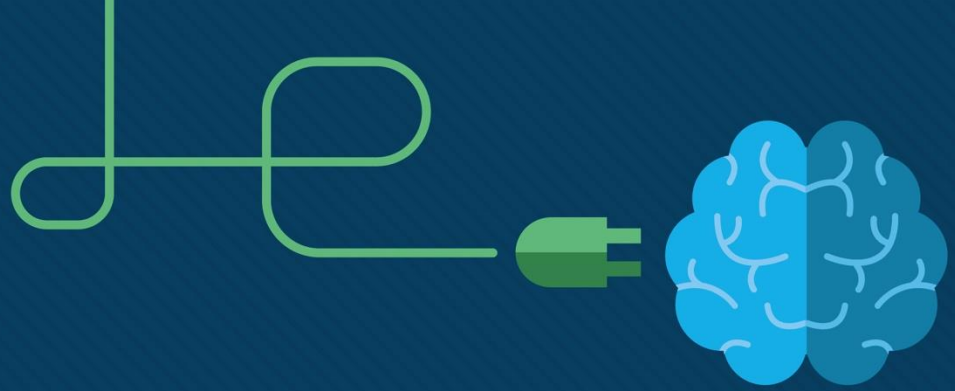


Chapter 14: QoS

Instructor Materials

CCNP Enterprise: Core Networking

Chapter 14 Content

This chapter covers the following content:

The Need for QoS - This section describes the leading causes of poor quality of service and how they can be alleviated by using QoS tools and mechanisms.

QoS Models - Three different models available for implementing QoS in a network: best effort, Integrated Services (IntServ), and Differentiated Services (DiffServ).

Classification and Marking - Classification is used to identify and assign IP traffic into different traffic classes. Marking is used to mark packets with a specified priority based on classification or traffic conditioning policies.

Policing and Shaping - Used to enforce rate limiting, where excess IP traffic is either dropped, marked, or delayed.

Congestion Management and Avoidance - A queueing mechanism is used to prioritize and protect IP traffic. Congestion avoidance involves discarding IP traffic to avoid network congestion.

The Need for QoS

- QoS is a network infrastructure technology that relies on a set of tools and mechanisms to assign different levels of priority to different IP traffic flows and provides special treatment to higher-priority IP traffic flows.
- For higher-priority IP traffic flows, it reduces packet loss during times of network congestion and also helps control delay (latency) and delay variation (jitter); for low-priority IP traffic flows, it provides a best-effort delivery service.
- Mechanisms used to achieve QoS goals include **c**lassification and marking, policing and shaping, congestion management and avoidance.

Causes and Results of Quality Issues

When packets are delivered using a best-effort delivery model, they may not arrive in order or in a timely manner, and they may be dropped.

- For video, this can result in pixelization of the image, pausing, choppy video, audio and video being out of sync, or no video at all.
- For audio, it could cause echo, talker overlap (a walkie-talkie effect where only one person can speak at a time), unintelligible and distorted speech, voice breakups, long silence gaps, and call drops.

The following are the leading causes of quality issues:

- Lack of bandwidth
- Latency and jitter
- Packet loss

Lack of Bandwidth

The available bandwidth on the data path from a source to a destination equals the capacity of the lowest-bandwidth link.

When the maximum capacity of the lowest-bandwidth link is surpassed, link congestion takes place, resulting in traffic drops.

The solution to this type of problem:

- Increase the link bandwidth capacity, but this is not always possible, due to budgetary or technological constraints.
- Implement QoS mechanisms such as policing and queueing to prioritize traffic according to level of importance.
  - Voice, video, and business-critical traffic should get prioritized forwarding and sufficient bandwidth to support their application requirements.
  - The least important traffic should be allocated the remaining bandwidth.

Latency and Jitter

One-way end-to-end delay, also known as network latency, is the time it takes for packets to travel across a network from a source to a destination.

Regardless of the application type, ITU Recommendation G.114 recommends:

- A network latency of 400 ms should not be exceeded,
- For real-time traffic, network latency should be less than 150 ms; however the ITU and Cisco have demonstrated that real-time traffic quality does not begin to significantly degrade until network latency exceeds 200 ms.

Network latency can be broken down into fixed and variable latency:

- Propagation delay (fixed)
- Serialization delay (fixed)
- Processing delay (fixed)
- Delay variation (variable)

Propagation Delay

Propagation delay is the time it takes for a packet to travel from the source to a destination at the speed of light over a medium such as fiber-optic cables or copper wires.

- The speed of light is 299,792,458 meters per second in a vacuum.
- The lack of vacuum conditions in a fiber-optic cable or a copper wire slows down the speed of light by a ratio known as the *refractive index*; the larger the refractive index value, the slower light travels.
- The average refractive index value of an optical fiber is about 1.5. The speed of light through a medium *v* is equal to the speed of light in a vacuum *c* divided by the refractive index *n*, or $v = c / n$. This means the speed of light through a fiber-optic cable with a refractive index of 1.5 is approximately 200,000,000 meters per second (that is, 300,000,000 / 1.5).
- If a single fiber-optic cable with a refractive index of 1.5 were laid out around the equatorial circumference of Earth, which is about 40,075 km, the propagation delay would be equal to the equatorial circumference of Earth divided by 200,000,000 meters per second. This is approximately 200 ms.

Serialization Delay/Processing Delay

Serialization delay is the time it takes to place all the bits of a packet onto a link.

- It is a fixed value that depends on the link speed; the higher the link speed, the lower the delay.
- The serialization delay *s* is equal to the packet size in bits divided by the line speed in bits per second.

Processing delay is the fixed amount of time it takes for a networking device to take the packet from an input interface and place the packet onto the output queue of the output interface.

The processing delay depends on factors such as the following:

CPU speed (for software-based platforms)
CPU utilization (load)
IP packet switching mode (process switching, software CEF, or hardware CEF)
Router architecture (centralized or distributed)
Configured features on both input and output interfaces

Delay Variation/Packet Loss

Delay variation, also referred to as jitter, is the difference in the latency between packets in a single flow. For example, if one packet takes 50 ms to traverse the network from the source to destination, and the following packet takes 70 ms, the jitter is 20 ms.

The major factors affecting variable delays are queuing delay, dejitter buffers, and variable packet sizes. Jitter is experienced due to the queueing delay experienced during periods of network congestion.

Packet loss is usually a result of congestion on an interface, and can be prevented by implementing one of the following approaches:

- Increase link speed.
- Implement QoS congestion-avoidance and congestion-management mechanism.
- Implement traffic policing to drop low-priority packets and allow high-priority traffic through.
- Implement traffic shaping to delay packets instead of dropping them since traffic may burst and exceed the capacity of an interface buffer. Traffic shaping is not recommended for real-time traffic because it relies on queuing that can cause jitter.

QoS Models

Three different QoS implementation models:

- **Best effort -** QoS is not enabled for this model. It is used for traffic that does not require any special treatment.
- **Integrated Services (IntServ) -** Applications signal the network to make a bandwidth reservation and to indicate that they require special QoS treatment.
- **Differentiated Services (DiffServ) -** The network identifies classes that require special QoS treatment.

IntServ Model

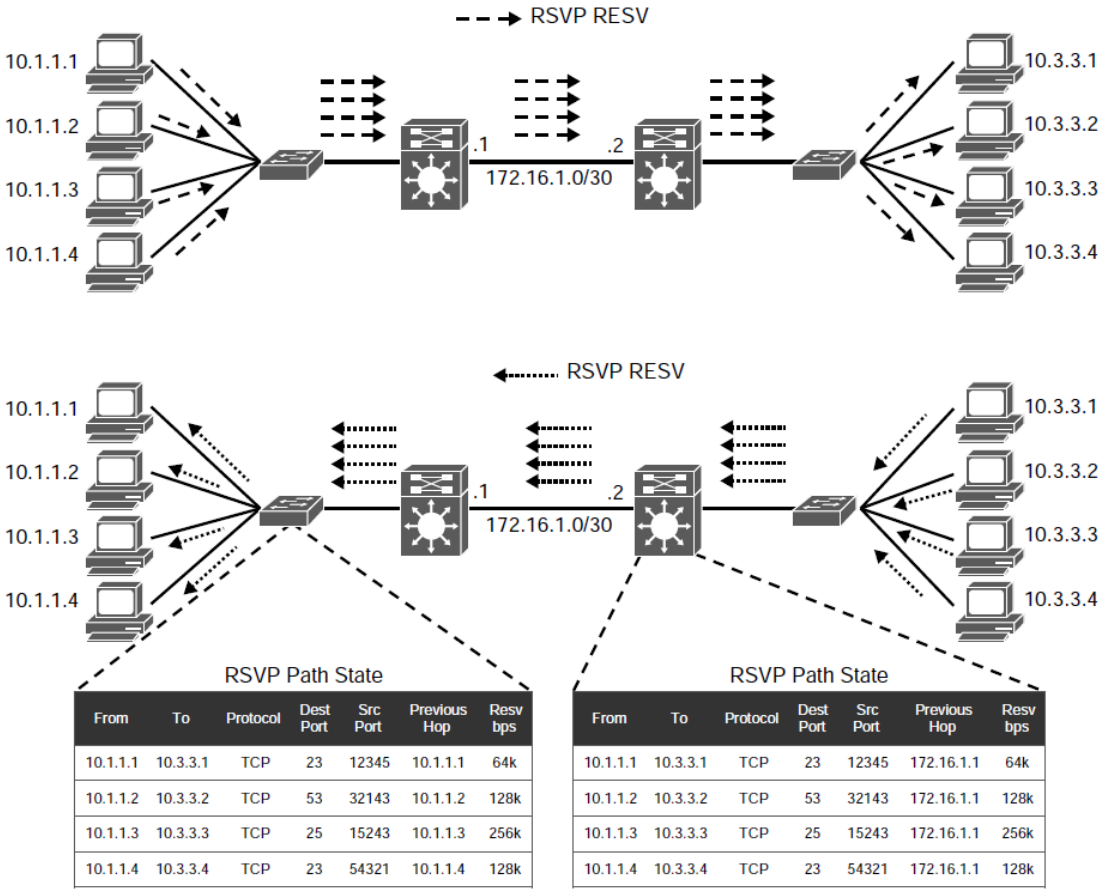
IntServ was created for real-time applications such as voice and video that require bandwidth, delay, and packet-loss guarantees to ensure both predictable and guaranteed service levels.

The network reserves the end-to-end resources (such as bandwidth) the application requires to provide an acceptable user experience.

- Resource Reservation Protocol (RSVP):
  - Used to reserve resources throughout a network
  - Provides call admission control (CAC) to guarantee that no other IP traffic can use the reserved bandwidth
  - Any bandwidth reserved and not used is wasted
- End-to-end QoS requires all nodes, including the endpoints running the applications, to support, build, and maintain RSVP path state for every single flow.
- Intserv does not scale well on large networks that might have thousands or millions of flows due to the large number of RSVP flows that would need to be maintained.

RSVP Reservation

- Hosts on the left side (senders) are attempting to establish a one-to-one bandwidth reservation to each of the hosts on the right side (receivers).
- The senders start by sending RSVP PATH messages to the receivers.
- RSVP PATH messages carry the receiver source address, the destination address, and the bandwidth they wish to reserve.
- This information is stored in the RSVP path state of each node.



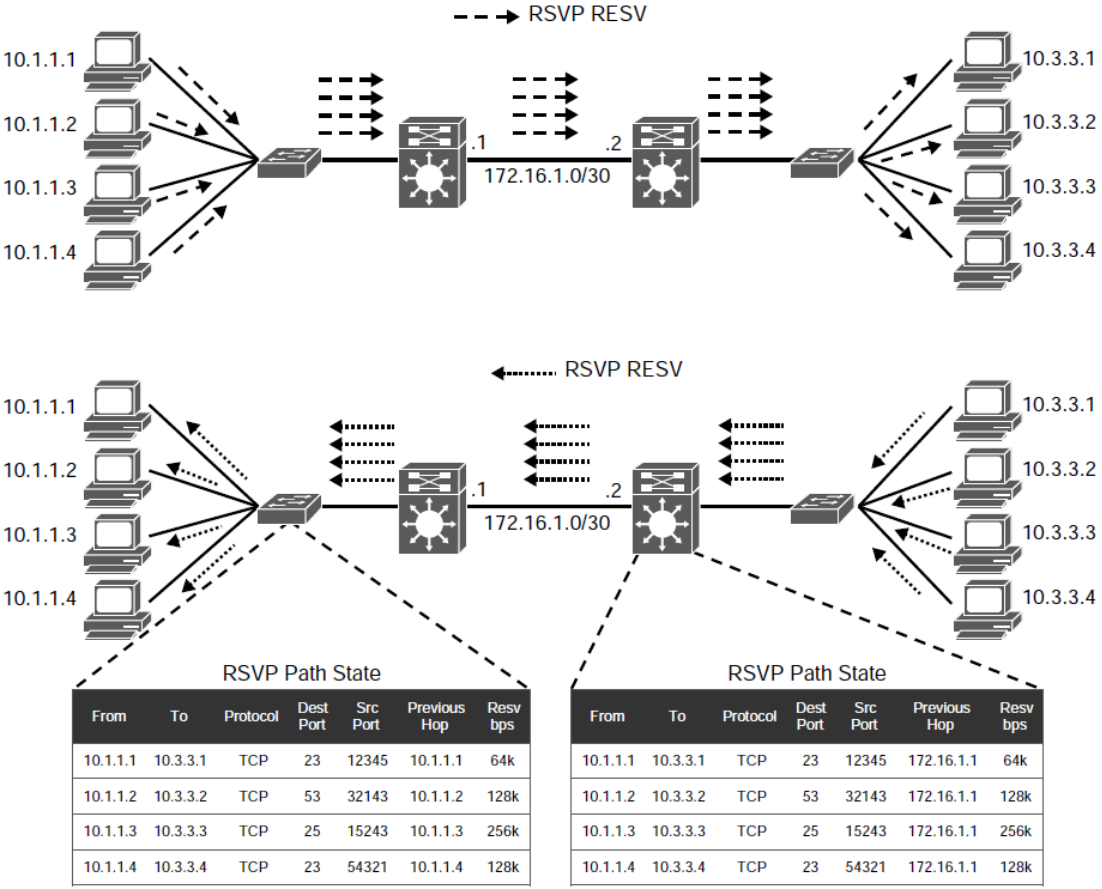
From	To	Protocol	Dest Port	Src Port	Previous Hop	Resv bps
10.1.1.1	10.3.3.1	TCP	23	12345	10.1.1.1	64k
10.1.1.2	10.3.3.2	TCP	53	32143	10.1.1.2	128k
10.1.1.3	10.3.3.3	TCP	25	15243	10.1.1.3	256k
10.1.1.4	10.3.3.4	TCP	23	54321	10.1.1.4	128k

From	To	Protocol	Dest Port	Src Port	Previous Hop	Resv bps
10.1.1.1	10.3.3.1	TCP	23	12345	172.16.1.1	64k
10.1.1.2	10.3.3.2	TCP	53	32143	172.16.1.1	128k
10.1.1.3	10.3.3.3	TCP	25	15243	172.16.1.1	256k
10.1.1.4	10.3.3.4	TCP	23	54321	172.16.1.1	128k

Figure 14-1 *RSVP Reservation Establishment*

RSVP Reservation (Cont.)

- Once the RSVP PATH messages reach the receivers, each receiver sends RSVP reservation request (RESV) messages in the reverse path of the data flow toward the receivers, hop-by-hop.
- At each hop, the IP destination address of a RESV message is the IP address of the previous-hop node, obtained from the RSVP path state of each node.
- As RSVP RESV messages cross each hop, they reserve bandwidth on each of the links for the traffic flowing from the receiver hosts to the sender hosts.



From	To	Protocol	Dest Port	Src Port	Previous Hop	Resv bps
10.1.1.1	10.3.3.1	TCP	23	12345	10.1.1.1	64k
10.1.1.2	10.3.3.2	TCP	53	32143	10.1.1.2	128k
10.1.1.3	10.3.3.3	TCP	25	15243	10.1.1.3	256k
10.1.1.4	10.3.3.4	TCP	23	54321	10.1.1.4	128k

From	To	Protocol	Dest Port	Src Port	Previous Hop	Resv bps
10.1.1.1	10.3.3.1	TCP	23	12345	172.16.1.1	64k
10.1.1.2	10.3.3.2	TCP	53	32143	172.16.1.1	128k
10.1.1.3	10.3.3.3	TCP	25	15243	172.16.1.1	256k
10.1.1.4	10.3.3.4	TCP	23	54321	172.16.1.1	128k

Figure 14-1 *RSVP Reservation Establishment*

DiffServ Model

DiffServ addresses the limitations of the best-effort and IntServ models.

- There is no need for a signaling protocol.
- It is highly scalable since there is no RSVP flow state to maintain.
- QoS characteristics (such as bandwidth and delay) are managed on a hop-by-hop basis.
- QoS policies are defined independently at each device in the network.
- DiffServ is not considered an end-to-end QoS solution because end-to-end QoS guarantees cannot be enforced.
- DiffServ divides IP traffic into classes and marks it based on business requirements.
- Each of the classes can be assigned a different level of service.
- Each of the network devices identifies the packet class by its marking and services the packets according to this class.

Classification and Marking

- IP traffic must first be identified and categorized into different classes, based on business requirements.
- Network devices use classification to identify IP traffic as belonging to a specific class.
- Marking can be used to mark or color individual packets so that other network devices can apply QoS mechanisms to those packets.

Classification

Once an IP packet is classified, packets can then be marked/re-marked, queued, policed, shaped, or any combination of these and other actions.

The following traffic descriptors are typically used for classification:

- **Internal -** QoS groups (locally significant to a router)
- **Layer 1 -** Physical interface, subinterface, or port
- **Layer 2 -** MAC address and 802.1Q/p Class of Service (CoS) bits
- **Layer 2.5 -** MPLS Experimental (EXP) bits
- **Layer 3 -** Differentiated Services Code Points (DSCP), IP Precedence (IPP), and source/destination IP address
- **Layer 4 -** TCP or UDP ports
- **Layer 7 -** Next Generation Network-Based Application Recognition (NBAR2)

Layer 7 Classification

NBAR2 is a deep packet inspection engine that can classify and identify a wide variety of protocols and applications using Layer 3 to Layer 7 data.

NBAR2 can recognize more than 1000 applications, and monthly protocol packs are provided for recognition of new and emerging applications, without requiring an IOS upgrade or router reload.

NBAR2 has two modes of operation:

- **Protocol Discovery -** Protocol Discovery enables NBAR2 to discover and get real-time statistics on applications currently running in the network. These statistics from the Protocol Discovery mode can be used to define QoS classes and policies using MQC configuration.
- **Modular QoS CLI (MQC) -** Using MQC, network traffic matching a specific network protocol such as Cisco Webex can be placed into one traffic class, while traffic that matches a different network protocol such as YouTube can be placed into another traffic class. After traffic has been classified in this way, different QoS policies can be applied to the different classes of traffic.

Marking

Packet marking is a QoS mechanism that colors a packet by changing a field within a packet or a frame header with a traffic descriptor so it is distinguished from other packets during the application of other QoS mechanisms (such as re-marking, policing, queuing, or congestion avoidance).

The following traffic descriptors are used for marking traffic:

- **Internal -** QoS groups
- **Layer 2 -** 802.1Q/p Class of Service (CoS) bits
- **Layer 2.5 -** MPLS Experimental (EXP) bits
- **Layer 3 -** Differentiated Services Code Points (DSCP) and IP Precedence (IPP)

QoS groups are used to mark packets as they are received and processed internally within the router and are automatically removed when packets egress the router. They are used only in special cases in which traffic descriptors marked or received on an ingress interface would not be visible for packet classification on egress interfaces due to encapsulation or de-encapsulation.

Layer 2 Marking

- The 802.1Q standard is an IEEE specification for implementing VLANs in Layer 2 switched networks. The 802.1Q specification defines two 2-byte fields: Tag Protocol Identifier (TPID) and Tag Control Information (TCI), which are inserted within an Ethernet frame following the Source Address field, as illustrated in Figure 14-2.

- The TPID value is a 16-bit field assigned the value 0x8100 that identifies it as an 802.1Q tagged frame.

- The TCI field is a 16-bit field composed of the following three fields:
  - Priority Code Point (PCP) field (3 bits)
  - Drop Eligible Indicator (DEI) field (1 bit)
  - VLAN Identifier (VLAN ID) field (12 bits)

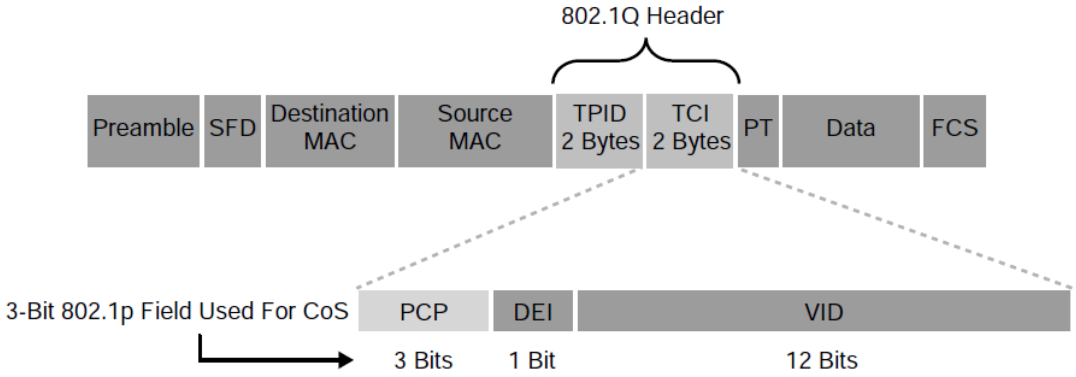


Figure 14-2 *802.1Q Layer 2 QoS Using 802.1p CoS*

Priority Code Point (PCP)

- The specifications of the 3-bit PCP field are defined by the IEEE 802.1p specification.
- This field is used to mark packets as belonging to a specific CoS.
- The CoS marking allows a Layer 2 Ethernet frame to be marked with eight different levels of priority values, 0 to 7, where 0 is the lowest priority and 7 is the highest.

PCP Value/Priority	Acronym	Traffic Type
0 (lowest)	BK	Background
1	BE	Best Effort
2	EE	Excellent Effort
3	CA	Critical Application
4	VI	Video with < 100 ms latency and jitter
5	VO	Voice with < 10 ms latency and jitter
6	IC	Internetwork Control
7 (highest)	NC	Network Control

Priority Code Point (PCP) (Cont.)

One drawback of using CoS is that frames lose their CoS markings when traversing a non-802.1Q link or a Layer 3 network. For this reason, packets should be marked with other higher-layer markings. This is typically accomplished by mapping a CoS marking into another marking.

For example, the CoS priority levels correspond directly to IPv4's IP Precedence Type of Service (ToS) values so they can be mapped directly to each other.

Drop Eligible Indicator (DEI): The DEI field is a 1-bit field that can be used independently or in conjunction with PCP to indicate frames that are eligible to be dropped during times of congestion. The default value for this field is 0 = not drop eligible; set to 1 = is drop eligible.

VLAN Identifier (VLAN ID): The VLAN ID field is a 12-bit field that defines the VLAN used by 802.1Q.

Layer 3 Marking

Using marking at Layer 3 provides a more persistent marker that is preserved end-to-end.

The ToS field is an 8-bit field. Only the first 3 bits of the ToS field, referred to as IP Precedence (IPP), are used for marking, and the rest of the bits are unused. IPP values, which range from 0 to 7, allow the traffic to be partitioned in up to six usable classes of service; IPP 6 and 7 are reserved for internal network use.

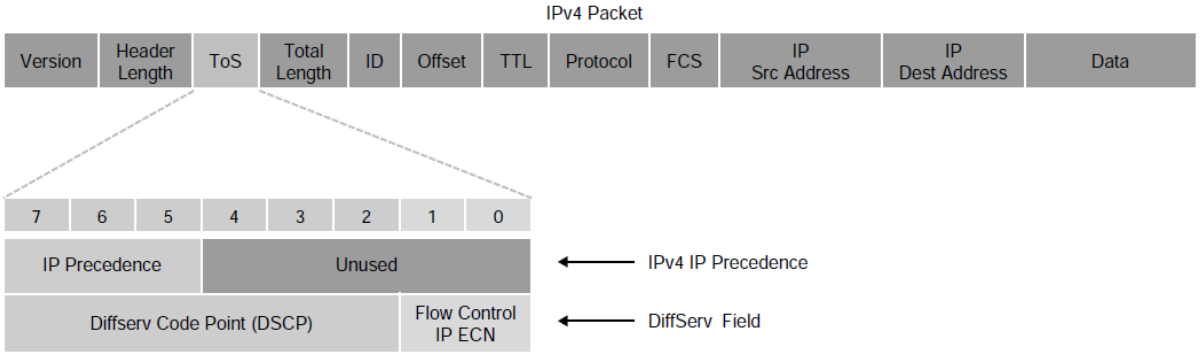


Figure 14-3 *IPv4 ToS/DiffServ Field*

Newer standards have redefined the IPv4 ToS and the IPv6 Traffic Class fields as an 8-bit Differentiated Services (DiffServ) field. The DiffServ field uses the same 8 bits that were previously used for the IPv4 ToS and the IPv6 Traffic Class fields, and this allows it to be backward compatible with IP Precedence. The DiffServ field uses the same 8 bits that were previously used for the IPv4 ToS and the IPv6 Traffic Class fields, and this allows it to be backward compatible with IP Precedence.

DSCP Per-Hop Behaviors

The DiffServ field is used to mark packets according to their classification into DiffServ Behavior Aggregates (BAs). A DiffServ BA is a collection of packets with the same DiffServ value crossing a link in a particular direction. Per-hop behavior (PHB) is the externally observable forwarding behavior (forwarding treatment) applied at a DiffServ-compliant node to a collection of packets with the same DiffServ value crossing a link in a particular direction (DiffServ BA).

PHB is expediting, delaying, or dropping a collection of packets by one or multiple QoS mechanisms on a per-hop basis, based on the DSCP value. A DiffServ BA could be multiple applications marked with the same DSCP value.

Four PHBs have been defined and characterized for general use:

- **Class Selector (CS) PHB -** The first 3 bits of the DSCP field are used as CS bits. The CS bits make DSCP backward compatible with IP Precedence because IP Precedence uses the same 3 bits to determine class.
- **Default Forwarding (DF) PHB -** Used for best-effort service.
- **Assured Forwarding (AF) PHB -** Used for guaranteed bandwidth service.
- **Expedited Forwarding (EF) PHB -** Used for low-delay service.

Class Selector PHB

Class Selector (CS) PHB RFC 2474 made the ToS field obsolete by introducing the DiffServ field, and the Class Selector (CS) PHB was defined to provide backward compatibility for DSCP with IP Precedence.

- The last 3 bits of the DSCP (bits 2 to 4), when set to 0, identify a Class Selector PHB, but the Class Selector bits 5 to 7 are the ones where IP Precedence is set. Bits 2 to 4 are ignored by non-DiffServ-compliant devices performing classification based on IP Precedence.
- There are eight CS classes, ranging from CS0 to CS7, that correspond directly with the eight IP Precedence values.

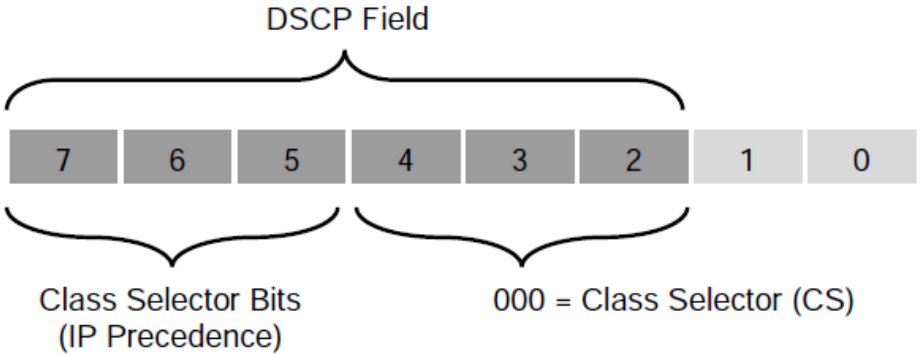


Figure 14-4 *Class Selector (CS) PHB*

Default Forwarding (DF) PHB

Default Forwarding (DF) PHB Default Forwarding (DF) and Class Selector 0 (CS0) provide best-effort behavior and use the DS value 000000.

- Default best-effort forwarding is also applied to packets that cannot be classified by a QoS mechanism such as queueing, shaping, or policing.
- This usually happens when a QoS policy on the node is incomplete or when DSCP values are outside the ones that have been defined for the CS, AF, and EF PHBs.

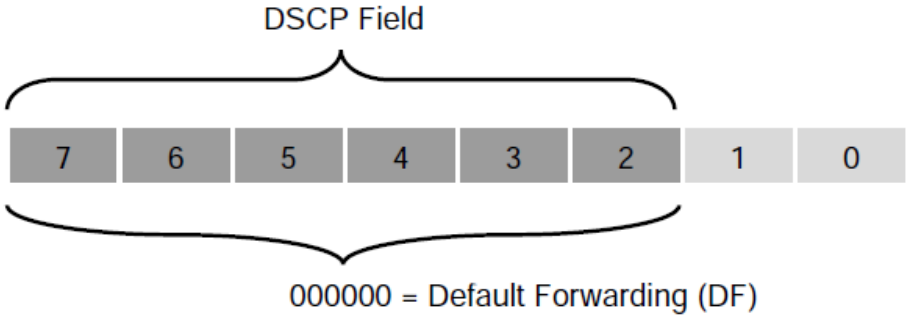


Figure 14-5 *Default Forwarding (DF) PHB*

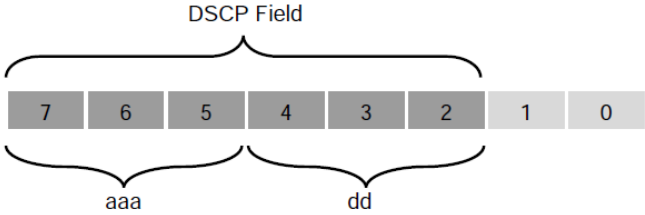
Assured Forwarding (AF) PHB

The AF PHB guarantees a certain amount of bandwidth to an AF class and allows access to extra bandwidth, if available.

Packets requiring AF PHB should be marked with DSCP value aaadd0, where aaa is the binary value of the AF class (bits 5 to 7), and dd (bits 2 to 4) is the drop probability where bit 2 is unused and always set to 0. Figure 14-6 illustrates the AF PHB.

There are four standard-defined AF classes: AF1, AF2, AF3, and AF4.

The AF class number does not represent precedence. AF4 does not get any preferential treatment over AF1. Each class should be treated independently.



AF Class Name	AF IP Precedence Bin	AF IP Precedence Dec (x)	Drop Probability Dec (y)	Drop Probability Bin
AF1	001	1	1	01
AF2	010	2	2	10
AF3	011	3	3	11
AF4	100	4		

Figure 14-6 *Assured Forwarding (AF) PHB*

Assured Forwarding (AF) PHB (Cont.)

Table 14-3 illustrates how each AF class is assigned an IP Precedence (under AF Class Value Bin) and has three drop probabilities: low, medium, and high.

- The AF Name (AF*xy*) is composed of the AF IP Precedence value in decimal (*x*) and the Drop Probability value in decimal (*y*).
- For example, AF41 is a combination of IP Precedence 4 and Drop Probability 1.
- To quickly convert the AF Name into a DSCP value in decimal, use the formula 8*x* + 2*y*. For example, the DSCP value for AF41 is 8(4) + 2(1) = 34.

AF Class Name	AF IP Procedure Dec (x)	AF IP Procedure Bin	Drop Probability	Drop Probability Value Bin	Drop Probability Value Dec (y)	AF Name (Afxy)	DSCP Value Bin	DSCP Value Dec
AF1	1	001	Low	01	1	AF11	001010	10
AF1	1	001	Medium	10	2	AF12	001100	12
AF1	1	001	High	11	3	AF13	001110	14
AF2	2	010	Low	01	1	AF21	010010	18
AF2	2	010	Medium	10	2	AF22	010100	20
AF2	2	010	High	11	3	AF23	010110	22
AF3	3	011	Low	01	1	AF31	011010	26
AF3	3	011	Medium	10	2	AF32	011100	28
AF3	3	011	High	11	3	AF33	011110	30
AF4	4	100	Low	01	1	AF41	100010	34
AF4	4	100	Medium	10	2	AF42	100100	36
AF4	4	100	High	11	3	AF43	100110	38

Assured Forwarding (AF) and WRED

- An AF implementation must detect and respond to long-term congestion within each class by dropping packets using a congestion-avoidance algorithm such as weighted random early detection (WRED).
- WRED uses the AF Drop Probability value within each class—where 1 is the lowest possible value, and 3 is the highest possible—to determine which packets should be dropped first during periods of congestion.
- It should also be able to handle short-term congestion resulting from bursts if each class is placed in a separate queue, using a queueing algorithm such as class-based weighted fair queueing (CBWFQ). The AF specification does not define the use of any particular algorithms to use for queueing and congestion avoidance, but it does specify the requirements and properties of such algorithms.

Expedited Forwarding (EF) PHB

The EF PHB can be used to build a low-loss, low-latency, low-jitter, assured bandwidth, end-to-end service.

- The EF PHB guarantees bandwidth by ensuring a minimum departure rate and provides the lowest possible delay by implementing low-latency queueing.
- It also prevents starvation of other applications or classes that are not using the EF PHB by policing EF traffic when congestion occurs.
- Packets requiring EF should be marked with DSCP binary value 101110 (46 in decimal). Bits 5 to 7 (101) of the EF DSCP value map directly to IP Precedence 5 for backward compatibility .

Scavenger Class

The scavenger class is intended to provide less than best-effort services.

Applications assigned to the scavenger class have little or no contribution to the business objectives of an organization and are typically entertainment-related applications. These include:

- peer-to-peer applications (such as Torrent),
- gaming applications (for example, Minecraft, Fortnite), and
- entertainment video applications (for example, YouTube, Vimeo, Netflix).

These types of applications are usually heavily rate limited or blocked entirely.

- Something very peculiar about the scavenger class is that it is intended to be lower in priority than a best-effort service.
- Best-effort traffic uses a DF PHB with a DSCP value of 000000 (CS0). Since there are no negative DSCP values, it was decided to use CS1 as the marking for scavenger traffic. This is defined in RFC 4594.

Trust Boundary

Packets should be marked by the endpoint or as close to the endpoint as possible.

- When an endpoint marks a frame or a packet with a CoS or DSCP value, the switch port it is attached to can be configured to accept or reject the CoS or DSCP values.
  - If the switch accepts the values, it means it trusts the endpoint and does not need to do any packet reclassification and re-marking for the received endpoint's packets.
  - If the switch does not trust the endpoint, it rejects the markings and reclassifies and re-marks the received packets with the appropriate CoS or DSCP value.
- For example, consider a campus network with IP telephony and host endpoints; the IP phones by default mark voice traffic with a CoS value of 5 and a DSCP value of 46 (EF), while incoming traffic from an endpoint (such as a PC) attached to the IP phone's switch port is re-marked to a CoS value of 0 and a DSCP value of 0.
- Even if the endpoint is sending tagged frames with a specific CoS or DSCP value, the default behavior for Cisco IP phones is to not trust the endpoint and zero out the CoS and DSCP values before sending the frames to the switch. When the IP phone sends voice and data traffic to the switch, the switch can classify voice traffic as higher priority than the data traffic, thanks to the high-priority CoS and DSCP markings for voice traffic.

Trust Boundary Example (Cont.)

- The IP phones by default mark voice traffic with a CoS value of 5 and a DSCP value of 46 (EF), while incoming traffic from an endpoint (such as a PC) attached to the IP phone's switch port is re-marked to a CoS value of 0 and a DSCP value of 0.
- Even if the endpoint is sending tagged frames with a specific CoS or DSCP value, the default behavior for Cisco IP phones is to not trust the endpoint and zero out the CoS and DSCP values before sending the frames to the switch. When the IP phone sends voice and data traffic to the switch, the switch can classify voice traffic as higher priority than the data traffic, thanks to the high-priority CoS and DSCP markings for voice traffic.
- Figure 14-7 illustrates trust boundaries at different points in a campus network, where 1 and 2 are optimal, and 3 is acceptable only when the access switch is not capable of performing classification.

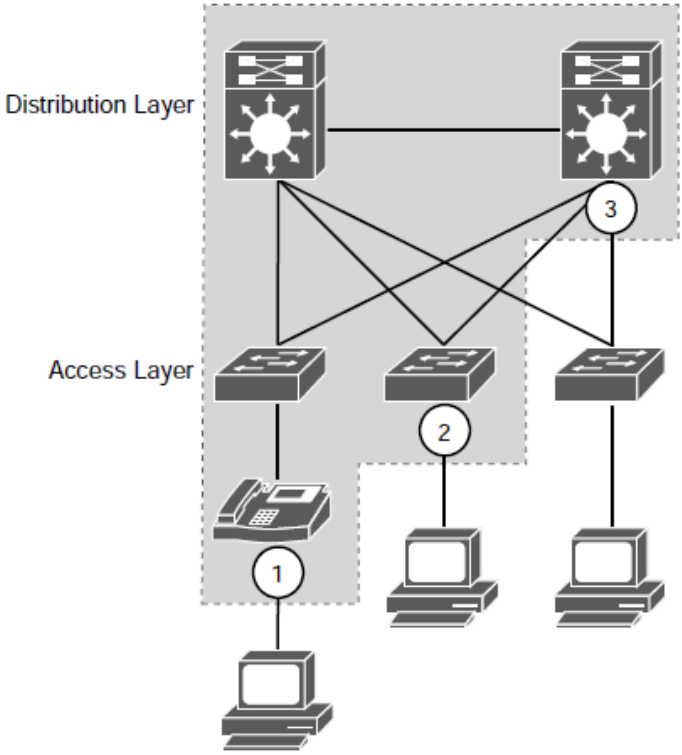


Figure 14-7 *Trust Boundaries*

A Practical Example: Wireless QoS

A wireless network can be configured to leverage the QoS mechanisms. For example, a wireless LAN controller (WLC) sits at the boundary between wireless and wired networks, so it becomes a natural location for a QoS trust boundary.

- Traffic entering and exiting the WLC can be classified and marked so that it can be handled appropriately as it is transmitted over the air and onto the wired network.
- Wireless QoS can be uniquely defined on each wireless LAN (WLAN), using the four traffic categories listed in table below. Notice that the category names are human-readable words that translate to specific 802.1p and DSCP values.

QoS Category	Traffic Type	8021.p Tag	DSCP Value
Platinum	Voice	5	46 (EF)
Gold	Video	4	34 (AF41)
Silver	Best Effort (Default)	0	0
Bronze	Background	1	10 (AF11)

A Practical Example: Wireless QoS (Cont.)

- When you create a new WLAN, its QoS policy defaults to Silver, or best-effort handling.
- In Figure 14-8, a WLAN named ‘voice’ has been created to carry voice traffic, so its QoS policy has been set to Platinum. Wireless voice traffic will then be classified for low latency and low jitter and marked with an 802.1p CoS value of 5 and a DSCP value of 46 (EF).

Figure 14-8 *Setting the QoS Policy for a Wireless LAN*

Policing and Shaping

- *Traffic policers and shapers* are traffic-conditioning QoS mechanisms used to classify traffic and enforce other QoS mechanisms such as rate limiting.
- *Traffic policers and shapers* classify traffic in an identical manner but differ in their implementation.
- Policers**:** Drop or re-mark incoming or outgoing traffic that goes beyond a desired traffic rate.
- Shapers**:** Buffer and delay egress traffic rates that momentarily peak above the desired rate until the egress traffic rate drops below the defined traffic rate. If the egress traffic rate is below the desired rate, the traffic is sent immediately.

Placing Policers and Shapers in the Network

Policers for incoming traffic are most optimally deployed at the edge of the network to keep traffic from wasting valuable bandwidth in the core of the network.

- Policers for outbound traffic are most optimally deployed at the edge of the network or core-facing interfaces on network edge devices.
- A downside of policing is that it causes TCP retransmissions when it drops traffic.

Shapers are used for egress traffic and typically deployed by enterprise networks on service provider (SP)–facing interfaces.

- Shaping is useful in cases where SPs are policing incoming traffic or when SPs are not policing traffic but do have a maximum traffic rate SLA, which, if violated, could incur monetary penalties.
- Shaping buffers and delays traffic rather than dropping it, and this causes fewer TCP retransmissions compared to policing.

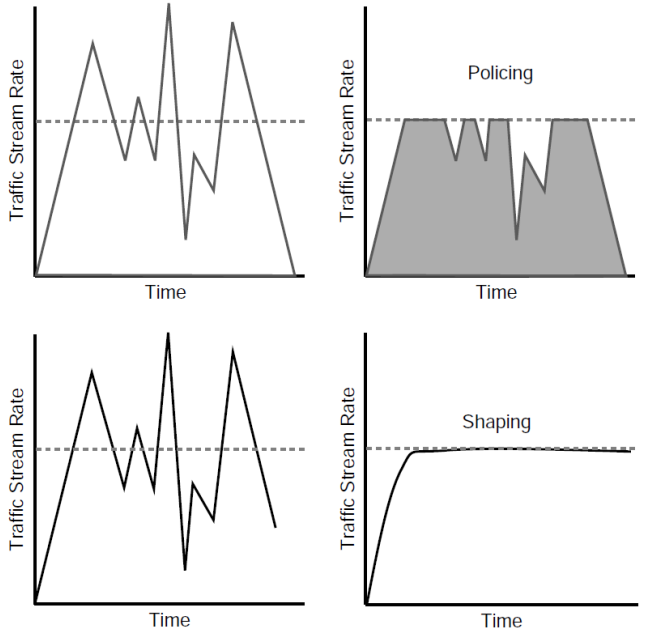


Figure 14-9 *Policing Versus Shaping*

Figure 14-9 illustrates the difference between traffic policing and shaping. Policers drop or re-mark excess traffic, while shapers buffer and delay excess traffic.

Markdown

When a desired traffic rate is exceeded, a policer can take one of the following actions:

- Drop the traffic.
- Mark down the excess traffic with a lower priority.

Marking down excess traffic involves re-marking the packets with a lower-priority class value:

- For example, excess traffic marked with AFx1 should be marked down to AFx2 (or AFx3 if using two-rate policing).
- After marking down the traffic, congestion avoidance mechanisms, such as DSCP-based weighted random early detection (WRED), should be configured throughout the network to drop AFx3 more aggressively than AFx2 and drop AFx2 more aggressively than AFx1.

Token Bucket Algorithms

Cisco IOS policers and shapers are based on token bucket algorithms. The following list includes definitions that are used to explain how token bucket algorithms operate:

- **Committed Information Rate (CIR) -** The policed traffic rate, in bits per second (bps), defined in the traffic contract.
- **Committed Time Interval (Tc) -** The time interval, in milliseconds (ms), over which the committed burst (Bc) is sent. Tc can be calculated with the formula Tc = (Bc [bits] / CIR [bps]) × 1000.
- **Committed Burst Size (Bc) -** The maximum size of the CIR token bucket, measured in bytes, and the maximum amount of traffic that can be sent within a Tc. Bc can be calculated with the formula Bc = CIR × (Tc / 1000).
- **Token -** A single token represents 1 byte or 8 bits.

Token Bucket Algorithms (Cont.)

Token bucket: A bucket that accumulates tokens until a maximum predefined number of tokens is reached (such as the Bc when using a single token bucket). These tokens are added into the bucket at a fixed rate (the CIR). Each packet is checked for conformance to the defined rate and takes tokens from the bucket equal to its packet size. For example, if the packet size is 1500 bytes, it takes 12,000 bits (1500 × 8) from the bucket.

If there are not enough tokens in the token bucket to send the packet, the traffic conditioning mechanism can take one of the following actions:

- Buffer the packets while waiting for enough tokens to accumulate in the token bucket (traffic shaping)
- Drop the packets (traffic policing)
- Mark down the packets (traffic policing)

Single Token Bucket Algorithm

It is recommended for the Bc value to be larger than or equal to the size of the largest possible IP packet in a traffic stream. Otherwise, there will never be enough tokens in the token bucket for larger packets, and they will always exceed the defined rate.

- If the bucket fills up to the maximum capacity, newly added tokens are discarded. Discarded tokens are not available for use in future packets.
- Token bucket algorithms may use one or multiple token buckets.
- For single token bucket algorithms, the measured traffic rate can conform to or exceed the defined traffic rate. The measured traffic rate is conforming if there are enough tokens in the token bucket to transmit the traffic. The measured traffic rate is exceeding if there are not enough tokens in the token bucket to transmit the traffic.

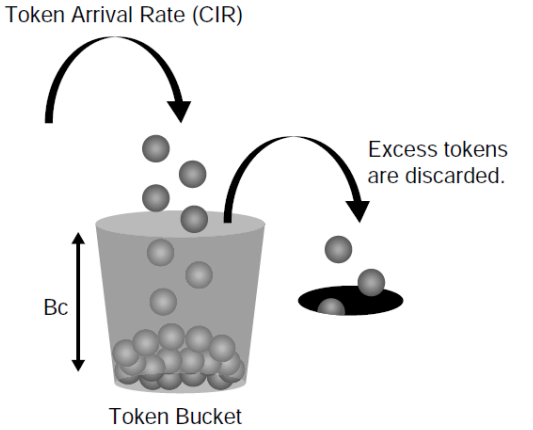


Figure 14-10 *Single Token Bucket Algorithm*

Single Token Bucket Operation

To understand how the single token bucket algorithms operate in more detail, assume that a 1 Gbps interface is configured with a policer defined with a CIR of 120 Mbps and a Bc of 12 Mb.

The Tc value cannot be explicitly defined in IOS, but it can be calculated as follows:

Tc = (Bc [bits] / CIR [bps]) × 1000

Tc = (12 Mb / 120 Mbps) × 1000

Tc = (12,000,000 bits / 120,000,000 bps) × 1000 = 100 ms

Once the Tc value is known, the number of Tcs within a second can be calculated as follows:

Tcs per second = 1000 / Tc

Tcs per second = 1000 ms / 100 ms = 10 Tcs

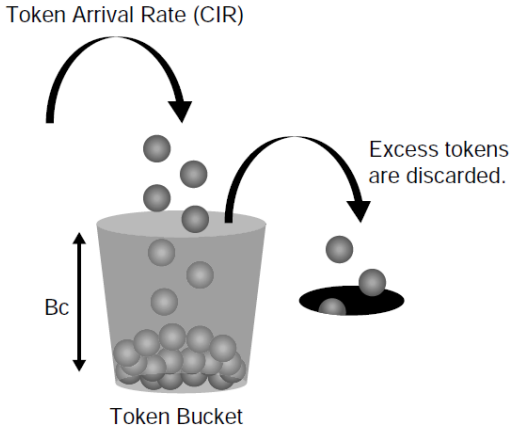


Figure 14-10 *Single Token Bucket Algorithm*

Single Token Bucket Operation (Cont.)

If a continuous stream of 1500-byte (12,000-bit) packets is processed by the token algorithm, only a Bc of 12 Mb can be taken by the packets within each Tc (100 ms). The number of packets that conform to the traffic rate and are allowed to be transmitted can be calculated as follows:

- Number of packets that conform within each Tc = Bc / packet size in bits (rounded down)
- Number of packets that conform within each Tc = 12,000,000 bits / 12,000 bits = 1000 packets

Any additional packets beyond 1000 will either be dropped or marked down.

To figure out how many packets would be sent in one second, the following formula can be used:

- Packets per second = Number of packets that conform within each Tc × Tcs per second
- Packets per second = 1000 packets × 10 intervals = 10,000 packets

CIR Calculation

To calculate the CIR for the 10,000, the following formula can be used:

- CIR = Packets per second × Packet size in bits
- CIR = 10,000 packets per second × 12,000 bits = 120,000,000 bps = 120 Mbps

To calculate the time interval it would take for the 1000 packets to be sent at interface line rate, the following formula can be used:

- Time interval at line rate = (Bc [bits] / Interface speed [bps]) × 1000
- Time interval at line rate = (12 Mb / 1 Gbps) × 1000
- Time interval at line rate = (12,000,000 bits / 1000,000,000 bps) × 1000 = 12 ms

CIR Calculation (Cont.)

Figure 14-11 illustrates how the Bc (1000 packets at 1500 bytes each, or 12Mb) is sent every Tc interval. After the Bc is sent, there is an interpacket delay of 113 ms (125 ms minus 12 ms) within the Tc where there is no data transmitted.

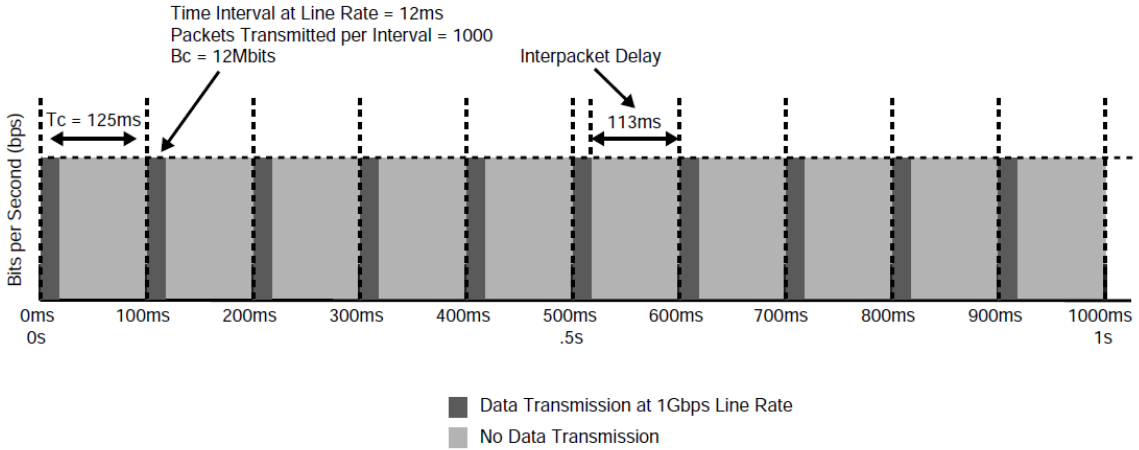


Figure 14-11 *Token Bucket Operation*

The recommended values for Tc range from 8 ms to 125 ms. Shorter Tcs, such as 8 ms to 10 ms, are necessary to reduce interpacket delay for real-time traffic such as voice. Tcs longer than 125 ms are not recommended for most networks because the interpacket delay becomes too large.

Single Rate Two-Color Markers/Policers

There are different policing algorithms, including single-rate two-color marker/policer, single-rate three-color marker/policer (srTCM), two-rate three-color marker/policer (trTCM). Single-rate, two-color model is based on the single token bucket algorithm. For this type of policer, traffic can be either conforming to or exceeding the CIR. Marking down or dropping actions can be performed for each of the two states.

Figure 14-12 illustrates different actions that the single-rate two-color policer can take.

- The section above the dotted line on the left side of the figure represents traffic that exceeded the CIR and was marked down.
- The section above the dotted line on the right side of the figure represents traffic that exceeded the CIR and was dropped.

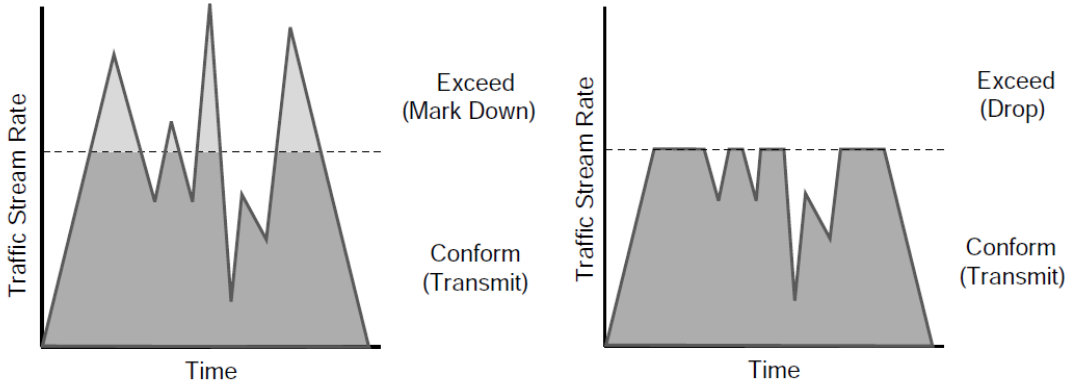


Figure 14-12 *Single-Rate Two-Color Marker/Policer*

Single Rate Three-Color Markers/Policers

Single-rate three-color policer algorithms are based on RFC 2697.

This type of policer uses two token buckets, and the traffic can be classified as either conforming to, exceeding, or violating the CIR. Marking down or dropping actions are performed for each of the three states of traffic.

The first token bucket operates very similarly to the single-rate two-color system; with a few differences:

- If there are any tokens left over in the bucket after each time period due to low or no activity, instead of discarding the excess tokens (overflow), the algorithm places them in a second bucket to be used later for temporary bursts that might exceed the CIR.
- Tokens placed in this second bucket are referred to as the *excess burst (Be)*, and Be is the maximum number of bits that can exceed the Bc burst size.

Single Rate Three-Color Markers/Policers (Cont.)

Traffic can be classified in three colors or states, as follows:

- **Conform -** Traffic under Bc is classified as conforming and green. Conforming traffic is usually transmitted and can be optionally re-marked.
- **Exceed -** Traffic over Bc but under Be is classified as exceeding and yellow. Exceeding traffic can be dropped or marked down and transmitted.
- **Violate -** Traffic over Be is classified as violating and red. This type of traffic is usually dropped but can be optionally marked down and transmitted.

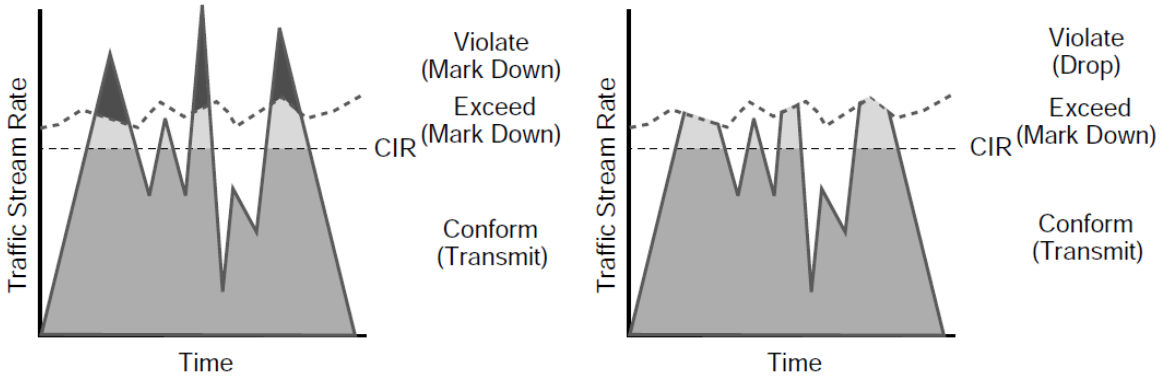


Figure 14-13 *Single-Rate Three-Color Marker/Policer*

Single Rate Three-Color Markers/Policers (Cont.)

- Figure 14-13 illustrates different actions that a single-rate three-color policer can take.
- The section below the straight dotted line on the left side of the figure represents the traffic that conformed to the CIR, the section right above the straight dotted line represents the exceeding traffic that was marked down, and the top section represents the violating traffic that was also marked down.
- The exceeding and violating traffic rates vary because they rely on random tokens spilling over from the Bc bucket into the Be.
- The section right above the straight dotted line on the right side of the figure represents traffic that exceeded the CIR and was marked down and the top section represents traffic that violated the CIR and was dropped.

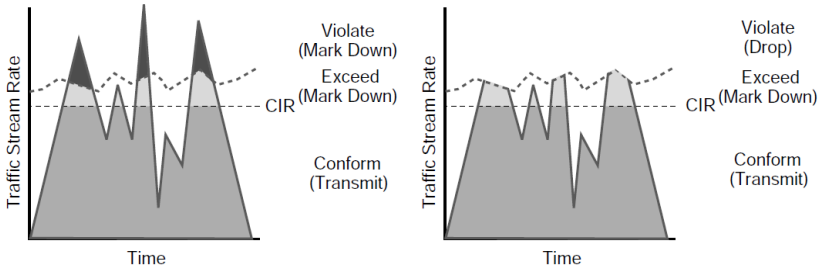


Figure 14-13 *Single-Rate Three-Color Marker/Policer*

Single Rate Three-Color Markers/Policers Parameters

The single-rate three-color marker/policer uses the following parameters to meter the traffic stream:

- **Committed Information Rate (CIR) -** The policed rate.
- **Committed Burst Size (Bc) -** The maximum size of the CIR token bucket, measured in bytes. Referred to as *Committed Burst Size (CBS)* in RFC 2697.
- **Excess Burst Size (Be) -** The maximum size of the excess token bucket, measured in bytes. Referred to as *Excess Burst Size (EBS)* in RFC 2697.
- **Bc Bucket Token Count (Tc) -** The number of tokens in the Bc bucket. Not to be confused with the committed time interval Tc.
- **Be Bucket Token Count (Te) -** The number of tokens in the Be bucket.
- **Incoming Packet Length (B) -** The packet length of the incoming packet, in bits.

Single Rate Three-Color Marker Uses

- The single-rate three-color policer's two bucket algorithm causes fewer TCP retransmissions and is more efficient for bandwidth utilization.
- It is the perfect policer to be used with AF classes (AFx1, AFx2, and AFx3).
- Using a three-color policer makes sense only if the actions taken for each color differ.
- If the actions for two or more colors are the same, for example, conform and exceed both transmit without re-marking, the single-rate two-color policer is recommended to keep things simpler.

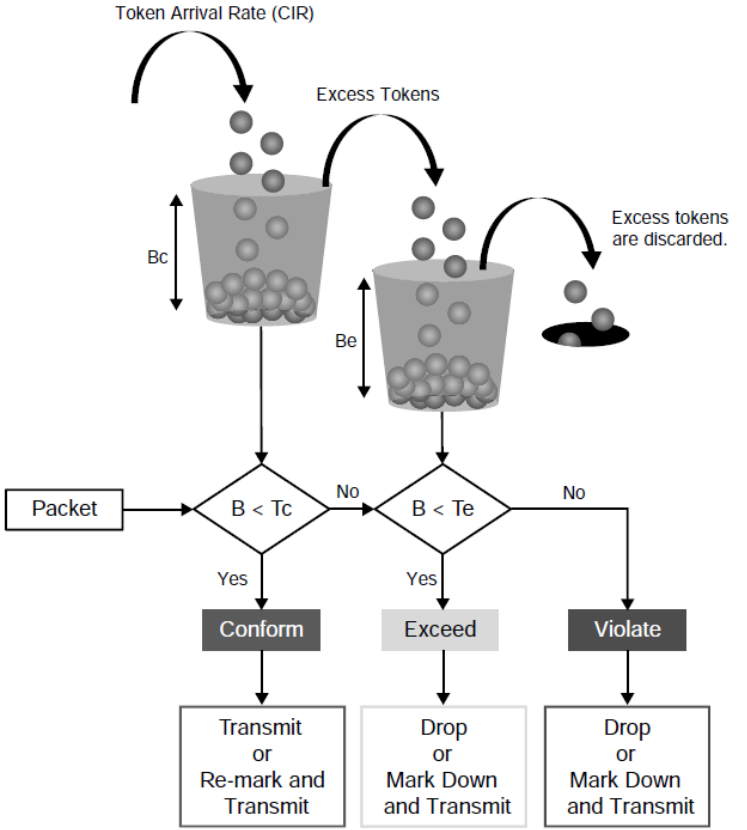


Figure 14-14 *Single-Rate Three-Color Marker/Policer Token Bucket Algorithm*

Two Rate Three-Color Markers/Policers

- The two-rate three-color marker/policer is based on RFC 2698 and is similar to the single-rate three-color policer.
- The difference is that single-rate three-color policers rely on excess tokens from the Bc bucket, which introduces a certain level of variability and unpredictability in traffic flows.
- The two-rate three-color marker/policers address this issue by using two distinct rates:
  - the CIR
  - the Peak Information Rate (PIR)
- The two-rate three-color marker/policer allows for a sustained excess rate based on the PIR that allows for different actions for the traffic exceeding the different burst values. For example, violating traffic can be dropped at a defined rate, and this is something that is not possible with the single-rate three-color policer.

Two Rate Three-Color Markers/Policers (Cont.)

- Figure 14-15 illustrates how violating traffic that exceeds the PIR can either be marked down (on the left side of the figure) or dropped (on the right side of the figure).
- Compare Figure 14-15 to Figure 14-13 to see the difference between the two-rate three-color policer and the single-rate three-color policer.

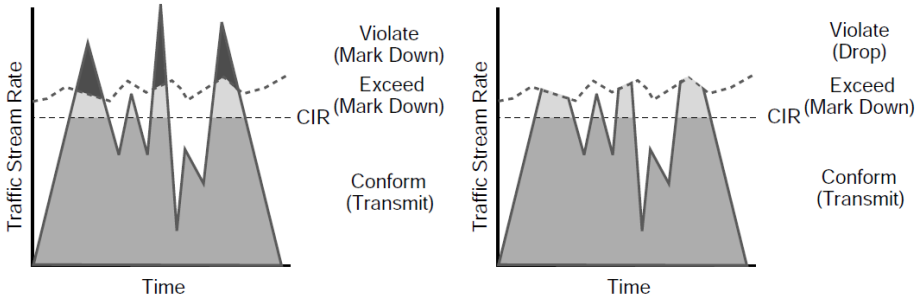


Figure 14-13 *Single-Rate Three-Color Marker/Policer*

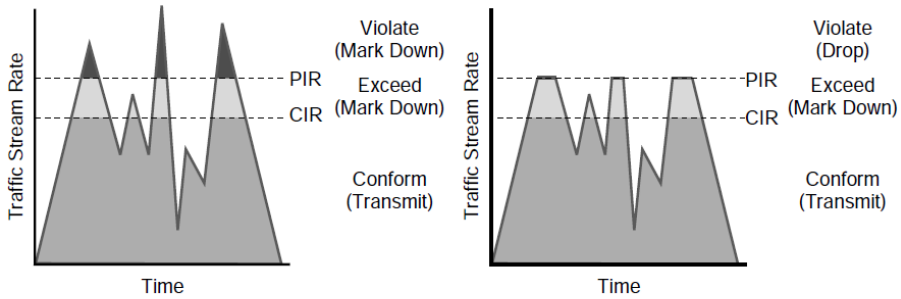


Figure 14-15 *Two-Rate Three-Color Marker/Policer Token Bucket Algorithm*

Two Rate Three-Color Markers/Policers Parameters

The two-rate three-color marker/policer uses the following parameters to meter the traffic stream:

- **Committed Information Rate (CIR) -** The policed rate.
- **Peak Information Rate (PIR) -** The maximum rate of traffic allowed. PIR should be equal to or greater than the CIR.
- **Committed Burst Size (Bc) -** The maximum size of the second token bucket, measured in bytes. Referred to as *Committed Burst Size (CBS)* in RFC 2698.
- **Peak Burst Size (Be) -** The maximum size of the PIR token bucket, measured in bytes. Referred to as *Peak Burst Size (PBS)* in RFC 2698. Be should be equal to or greater than Bc.
- **Bc Bucket Token Count (Tc) -** The number of tokens in the Bc bucket. Not to be confused with the committed time interval Tc.
- **Bp Bucket Token Count (Tp) -** The number of tokens in the Bp bucket.
- **Incoming Packet Length (B) -** The packet length of the incoming packet, in bits.

Two Rate Three-Color Markers/Policers

- The two-rate three-color policer also uses two token buckets.
- Instead of transferring unused tokens from the Bc bucket to the Be bucket, this policer has two separate buckets that are filled with two separate token rates.
- The Be bucket is filled with the PIR tokens, and the Bc bucket is filled with the CIR tokens. In this model, the Be represents the peak limit of traffic that can be sent during a subsecond interval.
- The logic varies further in that the initial check is to see whether the traffic is within the PIR. Only then is the traffic compared against the CIR. In other words, a violate condition is checked first, then an exceed condition, and finally a conform condition, which is the reverse of the logic of the single-rate three-color policer.

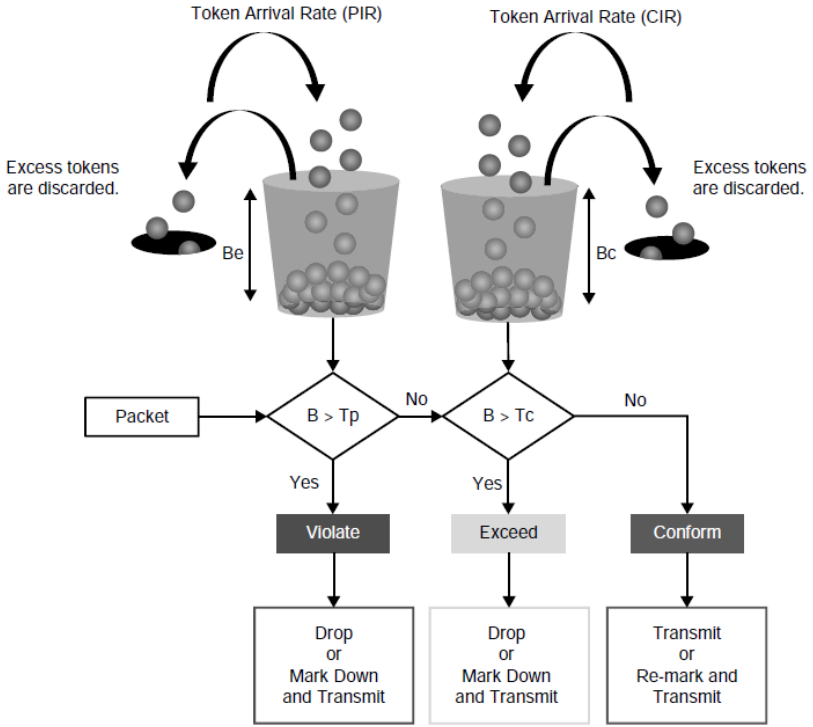


Figure 14-16 *Two-Rate Three-Color Marker/Policer Token Bucket Algorithm*

Two Rate Three-Color Markers/Policers (Cont.)

- Figure 14-16 illustrates the token bucket algorithm for the two-rate three-color marker/policer.
- Compare it to the token bucket algorithm of the single-rate three-color marker/policer in Figure 14-14 to see the differences between the two.

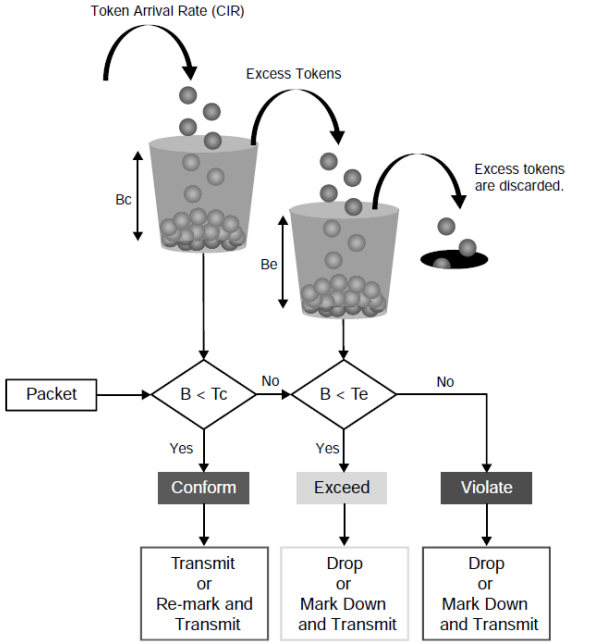


Figure 14-14 *Single-Rate Three-Color Marker/Policer Token Bucket Algorithm*

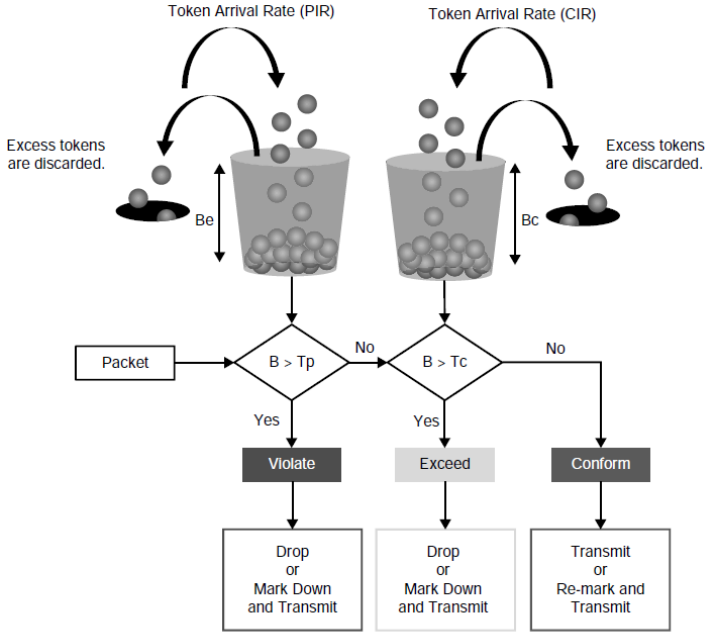


Figure 14-16 *Two-Rate Three-Color Marker/Policer Token Bucket Algorithm*

Congestion Management and Avoidance

- This section explores the queuing algorithms used for congestion management as well as packet drop techniques that can be used for congestion avoidance.
- These tools provide a way of managing excessive traffic during periods of congestion.

Congestion Management

Congestion management involves a combination of queuing and scheduling.

- Queuing (also known as buffering) is the temporary storage of excess packets.
- Queuing is activated when an output interface is experiencing congestion and deactivated when congestion clears.
  - Congestion is detected by the queuing algorithm when a Layer 1 hardware queue present on physical interfaces, known as the transmit ring (Tx-ring or TxQ), is full.
  - When the Tx-ring is not full anymore, this indicates that there is no congestion on the interface, and queueing is deactivated.
- Congestion can occur for one of these two reasons:
  - The input interface is faster than the output interface.
  - The output interface is receiving packets from multiple input interfaces.

Congestion Management: Legacy Queuing

- When congestion is taking place, the queues fill up, and packets can be reordered by some of the queuing algorithms so that higher-priority packets exit sooner than lower-priority ones.
- A scheduling algorithm decides which packet to transmit next. Scheduling is always active, regardless of whether the interface is experiencing congestion.
- There are many queuing algorithms available, but most of them are not adequate for modern rich-media networks. The legacy queuing algorithms that predate the MQC architecture include the following:

Legacy Queuing		
First-in, first-out queuing (FIFO)	Weighted round robin (WRR)	Priority queuing (PQ)
Round robin	Custom queuing (CQ)	Weighted fair queuing (WFQ)

Congestion Management: Current Queuing

The current queuing algorithms recommended for rich-media networks (and supported by MQC) combine the best features of the legacy algorithms. These algorithms provide real-time, delay-sensitive traffic bandwidth and delay guarantees while not starving other types of traffic. The recommended queuing algorithms include the following:

Current Queuing	
Class-based weighted fair queuing (CBWFQ)	CBWFQ enables the creation of up to 256 queues, serving up to 256 traffic classes. Each queue is serviced based on the bandwidth assigned to that class.
Low-latency queuing (LLQ)	LLQ is CBWFQ combined with priority queueing (PQ) and it was developed to meet the requirements of real-time traffic, such as voice.

CBWFQ with LLQ

- CBWFQ in combination with LLQ create queues into which traffic classes are classified.
- The CBWFQ queues are scheduled with a CBWFQ scheduler that guarantees bandwidth to each class. LLQ creates a high-priority queue that is always serviced first.
- During times of congestion, LLQ priority classes are policed to prevent the PQ from starving the CBWFQ non-priority classes (as legacy PQ does).
- When LLQ is configured, the policing rate must be specified as either a fixed amount of bandwidth or as a percentage of the interface bandwidth.
- LLQ allows for two different traffic classes to be assigned to it so that different policing rates can be applied to different types of high-priority traffic. For example, voice traffic could be policed during times of congestion to 10 Mbps, while video could be policed to 100 Mbps. This would not be possible with only one traffic class and a single policer.

CBWFQ with LLQ (Cont.)

Figure 14-17 illustrates the architecture of CBWFQ in combination with LLQ.

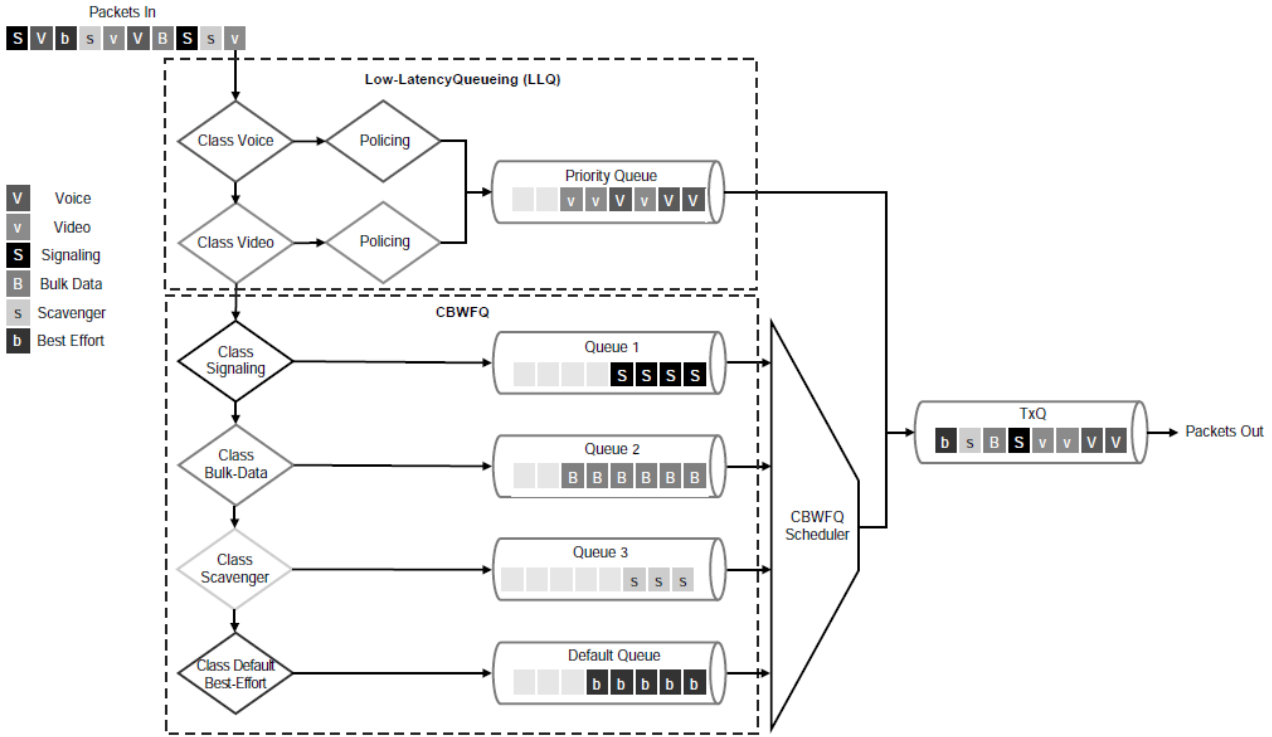


Figure 14-17 *CBWFQ with LLQ*

Congestion Avoidance Tools: RED

Congestion-avoidance techniques monitor network traffic loads to anticipate and avoid congestion by dropping packets.

The default packet dropping mechanism is tail drop.

- Tail drop treats all traffic equally and does not differentiate between classes of service. When the output queue buffers are full, all packets trying to enter the queue are dropped, regardless of their priority.
- Tail drop should be avoided for TCP traffic because it can cause TCP global synchronization, which results in significant link underutilization.

A better approach is to use a mechanism known as *random early detection (RED)*.

- RED provides congestion avoidance by randomly dropping packets before the queue buffers are full.
- Randomly dropping packets instead of dropping them all at once, as with tail drop, avoids global synchronization of TCP streams.
- RED monitors the buffer depth and performs early drops on random packets when the minimum defined queue threshold is exceeded.

Congestion Avoidance Tools: WRED

- The Cisco implementation of RED is known as Weighted RED (WRED).
- The difference between RED and WRED is that the randomness of packet drops can be manipulated by traffic weights denoted by either IP Precedence (IPP) or DSCP.
  - Packets with a lower IPP value are dropped more aggressively than are higher IPP values.
  - For example, IPP 3 would be dropped more aggressively than IPP 5 or DSCP, AFx3 would be dropped more aggressively than AFx2, and AFx2 would be dropped more aggressively than AFx1.
- WRED can also be used to set the IP Explicit Congestion Notification (ECN) bits to indicate that congestion was experienced in transit. ECN is an extension to WRED that allows for signaling to be sent to ECN-enabled endpoints, instructing them to reduce their packet transmission rates.

Prepare for the Exam

Prepare for the Exam

Key Topics for Chapter 14

Description	
QoS models	Marking traffic descriptors
Integrated Services (IntServ)	802.1Q/p
Differentiated Services (DiffServ)	802.1Q Tag Control Information (TCI) field
Classification	Priority Code Point (PCP) field
Classification traffic descriptors	Type of Service (ToS) field
Next Generation Network Based Application Recognition (NBAR2)	Differentiated Service Code Point (DSCP) field
Marking	

Prepare for the Exam

Key Topics for Chapter 14 (Cont.)

Description	
Per-hop behavior (PHB) definition	Token bucket algorithm key definitions
Available PHBs	Policing Algorithms
Trust boundary	Legacy queuing algorithms
Policing and shaping definition	Current queuing algorithms
Markdown	Weighted Random Early Detection (WRED)

Key Terms for Chapter 14

Key Terms
802.1Q
802.1p
Differentiated Services (DiffServ)
Differentiated Services Code Point (DSCP)
per-hop behavior (PHB)
Type of Service (TOS)


CISCO