


Figure: Smoothing the output stream by buffering packets

3. Traffic shaping
    a. Service level agreement done and signed. Traffic policing done periodically.
    b. Leaky bucket algorithm and token bucket algorithm can be implemented as part of traffic shaping
4. Leaky bucket algorithm
    a. Uneven flow into the host and even flow of packets into the network
    b. Reduces the chance of congestion and smooth out bursts
5. Token Bucket Algorithm
    a. Works on the basis of token where each packet assigned a token.
6. Resource Reservation

   a. First all packets to the same route and flown in one direction. Then the resources are reserved. Commonly used resources are bandwidth, buffer space and CPU cycles are used as resources.

7. Admission control

a. Incoming traffic from some flow is well shaped and can potentially follow a single route in which capacity can be reserved in advance on the routers along the path.
b. When such a flow is offered to a router, it has to decide, based on its capacity and how many commitments it has already made for other flows, whether to admit or reject the flow.
c. The decision to accept or reject a flow is not a simple matter of comparing the (bandwidth, buffers and cycles) requested by the flow with the router's excess capacity in those three dimensions. A set of such parameters is called a **flow specification**. Typically, the sender (e.g., the video server) produces a flow specification proposing the parameters it would like to use. As the specification propagates along the route, each router examines it and modifies the parameters as need be. The modifications can only reduce the flow, not increase it (e.g., a lower data rate, not a higher one). When it gets to the other end, the parameters can be established.

8. Proportional Routing
   a. Most routing algorithms try to find best paths for each destination and send all traffic to that destination over the best path.
9. Packet Scheduling
   a. Fair queuing algorithm used here
   b. The essence of the algorithm is that routers have separate queues for each output line, one for each flow. When a line becomes idle, the router scans the queues round robin, taking the first packet on the next queue. In this way, with *n* hosts competing for a given output line, each host gets to send one out of every *n* packets. Sending more packets will not improve this fraction.

Token Bucket Algorithm

The leaky bucket algorithm described above, enforces a rigid pattern at the output stream, irrespective of the pattern of the input. For many applications it is better to allow the output to speed up somewhat when a larger burst arrives than to loose the data. Token Bucket algorithm provides such a solution. In this algorithm leaky bucket holds token, generated at regular intervals. Main steps of this algorithm can be described as follows:

- In regular intervals tokens are thrown into the bucket.
- The bucket has a maximum capacity.
- If there is a ready packet, a token is removed from the bucket, and thepacket is send.
- If there is no token in the bucket, the packet cannot be send.

Figure 2.1 shows the two scenarios before and after the tokens present in the bucket have been consumed. In Fig. 2.1(a) the bucket holds two tokens, and three packets are waiting to be sent out of the interface, in Fig. 2.1(b) two packets have been sent out by consuming two tokens, and 1 packet is still left.

The token bucket algorithm is less restrictive than the leaky bucket algorithm, in a sense that it allows bursty traffic. However, the limit of burst is restricted by the number of tokens available in the bucket at a particular instant of time.

The implementation of basic token bucket algorithm is simple; a variable is used just to count the tokens. This counter is incremented every *t seconds* and is decremented whenever a packet is sent. Whenever this counter reaches zero, no further packet is sent out as shown in Fig. 1.2

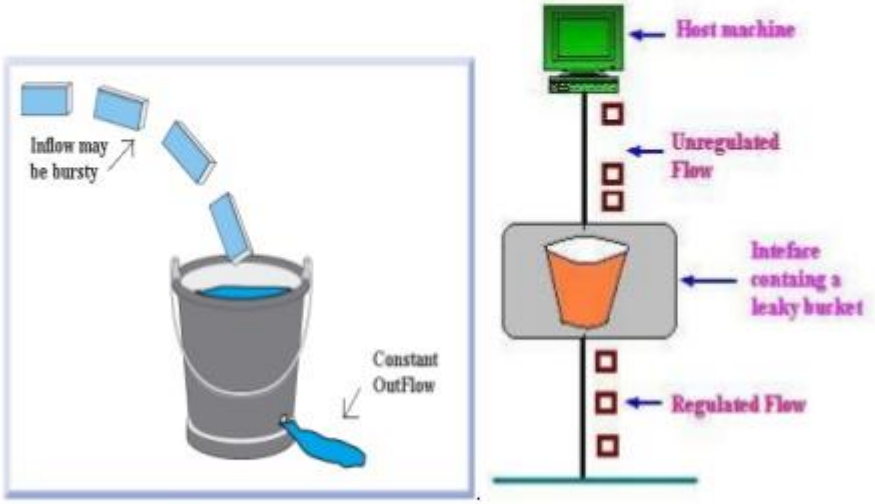
Leaky Bucket Algorithm

Consider a Bucket with a small hole at the bottom, whatever may be the rate of water pouring into the bucket, the rate at which water comes out from that small hole is constant. This scenario is depicted in figure 1(a). Once the bucket is full, any additional water entering it spills over the sides and is lost (i.e. it doesn't appear in the output stream through the hole underneath).

The same idea of leaky bucket can be applied to packets, as shown in Fig. 1(b). Conceptually each network interface contains a *leaky bucket*. And the following steps are performed:

- When the host has to send a packet, the packet is thrown into the bucket.
- The bucket leaks at a constant rate, meaning the network interface transmits packets at a constant rate.
- Bursty traffic is converted to a uniform traffic by the leaky bucket.
- In practice the bucket is a finite queue that outputs at a finite rate.

This arrangement can be simulated in the operating system or can be built into the hardware. Implementation of this algorithm is easy and consists of a finite queue. Whenever a packet arrives, if there is room in the queue it is queued up and if there is no room then the packet is discarded



(a) (b)

Figure 1.(a) Leaky bucket (b) Leaky bucket implementation

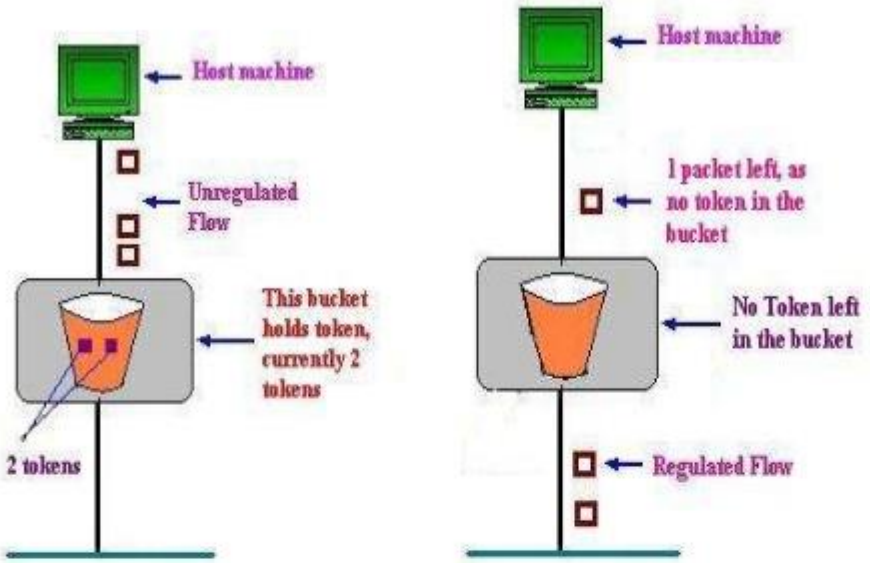


Figure 2.1(a) Token bucket holding two tokens, before packets are send out, (b) Token bucket after two packets are send, one packet still remains as no token is left

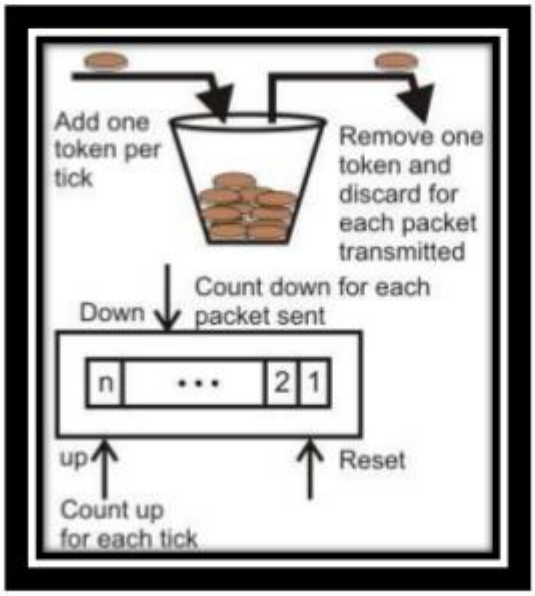


Figure 2.2 Implementation of the Token bucket algorithm

Integrated services

- Architecture for streaming multimedia
- Unicasting and multicasting uses this
- Resource reSerVation Protocol(RSVP)
    - It creates a spanning tree.
    - Procedure is as follows
        - Sender sends RSVP message to the receiver
        - Receiver sends back RSVP reservation message back with specification about the flow
        - If reservation not possible error send back
        - When reservation RSVP reaches the sender the complete path is reserved, the sender begins to send data.
        - To delete outdated reservations timeouts are defined.

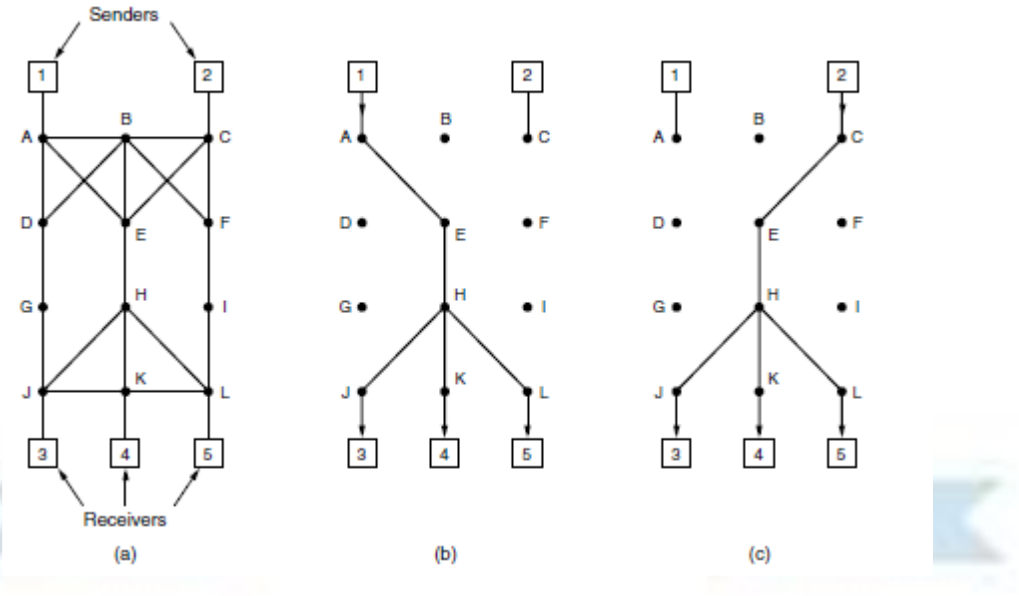


Figure: a) A network b) Multicast spanning tree for host 1 c) Multicast spanning tree for host 2

An example of such a reservation is shown in figure. Here host 3 has requested a channel to host 1. Once it has been established, packets can flow from 1 to 3 without congestion. Now consider what happens if host 3 next reserves a channel to the other sender, host 2, so the user can watch two television programs at once. A second path is reserved, as illustrated in figure (b). Note that two separate channels are needed from host 3 to router *E* because two independent streams are being transmitted.

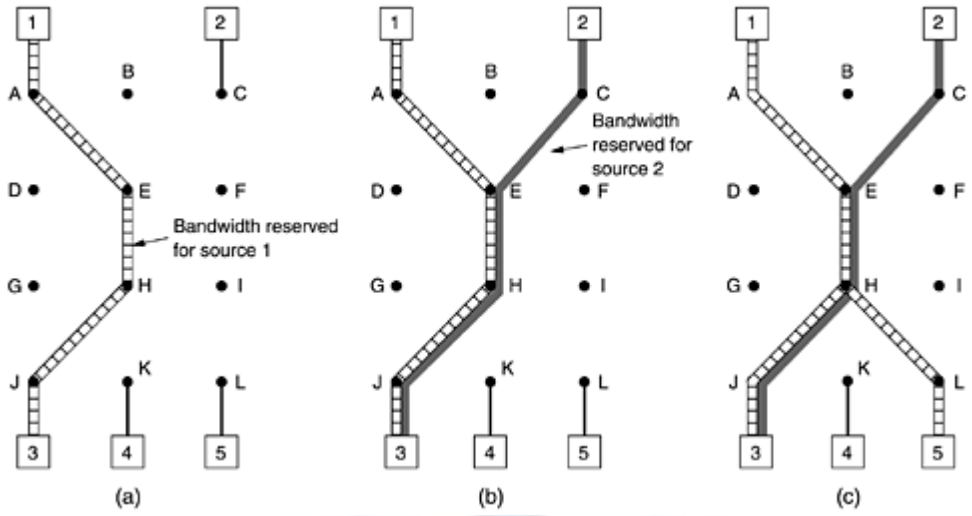


Figure: (a) Host 3 requests a channel to host 1. (b) Host 3 then requests a second channel, to host 2. (c) Host 5 requests a channel to host 1.

Finally, in figure:c, host 5 decides to watch the program being transmitted by host 1 and also makes a reservation. First, dedicated bandwidth is reserved as far as router *H*. However, this router sees that it already has a feed from host 1, so if the necessary bandwidth has already been reserved, it does not have to reserve any more. Note that hosts 3 and 5 might have asked for different amounts of bandwidth (e.g., 3 has a black-and-white television set, so it does not want the color information), so the capacity reserved must be large enough to satisfy the greediest receiver.

Differentiated Services

- Integrated services need connection set up
- Solution to this is local QoS decision making done by differentiated services.
- There are class of application like Low, Medium, High.
- For each class the resources are reserved. Only that class allowed the resources to use at particular instant.
- Here **expedited forwarding** used. In that the expedited packets are passed in such a way as if there would be no or only little traffic. For this routers manage separate queues for these packet types (Weighted Fair Queuing).

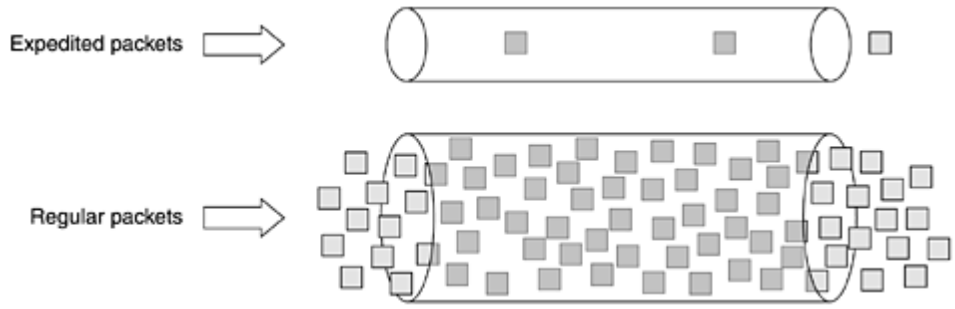


Figure: Expedited packets experience traffic free network

- **Assured Forwarding :** 12 different service classes are used. There are three possibilities for discarding a packet(Low,Medium,High).
- Procedure is as follows
  - First determine the capacity of the routers
  - During high loads low priority packets are discarded
  - By suitable selection of probabilities lower priority level is still forwarded, while packets are discarded.

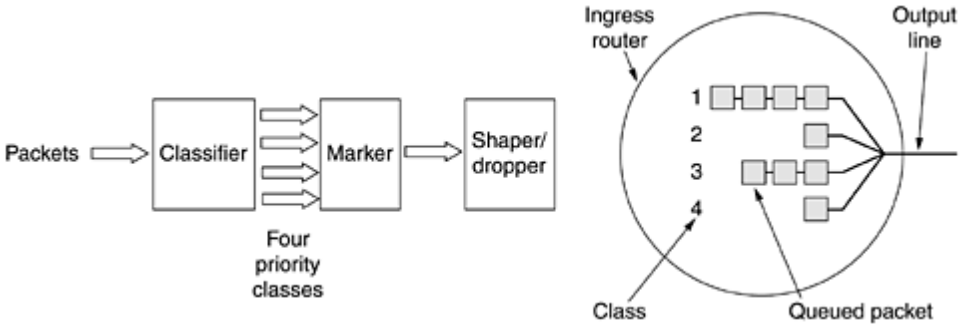


Figure: A possible implementation for Assured forwarding

Label Switching: MPLS

- Each packet assigned a label
- Procedure is as follows
  - First packet determines the path. Routers store the albel information and sends the packets with this labeloner the network.
  - Routers decide on the basis of the label on which outgoing link a packet is forwarded.
  - Ath the same time a new lable is set which instructs the next router.
  - Router manages different queues.
  - At the end of the network the label can be removed and IP address is used for path choice.

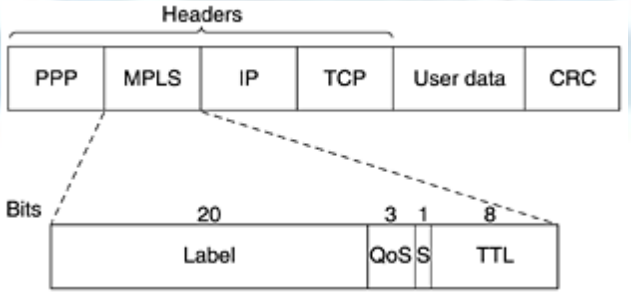
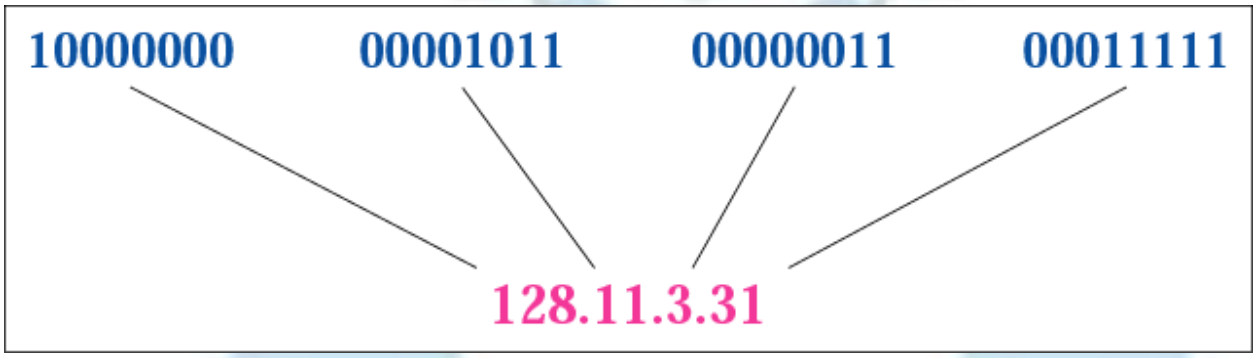


Figure: Transmitting a TCP segment using IP, MPLS, and PPP

IP Addressing

- The identifier used in the IP layer of the TCP/IPprotocol suite to identify each device connected to the Internet is called the Internet address or IP address.
- An IP address is a 32-bit address that uniquely anduniversally defines the connection of a host or a router to the Internet.
- The address space in a protocol that uses N-bits to define an Address is: 2^N
- The address space of IPv4 is 2^{32} or 4,294,967,296.
- Three common notations to show an IP address:
    - Binary Notation
    - Dotted Decimal Notation
    - Hexadecimal Notation
- Binary Notation
    - In binary notation IP address is displayed as 32 bits.
    - To make address more readable, one or more space is usually inserted between each octet.
    - IP address is also referred as 32 bit address, a 4 octet address or a 4- byte address.
    - Eg:01110101 10010101 00011101 11101010

- Dotted-decimal notation
    - Internet addresses are usually written in decimal form with a decimal point separating the bytes.
    - Each number in the dotted- decimal notation is between 0 and 255.



- Hexadecimal Notation
    - Each hexadecimal digit is equivalent to four bits.
    - So a 32- bit address has 8 hexadecimal digits.
    - This notation is used in network programming.
    - Ex.
      1100001 10000011 00011011 11111111 can be represented as
      OXC1831CFF or $C1831CFF_{16}$

      0111 0101 1001 0101 0001 1101 1110 1010

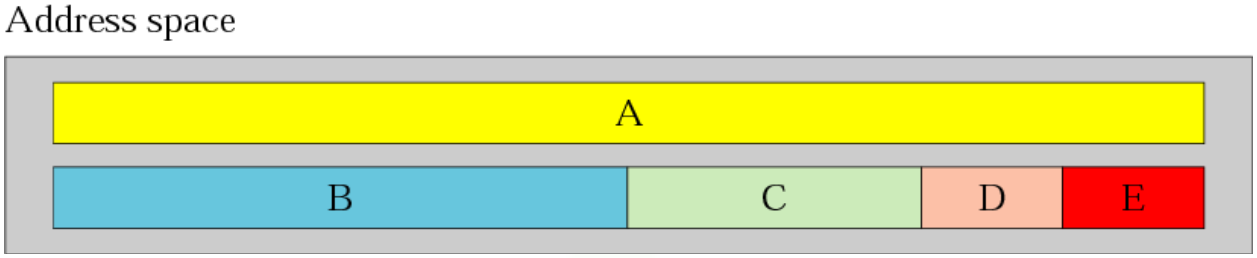
      75 95 1D EA

      0x75951DEA

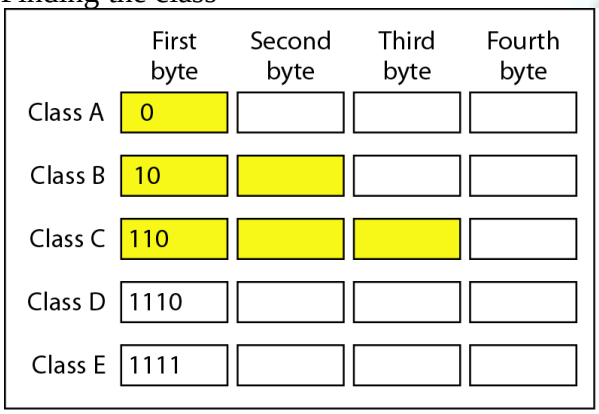
Qn:Change the following IP address from binary notation to dotted-decimal notation.
10000001 00001011 00001011 11101111
Solution ***129.11.11.239***

CLASSFULADDRESSING

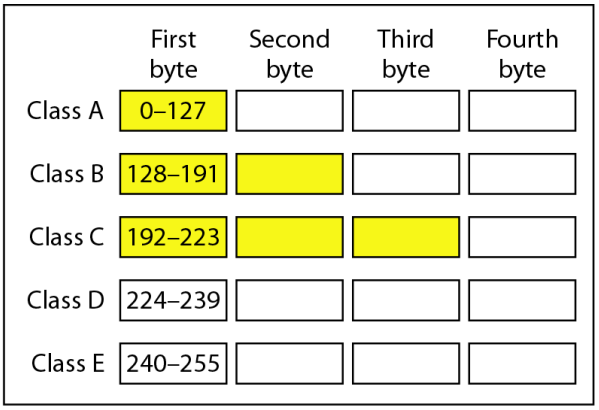
- In classful addressing, the IP address space is divided into five classes: A,B,C,D and E.
- Each part occupies some part of the whole address space.



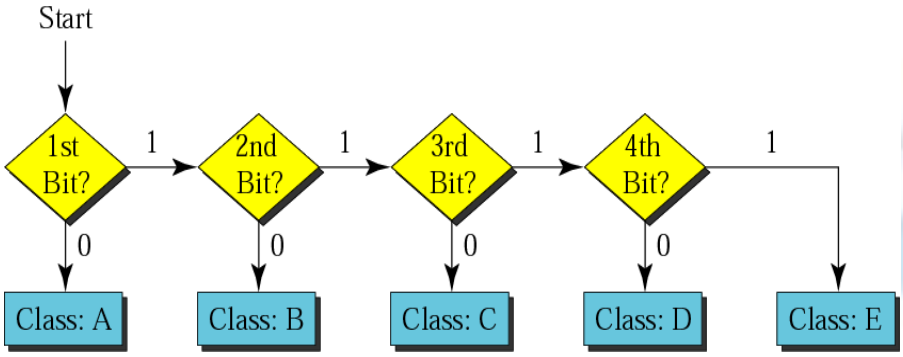
- Finding the class

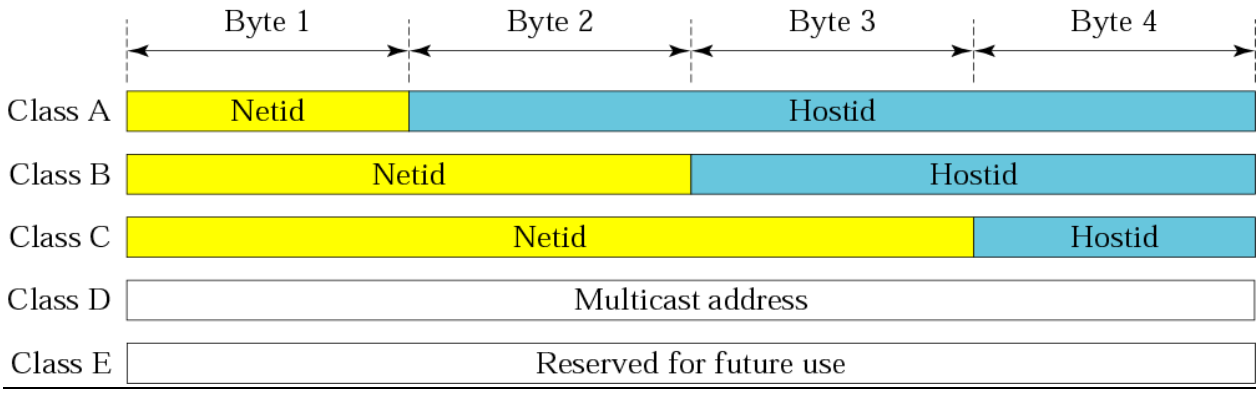


a. Binary notation



b. Dotted-decimal notation

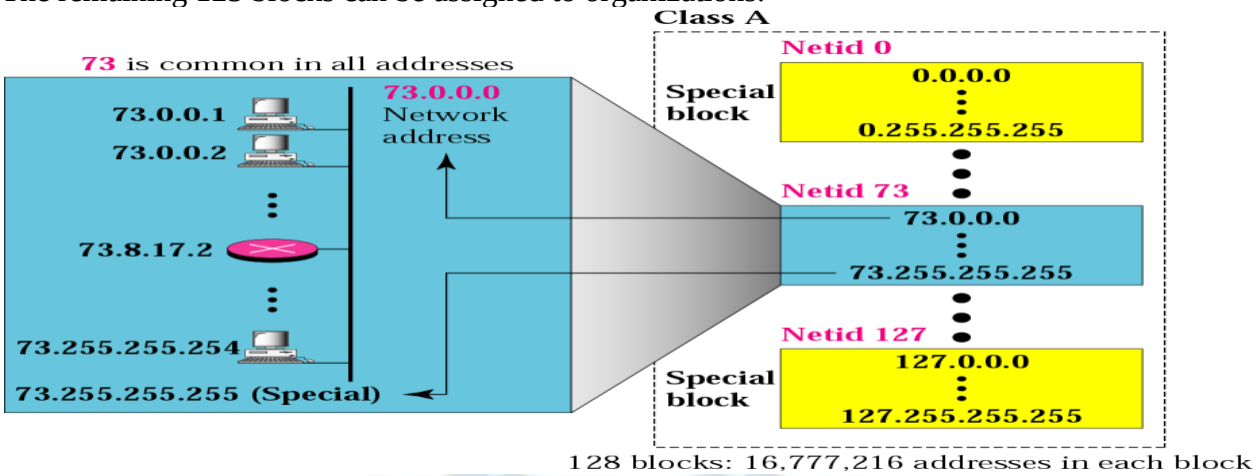
- 

- In classful addressing, an IP address in classes A,B and C is divided into netid and hostid.
- 

- One problem with classful addressing is that each class is divided into fixed number of blocks with each block having a fixed size.

Class A

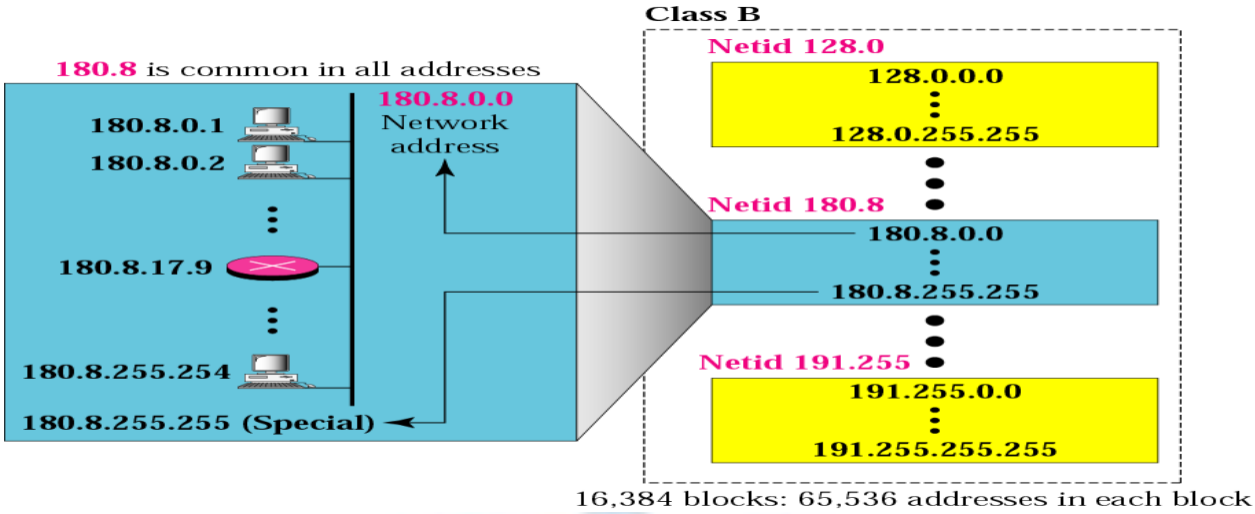
- Class A is divided into 128 blocks with each block having different net id.
- The first block covers addresses from 0.0.0.0 to 255.255.255 (netid 0)
- Each block of addresses the first addresses is same, but the other 3 bytes can take any value.
- The first and last blocks in this class are reserved for special purposes.
- One block (netid 10) is used for private addresses.
- The remaining 125 blocks can be assigned to organizations.



- The first address in the block is used identify the organization to the rest of internet. This address is called network address.
- Class A addresses were designed for large organizations
- Millions of Class A addresses are wasted.

Class B

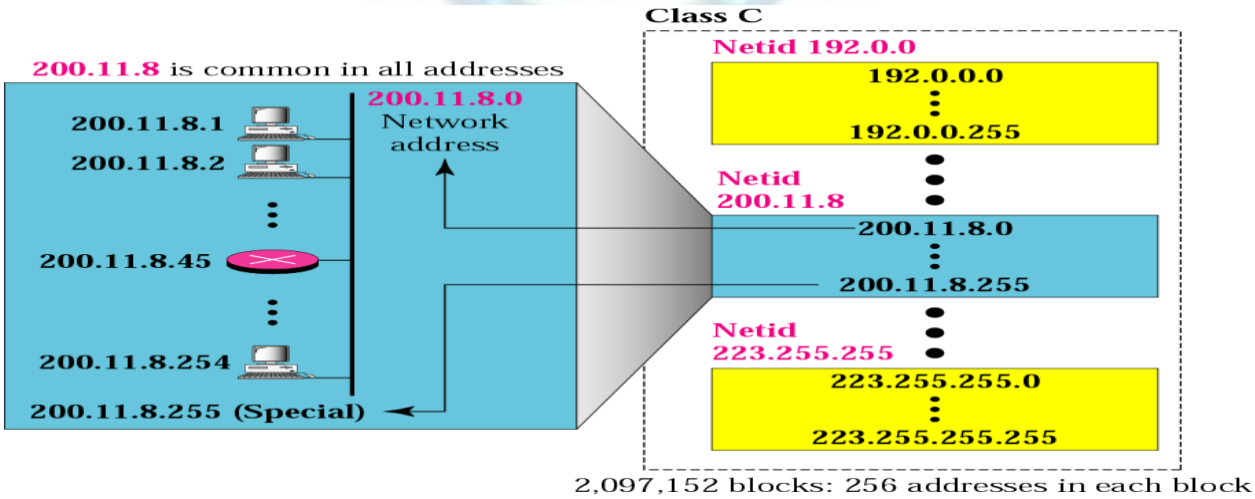
- Class B is divided into 16,384 blocks with each block having different netid.
- 16 blocks are reserved for private addresses, remaining 16368 blocks can be assigned to organizations.
- For each block the first two bytes (netid) are same.
- Class B addresses were designed for mid- sized organizations that may have ten thousands of host or routers.



- The no: of addresses in each block is 65,536, is larger than needs of most organizations.
- So many Class B addresses are wasted.

Class C

- Class C is divided into 2,097,152 blocks with each block having different netid.
- 256 blocks are reserved for private addresses, leaving 2,096,896 blocks can be assigned to organizations.
- Each block in this class has 256 addresses, which means the organizations should be small enough to need less than 256 addresses.



- Class C addresses were designed for small organizations with a smaller number of hosts or routers.
- The number of addresses in Class C is smaller than the needs of most organizations

Class D

- There is only one block in class D addresses.
- It is designed for multicasting.
- Each address in this class is used to define one group of host on the internet.

Class E

- There is only one block in class E addresses.

- It is used as reserved address
- Each address in this class is used to define one group of host on the internet.

Network Addresses

- A network address has several properties:
  - The network address is the first address in the block.
  - The network address defines the network to the rest of the Internet.
  - Given the network address, we can find the class of the address , the block and the range of the addresses in the block.

Mask

- If an address is given, we can find the network address.
- This is important because to route a packet to correct network, a router needs to extract a network address from dest. address in the packet header.
- A mask is a 32- bit binary number that gives the first address in the block when bitwise ANDed with an address in the block.
-

-

Class	*Mask in binary*	*Mask in dotted-decimal*
A	11111111 00000000 00000000 00000000	255.0.0.0
B	11111111 11111111 00000000 00000000	255.255.0.0
C	11111111 11111111 11111111 00000000	255.255.255.0

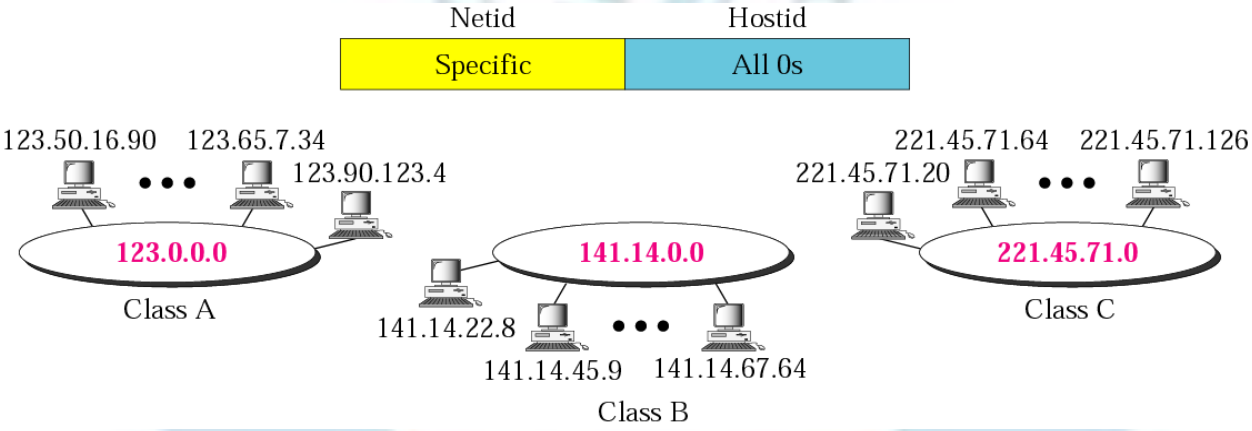
Table : Default masks

- **Special Addresses**

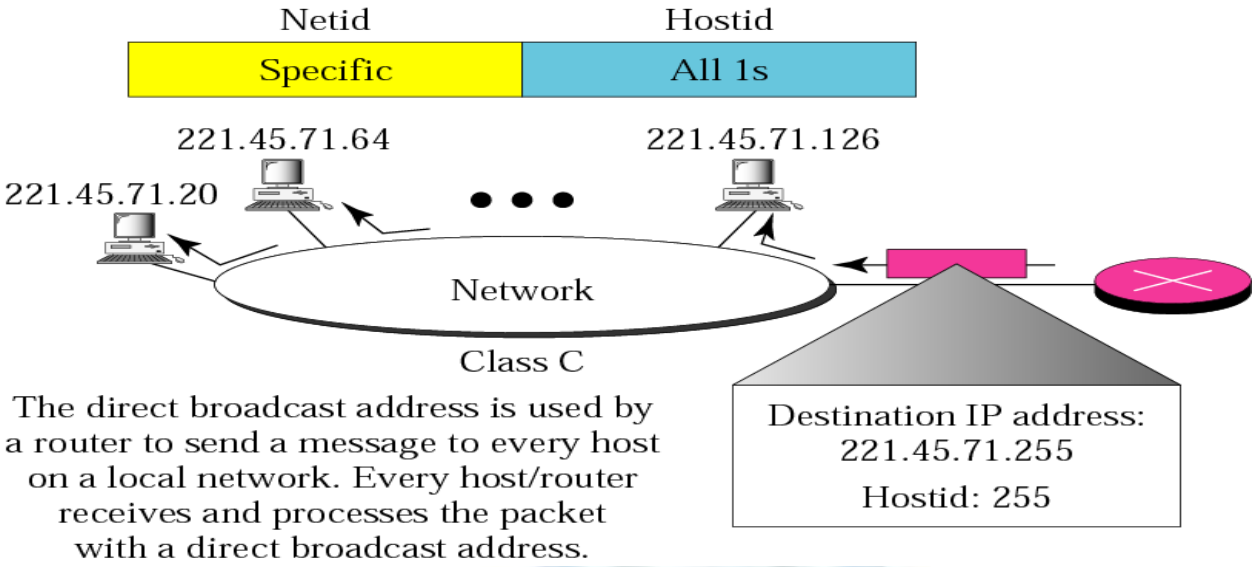
Special Address	*Netid*	*Hostid*	*Source or Destination*
Network address	Specific	All 0s	None
Direct broadcast address	Specific	All 1s	Destination
Limited broadcast address	All 1s	All 1s	Destination
This host on this network	All 0s	All 0s	Source
Specific host on this network	All 0s	Specific	Destination
Loopback address	127	Any	Destination

Table : Special addresses

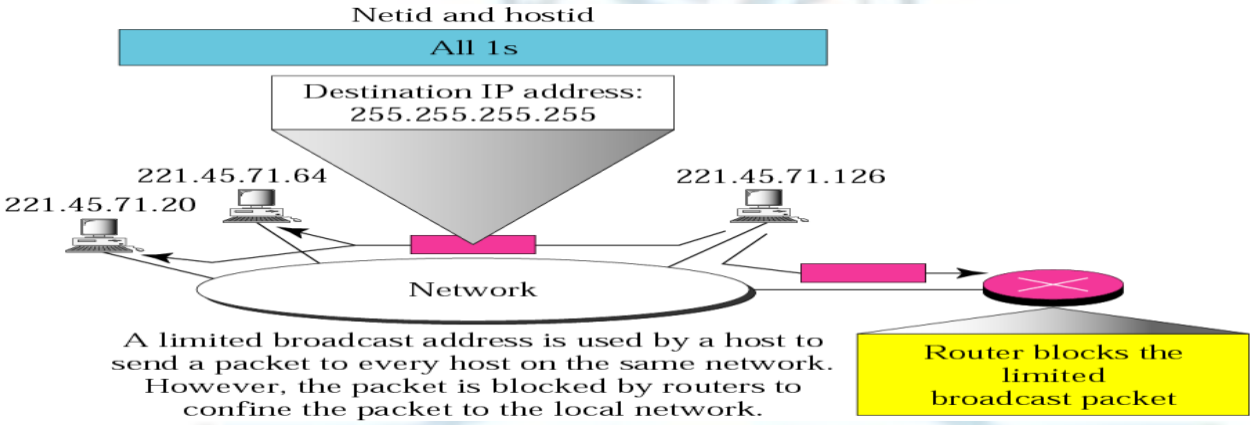
- **Network address**
  - The first address in a block defines the network address



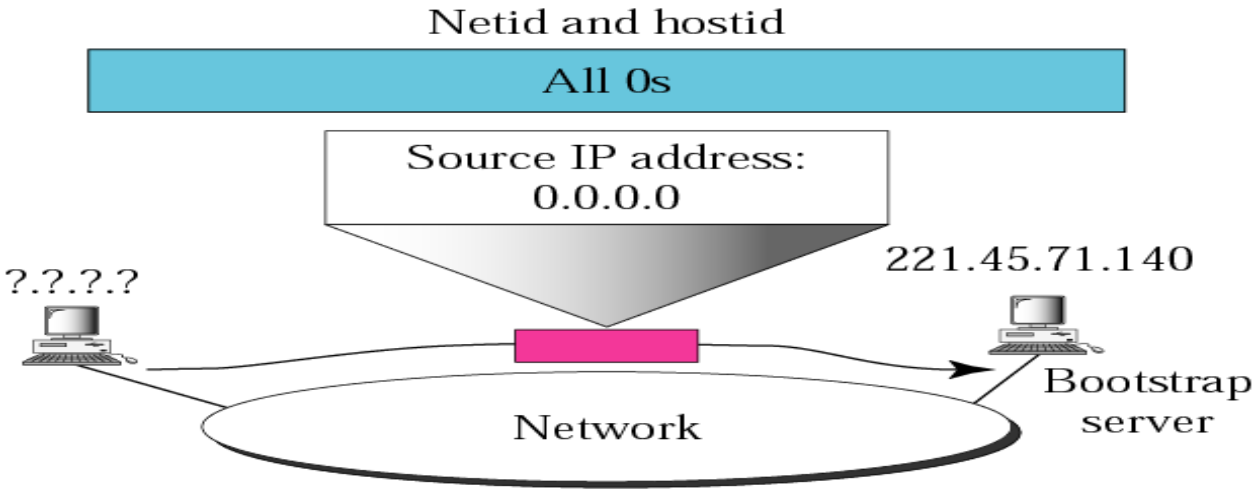
- **Direct Broadcast Address**
  - It is used by a router to send a packet to all host in a specific network



- **Limited Broadcast Address**
  - A host that wants to send a message to every other host can use this address as dest. address in a packet

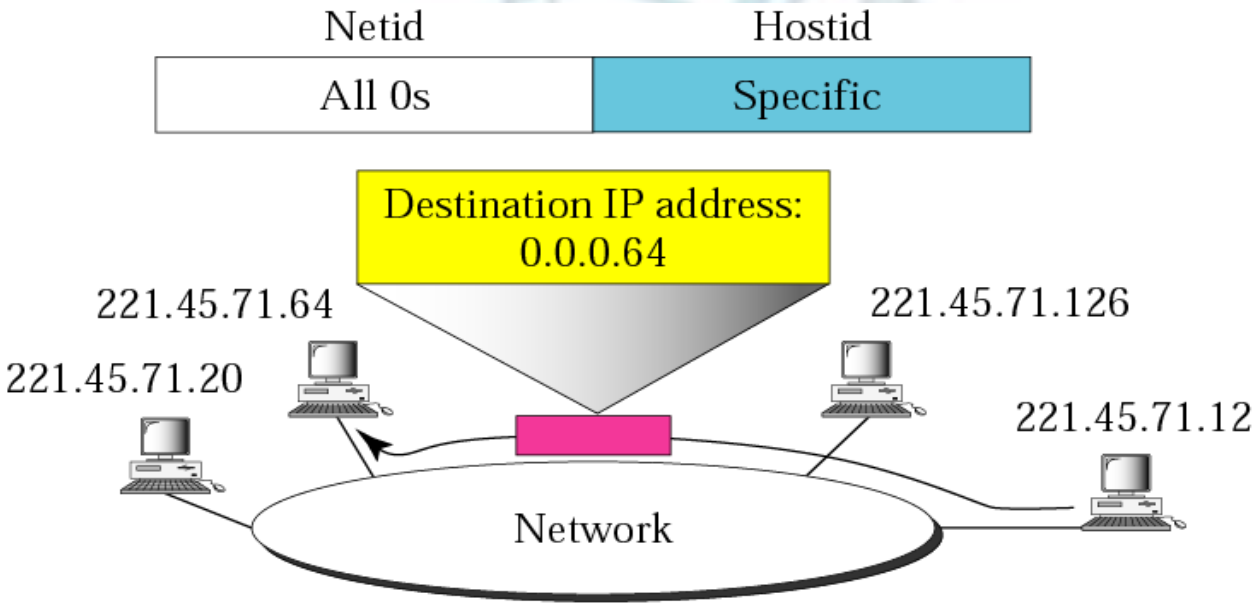


- **This Host on This Network**



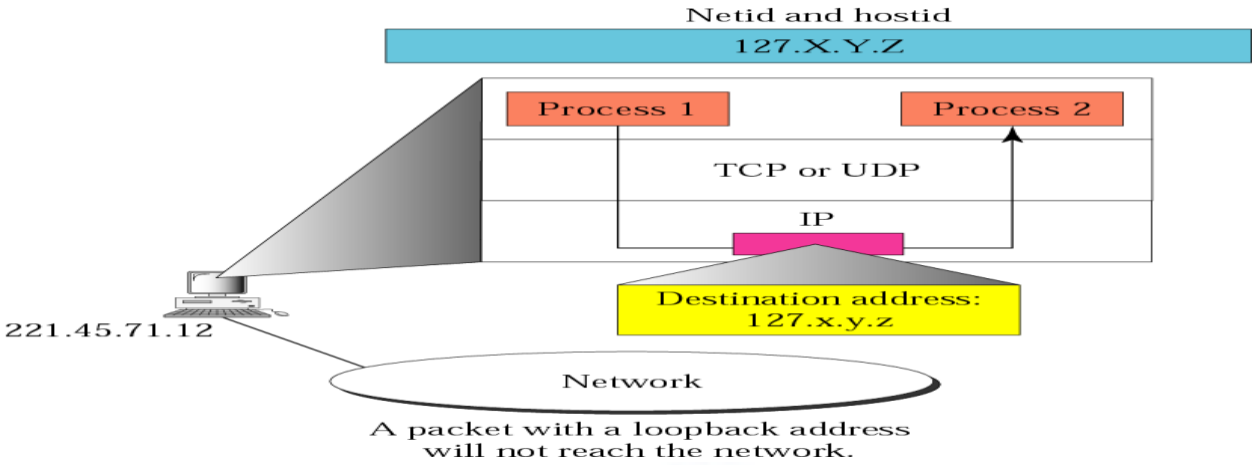
A host that does not know its IP address uses the IP address 0.0.0.0 as the source address and 255.255.255.255 as the destination address to send a message to a bootstrap server.

- **Specific Host on This Network**



This address is used by a router or host to send a message to a specific host on the same network.

- **Loopback address**
  - It is used test the software on a machine.

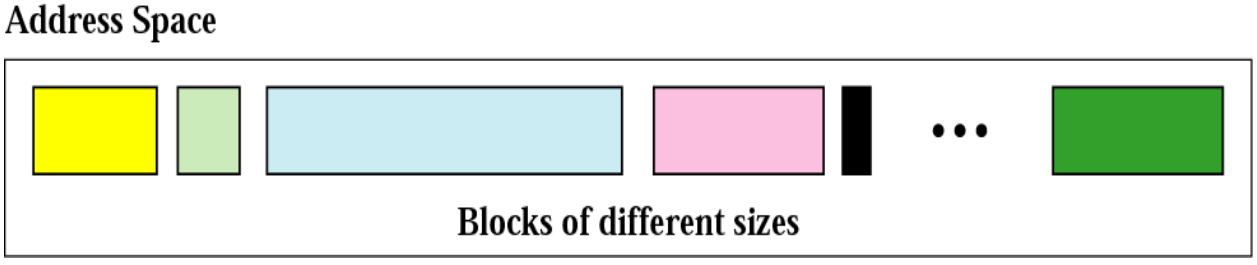


- **Private Address**
  - A number of blocks in each class are assigned for private use

Class	*Netids*	*Blocks*
A	10.0.0	1
B	172.16 to 172.31	16
C	192.168.0 to 192.168.255	256

CLASSLESS ADDRESSING

- In the mid of 1990s, a range of addresses meant a block of addresses in class A,B, or C.
- The minimum number of addresses granted to an organization was 256 (class C).
- An ISP is an organization that provides Internet addresses to individuals.
- Classless addressing, announced in 1996, allows an ISP to assign as few or as many IP addresses as requested.
- In classless addressing the entire 2^32 address space is divided into variable-sized blocks, which are multiples of powers of 2.
- The first address must be evenly divisible by the number of addresses. For example, if a block contains 4 addresses, the first address must be divisible by 4.
-



Mask

- When an organization is given a block, it is given the first address and the mask.
- The mask is made of some 1s at left followed by some 0s at right.
- So instead of mask 255.255.255.224, we can say that the mask has 27 1s.
- This number is attached at the end of classless address. This is called slash notation or CIDR notation (Classless Inter Domain Routing).

$$A.B.C.D/n$$

- Prefix and Prefix Length
- Two terms often used in classless addressing
  - Prefix – another name for the common part of the address range (netid)
  - Prefix length – the length of the prefix (n in slash notation).
- Prefix lengths

/n	Mask	/n	Mask	/n	Mask	/n	Mask
/1	128.0.0.0	/9	255.128.0.0	/17	255.255.128.0	/25	255.255.255.128
/2	192.0.0.0	/10	255.192.0.0	/18	255.255.192.0	/26	255.255.255.192
/3	224.0.0.0	/11	255.224.0.0	/19	255.255.224.0	/27	255.255.255.224
/4	240.0.0.0	/12	255.240.0.0	/20	255.255.240.0	/28	255.255.255.240
/5	248.0.0.0	/13	255.248.0.0	/21	255.255.248.0	/29	255.255.255.248
/6	252.0.0.0	/14	255.252.0.0	/22	255.255.252.0	/30	255.255.255.252
/7	254.0.0.0	/15	255.254.0.0	/23	255.255.254.0	/31	255.255.255.254
/8	255.0.0.0	/16	255.255.0.0	/24	255.255.255.0	/32	255.255.255.255

** study the problems worked out in class**

SUBNETTING AND SUPERNETTING

- In ***subnetting***, a network is divided into smaller subnetworks with each subnet having its own subnet address
- In ***supernetting***, a organization can combine several class C to create a large range of addresses. In other word, several networks are combined to create a supernetwork
- Generally: class A, B, C → two levels of hierarchy. Sometimes two levels is **not enough**

- Solution - subnetting

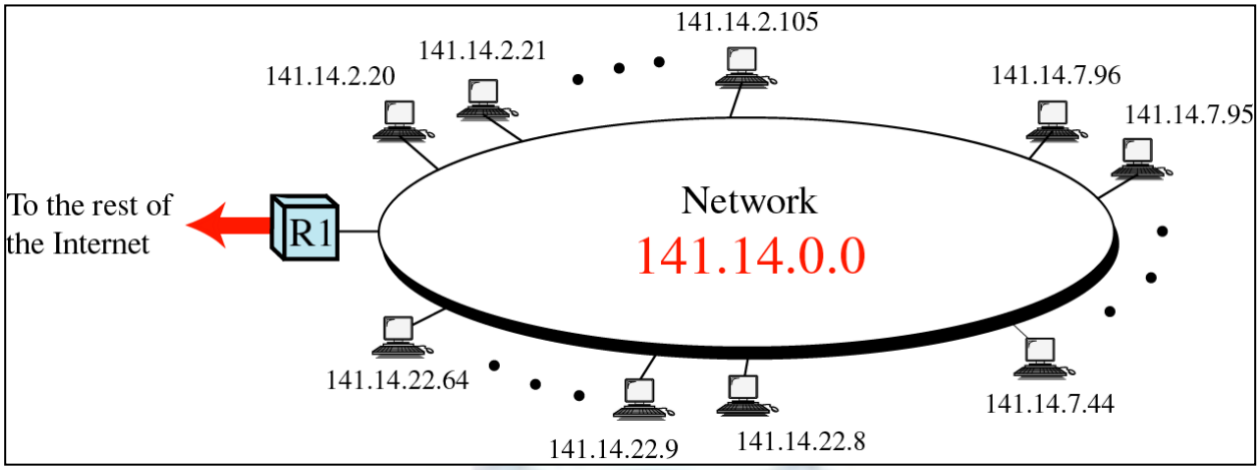


Figure: Without subnetting

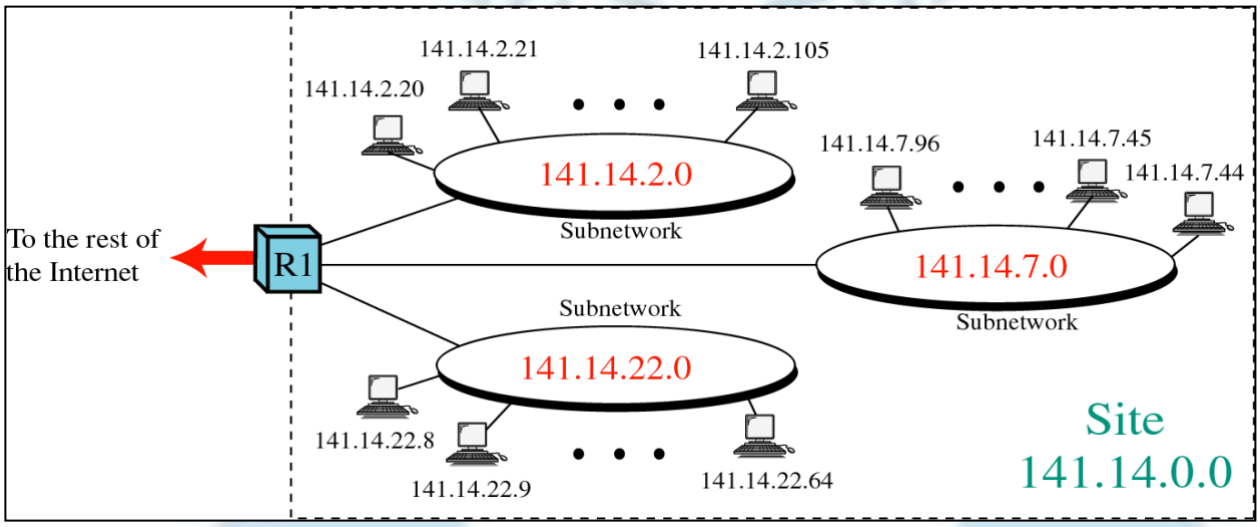
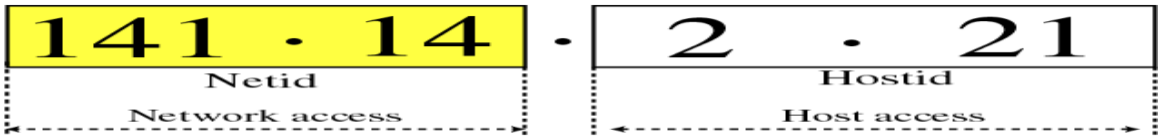


Figure: with subnetting

- **Three levels of heirachy**

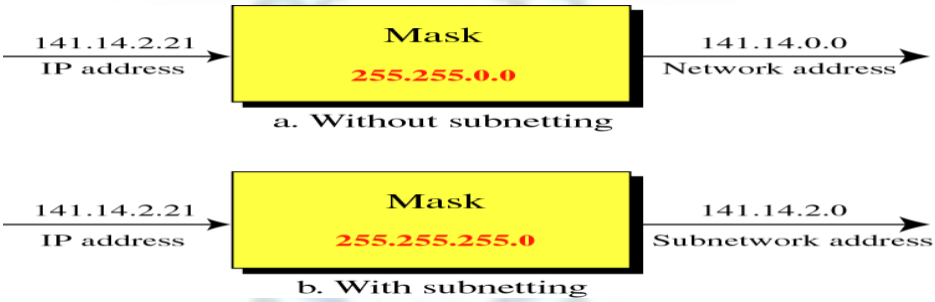


a. Without subnetting

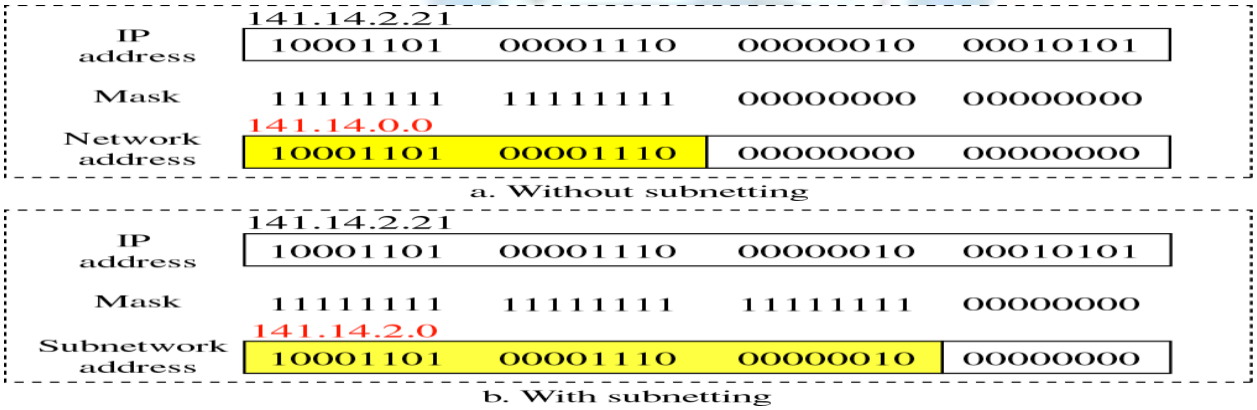
b. With subnetting

- Hierachy concept in a telephone number

- **Masking**
  - A process that extracts the address of the physical network from IP address
  - Can be done whether we have subnet or not.
  - Not have subnet: masking extract the network address from an IP address.
  - Have subnet: masking extract the subnet address from IP address.
- **Subnet Mask**
  - A 32 bit number used for masking
  - Part of masking ***containing 1s*** defines netid orcombination of netid and subnetid
  - Part of making ***containing 0s*** defines hostid
  - To get the network and subnet address, we must apply the ***bit-wise-and*** operation on IP address and the mask



- **Applying bit-wise-and operation to achieve masking**



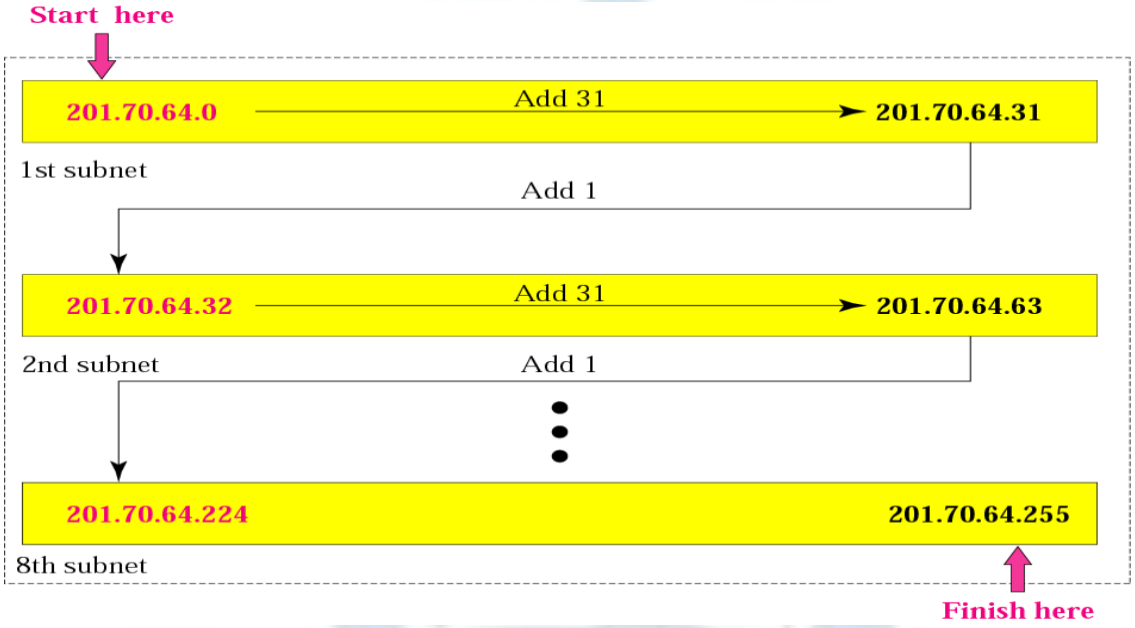
Subnetting

- In subnetting, add bits from the host part to network part to define subnet address

Designing Subnets

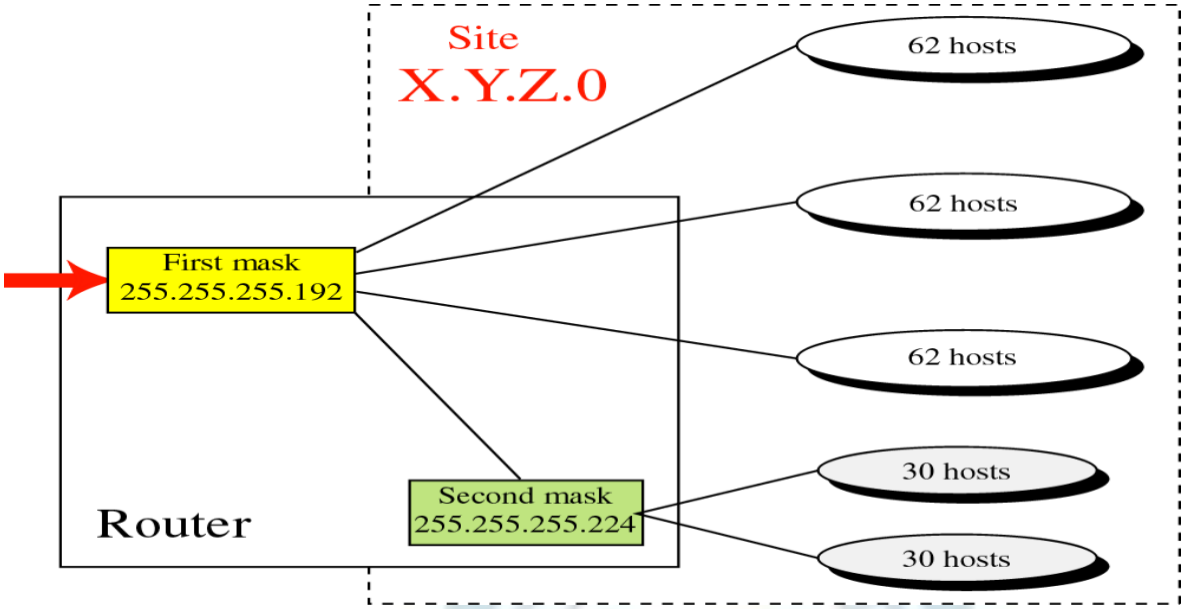
- Deciding the number of subnets
  The number of subnets must be a power of 2

- Finding the subnet mask
- Finding the range of address in each subnet
- **Example :**
  A organization with a class A needs a least 1000 subnetworks. Find the subnet mask and configuration of each network
- **Solution:**
  This means that the minimum number of bits to be allocated for subnetting should be 10 ($2^9 < 1,000 < 2^{10}$)
  Fourteen bits are left to define the hostid
- **Example :**
  A company is granted the site address 201.70.64.0 (class C). The company needs 8 subnets. Design the subnets.



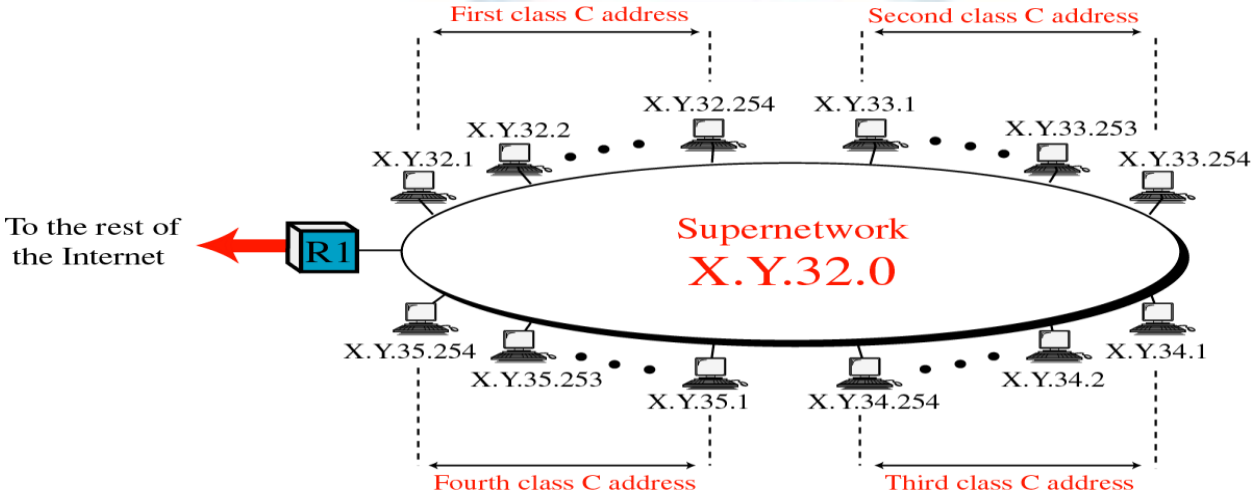
Variable-length subnetting

- The internet allows a site to use variable-length subnetting.
- For an example of when this may be desirable, consider a site that is granted a class C address and needs to have five subnets with the following number of host: 60,60,60,30,30
  - This site can not use a subnet mask with only two bits in the subnets section because this allows only four subnet with 62 hosts (256/4 - 2 = 62).
  - Nor can the site use a subnet mask with three bits in the subnet section, because this allows 8 subnets with 30 hosts (256/8 -2 = 30)
  - Solution for the problem: *variable length subnetting.*
  - The router uses two different masks, one applied after the other
  - First mask uses the mask with 26 1s (11111111 11111111 11111111 1100000 or 255.255.255.255.192) to divide the network into four subnets
  - Then it applies the second mask with 27 1s (255.255.255.224) to one of the subnets to divide it into two smaller subnets



Supernetting

- Depend on the need of an organization
- One or more classes c can be jointed to make one supernetwork
- Example: an organization that needs 1000 address can be granted four class c addresses. The organization can then use these address in one supernetwork, in four network, or in more then four networks.
-



- **Assigning addresses**

When combine set of c blocks into a large network 2 choices

1. choose the blocks randomly based on some rules if do so, the routers outside the organaization treat each block separately

2. Make a superblock out of the block, so that each routing table has only one entry in the table follow a set of rules

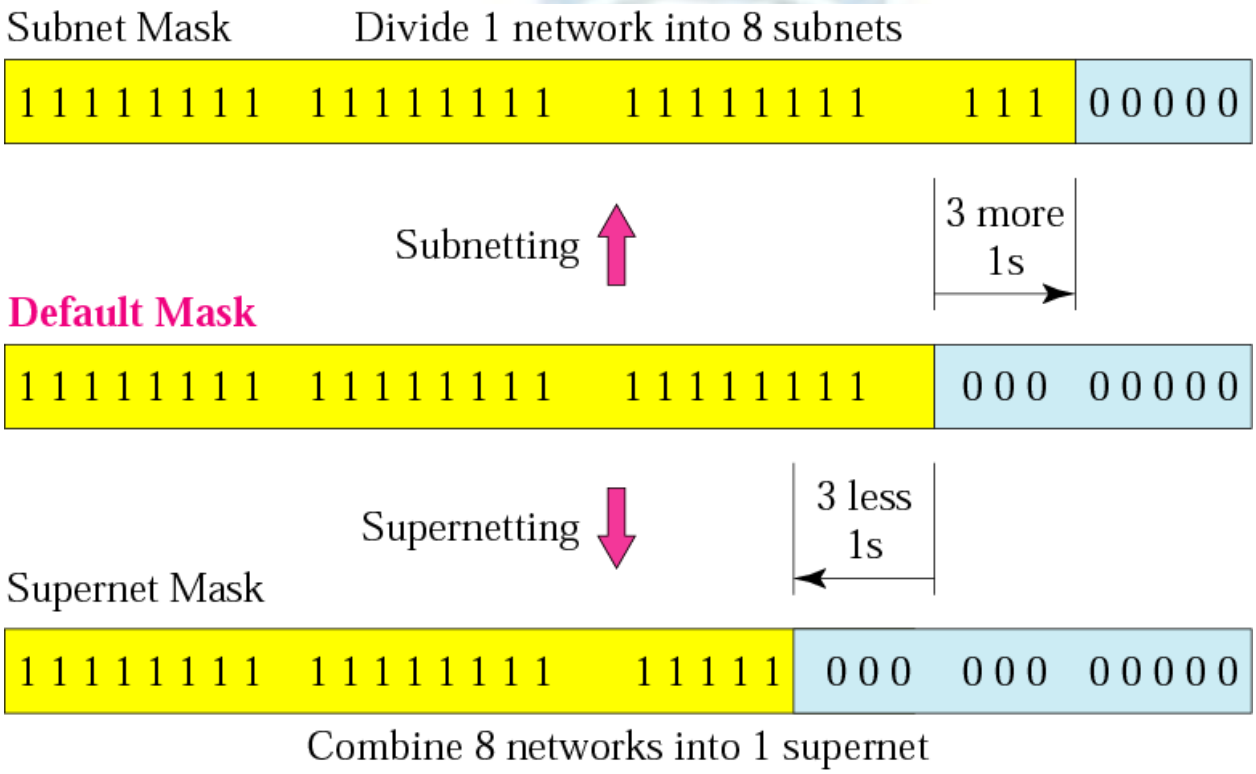
I. The number of blocks must be a power of 2

II. The blocks must be contiguous in the address space

III. The third byte of the first address in the superblock must be evenly divisible by the number of blocks

Supernet mask

- In supernetting, need the first address of the supernet and the supernet mask to define the range of addresses.
- **Comparison of subnet, default, and supernet masks**



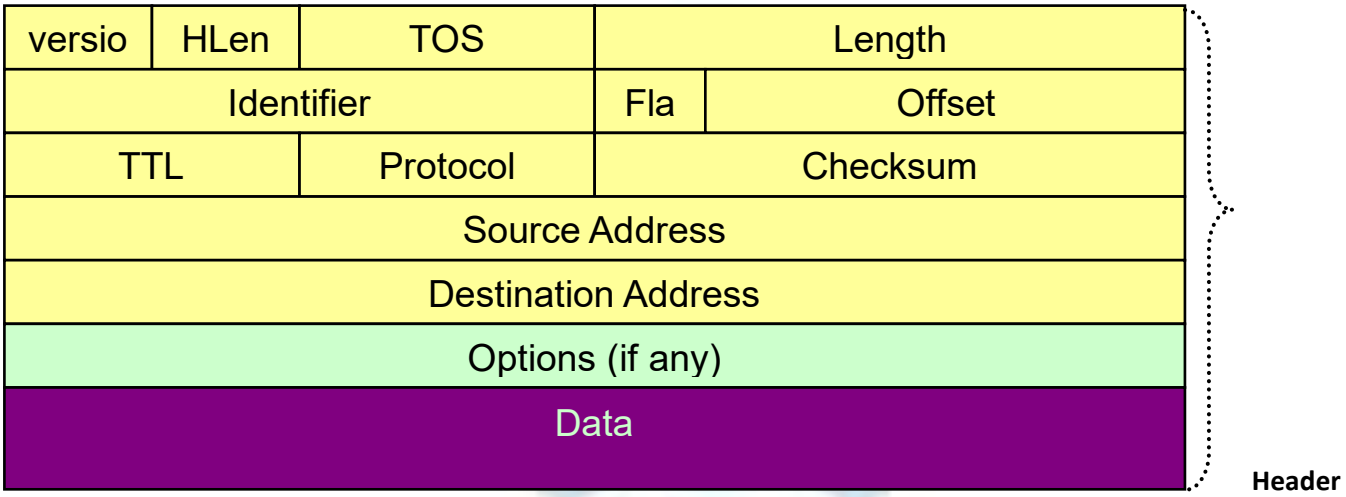
- **Example :**
  We need to make a supernetwork out of 16 class C blocks. What is the supernet mask?
  Solution:
  We need 16 blocks. For 16 blocks we need to change four 1s to 0s in the default mask. So the mask is
  11111111 11111111 11110000 00000000
  or
  255.255.240.0
- **Example :**

A supernet has a first address of 205.16.32.0 and a supernet mask of 255.255.248.0. How many blocks are in this supernet and what is the range of addresses?

Solution:

The supernet has 21 1s. The default mask has 24 1s. Since the difference is 3, there are 2^3 or 8 blocks in this supernet. The blocks are 205.16.32.0 to 205.16.39.0. The first address is 205.16.32.0. The last address is 205.16.39.255.

IPv4 PacketFormat



- Version: IP Version
  - 4 for IPv4
- HLen: Header Length
  - 32-bit words (typically 5)
- TOS: Type of Service
  - Priority information

- Length: Packet Length
  - Bytes (including header)
- Header format can change with versions
  - First byte identifies version
- Length field limits packets to 65,535 bytes
- Identifier, flags, fragment offset → used primarily for fragmentation
- Time to live
  - Must be decremented at each router
  - Packets with TTL=0 are thrown away
  - Ensure packets exit the network
- Protocol
  - Demultiplexing to higher layer protocols
  - TCP = 6, ICMP = 1, UDP = 17…
- Header checksum

- Source Address
  - 32-bit IP address of sender

- Like the addresses on an envelope
- Globally unique identification of sender & receiver