/

3.1 Queuing Delay and packet lost:-

At the queue, the packet experiences a queuing delay as it waits to be transmitted onto the link. The queuing delay of a specific packet will depend on the number of other, earlier-arriving packets that are queued and waiting for transmission across the link. The delay of a given packet can vary significantly from packet to packet. If the queue is empty and no other packet is currently being transmitted, then our packet's queuing delay is zero. On the other hand, if the traffic is heavy and many other packets are also waiting to be transmitted, the queuing delay will be long. We will see shortly that the number of packets that an arriving packet might expect to find on arrival is a function of the intensity and nature of the traffic arriving to the queue. Queuing delays can be on the order of milliseconds to microseconds in practice.

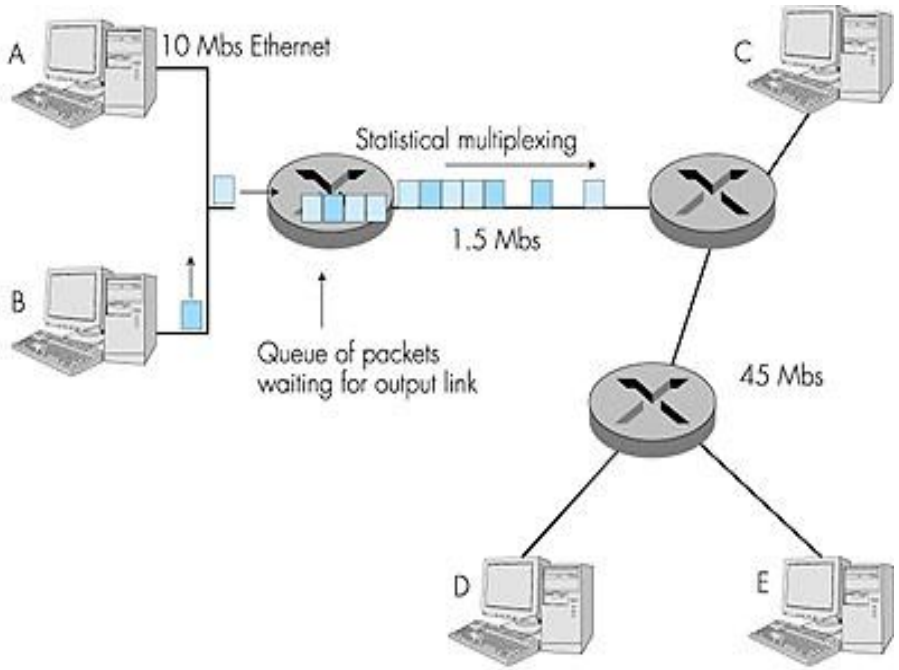

Figure-3-1 Queuing Delay and packet lost

To find out when is the queuing delay large and when it is insignificant depends largely on the rate at which traffic arrives to the queue, the transmission rate of the link, and the nature of the arriving traffic, that is, whether the traffic arrives periodically or whether it arrives in bursts. To

gain some insight here, let a denote the average rate at which packets arrive to the queue (a is in units of packets/sec). Recall that R is the transmission rate, that is, it is the rate (in bits/sec) at which bits are pushed out of the queue. Also suppose, for simplicity, that all packets consist of L bits. Then the average rate at which bits arrive to the queue is La bits/sec. Finally, assume that the queue is very big, so that it can hold essentially an infinite number of bits. The ratio La/R, called the traffic intensity, often plays an important role in estimating the extent of the queuing delay. If La/R > 1, then the average rate at which bits arrive to the queue exceeds the rate at which the bits can be transmitted from the queue. In this unfortunate situation, the queue will tend to increase without bound and the queuing delay will approach infinity! Therefore, one of the golden rules in traffic engineering is: Design your system so that the traffic intensity is no greater than 1.

Figure-3-2: Dependence of average queuing delay on traffic density

3.2 Congestion Management:

Congestion happens when the rate of input (incoming traffic switched) to an interface exceeds therate of output (outgoing traffic) from an interface. this happen because Sometimes traffic enters a device from a high-speed interface and it has to depart from a lower-speed interface; this can cause congestion on the egress lower-speed interface, and it is referred to as the speed mismatch problem. If traffic from many interfaces aggregates into a single interface that does not have enough capacity, congestion is likely; this is called the aggregation problem. Finally, if joining of multiple traffic streams causes congestion on an interface, it is referred to as theconfluence problem.

3.2.1FIFO Queuing

By default, most interfaces use FIFO queuing—there is just one software queue, and traffic is buffered and then scheduled onto the interface in the order it is received.

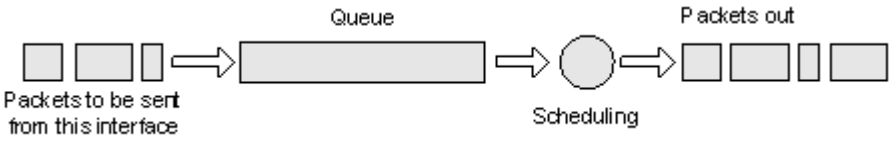

Figure-3-3 FIFO Queuing

3.2.2Priority Queuing

With Priority queuing, queues are assigned different priority values andplaced in one of four queues. The high-priority queue is a strict priorityqueue, which means that it gets serviced before anything else until it isempty. After that, each queue is serviced in turn, as long as the priorityqueue remains empty. The lower-priority queues may never be servicedif there is sufficient traffic in higher-priority queues (a condition called“starvation”).

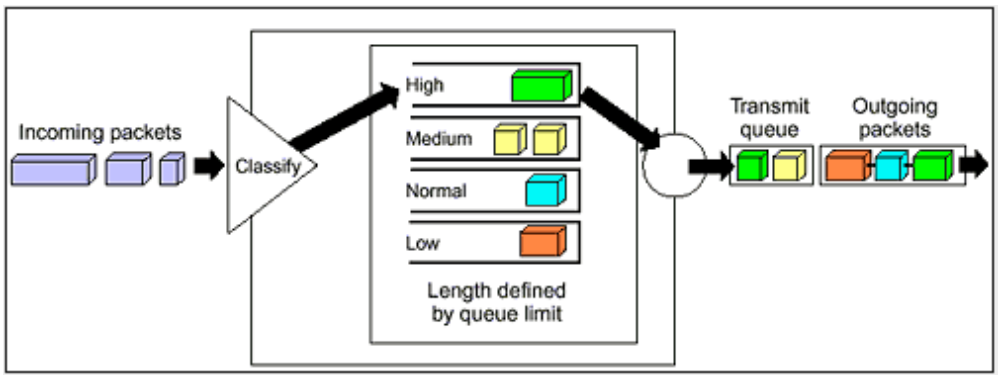

Figure-3-4 Priority Queuing

3.2.3 Round Robin Queuing

Round Robin queuing takes one packet from each queue and then startsover. Each queue is serviced, none starve, but there is no way to prioritize
any of the traffic or apply any sort of differential treatment to it.During times of interface congestion, Weighted Round Robin (WRR)queuing weights queues, and more packets are sent from higherweighted queues, thus giving them more bandwidth. However, thebandwidth allocations are done in a way that might lead to more bytesbeing sent from some queues than desired, which causes other packetsto be delayed.

3.2.4Weighted Fair Queuing

Weighted Fair Queuing (WFQ) attempts to address some of the failing of FIFO and Priority queuing by allowing all traffic some access to theInterface. Some characteristics of WFQ include:
Queues traffic by flow or conversation; flows are identified by header information, such as source and destination IP address, protocol, source and destination ports, andtype of service field value.
These are used by a hash algorithm tocreate a queue index number.
Each interface has a limited number of WFQ queues. If the number of flows exceeds the number of queues, multipleflows are placed in the same queue, resulting in less bandwidthper flow.
Provides queues for system traffic and RSVP traffic separate fromthe WFQ queues.[9]
Traffic is weighted by flow, based on IP precedence.WFQ schedules small interactive flows before high-bandwidth flows. Allows lower-weighted flows relatively more bandwidth than higher weighted conversations.
Drops packets from high-volume flows more aggressively thanthose of low-volume flows.The hold-queue limit determines how many packets can be held in the WFQ system before all new packets are dropped (tail drop).
The congestive discard threshold (CDT) determines how manypackets can be held by the WFQ before it begins dropping packetsfrom high-volume flows. Packets are dropped from the queueswith the most packets first. Packets are not dropped from lowvolumeconversations.

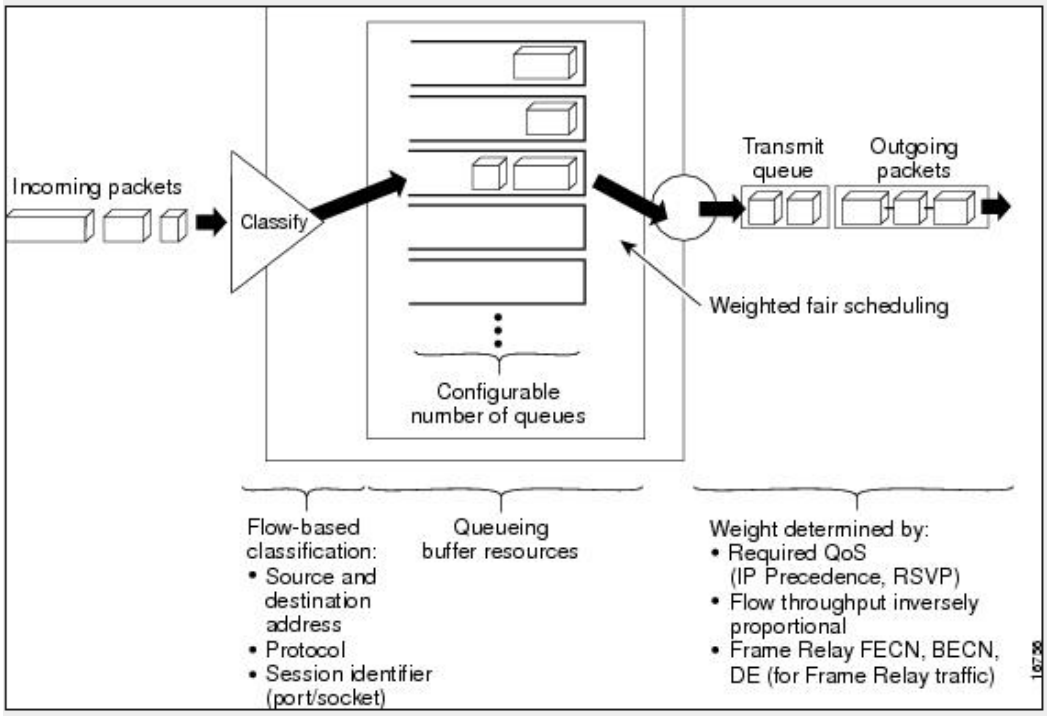

Figure-3-5 Weighted Fair Queuing

3-3 Differentiated Services Model:-

Differentiated Services (DiffServ) is the newest of the three QoS models, and its development has aimed to overcome the limitations of its predecessors. DiffServ is not a guaranteed QoS model, but it is a highly scalable one. The Internet Engineering Task Force (IETF) description and discussion on DiffServ are included in RFCs 2474 and 2475. Whereas IntServ has been called the "Hard QoS" model, DiffServ has been called the "Soft QoS model. IntServ, through usage of signaling and admission control, is able to either deny application of requested resources or admit it and guarantee the requested resources. Pure DiffServ does not use signaling; it is based on per-hop behavior (PHB). PHB means that each hop in a network must be preprogrammed to provide a specific level of service for each class of traffic. PHB then does not require signaling as long as the traffic is marked to be identified as one of the expected traffic classes. [16].

This model is more scalable because signaling and status monitoring (overhead) for each flow are not necessary. Each node (hop) is prepared to deal with a limited variety of traffic classes.

This means that even if thousands of flows become active, they are still categorized as one of the predefined classes, and each flow will receive the service level that is appropriate for its class. The number of classes and the service level that each traffic class should receive are decided based on business requirements. Within the DiffServ model, traffic is first classified and marked.

As the marked traffic flows through the network nodes, the type of service it receives depends on its marking. DiffServ can protect the network from oversubscription by using policing and admission control techniques as well. For example, in a typical DiffServ network, voice traffic is assigned to a priority queue that has reserved bandwidth (through LLQ) on each node. To prohibit too many voice calls from becoming active concurrently, you can deploy CAC. Note that all the voice packets that belong to the admitted calls are treated as one class. The main benefit of the DiffServ model is its scalability.

The second benefit of the DiffServ model is that it provides a flexible framework for you to define as many service levels as your business requirements demand. The main drawback of the DiffServ model is that it does not provide an absolute guarantee of service. That is why it is associated with the term Soft QoS. The other drawback of this model is that several complex mechanisms must be set up consistently on all the elements of the network for the model to yield the desired results. Following are the benefits of DiffServ:

■ Scalability

■ Ability to support many different service levels The drawbacks of DiffServ are as follows:

■ It cannot provide an absolute service guarantee.

■ It requires implementation of complex mechanisms through the network.

3-4Differentiated services architectural mode:-

The architectural features of Diffserv are encapsulated within a Diffserv domain. Defines the diffserv architectural as shown in figure 3.6Packets are with the same value in the DS field in the Ds field in the ip header.

This Value of the Ds field is called the differentiated service code point(DSCP).each of the different combinations of DSCP bits is expected to stimulate every network device along the traffic path to behave in a certain way and to provide a particular QoS treatment to the traffic. Therefore, within the DiffServ framework, you set the DSCP value on the IP packet header to select a per-hop behavior (PHB).*PHB* is formally defined as an externally observable forwarding behavior of a network node toward a group of IP packets that have the same DSCP value.

The group of packets with a common DSCP value (belonging to the same or different sources and applications), which receive similar PHB from a DiffServ node, is called a behavior aggregate (BA). The PHB toward a packet ,including how it is scheduled, queued, policed, and so on, is based on the BA that the packet belongs to and the implemented service level agreement (SLA)or policy.

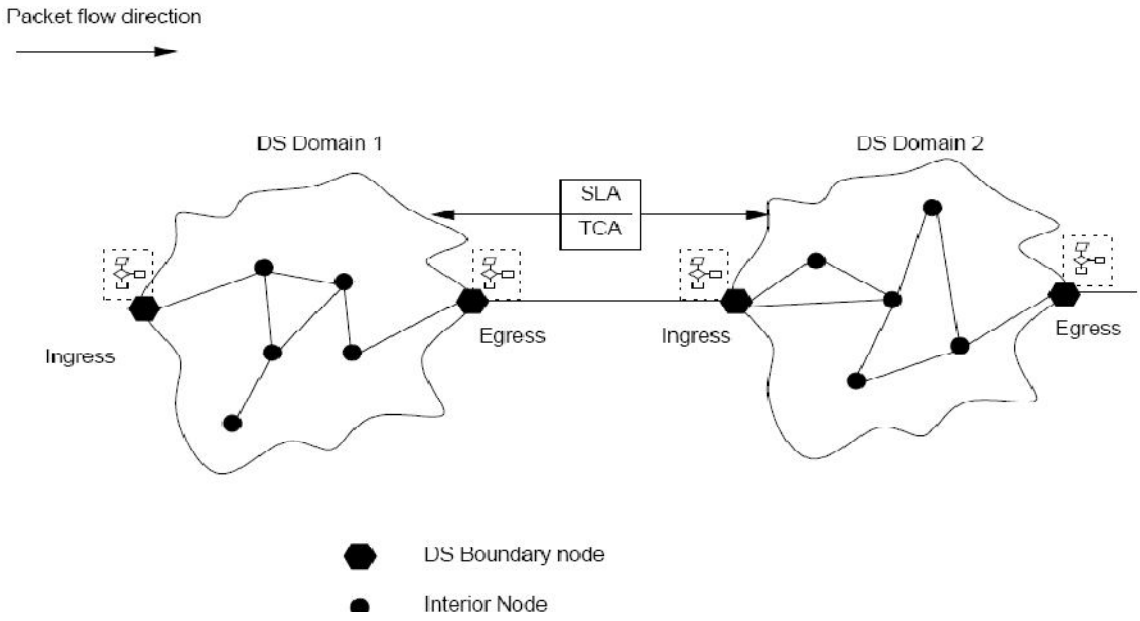

Figure 3.6: Differentiated Services model

Scalability is a main goal of the DiffServ model. Complex traffic classification is performed as close to the source as possible. Traffic marking is performed subsequent to classification. If marking is done by a device under control of the network administration, the marking is said to be trusted. It is best if the complex classification task is not repeated, and the PHB of the transit network devices will solely depend on the trusted traffic marking. This way, the DiffServ model has a coarse level of classification, and the marking-based PHB is applied to traffic aggregates or behavior aggregates (BAs), with no per-flow state in the core .Application-generated signaling (IntServ style) is not part of the DiffServ framework, and this boosts the scalability of the DiffServ model. Most applications do not have signaling and Resource Reservation Protocol (RSVP) capabilities. The DiffServ model provides specific services and QoS treatments to groups of packets with common DSCP values (BAs). These packets can, and in large scale do, belong to multiple flows. The services and QoS treatments that are provided to traffic aggregates based on their common DSCP values are a set of actions and guarantees such as queue in section policy, drop preference, and bandwidth guarantee. The DiffServ mode l provides particular service classes to traffic aggregates by classifying and marking the traffic first, followed by PHB toward the marked traffic within the network core.

3-5 Differentiated Services Domain:

A Diffserv domain is a contiguous set of DS nodes with the same service provisioning policy and set of PHBs. It has a well-defined boundary. Boundary nodes apply rules to traffic that enter the DS domain. The Traffic that enters the domain is called ingress and egress traffic, respectively. Internal nodes select a forwarding behavior (PHB) according to the value of the DS code point.

A DS domain consists of both boundary and interior nodes. Boundary nodes connect the DS domain to other DS domains or to non-DS domains. Interior nodes only connect to other interior or boundary nodes in the same domain.

Boundary nodes must be able to perform traffic conditioning based on a Traffic Conditioning Specification (TCS). A TCS specifies, for example, service performance parameters (e.g. Throughput and drop probability), and traffic profiles within which the service is offered (e.g. Token bucket parameters), the period over which the service is offered, and how packets are marked.

A boundary node can act as both ingress and egress for traffic, depending on the direction of the stream. Boundary node that acts as an ingress node must ensure that the traffic conforms to the TCS that was set up between the customer and provider domain. The egress node can condition traffic exiting the domain to ensure that it conforms to the TCS of the downstream domain.

3-6 Traffic Classification and Conditioning:

Differentiated services are extended across a DS domain boundary by establishing a Service Level Specification (SLS) between the upstream and the downstream domains. The SLS defines packet classification and re-marking rules, traffic profiles, and actions to be performed for in and out-of-profile streams. The previously mentioned TCS forms an integral part of the SLS.

3-6-1Classifiers:

A classifier selects packets in a traffic stream, based on the contents of a portion of the packet header. In the Diffserv context there are two types of classifiers: BA classifiers and multi-field classifiers. BA classifiers classify packets based on only the value of the DS code point. Multi-field classifiers use additional fields in the packet header to classify the packet.

Classifiers are used to 'steer' certain packets (matching some rule) to an element of a traffic conditioner for further processing. Classifiers are configured by some management procedure according to a TCS. Traffic profile specifies the properties of a traffic stream selected by a classifier. It also provides rules for determining whether a packet is in- or out-of-profile. A profile could, for example, specify that all packets marked with code point *X* should be measured against a token bucket meter with rate *r* and burst size *b*.

Different so-called conditioning actions may be applied to in- and out-of-profile traffic. These actions are carried out by components called traffic conditioners, as discussed in the next section.

Figure3-7 gives a logical representation of a traffic classifier and the most important functional components of a conditioner. The various components in the figure are related as follows.

A meter measures the properties of packet stream that has been selected by a classifier against a profile. The meter then passes state information to other conditioning functions to trigger particular actions.

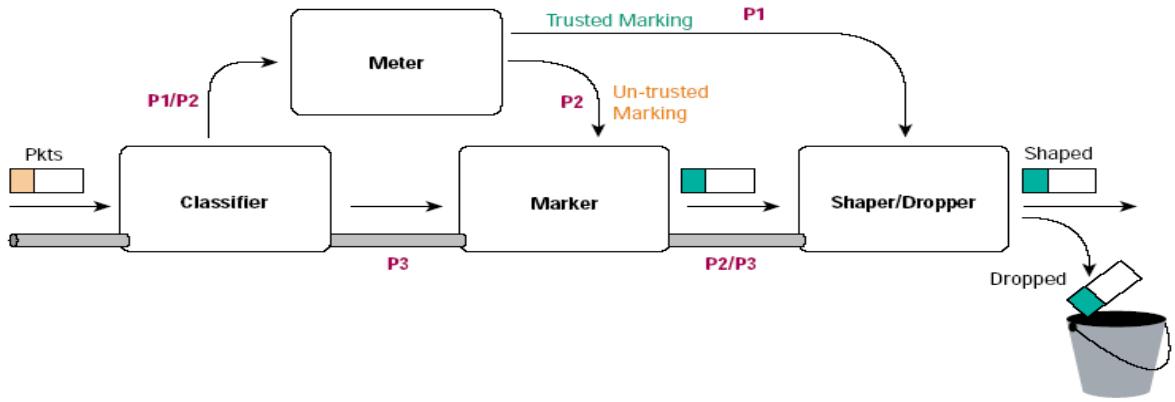

Figure 3-7 a logical representation of a traffic classifier

3-6-2Traffic Conditioners:

A traffic stream is selected by a classifier which steers the packets to a logical instance of a traffic conditioner. A traffic conditioner may contain components called a meter, a shaper, a dropper, and a marker. A meter is used to measure the traffic stream's characteristics against a profile. The state of the meter regarding a specific packet is used to ascertain whether or not the packet is out-of-profile.

Out-of-profile traffic could be acted upon in a variety of ways. It might be queued until it is in-profile (shaped); it might be discarded (dropped or policed); it might be marked with a new code point (re-marked); or it might be forwarded without change but then also trigger an accounting procedure.

A marker sets the DS field of a packet to a particular code point, thereby adding it to a particular BA. Shapers delay packets in a stream to bring them in compliance with a given profile. A shaper has finite buffer space and will discard packets if the buffer becomes full. On the other hand, droppers discard packets in a stream to bring them in compliance with a profile. This is called 'policing' the stream.

Traffic conditioners are usually located within DS ingress and egress boundary nodes. The latter case usually occurs when one wants to ensure that the egress traffic conforms to some specification before it is sent to a downstream DS domain. Traffic conditioners can, however, also be located within interior nodes or in a non-DS-capable domain.

3-7Marking at Layer 3:-

The concept behind DiffServ (DS) is to group traffic into classes and mark it once at the edge of the network. DiffServ was created to behighly scalable by separating classification from policy creation and byservicing aggregate classes of traffic rather than individual flows.

DiffServ uses Layer 3 markings, setting the eight-bit ToS field in the IP header. Unlike the Layer 2 header, this marking remains with thepacket as it traverses the network, and changes only if some device overwrites the value of these bits. You can set either IP Precedence,using the top three bits, or Differentiated Services Code Points (DSCP),using the top six bits of the field. The bottom two bits can be used forcongestion notification. The default DSCP value is zero, which corresponds to best-effort delivery. When properly configured, DSCP isbackward compatible with IP Precedence. IP packet shown in Figure-3-8

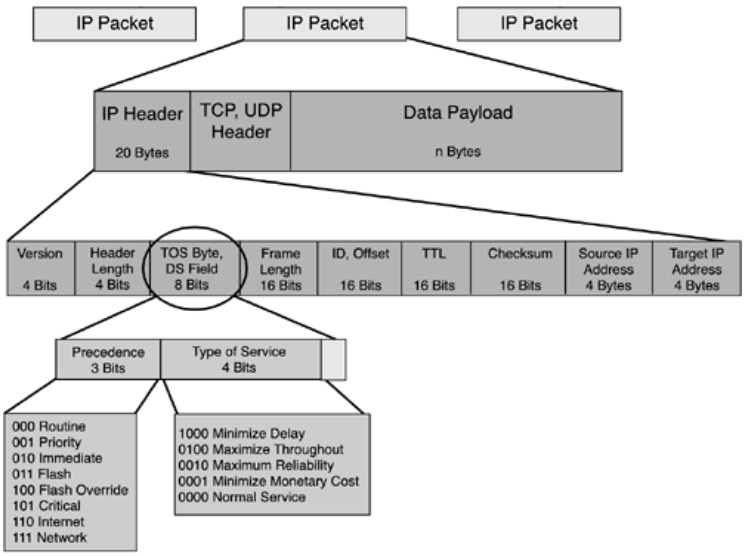

Figure-3-8IP packet

Table3-1ToS Bits and Their Corresponding Decimal Values and Definitions

ToS (bits)	ToS (in Decimal)	IETF RFC791	Application
000	0	Routine	Best-Effort Data
001	1	Priority	Medium Priority Data
010	2	Immediate	High Priority Data
011	3	Flash	Call Signaling
100	4	Flash-Override	Video Conferencing
101	5	Critical	Voice Bearer
110	6	Internet	Reserved (inter-network control)
111	7	Network	Reserved (network control)

Each hop in the network is provisioned to treat traffic differently based on its markings; this is called “per-hop behavior” (PHB). RFC 2475 defines PHB as “the externally observable forwarding behavior appliedat a DS-compliant node to a DS behavior aggregate.” A behavioraggregate is a logical grouping of traffic that needs similar service levels. It is also referred to as a *service class*.

3-8Per-hop Behaviors:-

When a DS node receives a packet, it examines the code point and takes certain actions, depending on the particular behavior aggregate. The per-hop behavior of the node is the resulting externally observable forwarding behavior of the traffic at the node. The PHB may be specified in terms of the resources available to it (buffer, bandwidth); Its priority relative to other PHBs; or the observable traffic characteristics (delay, loss) associated with it, expressed in relative terms. PHBs are used as the building blocks for allocating resources in the forwarding path. They are implemented on nodes by buffer management and packet scheduling mechanisms. While a PHB only describes behavior at a single hop in DS domain, a PHB describes how to configure a DS domain. The PHB is therefore the specification of how the DS domain is configured as well as the quantifiable behavior that is expected from the domain. Example of Per hop Behaviors, two example PHB definitions are given. The one PHB allows for distinguishing between different classes of traffic while the other provides the ability to create a virtual leased-line.

3-8-1 Assured Forwarding (AF) PHB:

This section describes the Assured Forwarding (AF) PHB group .The requirement is to be sure that IP packets are forwarded with a high probability as long as the aggregate traffic does not exceed the specified traffic profile. Packets should also not be re-ordered within a micro-flow.

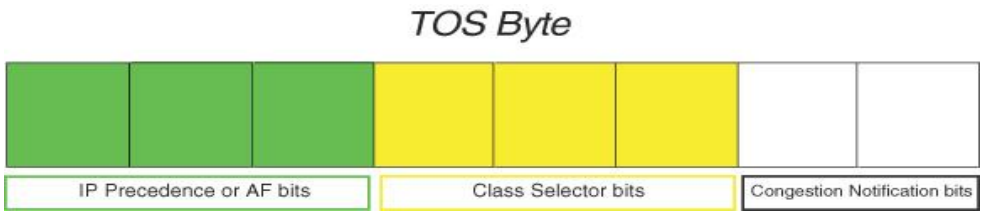

Figure 3-9 shows the TOS byte in the IP header.

The six DSCP bits can be broken down into two sections: The highest three bits define the DiffServ Assured Forwarding (AF) class (the area in green), and the next three bits are called the “Class Selector (CS)” (the area in yellow).

When the three CS bits are all zero, you have a value that is equal to IP Precedence. The lowest two bits (the area in white) are not used in DiffServ markingthey allow the sending of congestion notification information. Each AF class becomes its own queue at the interface. AF

uses the first two CS bits to define the drop probability within that queue. The last bit is always zero and is not used in calculating drop probability values.AF classes 1–4 are defined and within each class, 1 is low drop probability,2 is medium, and 3 is high (meaning that traffic is more likely to get dropped if there is congestion).AF guarantees a specified amount of bandwidth to a class. By default, it allows the traffic to burst above that amount if there is extra bandwidth available, although this can be policed. Table 4-3 lists the classes and their associated AF values.

AF classes 1–4 are defined and within each class, 1 is low drop probability,2 is medium, and 3 is high (meaning that traffic is more likely to get dropped if there is congestion).

AF guarantees a specified amount of bandwidth to a class. By default, it allows the traffic to burst above that amount if there is extra bandwidth available, although this can be policed.

Table 3-2 lists the classes and their associated AF values.

Class / Dropprecedence	Class 1	Class 2	Class 3	Class 4
Low Drop	001010 (AF11)	010010 (AF21)	011010 (AF31)	100010 (AF41)
Medium Drop	001100 (AF12)	010100 (AF22)	011100 (AF32)	100100 (AF42)
High Drop	001110 (AF13)	010110 (AF23)	011110 (AF33)	100110 (AF43)

3-8-2 Expedited Forwarding:-

In this section the Expedited Forwarding (EF) PHB is described as defined by Davie etal.It was originally described by Jacobson et al. but some problems in the specification led to an updated specification. The EF PHB can be used to build a low loss, low latency, low jitter, assured bandwidth, end-to-end service through DS domains. Loss, jitter, and latency are caused by the queues that traffic experience. To provide low loss, latency, and jitter, traffic should encounter small queues. To

achieve this, the aggregate's maximum arrival rate must be less than its minimum departure rate. To create such a service two things are needed: an aggregate should have a well- defined minimum departure rate and the arrival rate should be less than the configured minimum departure rate. The EF PHB provides the first requirement.By appropriately shaping, dropping or marking packets, the traffic conditioners attempt to provide for the second requirement. One way of implementing the EF PHB is to use an output buffered device which delivers packet immediately to the appropriate output queue with a priority queue for EF traffic. A recent proposal by Nichols et al. (2004) for a 'virtual wire' service across a Diffserv domain defines a per-domain behavior based on the previous EF research. Components of a Diffservrouter.

3-9 Components of a Diffserv router:-

In this section an informal conceptual model of a Diffserv router is provided. The model is defined in Bernetetal. (2002).

Figure 3-10 gives a graphical representation of the major functional blocks of a DS router.

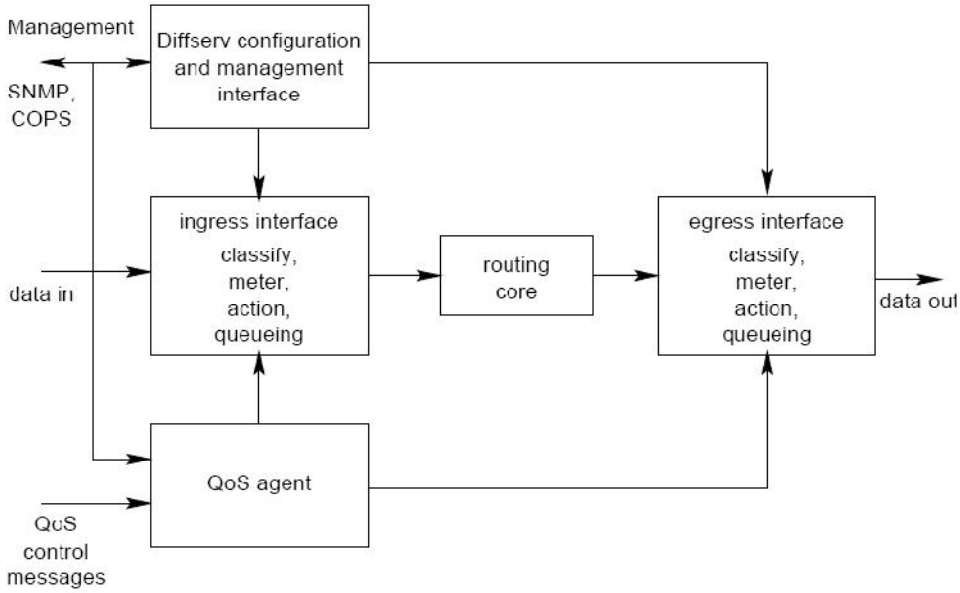

Figure 3-10 Major functional blocks in a Diffserv router

The inner block labeled 'routing core' represents the normal routing and switching functionality (outside Diffserv). In addition, however, the

diagram shows ingress and egress interface blocks that may contain the following Diffserv functions:

a. Traffic Classification elements.
b. Metering functions.
c. Action elements providing for marking, dropping, counting, and multiplexing.
d. Queuing elements, including capabilities of dropping and scheduling.
e. Certain combinations of the above functional data path elements into higher-level blocks known as Traffic Conditioning Blocks (TCBs).

These building blocks of the router's functionality need to be managed by Diffserv configuration and management tools. An interface to support such management is depicted in the upper left block of the diagram.

Configuration and Management Interface Diffserv operating parameters are monitored and provisioned through this interface. Monitoring includes gathering statistics regarding traffic forwarded at different Diffserv service levels. These can be used for accounting purposes as well as for tracking compliance to Traffic Conditioning Agreements (TCAs).

Configuration parameters include parameters for classifiers and meters; for PHB; and parameters for action and queuing elements. The interfaces to these are provided by management protocols such as Simple Network Management Protocol (SNMP) or common Open Policy Service (COPS).

The following section provides more detail about the remaining interface blocks in the diagram, namely the ingress and egress interfaces which provide certain Diffserv functions.

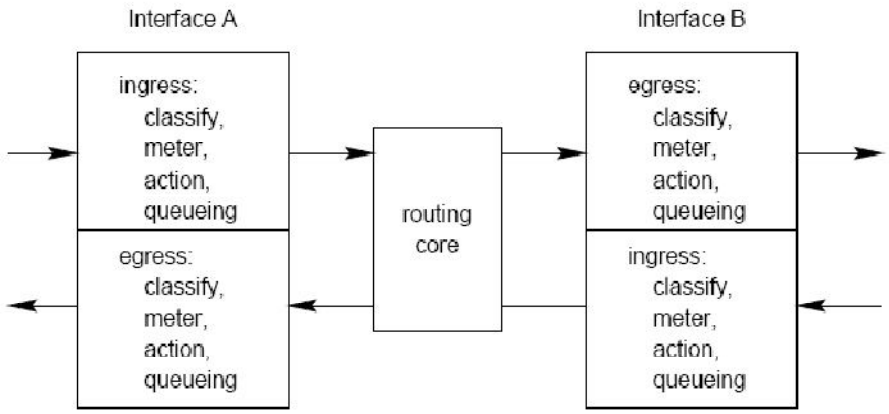

Figure3-11Diffserv interface between A and B

3-10Diffserv Functions at Ingress and Egress:-

Figure 4.6 shows a high-level view of the ingress and egress interfaces on a Diffserv router. The figure illustrates two Diffserv interfaces. It also shows the functions that might be instantiated on each interface.

Each router should perform the following four QoS control functions on traffic in the data path.

Classify each packet according to some set of rules.

Determine if the traffic stream to which the packet belongs is in- or out-of-profile by metering the stream.

Perform a set of resulting actions. These might include applying a drop policy or marking of traffic with a DS code point.

in queue the packet in the appropriate queue. The scheduling of the queue may lead to shaping of the packet stream or cause the packets to be forwarded at some minimum rate.

Not all functions will be implemented on all interfaces of all routers. Ingress and egress routers may differ in the same way that core and edge routers may differ.

3-11Related Work:

About	Authors	Problem	Solution
Performance Analysis of Integrated Service Over Differentiated Service for Next Generation Internet (2010). [17]	ShammiAkhtar, Emdad Ahmed, AlokekumarSaha, and KaziShamsulArefin	on next generation Internet how we can provide Quality of Services (QoS) to the users while today's internet provides the BES (Best Effort Services), that does not guaranteed the QoS	set up a network that carries three applications: FTP, Video, and VoIP and designed the architecture using OPNET ITGURU Academic edition. Besides this, we generated graphs for three different applications and examined these graphs and compared with each other, which can provide a better solutionfor next generation internet
DiffServ Extension Allowing User Applications to Effect QoS Control (2011). [18]	Jiri Hosek, Karol Molnar, LukasRucka	new QoS control system where the user application is able to cooperate with the edge node of the DiffServ domain and can affect the allocation of network resources	The model is composed from a special SNMP manager and SNMP agent. The simulation model was focused on the evaluation of the SNMP-based communication process.
The Effects of Different Queuing Algorithms within the Router on QoS VoIP application Using OPNET(2.013) .[19]	Dr.Hussein Mohammed, Dr.AdnanHussein Ali, HawraaJassim Mohammed	study the effect of different queuing algorithms within the router on VoIP QoS.	a comparison was carried out between different queue algorithms, and it was found that PQ and WFQ algorithms are the most appropriate to improve VoIP QoS

3.12 Algorithm:

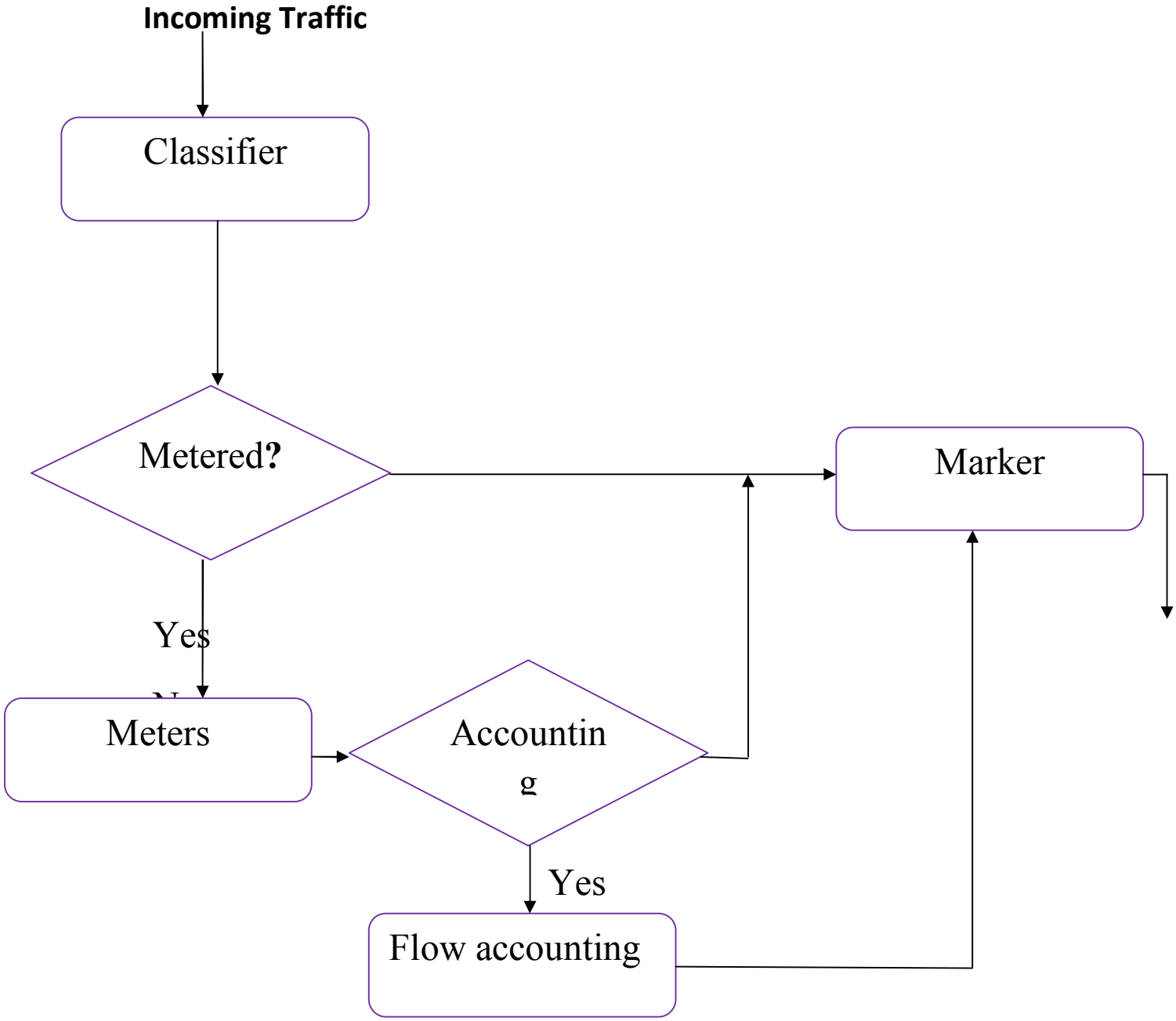

Figure 3-12 Traffic Flow through the IPQoS Implementation of the Diffserv Model

This figure illustrates a common traffic flow sequence on an IPQoS-enabled machine:

1. The classifier selects from the packet stream all packets that match the filtering criteria in the system's QoS policy.

2. The selected packets are then evaluated for the next action to be taken.
3. The classifier sends to the marker any traffic that does not require flow control.
4. Traffic to be flow-controlled is sent to the meter.
5. The meter enforces the configured rate. Then, the meter assigns a traffic conformance value to the flow-controlled packets.
6. The flow-controlled packets are then evaluated to determine if any packets require accounting.
7. The meter sends to the marker any traffic that does not require flow accounting.
8. The flow-accounting module gathers statistics on received packets. The module then sends the packets to the marker.
9. The marker assigns a DS codepoint to the packet header. This DSCP indicates the per-hop behavior that a Diffserv-aware system must apply to the packet.