CHAPTER 30

Configuring QoS

This chapter describes how to configure quality of service (QoS) by using automatic QoS (auto-QoS) commands or by using standard QoS commands. With QoS, you can give preferential treatment to certain types of traffic at the expense of others. Without QoS, the Catalyst 3550 switch offers best-effort service to each packet, regardless of the packet contents or size. It sends the packets without any assurance of reliability, delay bounds, or throughput.

Note For complete syntax and usage information for the commands used in this chapter, see the command reference for this release.

This chapter consists of these sections:

- Understanding QoS, page 30-2
- Configuring Auto-QoS, page 30-17
- Displaying Auto-QoS Information, page 30-23
- Auto-QoS Configuration Example, page 30-24
- Configuring Standard QoS, page 30-26
- Displaying Standard QoS Information, page 30-71
- Standard QoS Configuration Examples, page 30-71

Note When you are configuring QoS parameters for the switch, in order to allocate system resources to maximize the number of possible QoS access control entries (ACEs) allowed, you can use the **sdm prefer access** global configuration command to set the Switch Database Management feature to the access template. For more information on the SDM templates, see the "Optimizing System Resources for User-Selected Features" section on page 6-26.

The switch supports some of the modular QoS CLI (MQC) commands. For more information about the MQC commands, see the "Modular Quality of Service Command Line Interface Overview" at this URL:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios122/122cgcr/fqos_c/fqcprt8/qcfmdcli.htm#89799

Understanding QoS

Typically, networks operate on a best-effort delivery basis, which means that all traffic has equal priority and an equal chance of being delivered in a timely manner. When congestion occurs, all traffic has an equal chance of being dropped.

When you configure QoS, you can select specific network traffic, prioritize it according to its relative importance, and use congestion-management and congestion-avoidance techniques to give preferential treatment. Implementing QoS in your network makes network performance more predictable and bandwidth utilization more effective.

The QoS implementation is based on the DiffServ architecture, an emerging standard from the Internet Engineering Task Force (IETF). This architecture specifies that each packet is classified upon entry into the network. The classification is carried in the IP packet header, using 6 bits from the deprecated IP type of service (ToS) field to carry the classification (*class*) information. Classification can also be carried in the Layer 2 frame. These special bits in the Layer 2 frame or in the Layer 3 packet are described here and shown in Figure 30-1:

- Prioritization bits in Layer 2 frames:

  Layer 2 Inter-Switch Link (ISL) frame headers have a 1-byte User field that carries an IEEE 802.1p class of service (CoS) value in the three least-significant bits. On interfaces configured as Layer 2 ISL trunks, all traffic is in ISL frames.

  Layer 2 IEEE 802.1Q frame headers have a 2-byte Tag Control Information field that carries the CoS value in the three most-significant bits, which are called the User Priority bits. On interfaces configured as Layer 2 IEEE 802.1Q trunks, all traffic is in IEEE 802.1Q frames except for traffic in the native VLAN.

  Other frame types cannot carry Layer 2 CoS values.

  Layer 2 CoS values range from 0 for low priority to 7 for high priority.

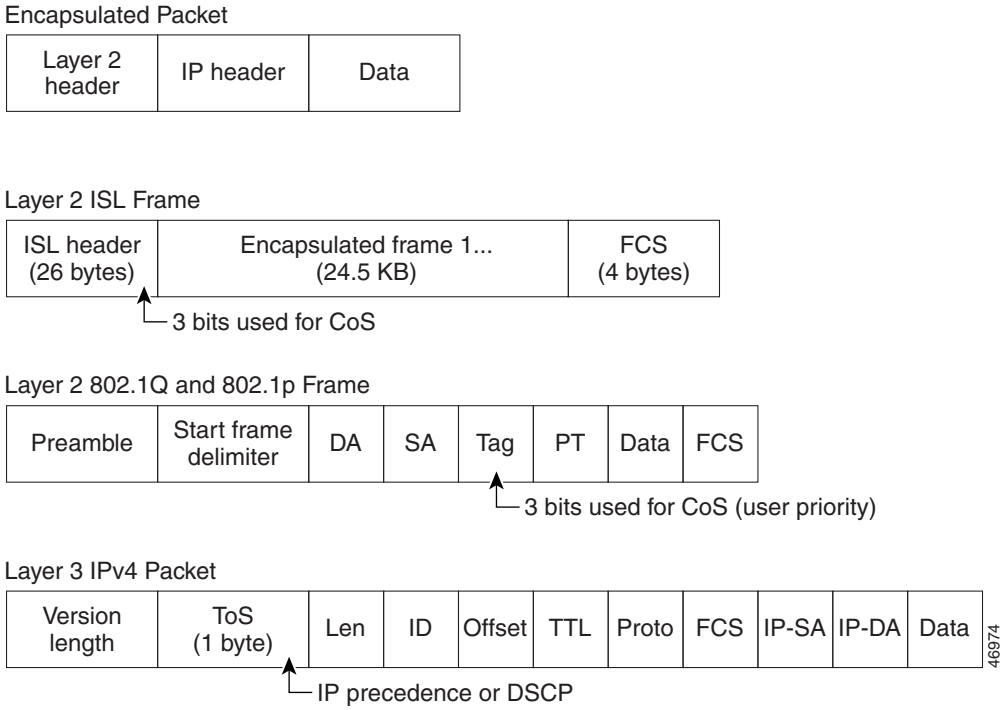
- Prioritization bits in Layer 3 packets:

  Layer 3 IP packets can carry either an IP precedence value or a Differentiated Services Code Point (DSCP) value. QoS supports the use of either value because DSCP values are backward-compatible with IP precedence values.

  IP precedence values range from 0 to 7.

  DSCP values range from 0 to 63.

Figure 30-1 QoS Classification Bits in Frames and Packets



Note Layer 3 IPv6 packets are treated as non-IP packets and are bridged by the switch.

To give the same forwarding treatment to packets with the same class information and different treatment to packets with different class information, all switches and routers that access the Internet rely on class information. Class information in the packet can be assigned by end hosts or by switches or routers along the way, based on a configured policy, detailed examination of the packet, or both. Detailed examination of the packet is expected to happen closer to the network edge so that core switches and routers are not overloaded.

Switches and routers along the path can use class information to limit the amount of resources allocated per traffic class. The behavior of an individual device when handling traffic in the DiffServ architecture is called per-hop behavior. If all devices along a path have a consistent per-hop behavior, you can construct an end-to-end QoS solution.

Implementing QoS in your network can be a simple or complex task and depends on the QoS features offered by your internetworking devices, the traffic types and patterns in your network, and the granularity of control that you need over incoming and outgoing traffic.

These sections describe the QoS stages and how they work:

- Basic QoS Model, page 30-4
- Classification, page 30-5
- Policing and Marking, page 30-8
- Mapping Tables, page 30-10
- Queueing and Scheduling, page 30-11
- Packet Modification, page 30-17

Basic QoS Model

Figure 30-2 shows the basic QoS model. Actions at the ingress interface include classifying traffic, policing, and marking:

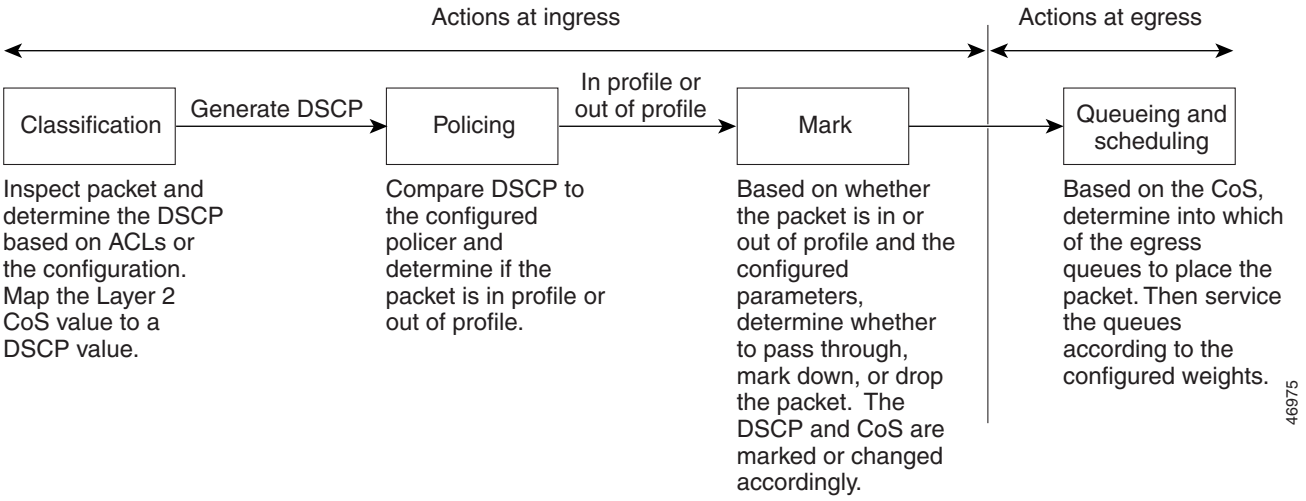
- Classifying distinguishes one kind of traffic from another. The process generates an internal DSCP for a packet, which identifies all the future QoS actions to be performed on this packet. For more information, see the "Classification" section on page 30-5.
- Policing determines whether a packet is in or out of profile by comparing the internal DSCP to the configured policer. The policer limits the bandwidth consumed by a flow of traffic. The result of this determination is passed to the marker. For more information, see the "Policing and Marking" section on page 30-8.
- Marking evaluates the policer and the configuration information for the action to be taken when a packet is out of profile and decides what to do with the packet (pass through a packet without modification, mark down the DSCP value in the packet, or drop the packet). For more information, see the "Policing and Marking" section on page 30-8.

Actions at the egress interface include queueing and scheduling:

- Queueing evaluates the internal DSCP and determines which of the four egress queues in which to place the packet. The DSCP value is mapped to a CoS value, which selects one of the queues. For more information, see the "Mapping Tables" section on page 30-10.
- Scheduling services the four egress queues based on their configured weighted round robin (WRR) weights and thresholds. One of the queues can be the expedite queue, which is serviced until empty before the other queues are serviced. Congestion avoidance techniques include tail drop and Weighted Random Early Detection (WRED) on Gigabit-capable Ethernet ports and tail drop (with only one threshold) on 10/100 Ethernet ports. For more information, see the "Queueing and Scheduling" section on page 30-11.

Note Policing and marking also can occur on egress interfaces.

Figure 30-2 Basic QoS Model



Classification

Classification is the process of distinguishing one kind of traffic from another by examining the fields in the packet. Classification is enabled only if QoS is globally enabled on the switch. By default, QoS is globally disabled, so no classification occurs.

Note Classification occurs on a physical interface or on a per-port per-VLAN basis. No support exists for classifying packets at the switch virtual interface level.

You specify which fields in the frame or packet that you want to use to classify incoming traffic.

For non-IP traffic, these are the classification options as shown in Figure 30-3:

- Use the port default. If the frame does not contain a CoS value, the switch assigns the default port CoS value to the incoming frame. Then, the switch uses the configurable CoS-to-DSCP map to generate the internal DSCP value. The switch uses the internal DSCP value to generate a CoS value representing the priority of the traffic.
- Trust the CoS value in the incoming frame (configure the port to trust CoS). Then, the switch uses the configurable CoS-to-DSCP map to generate the internal DSCP value. Layer 2 ISL frame headers carry the CoS value in the three least-significant bits of the 1-byte User field. Layer 2 IEEE 802.1Q frame headers carry the CoS value in the three most-significant bits of the Tag Control Information field. CoS values range from 0 for low priority to 7 for high priority.
- The trust DSCP and trust IP precedence configurations are meaningless for non-IP traffic. If you configure a port with either of these options and non-IP traffic is received, the switch assigns the default port CoS value and generates the internal DSCP from the CoS-to-DSCP map.
- Perform the classification based on the configured Layer 2 MAC access control list (ACL), which can examine the MAC source address, the MAC destination address, and the Ethertype field. If no ACL is configured, the packet is assigned the default DSCP of 0, which means best-effort traffic; otherwise, the policy map specifies the DSCP to assign to the incoming frame.

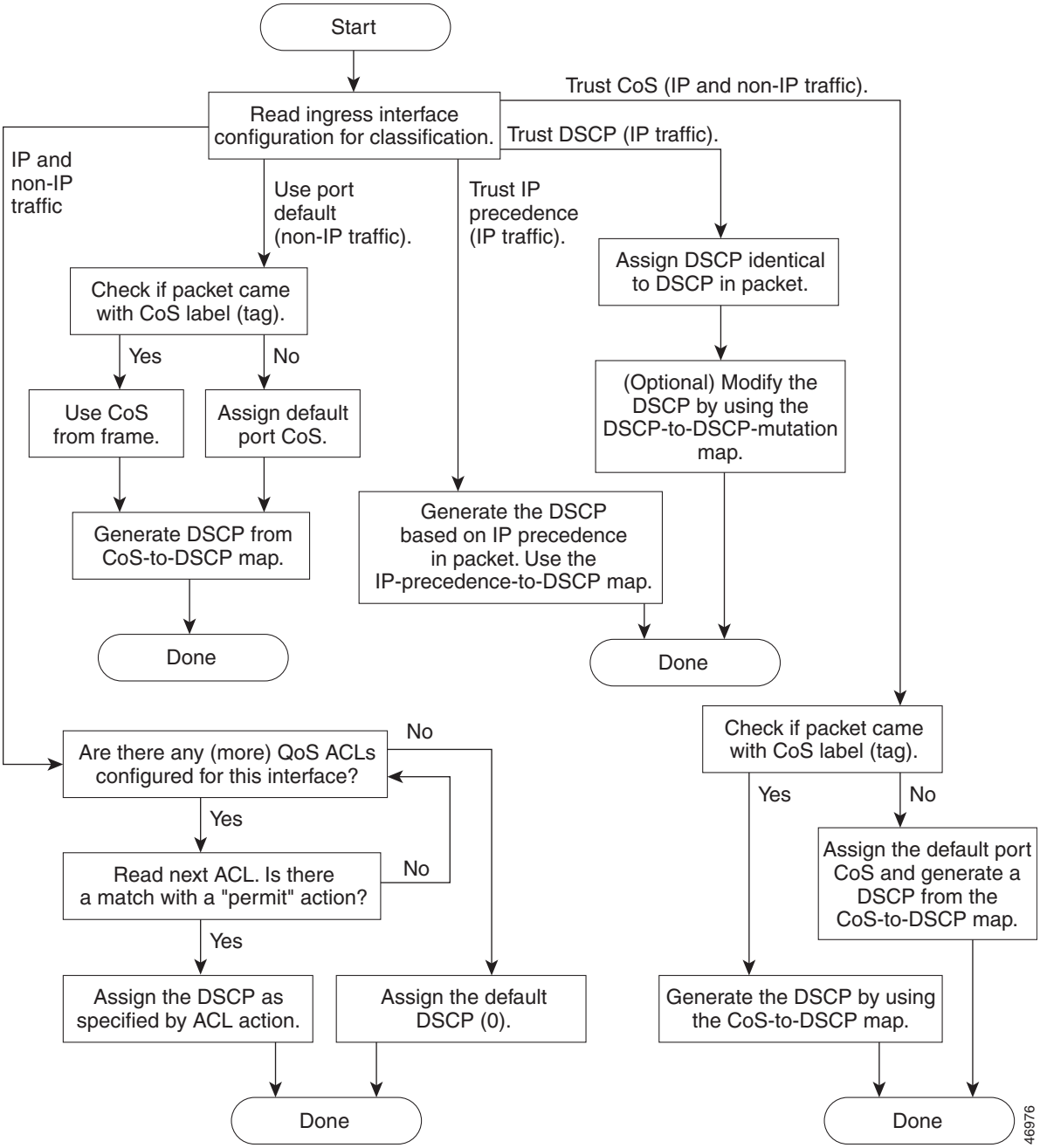
For IP traffic, these are the classification options as shown in Figure 30-3:

- Trust the IP DSCP in the incoming packet (configure the port to trust DSCP), and assign the same DSCP to the packet for internal use. The IETF defines the 6 most-significant bits of the 1-byte ToS field as the DSCP. The priority represented by a particular DSCP value is configurable. DSCP values range from 0 to 63.

  For ports that are on the boundary between two QoS administrative domains, you can modify the DSCP to another value by using the configurable DSCP-to-DSCP-mutation map.
- Trust the IP precedence in the incoming packet (configure the port to trust IP precedence), and generate a DSCP by using the configurable IP-precedence-to-DSCP map. The IP Version 4 specification defines the three most-significant bits of the 1-byte ToS field as the IP precedence. IP precedence values range from 0 for low priority to 7 for high priority.
- Trust the CoS value (if present) in the incoming packet, and generate the DSCP by using the CoS-to-DSCP map.
- Perform the classification based on a configured IP standard or an extended ACL, which examines various fields in the IP header. If no ACL is configured, the packet is assigned the default DSCP of 0, which means best-effort traffic; otherwise, the policy map specifies the DSCP to assign to the incoming frame.

For information on the maps described in this section, see the "Mapping Tables" section on page 30-10. For configuration information on port trust states, see the "Configuring Classification By Using Port Trust States" section on page 30-30.

Figure 30-3 Classification Flowchart



Classification Based on QoS ACLs

You can use IP standard, IP extended, and Layer 2 MAC ACLs to define a group of packets with the same characteristics (*class*). In the QoS context, the permit and deny actions in the access control entries (ACEs) have different meanings than with security ACLs:

- If a match with a permit action is encountered (first-match principle), the specified QoS-related action is taken.
- If a match with a deny action is encountered, the ACL being processed is skipped, and the next ACL is processed.
- If no match with a permit action is encountered and all the ACEs have been examined, no QoS processing occurs on the packet, and the switch offers best-effort service to the packet.
- If multiple ACLs are configured on an interface, the lookup stops after the packet matches the first ACL with a permit action, and QoS processing begins.

Note When creating an access list, remember that, by default, the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.

After a traffic class has been defined with the ACL, you can attach a policy to it. A policy might contain multiple classes with actions specified for each one of them. A policy might include commands to classify the class as a particular aggregate (for example, assign a DSCP) or rate-limit the class. This policy is then attached to a particular port on which it becomes effective.

You implement IP ACLs to classify IP traffic by using the **access-list** global configuration command; you implement Layer 2 MAC ACLs to classify non-IP traffic by using the **mac access-list extended** global configuration command. For configuration information, see the "Configuring a QoS Policy" section on page 30-37.

Classification Based on Class Maps and Policy Maps

A class map is a mechanism that you use to name and to isolate a specific traffic flow (or class) from all other traffic. The class map defines the criteria used to match against a specific traffic flow to further classify it; the criteria can include matching the access group defined by the ACL, matching a specific list of DSCP or IP precedence values, or matching a specific list of VLAN IDs associated with another class map that defines the actual criteria (for example, to match a standard or extended ACL). If you have more than one type of traffic that you want to classify, you can create another class map and use a different name. After a packet is matched against the class-map criteria, you further classify it through the use of a policy map.

A policy map specifies which traffic class to act on. Actions can include trusting the CoS, DSCP, or IP precedence values in the traffic class; setting a specific DSCP or IP precedence value in the traffic class; or specifying the traffic bandwidth limitations and the action to take when the traffic is out of profile. Before a policy map can be effective, you must attach it to an interface.

You create a class map by using the **class-map** global configuration command or the **class** policy-map configuration command; you should use the **class-map** command when the map is shared among many ports. When you enter the **class-map** command, the switch enters the class-map configuration mode. In this mode, you define the match criterion for the traffic by using the **match** class-map configuration command.

You create and name a policy map by using the **policy-map** global configuration command. When you enter this command, the switch enters the policy-map configuration mode. In this mode, you specify the actions to take on a specific traffic class by using the **class**, **trust**, or **set** policy-map configuration and policy-map class configuration commands. To make the policy map effective, you attach it to an interface by using the **service-policy** interface configuration command.

The policy map also can contain commands that define the policer, the bandwidth limitations of the traffic, and the action to take if the limits are exceeded. For more information, see the "Policing and Marking" section on page 30-8.

A policy map has these characteristics:

- A policy map can contain multiple class statements.
- A separate policy-map class can exist for each type of traffic received through an interface.
- The policy-map trust state and an interface trust state are mutually exclusive, and whichever is configured last takes affect.

For configuration information, see the "Configuring a QoS Policy" section on page 30-37.

Policing and Marking

After a packet is classified and has an internal DSCP value assigned to it, the policing and marking process can begin as shown in Figure 30-4.

Policing involves creating a policer that specifies the bandwidth limits for the traffic. Packets that exceed the limits are *out of profile* or *nonconforming*. Each policer specifies the action to take for packets that are in or out of profile. These actions, carried out by the marker, include passing through the packet without modification, dropping the packet, or marking down the packet with a new DSCP value that is obtained from the configurable policed-DSCP map. For information on the policed-DSCP map, see the "Mapping Tables" section on page 30-10.

You can create these types of policers:

- Individual

  QoS applies the bandwidth limits specified in the policer separately to each matched traffic class. You configure this type of policer within a policy map by using the **police** policy-map configuration command.

- Aggregate

  QoS applies the bandwidth limits specified in an aggregate policer cumulatively to all matched traffic flows. You configure this type of policer by specifying the aggregate policer name within a policy map by using the **police aggregate** policy-map configuration command. You specify the bandwidth limits of the policer by using the **mls qos aggregate-policer** global configuration command. In this way, the aggregate policer is shared by multiple classes of traffic within a policy map.

Policing uses a token bucket algorithm. As each frame is received by the switch, a token is added to the bucket. The bucket has a hole in it and leaks at a rate that you specify as the average traffic rate in bits per second. Each time a token is added to the bucket, the switch performs a check to determine if there is enough room in the bucket. If there is not enough room, the packet is marked as nonconforming, and the specified policer action is taken (dropped or marked down).

How quickly the bucket fills is a function of the bucket depth (burst-byte), the rate at which the tokens are removed (rate-bps), and the duration of the burst above the average rate. The size of the bucket imposes an upper limit on the burst length and determines the number of frames that can be sent back-to-back. If the burst is short, the bucket does not overflow, and no action is taken against the traffic flow. However, if a burst is long and at a higher rate, the bucket overflows, and the policing actions are taken against the frames in that burst.

You configure the bucket depth (the maximum burst that is tolerated before the bucket overflows) by using the *burst-byte* option of the **police** policy-map class configuration command or the **mls qos aggregate-policer** global configuration command. You configure how fast (the average rate) that the tokens are removed from the bucket by using the *rate-bps* option of the **police** policy-map class configuration command or the **mls qos aggregate-policer** global configuration command.

When configuring policing and policers, keep these items in mind:

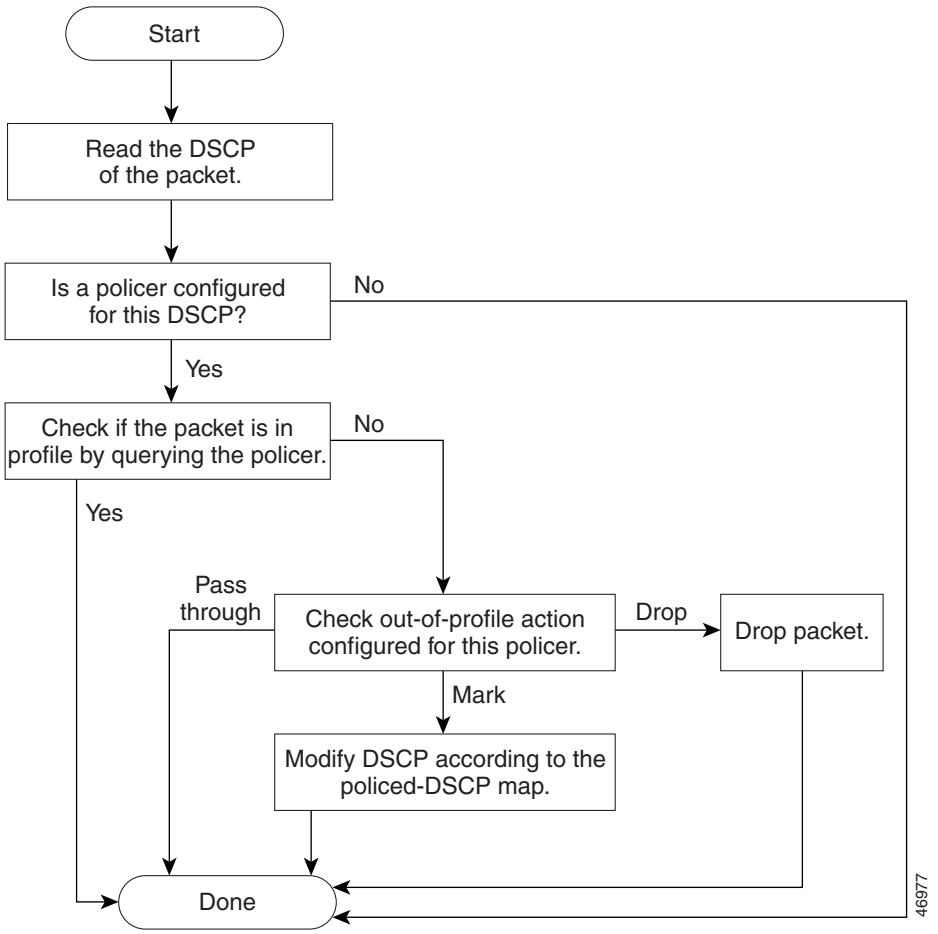
- By default, no policers are configured.
- Policers can be configured only on a physical port or on a per-port per-VLAN basis (specifies the bandwidth limits for the traffic on a per-VLAN basis, for a given port). Per-port per-VLAN policing is not supported on routed ports or on virtual (logical) interfaces. It is supported only on an ingress port configured as a trunk or as a static-access port.
- Only one policer can be applied to a packet per direction.
- Only the average rate and committed burst parameters are configurable.
- Policing can occur on ingress and egress interfaces:

  Note Per-port per-VLAN policing is supported only on ingress interfaces.

  - 128 policers are supported on ingress Gigabit-capable Ethernet ports.
  - 8 policers are supported on ingress 10/100 Ethernet ports.
  - 8 policers are supported on all egress ports.
  - Ingress policers can be individual or aggregate.
- On an interface configured for QoS, all traffic received through the interface is classified, policed, and marked according to the policy map attached to the interface. On a trunk interface configured for QoS, traffic in *all* VLANs received through the interface is classified, policed, and marked according to the policy map attached to the interface.

After you configure the policy map and policing actions, attach the policy to an ingress or egress interface by using the **service-policy** interface configuration command. For configuration information, see the "Classifying, Policing, and Marking Traffic by Using Policy Maps" section on page 30-44 and the "Classifying, Policing, and Marking Traffic by Using Aggregate Policers" section on page 30-50.

Figure 30-4 Policing and Marking Flowchart



Mapping Tables

During QoS processing, the switch represents the priority of all traffic (including non-IP traffic) with an internal DSCP value:

- During classification, QoS uses configurable mapping tables to derive the internal DSCP (a 6-bit value) from received CoS or IP precedence (3-bit) values. These maps include the CoS-to-DSCP map and the IP-precedence-to-DSCP map.

  On an ingress interface configured in the DSCP-trusted state, if the DSCP values are different between the QoS domains, you can apply the configurable DSCP-to-DSCP-mutation map to the interface that is on the boundary between the two QoS domains.

- During policing, QoS can assign another DSCP value to an IP or non-IP packet (if the packet is out of profile and the policer specifies a marked down DSCP value). This configurable map is called the policed-DSCP map.

- Before the traffic reaches the scheduling stage, QoS uses the configurable DSCP-to-CoS map to derive a CoS value from the internal DSCP value. Through the CoS-to-egress-queue map, the CoS values select one of the four egress queues for output processing.

The CoS-to-DSCP, DSCP-to-CoS, and the IP-precedence-to-DSCP map have default values that might or might not be appropriate for your network.

The default DSCP-to-DSCP-mutation map and the default policed-DSCP map are null maps; they map an incoming DSCP value to the same DSCP value. The DSCP-to-DSCP-mutation map is the only map you apply to a specific Gigabit-capable Ethernet port or to a group of 10/100 Ethernet ports. All other maps apply to the entire switch.

For configuration information, see the "Configuring DSCP Maps" section on page 30-53.

Queueing and Scheduling

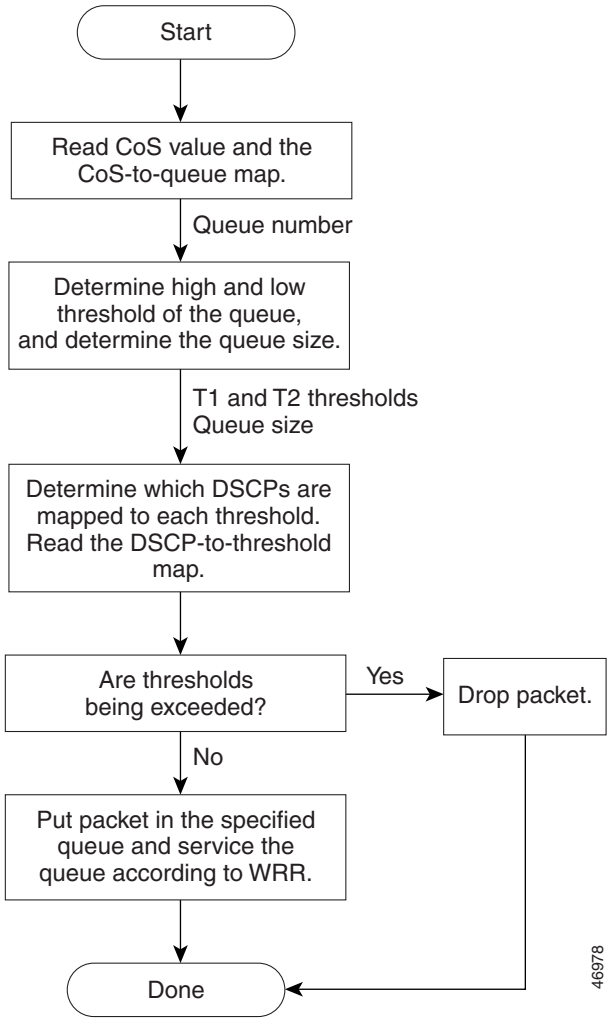
After a packet is policed and marked, the queueing and scheduling process begins as described in these sections:

- Queueing and Scheduling on Gigabit-Capable Ports, page 30-11
- Queueing and Scheduling on 10/100 Ethernet Ports, page 30-15

Queueing and Scheduling on Gigabit-Capable Ports

Figure 30-5 shows the queueing and scheduling flowchart for Gigabit-capable Ethernet ports.

Figure 30-5 Queueing and Scheduling Flowchart for Gigabit-Capable Ethernet Ports



Note If the expedite queue is enabled, WRR services it until it is empty before servicing the other three queues.

During the queueing and scheduling process, the switch uses egress queues and WRR for congestion management, and tail drop or WRED algorithms for congestion avoidance on Gigabit-capable Ethernet ports.

Each Gigabit-capable Ethernet port has four egress queues, one of which can be the egress expedite queue. You can configure the buffer space allocated to each queue as a ratio of weights by using the **wrr-queue queue-limit** interface configuration command, where the relative size differences in the numbers show the relative differences in the queue sizes. To display the absolute value of the queue size, use the **show mls qos interface** *interface-id* **statistics** privileged EXEC command, and examine the FreeQ information.

You assign two drop thresholds to each queue, map DSCPs to the thresholds through the DSCP-to-threshold map, and enable either tail drop or WRED on the interface. The queue size, drop thresholds, tail-drop or WRED algorithm, and the DSCP-to-threshold map work together to determine when and which packets are dropped when the thresholds are exceeded. You configure the drop percentage thresholds by using either the **wrr-queue threshold** interface configuration command for tail drop or the **wrr-queue random-detect max-threshold** interface configuration command for WRED; in either case, you map DSCP values to the thresholds (DSCP-to-threshold map) by using the **wrr-queue dscp-map** interface configuration command. For more information, see the "Tail Drop" section on page 30-13 and "WRED" section on page 30-14.

The available bandwidth of the egress link is divided among the queues. You configure the queues to be serviced according to the ratio of WRR weights by using the **wrr-queue bandwidth** interface configuration command. The ratio represents the importance (weight) of a queue relative to the other queues. WRR scheduling prevents low-priority queues from being completely neglected during periods of high-priority traffic by sending some packets from each queue in turn. The number of packets sent corresponds to the relative importance of the queue. For example, if one queue has a weight of 3 and another has a weight of 4, three packets are sent from the first queue for every four that are sent from the second queue. By using this scheduling, low-priority queues can send packets even though the high-priority queues are not empty. Queues are selected by the CoS value that is mapped to an egress queue (CoS-to-egress-queue map) through the **wrr-queue cos-map** interface configuration command.

All four queues participate in the WRR unless the expedite queue is enabled, in which case, the fourth bandwidth weight is ignored and not used in the ratio calculation. The expedite queue is a priority queue, and it is serviced until empty before the other queues are serviced. You enable the expedite queue by using the **priority-queue out** interface configuration command.

You can combine the commands described in this section to prioritize traffic by placing packets with particular DSCPs into certain queues, allocate a larger queue size or service the particular queue more frequently, and adjust queue thresholds so that packets with lower priorities are dropped. For configuration information, see the "Configuring Egress Queues on Gigabit-Capable Ethernet Ports" section on page 30-59.

Tail Drop

Tail drop is the default congestion-avoidance technique on Gigabit-capable Ethernet ports. With tail drop, packets are queued until the thresholds are exceeded. Specifically, all packets with DSCPs assigned to the first threshold are dropped until the threshold is no longer exceeded. However, packets assigned to the second threshold continue to be queued and sent as long as the second threshold is not exceeded.

You can modify the two tail-drop threshold percentages assigned to the four egress queues by using the **wrr-queue threshold** interface configuration command. Each threshold value is a percentage of the total number of allocated queue descriptors for the queue. The default threshold is 100 percent for thresholds 1 and 2.

You modify the DSCP-to-threshold map to determine which DSCPs are mapped to which threshold ID by using the **wrr-queue dscp-map** interface configuration command. By default, all DSCPs are mapped to threshold 1, and when this threshold is exceeded, all the packets are dropped.

If you use tail-drop thresholds, you cannot use WRED, and vice versa. If tail drop is disabled, WRED is automatically enabled with the previous configuration (or the default if it was not previously configured).

WRED

Cisco's implementation of Random Early Detection (RED), called Weighted Random Early Detection (WRED), differs from other congestion-avoidance techniques because it attempts to anticipate and avoid congestion, rather than controlling congestion when it occurs.

WRED takes advantage of the Transmission Control Protocol (TCP) congestion control to try to control the average queue size by indicating to end hosts when they should temporarily stop sending packets. By randomly dropping packets before periods of high congestion, it tells the packet source to decrease its transmission rate. Assuming the packet source is using TCP, WRED tells it to decrease its transmission rate until all the packets reach their destination, meaning that the congestion is cleared.

WRED reduces the chances of tail drop by selectively dropping packets when the output interface begins to show signs of congestion. By dropping some packets early rather than waiting until the queue is full, WRED avoids dropping large numbers of packets at once. Thus, WRED allows the transmission line to be fully used at all times. WRED also drops more packets from large users than small. Therefore, sources that generate the most traffic are more likely to be slowed down versus sources that generate little traffic.

You can enable WRED and configure the two threshold percentages assigned to the four egress queues on a Gigabit-capable Ethernet port by using the **wrr-queue random-detect max-threshold** interface configuration command. Each threshold percentage represents where WRED starts to randomly drop packets. After a threshold is exceeded, WRED randomly begins to drop packets assigned to this threshold. As the queue limit is approached, WRED continues to drop more and more packets. When the queue limit is reached, WRED drops all packets assigned to the threshold. By default, WRED is disabled.

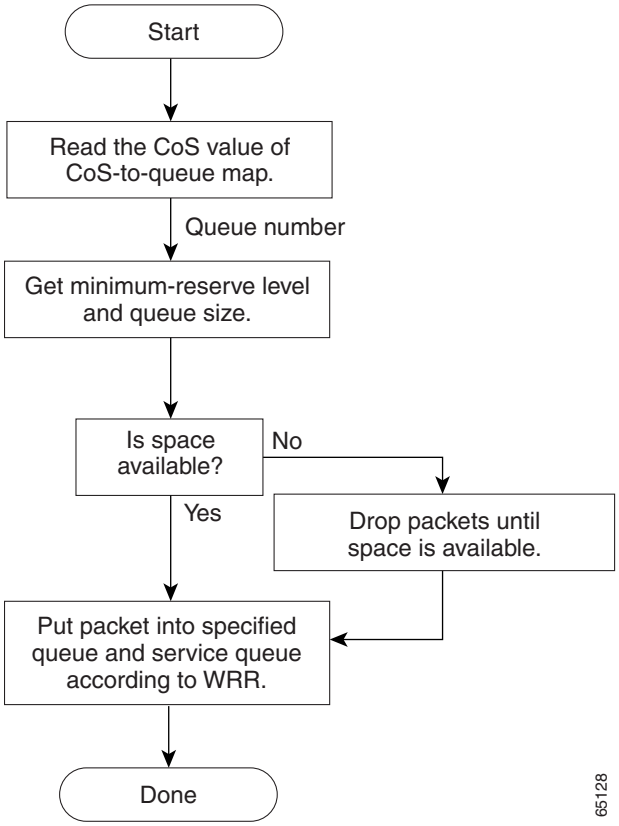
You modify the DSCP-to-threshold map to determine which DSCPs are mapped to which threshold ID by using the **wrr-queue dscp-map** interface configuration command. By default, all DSCPs are mapped to threshold 1, and when this threshold is exceeded, all the packets are randomly dropped.

If you use WRED thresholds, you cannot use tail drop, and vice versa. If WRED is disabled, tail drop is automatically enabled with the previous configuration (or the default if it was not previously configured).

Queueing and Scheduling on 10/100 Ethernet Ports

Figure 30-6 shows the queueing and scheduling flowchart for 10/100 Ethernet ports.

Figure 30-6 Queueing and Scheduling Flowchart for 10/100 Ethernet Ports



Note If the expedite queue is enabled, WRR services it until it is empty before servicing the other three queues.

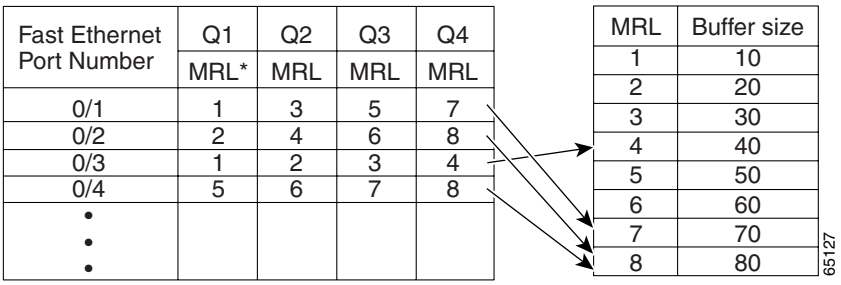
During the queueing and scheduling process, the switch uses egress queues (to select the minimum-reserve level and buffer size) and WRR for congestion management.

Each 10/100 Ethernet port has four egress queues, one of which can be the egress expedite queue. Each queue can access one of eight minimum-reserve levels; each level has 100 packets of buffer space by default for queueing packets. When the buffer specified for the minimum-reserve level is full, packets are dropped until space is available.

Figure 30-7 is an example of the 10/100 Ethernet port queue assignments, minimum-reserve levels, and buffer sizes. The figure shows four egress queues per port, with each queue assigned to a minimum-reserve level. For example, for Fast Ethernet port 0/1, queue 1 is assigned to minimum-reserve level 1, queue 2 is assigned to minimum-reserve level 3, queue 3 is assigned to minimum-reserve level 5, and queue 4 is assigned to minimum-reserve level 7. You assign the minimum-reserve level to a queue by using the **wrr-queue min-reserve** interface configuration command.

Each minimum-reserve level is configured with a buffer size. As shown in the figure, queue 4 of Fast Ethernet port 1 has a buffer size of 70 packets, queue 4 of Fast Ethernet port 2 has a buffer size of 80 packets, queue 4 of Fast Ethernet port 3 has a buffer size of 40 packets, and Fast Ethernet port 4 has a buffer size of 80 packets. You configure the buffer size by using the **mls qos min-reserve** global configuration command.

Figure 30-7 10/100 Ethernet Port Queue Assignment, Minimum-Reserve Levels, and Buffer Size



* MRL = Minimum-reserve level

The available bandwidth of the egress link is divided among the queues. You configure the queues to be serviced according to the ratio of WRR weights by using the **wrr-queue bandwidth** interface configuration command. The ratio represents the importance (weight) of a queue relative to the other queues. WRR scheduling prevents low-priority queues from being completely neglected during periods of high-priority traffic by sending some packets from each queue in turn. The number of packets sent corresponds to the relative importance of the queue. For example, if one queue has a weight of 3 and another has a weight of 4, three packets are sent from the first queue for every four that are sent from the second queue. By using this scheduling, low-priority queues can send packets even though the high-priority queues are not empty. Queues are selected by the CoS value that is mapped to an egress queue (CoS-to-egress-queue map) through the **wrr-queue cos-map** interface configuration command.

All four queues participate in the WRR unless the egress expedite queue is enabled, in which case, the fourth bandwidth weight is ignored and not used in the ratio calculation. The expedite queue is a priority queue, and it is serviced until empty before the other queues are serviced. You enable the expedite queue by using the **priority-queue out** interface configuration command.

You can combine the commands described in this section to prioritize traffic by placing packets with particular DSCPs into certain queues, allocate a larger minimum-reserve buffer size, and service a particular queue more frequently. For configuration information, see the "Configuring Egress Queues on 10/100 Ethernet Ports" section on page 30-66.

Packet Modification

A packet is classified, policed, and queued for QoS. Packet modifications can occur during this process:

- For IP packets, classification involves assigning a DSCP to the packet. However, the packet is not modified at this stage; only an indication of the assigned DSCP is carried along. The reason for this is that QoS classification and ACL lookup occur in parallel, and it is possible that the ACL specifies that the packet should be denied and logged. In this situation, the packet is forwarded with its original DSCP to the CPU, where it is again processed through ACL software. However, route lookup is performed based on classified DSCPs.
- For non-IP packets, classification involves assigning an internal DSCP to the packet, but because there is no DSCP in the non-IP packet, no overwrite occurs. Instead, the internal DSCP is translated to the CoS and is used both for queueing and scheduling decisions and for writing the CoS priority value in the tag if the packet is being sent on either an ISL or IEEE 802.1Q trunk port. Because the CoS priority is written in the tag, Catalyst 3500 series XL switches that use the IEEE 802.1p priority can interoperate with the QoS implementation on the Catalyst 3550 switches.
- During policing, IP and non-IP packets can have another DSCP assigned to them (if they are out of profile and the policer specifies a markdown DSCP). For IP packets, the packet modification occurs at a later stage; for non-IP packets the DSCP is converted to CoS and used for queueing and scheduling decisions.

Configuring Auto-QoS

You can use the auto-QoS feature to simplify the deployment of existing QoS features. Auto-QoS makes assumptions about the network design, and as a result, the switch can prioritize different traffic flows and appropriately use the egress queues instead of using the default QoS behavior. (The default is that QoS is disabled. The switch then offers best-effort service to each packet, regardless of the packet contents or size, and sends it from a single queue.)

When you enable auto-QoS, it automatically classifies traffic based on the traffic type and ingress packet label. The switch uses the resulting classification to choose the appropriate egress queue.

You use auto-QoS commands to identify ports connected to Cisco IP Phones and to devices running the Cisco SoftPhone application. You also use the commands to identify ports that receive trusted traffic through an uplink. Auto-QoS then performs these functions:

- Detects the presence or absence of Cisco IP Phones
- Configures QoS classification
- Configures egress queues

These sections describe how to configure auto-QoS on your switch:

- Generated Auto-QoS Configuration, page 30-18
- Effects of Auto-QoS on the Configuration, page 30-21
- Configuration Guidelines, page 30-21
- Upgrading from a Previous Software Release, page 30-22
- Enabling Auto-QoS for VoIP, page 30-22

Generated Auto-QoS Configuration

By default, auto-QoS is disabled on all interfaces.

When auto-QoS is enabled, it uses the ingress packet label to categorize traffic and to configure the egress queues as shown in Table 30-1.

Table 30-1 Traffic Types, Packet Labels, and Egress Queues

	VoIP[1] Data Traffic	VoIP Control Traffic	Routing Protocol Traffic	STP BPDU[2] Traffic	Real-Time Video Traffic	All Other Traffic	
DSCP	46	24, 26	48	56	34	–	
CoS	5	3	6	7	4		
CoS-to-Queue Map	5	3, 6, 7			4	2	0, 1
Egress Queue	Expedite (queue 4)	70% WRR (queue 3)			20% WRR (queue 2)	20% WRR (queue 2)	10% WRR (queue 1)

1. VoIP = voice over IP
2. BPDU = bridge protocol data unit

Table 30-2 shows the generated auto-QoS configuration for the egress queues.

Table 30-2 Auto-QoS Configuration for the Egress Queues

Egress Queue	Queue Number	CoS-to-Queue Map	Queue Weight	Queue Size for Gigabit-Capable Ports	Queue Size (in packets) for 10/100 Ethernet Ports
Expedite	4	5	–	10 percent	34 (10 percent)
70% WRR	3	3, 6, 7	70 percent	15 percent	51 (15 percent)
20% WRR	2	2, 4	20 percent	25 percent	82 (25 percent)
10% WRR	1	0, 1	10 percent	50 percent	170 (50 percent)

When you enable the auto-QoS feature on the first interface, these automatic actions occur:

- QoS is globally enabled (**mls qos** global configuration command).
- When you enter the **auto qos voip cisco-phone** interface configuration command on a port at the edge of the network that is connected to a Cisco IP Phone, the switch enables the trusted boundary feature. The switch uses the Cisco Discovery Protocol (CDP) to detect the presence or absence of a Cisco IP Phone. When a Cisco IP Phone is detected, the ingress classification on the interface is set to trust the QoS label received in the packet. When a Cisco IP Phone is absent, the ingress classification is set to not trust the QoS label in the packet. The switch configures egress queues on the port according to the settings in Table 30-2.
- When you enter the **auto qos voip cisco-softphone** interface configuration command on a port at the edge of the network that is connected to a device running the Cisco SoftPhone, the switch uses policing to determine whether a packet is in or out of profile and to specify the action on the packet. If the packet does not have a DSCP value of 24, 26, or 46 or is out of profile, the switch changes the DSCP value to 0. The switch configures egress queues on the port according to the settings in Table 30-2.

- When you enter the **auto qos voip trust** interface configuration command on a port connected to the interior of the network, the switch trusts the CoS value for nonrouted ports or the DSCP value for routed ports in ingress packets (the assumption is that traffic has already been classified by other edge devices). The switch configures egress queues on the port according to the settings in Table 30-2.

  For information about the trusted boundary feature, see the "Configuring a Trusted Boundary to Ensure Port Security" section on page 30-33.

When you enable auto-QoS by using the **auto qos voip cisco-phone**, the **auto qos voip cisco-softphone**, or the **auto qos voip trust** interface configuration command, the switch automatically generates a QoS configuration based on the traffic type and ingress packet label and applies the commands listed in Table 30-3 to the interface.

Table 30-3 Generated Auto-QoS Configuration

Description	Automatically Generated Command
The switch automatically enables standard QoS and configures the CoS-to-DSCP map (maps CoS values in incoming packets to a DSCP value) as shown in Table 30-1 on page 30-18.	`Switch(config)# mls qos` `Switch(config)# mls qos map cos-dscp 0 8 16 26 32 46 48 56`

If 10/100 Ethernet ports are present, the switch automatically configures the buffer size of the minimum-reserve levels 5, 6, 7, and 8:
• Level 5 can hold 170 packets.
• Level 6 can hold 85 packets.
• Level 7 can hold 51 packets.
• Level 8 can hold 34 packets.
	`Switch(config)# mls qos min-reserve 5 170`
	`Switch(config)# mls qos min-reserve 6 85`
	`Switch(config)# mls qos min-reserve 7 51`
	`Switch(config)# mls qos min-reserve 8 34`

If you entered the **auto qos voip trust** command, the switch automatically sets the ingress classification to trust the CoS value received in the packet on a nonrouted port or to trust the DSCP value received in the packet on a routed port.	`Switch(config-if)# mls qos trust cos` `Switch(config-if)# mls qos trust dscp`
If you entered the **auto qos voip cisco-phone** command, the switch automatically enables the trusted boundary feature, which uses the CDP to detect the presence or absence of a Cisco IP Phone.	`Switch(config-if)# mls qos trust device cisco-phone`

If you entered the **auto qos voip cisco-softphone** command, the switch automatically creates class maps and policy maps.	`Switch(config)# mls qos map policed-dscp 24 26 46 to 0`
	`Switch(config)# class-map match-all AutoQoS-VoIP-RTP-Trust`
	`Switch(config-cmap)# match ip dscp 46`
	`Switch(config)# class-map match-all AutoQoS-VoIP-Control-Trust`
	`Switch(config-cmap)# match ip dscp 24 26`
	`Switch(config)# policy-map AutoQoS-Police-SoftPhone`
	`Switch(config-pmap)# class AutoQoS-VoIP-RTP-Trust`
	`Switch(config-pmap-c)# set dscp 46`
	`Switch(config-pmap-c)# police 320000 8000 exceed-action policed-dscp-transmit`
	`Switch(config-pmap)# class AutoQoS-VoIP-Control-Trust`
	`Switch(config-pmap-c)# set dscp 24`
	`Switch(config-pmap-c)# police 32000 8000 exceed-action policed-dscp-transmit`

Table 30-3 Generated Auto-QoS Configuration (continued)

Description	Automatically Generated Command
After creating the class maps and policy maps, the switch automatically applies the policy map called *AutoQoS-Police-SoftPhone* to an ingress interface on which auto-QoS with the Cisco SoftPhone feature is enabled.	`Switch(config-if)# service-policy input AutoQoS-Police-SoftPhone`

The switch automatically assigns egress queue usage (as shown in Table 30-2 on page 30-18) on this interface.
The switch enables the egress expedite queue and assigns WRR weights to queues 1, 2, and 3. (The lowest value for a WRR queue is 1. When the WRR weight of a queue is set to 0, this queue becomes an expedite queue.)
The switch configures the CoS-to-egress-queue map:
• CoS values 0 and 1 select queue 1.
• CoS values 2 and 4 select queue 2.
• CoS values 3, 6, and 7 select queue 3.
• CoS value 5 selects queue 4 (expedite queue).
Because the expedite queue (queue 4) contains the VoIP data traffic, the queue is serviced until empty.
	`Switch(config-if)# wrr-queue bandwidth 10 20 70 1`
	`Switch(config-if)# no wrr-queue cos-map`
	`Switch(config-if)# wrr-queue cos-map 1 0 1`
	`Switch(config-if)# wrr-queue cos-map 2 2 4`
	`Switch(config-if)# wrr-queue cos-map 3 3 6 7`
	`Switch(config-if)# wrr-queue cos-map 4 5`
	`Switch(config-if)# priority-queue out`

On Gigabit-capable Ethernet ports only, the switch automatically configures the ratio of the sizes of the WRR egress queues: • Queue 1 is 50 percent. • Queue 2 is 25 percent. • Queue 3 is 15 percent. • Queue 4 is 10 percent.	`Switch(config-if)# wrr-queue queue-limit 50 25 15 10`

On 10/100 Ethernet ports only, the switch automatically configures minimum-reserve levels for the egress queues:
• Queue 1 selects the minimum-reserve level 5.
• Queue 2 selects the minimum-reserve level 6.
• Queue 3 selects the minimum-reserve level 7.
• Queue 4 selects the minimum-reserve level 8.
	`Switch(config-if)# wrr-queue min-reserve 1 5`
	`Switch(config-if)# wrr-queue min-reserve 2 6`
	`Switch(config-if)# wrr-queue min-reserve 3 7`
	`Switch(config-if)# wrr-queue min-reserve 4 8`

Effects of Auto-QoS on the Configuration

When auto-QoS is enabled, the **auto qos voip** interface configuration command and the generated configuration are added to the running configuration.

The switch applies the auto-QoS-generated commands as if the commands were entered from the CLI. An existing user configuration can cause the application of the generated commands to fail or to be overridden by the generated commands. These actions occur without warning. If all the generated commands are successfully applied, any user-entered configuration that was not overridden remains in the running configuration. Any user-entered configuration that was overridden can be retrieved by reloading the switch without saving the current configuration to memory. If the generated commands fail to be applied, the previous running configuration is restored.

Configuration Guidelines

Before configuring auto-QoS, you should be aware of this information:

- In releases earlier than Cisco IOS Release 12.1(20)EA2, auto-QoS configures the switch for VoIP only with Cisco IP Phones on nonrouted ports.
- In Cisco IOS Release 12.1(20)EA2 or later, auto-QoS configures the switch for VoIP with Cisco IP Phones on nonrouted and routed ports. Auto-QoS also configures the switch for VoIP with devices running the Cisco SoftPhone application.

Note When a device running Cisco SoftPhone is connected to a nonrouted or routed port, the switch supports only one Cisco SoftPhone application per port.

- To take advantage of the auto-QoS defaults, you should enable auto-QoS before you configure other QoS commands. If necessary, you can fine-tune the QoS configuration, but we recommend that you do so only after the auto-QoS configuration is completed. For more information, see the "Effects of Auto-QoS on the Configuration" section on page 30-21.
- After auto-QoS is enabled, do not modify a policy map or aggregate policer that includes *AutoQoS* in its name. If you need to modify the policy map or aggregate policer, make a copy of it, and change the copied policy map or policer. To use this new policy map instead of the generated one, remove the generated policy map from the interface, and apply the new policy map to the interface.
- You can enable auto-QoS on static, dynamic-access, voice VLAN access, and trunk ports.
- By default, the CDP is enabled on all interfaces. For auto-QoS to function properly, do not disable the CDP.
- When enabling auto-QoS with a Cisco IP Phone on a routed port, you must assign a static IP address to the IP phone.
- This release supports only Cisco IP SoftPhone Version 1.3(3) or later.
- Connected devices must use Cisco Call Manager Version 4 or later.

Upgrading from a Previous Software Release

In Cisco IOS Release 12.2(20)EA2, the implementation for auto-QoS changed from the previous release. The generated auto-QoS configuration was changed, support for the Cisco SoftPhone feature was added, and support for Cisco IP Phones on routed ports was added.

If auto-QoS is configured on the switch, if your switch is running a release earlier than Cisco IOS Release 12.2(20)EA2, and if you upgrade to Cisco IOS Release 12.2(20)EA2 or later, the configuration file will not contain the new configuration, and auto-QoS will not operate. Follow these steps to update the auto-QoS settings in your configuration file:

1. Upgrade your switch to Cisco IOS Release 12.2(20)EA2 or later.
2. Disable auto-QoS on all ports on which auto-QoS was enabled.
3. Return all the global auto-QoS settings to their default values by using the **no** commands.
4. Re-enable auto-QoS on the ports on which auto-QoS was disabled in Step 2. Configure the ports with the same auto-QoS settings as the previous ones.

Enabling Auto-QoS for VoIP

Beginning in privileged EXEC mode, follow these steps to enable auto-QoS for VoIP within a QoS domain:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface that is connected to a Cisco IP Phone or the uplink interface that is connected to another trusted switch or router in the interior of the network, and enter interface configuration mode.
Step 3	**auto qos voip** {**cisco-phone** | **cisco-softphone** | **trust**}	Enable auto-QoS. The keywords have these meanings: • **cisco-phone**—If the interface is connected to a Cisco IP Phone, the QoS labels of incoming packets are trusted only when the telephone is detected. • **cisco-softphone**—The port is connected to device running the Cisco SoftPhone feature. **Note** The **cisco-softphone** keyword is supported only in Cisco IOS Release 12.2(20)EA2 or later. • **trust**—The uplink interface is connected to a trusted switch or router, and the VoIP traffic classification in the ingress packet is trusted.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show auto qos interface** *interface-id*	Verify your entries. This command displays the QoS commands on the interface on which auto-QoS was enabled. You can use the **show running-config** privileged EXEC command to display the auto-QoS configuration and the user modifications.

To display the QoS commands that are automatically generated when auto-QoS is enabled or disabled, enter the **debug auto qos** privileged EXEC command before enabling auto-QoS. For more information, see the "Using the debug auto qos Command" section on page 38-18.

To disable auto-QoS on an interface, use the **no auto qos voip** interface configuration command. When you enter this command, the switch changes the auto-QoS settings to the standard-QoS default settings for that interface.

To disable auto-QoS on the switch, use the **no mls qos** global configuration command. When you enter this command, the switch disables QoS on all interfaces and enables pass-through mode.

This example shows how to enable auto-QoS and to trust the QoS labels in incoming packets when the device connected to the interface is detected as a Cisco IP Phone:

```
Switch(config)# interface fastethernet0/1
Switch(config-if)# auto qos voip cisco-phone
```

This example shows how to enable auto-QoS and to trust the QoS labels in incoming packets when the switch or router connected to the interface is a trusted device:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# auto qos voip trust
```

Displaying Auto-QoS Information

To display the initial auto-QoS configuration, use the **show auto qos** [**interface** [*interface-id*]] privileged EXEC command. To display any user changes to that configuration, use the **show running-config** privileged EXEC command. You can compare the **show auto qos** and the **show running-config** command output to identify the user-defined QoS settings.

To display information about the QoS configuration that might be affected by auto-QoS, use one of these commands:

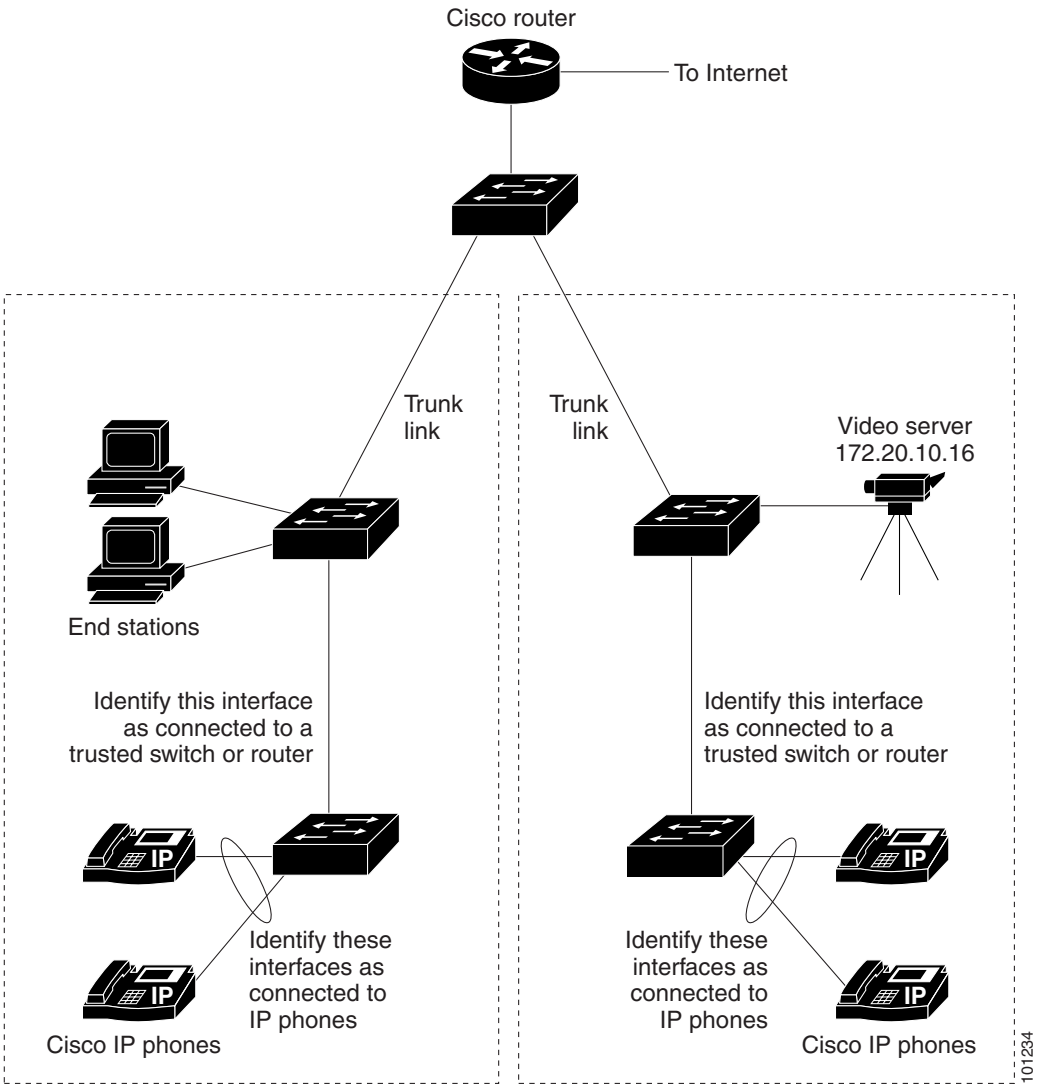
- **show mls qos**
- **show mls qos map cos-dscp**
- **show mls qos interface** [*interface-id*] [**buffers** | **queueing**]

For more information about these commands, see the command reference for this release.

Auto-QoS Configuration Example

This section describes how you could implement auto-QoS in a network, as shown in Figure 30-8. For optimum QoS performance, auto-QoS should be enabled on all the devices in the network.

Figure 30-8 Auto-QoS Configuration Example Network



The intelligent wiring closets in Figure 30-8 are composed of Catalyst 2950 switches running the EI and Catalyst 3550 switches. The object of this example is to prioritize the VoIP traffic over all other traffic. To do so, enable auto-QoS on the switches at the edge of the QoS domains in the wiring closets.

Note You should not configure any standard QoS commands before entering the auto-QoS commands. You can fine-tune the QoS configuration, but we recommend that you do so only after the auto-QoS configuration is completed.

Beginning in privileged EXEC mode, follow these steps to configure the switch at the edge of the QoS domain to prioritize the VoIP traffic over all other traffic:

	Command	Purpose
Step 1	**debug auto qos**	Enable debugging for auto-QoS. When debugging is enabled, the switch displays the QoS configuration that is automatically generated when auto-QoS is enabled.
Step 2	**configure terminal**	Enter global configuration mode.
Step 3	**cdp enable**	Enable CDP globally. By default, it is enabled.
Step 4	**interface** *interface-id*	Specify the switch port connected to the Cisco IP Phone, and enter interface configuration mode.
Step 5	**auto qos voip cisco-phone**	Enable auto-QoS on the interface, and specify that the interface is connected to a Cisco IP Phone. The QoS labels of incoming packets are trusted only when the IP phone is detected.
Step 6	**exit**	Return to global configuration mode.
Step 7		Repeat Steps 4 to 6 for as many ports as are connected to the Cisco IP Phone.
Step 8	**interface** *interface-id*	Specify the switch port identified as connected to a trusted switch or router, and enter interface configuration mode. See Figure 30-8.
Step 9	**auto qos voip trust**	Enable auto-QoS on the interface, and specify that the interface is connected to a trusted router or switch.
Step 10	**end**	Return to privileged EXEC mode.
Step 11	**show auto qos**	Verify your entries. This command displays the auto-QoS command on the interface on which auto-QoS was enabled. You can use the **show running-config** privileged EXEC command to display the auto-QoS configuration and the user modifications. For information about the QoS configuration that might be affected by auto-QoS, see the "Displaying Auto-QoS Information" section on page 26-12.
Step 12	**copy running-config startup-config**	Save the **auto qos voip** interface configuration commands and the generated auto-QoS configuration in the configuration file.

Configuring Standard QoS

Before configuring standard QoS, you must have a thorough understanding of these items:

- The types of applications used and the traffic patterns on your network.
- Traffic characteristics and needs of your network. Is the traffic bursty? Do you need to reserve bandwidth for voice and video streams?
- Bandwidth requirements and speed of the network.
- Location of congestion points in the network.

These sections describe how to configure standard QoS on your switch:

- Default Standard QoS Configuration, page 30-26
- Standard QoS Configuration Guidelines, page 30-27
- Enabling QoS Globally, page 30-29
- Configuring Classification By Using Port Trust States, page 30-30
- Configuring a QoS Policy, page 30-37
- Configuring DSCP Maps, page 30-53
- Configuring Egress Queues on Gigabit-Capable Ethernet Ports, page 30-59
- Configuring Egress Queues on 10/100 Ethernet Ports, page 30-66

Default Standard QoS Configuration

Table 30-4 shows the default standard QoS configuration when QoS is disabled.

Table 30-4 Default Standard QoS Configuration when QoS is Disabled

Port Type	QoS State	Egress traffic (DSCP and CoS Value)	Queue	Queue Weights	Tail-drop Thresholds	CoS Mapping to Queue
Gigabit-capable Ethernet ports	Disabled	Pass through.	All of the queue RAM is allocated to queue 1 (no expedite queue).	–	100%, 100% WRED is disabled.	All CoS values map to queue 1.
10/100 Ethernet ports	Disabled	Pass through.	Each of the eight minimum-reserve levels have a buffer size of 100 packets. The queue selects the level.	–	–	All CoS values map to queue 1.

When QoS is disabled, there is no concept of trusted or untrusted ports because the packets are not modified (the CoS, DSCP, and IP precedence values in the packet are not changed).

Table 30-5 shows the default standard QoS configuration without any further configuration when QoS is enabled.

Table 30-5 Default Standard QoS Configuration when QoS is Enabled

Port Type	QoS State	Egress traffic (DSCP and CoS Value)	Queue	Queue Weights	Tail-drop Thresholds	CoS Mapping to Queue
Gigabit-capable Ethernet ports	Enabled (no policing)	DSCP=0 CoS=0 (0 means best-effort delivery.)	Four queues are available (no expedite queue).	Each queue has the same weight.	100%, 100% WRED is disabled.	0, 1: queue 1 2, 3: queue 2 4, 5: queue 3 6, 7: queue 4
10/100 Ethernet ports	Enabled (no policing)	DSCP=0 CoS=0 (0 means best-effort delivery.)	Each of the eight minimum-reserve levels have a buffer size of 100 packets. The queue selects the level.	Each queue has the same weight.	–	0, 1: queue 1 2, 3: queue 2 4, 5: queue 3 6, 7: queue 4

The default port CoS value is 0.

The default port trust state on all ports is untrusted.

No policy maps are configured.

No policers are configured.

The default CoS-to-DSCP map is shown in Table 30-6 on page 30-54.

The default IP-precedence-to-DSCP map is shown in Table 30-7 on page 30-55.

The default DSCP-to-CoS map is shown in Table 30-8 on page 30-57.

The default DSCP-to-DSCP-mutation map is a null map, which maps an incoming DSCP value to the same DSCP value.

The default policed-DSCP map is a null map, which maps an incoming DSCP value to the same DSCP value (no markdown).

The default DSCP-to-switch-priority map maps DSCPs 0 to 15 to priority 0, DSCPs 16 to 31 to priority 1, DSCPs 32 to 47 to priority 2, and DSCPs 48 to 63 to priority 3.

Standard QoS Configuration Guidelines

Before beginning the QoS configuration, you should be aware of this information:

- You must disable the IEEE 802.3x flow control on all ports before enabling QoS on the switch. To disable it, use the **flowcontrol receive off** and **flowcontrol send off** interface configuration commands.

Note If QoS is disabled and you enter the **mls qos** global configuration command, this message appears:

```
QoS:ensure flow-control on all interfaces are OFF for proper operation.
```

- If you have EtherChannel ports configured on your switch, you must configure QoS classification, policing, mapping, and queueing on the individual physical ports that comprise the EtherChannel. You must decide whether the QoS configuration should match on all ports in the EtherChannel.
- You can classify traffic on an ingress physical port or on a per-ingress-port per-VLAN basis. You cannot classify traffic at the switch virtual interface level.
- Only one ACL per class map and only one **match** class-map configuration command per class map are supported. The ACL can have multiple access control entries, which are commands that match fields against the contents of the packet.
- When classifying traffic on a per-port per-VLAN basis, you must use the **match-all** keyword with the **class-map** global configuration command. For more information, see the "Classifying Traffic on a Per-Port Per-VLAN Basis by Using Class Maps" section on page 30-42.
- The switch has only 256 VLAN labels (a few are always used internally for defaults), which are shared between VLAN maps and per-port per-VLAN policing. If a large number of VLANs are used in class maps and either different ACL actions are performed on them or they have different VLAN maps applied, the available VLAN labels might be insufficient. As a consequence, the TCAM entries are not programmed, and the feature does not work. Use the **show tcam qos** *tcam-id* **port-labels vlan-labels** privileged EXEC command to display how many VLAN labels are in use by this QoS feature.
- It is not possible to match IP fragments against configured IP extended ACLs to enforce QoS. IP fragments are sent as best-effort. IP fragments are denoted by fields in the IP header.
- You can match IP options against configured IP extended ACLs to enforce QoS. These packets are sent to the CPU and processed by software. IP options are denoted by fields in the IP header.
- You can configure a policer on an ingress or egress physical port; you can configure a per-port per-VLAN policer only on an ingress port (specifies the bandwidth limits for the traffic on a per-VLAN basis, for a given port). You cannot police at the switch virtual interface level.

  You cannot configure per-port per-VLAN policing on routed ports or on virtual (logical) interfaces. It is supported only on an ingress port configured as a trunk or as a static-access port.

  The switch does not support per-VLAN QoS or VLAN QoS policing across the entire switch.
- Use only the **match ip dscp** *dscp-list* class-map configuration command in a policy map that is attached to an egress interface.
- You cannot classify traffic by using a port trust state (for example, **mls qos trust** [**cos** | **dscp** | **ip-precedence**] and by using a policy map (for example, **service-policy input** *policy-map-name*) at the same time on an interface. These commands are mutually exclusive. The last one configured overwrites the previous configuration.
- You cannot use the **service-policy** interface configuration command to attach policy maps that contain these elements to an egress interface:
  - **set** or **trust** policy-map class configuration commands. Instead, you can use the **police** policy-map class configuration command to mark down (reduce) the DSCP value at the egress interface.
  - Access control list (ACL) classification.
  - Per-port per-VLAN classification.

  The only match criterion in a policy map that can be attached to an egress interface is the **match ip dscp** *dscp-list* class-map configuration command.
- You can create an aggregate policer that is shared by multiple traffic classes within the same policy map. However, you cannot use the aggregate policer across different policy maps or interfaces.

- All ingress QoS processing actions apply to control traffic (such as spanning-tree bridge protocol data units [BPDUs] and routing update packets) that the switch receives.
- Layer 3 QoS ACLs and other QoS features related to Layer 3 information are not supported on tunnel ports. MAC-based QoS is supported on tunnel ports. When applied to trunk ports, Layer 3 QoS ACLs do not work for VLANs that include tunnel ports.
- Do not use the **show policy-map interface** privileged EXEC command to display classification information for incoming traffic. The **interface** keyword is not supported, and you should ignore the statistics shown in the display. Instead, you should specify the DSCPs to be monitored by using the **mls qos monitor dscp** *dscp1 ... dscp8* interface configuration command, and then you should use the **show mls qos interface** *interface-id* **statistics** privileged EXEC command. For more information about these commands, see the command reference for this release.

Enabling QoS Globally

By default, QoS is disabled on the switch, which means that the switch offers best-effort service to each packet regardless of the packet contents or size. All CoS values map to egress queue 1 with both tail-drop thresholds set to 100 percent of the total queue size for Gigabit-capable Ethernet ports. On 10/100 Ethernet ports, all CoS values map to egress queue 1, which uses minimum-reserve level 1 and can hold up to 100 packets. When the buffer is full, packets are dropped.

Beginning in privileged EXEC mode, follow these steps to enable QoS:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface range** *port-range*	Enter interface configuration mode, and execute a command on multiple interfaces. You can define up to five interface ranges with a single command, with each range separated by a comma. All interfaces in a range must be the same type; that is, all Fast Ethernet ports or all Gigabit Ethernet ports.
Step 3	**flowcontrol receive off** **flowcontrol send off**	Disable flow control on all interfaces.
Step 4	**exit**	Return to global configuration mode.
Step 5	**mls qos**	Enable QoS globally.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show mls qos**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

After QoS is enabled, the default settings are as shown in Table 30-4 on page 30-26.

To disable QoS, use the **no mls qos** global configuration command.

Configuring Classification By Using Port Trust States

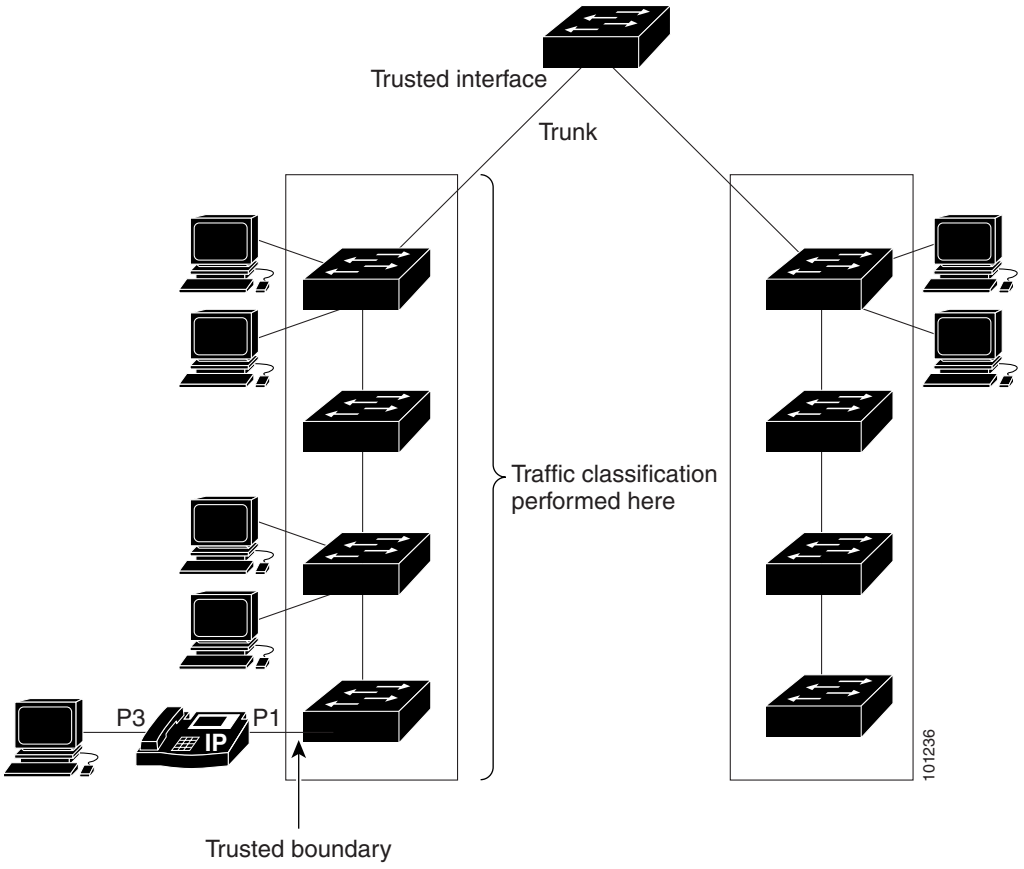
These sections describe how to classify incoming traffic by using port trust states:

- Configuring the Trust State on Ports within the QoS Domain, page 30-30
- Configuring the CoS Value for an Interface, page 30-32
- Configuring a Trusted Boundary to Ensure Port Security, page 30-33
- Enabling Pass-Through Mode, page 30-34
- Configuring the DSCP Trust State on a Port Bordering Another QoS Domain, page 30-35

Configuring the Trust State on Ports within the QoS Domain

Packets entering a QoS domain are classified at the edge of the QoS domain. When the packets are classified at the edge, the switch port within the QoS domain can be configured to one of the trusted states because there is no need to classify the packets at every switch within the QoS domain. Figure 30-9 shows a sample network topology.

Figure 30-9 Port Trusted States within the QoS Domain



Beginning in privileged EXEC mode, follow these steps to configure the port to trust the classification of the traffic that it receives:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS globally.
Step 3	**interface** *interface-id*	Specify the interface to be trusted, and enter interface configuration mode. Valid interfaces include physical interfaces.
Step 4	**mls qos trust** {**cos** | **dscp** | **ip-precedence**}	Configure the port trust state. By default, the port is not trusted. The keywords have these meanings: • **cos**—Classifies ingress packets with the packet CoS values. For untagged packets, the port default CoS value is used. The default port CoS value is 0. • **dscp**—Classifies ingress packets with packet DSCP values. For non-IP packets, the packet CoS value is used if the packet is tagged; for untagged packets, the default port CoS is used. Internally, the switch maps the CoS value to a DSCP value by using the CoS-to-DSCP map. • **ip-precedence**—Classifies ingress packets with the packet IP-precedence values. For non-IP packets, the packet CoS value is used if the packet is tagged; for untagged packets, the default port CoS is used. Internally, the switch maps the CoS value to a DSCP value by using the CoS-to-DSCP map. Use the **cos** keyword setting if your network is composed of Ethernet LANs, Catalyst 3500 XL and 2900 XL switches, and has no more than two types of traffic. Recall that on Catalyst 3500 XL and 2900 XL switches, CoS configures each transmitting port with a normal-priority transmit queue and a high-priority transmit queue. Use the **dscp** or **ip-precedence** keyword if your network is not composed of only Ethernet LANs and if you are familiar with sophisticated QoS features and implementations.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show mls qos interface**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return a port to its untrusted state, use the **no mls qos trust** interface configuration command.

For information on how to change the default CoS value, see the "Configuring the CoS Value for an Interface" section on page 30-32. For information on how to configure the CoS-to-DSCP map, see the "Configuring the CoS-to-DSCP Map" section on page 30-54.

Configuring the CoS Value for an Interface

QoS assigns the CoS value specified with the **mls qos cos** interface configuration command to untagged frames received on trusted and untrusted ports.

Beginning in privileged EXEC mode, follow these steps to define the default CoS value of a port or to assign the default CoS to all incoming packets on the port:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS globally.
Step 3	**interface** *interface-id*	Specify the interface to be trusted, and enter interface configuration mode. Valid interfaces include physical interfaces.
Step 4	**mls qos cos** {*default-cos* | **override**}	Configure the default CoS value for the port. • For *default-cos*, specify a default CoS value to be assigned to a port. If the port is CoS trusted and packets are untagged, the default CoS value becomes the CoS value for the packet. The CoS range is 0 to 7. The default is 0. • Use the **override** keyword to override the previously configured trust state of the incoming packets and to apply the default port CoS value to all incoming packets. By default, CoS override is disabled. Use the **override** keyword when all incoming packets on certain ports deserve higher or lower priority than packets entering from other ports. Even if a port was previously set to trust DSCP, CoS, or IP precedence, this command overrides the previously configured trust state, and all the incoming CoS values are assigned the default CoS value configured with this command. If an incoming packet is tagged, the CoS value of the packet is modified with the default CoS of the port at the ingress port.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show mls qos interface**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default setting, use the **no mls qos cos** {*default-cos* | **override**} interface configuration command.

Configuring a Trusted Boundary to Ensure Port Security

In a typical network, you connect a Cisco IP Phone to a switch port as shown in Figure 30-9 on page 30-30, and cascade devices that generate data packets from the back of the telephone. The Cisco IP Phone guarantees the voice quality through a shared data link by marking the CoS level of the voice packets as high priority (CoS = 5) and by marking the data packets as low priority (CoS = 0). Traffic sent from the telephone to the switch is typically marked with a tag that uses the IEEE 802.1Q header. The header contains the VLAN information and the class of service (CoS) 3-bit field, which is the priority of the packet.

For most Cisco IP Phone configurations, the traffic sent from the telephone to the switch should be trusted to ensure that voice traffic is properly prioritized over other types of traffic in the network. By using the **mls qos trust cos** interface configuration command, you configure the switch port to which the telephone is connected to trust the CoS labels of all traffic received on that port. Use the **mls qos trust dscp** interface configuration command to configure a routed port to which the telephone is connected to trust the DSCP labels of all traffic received on that port.

With the trusted setting, you also can use the trusted boundary feature to prevent misuse of a high-priority queue if a user bypasses the telephone and connects the PC directly to the switch. Without trusted boundary, the CoS labels generated by the PC are trusted by the switch (because of the trusted CoS setting). By contrast, trusted boundary uses CDP to detect the presence of a Cisco IP Phone (such as the Cisco IP Phone 7910, 7935, 7940, and 7960) on a switch port. If the telephone is not detected, the trusted boundary feature disables the trusted setting on the switch port and prevents misuse of a high-priority queue. Note that the trusted boundary feature is not effective if the PC and Cisco IP Phone are connected to a hub that is connected to the switch.

In some situations, you can prevent a PC connected to the Cisco IP Phone from taking advantage of a high-priority data queue. You can use the **switchport priority extend cos** interface configuration command to configure the telephone through the switch CLI to override the priority of the traffic received from the PC.

Beginning in privileged EXEC mode, follow these steps to enable trusted boundary on a port:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS globally.
Step 3	**cdp run**	Enable CDP globally. By default, CDP is enabled.
Step 4	**interface** *interface-id*	Specify the interface connected to the IP phone, and enter interface configuration mode. Valid interfaces include physical interfaces.
Step 5	**cdp enable**	Enable CDP on the interface. By default, CDP is enabled.
Step 6	**mls qos trust cos** or **mls qos trust dscp**	Configure the switch port to trust the CoS value in traffic received from the Cisco IP Phone. or Configure the routed port to trust the DSCP value in traffic received from the Cisco IP Phone. By default, the port is not trusted.

	Command	Purpose
Step 7	**mls qos trust device cisco-phone**	Specify that the Cisco IP Phone is a trusted device. You cannot enable both trusted boundary and auto-QoS (**auto qos voip** interface configuration command) at the same time; they are mutually exclusive.
Step 8	**end**	Return to privileged EXEC mode.
Step 9	**show mls qos interface**	Verify your entries.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the trusted boundary feature, use the **no mls qos trust device** interface configuration command.

Enabling Pass-Through Mode

You can use the pass-through mode to enable the CoS and DSCP setting to be independent for packets that contain both values. Use the pass-through mode when you do not want the other value (CoS or DSCP) to be modified when using the **mls qos trust** [**cos** | **dscp**] interface configuration command.

By default, in software releases earlier than Cisco IOS Release 12.1(11)EA1, if you configure the interface to trust the DSCP, the switch does not modify the DSCP field of the IP packet. However, the switch modifies the CoS value of the packet according to the DSCP-to-CoS map. If you configure the interface to trust the CoS, the switch does not modify the CoS field of the packet. However, the switch modifies the DSCP according to the CoS-to-DSCP map if the packet is an IP packet.

In Cisco IOS Release 12.1(11)EA1 or later, you configure the interface for pass-through mode. The interface trusts the DSCP, and the switch sends the packet without modifying the CoS value (the DSCP-to-CoS map is ignored). Otherwise, the interface trusts the CoS, and the switch sends the packet without modifying the DSCP value. The CoS-to-DSCP map is ignored.

Beginning in privileged EXEC mode, follow these steps to enable pass-through mode on an interface:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**interface** *interface-id*	Specify the interface on which pass-through mode is enabled, and enter interface configuration mode. Valid interfaces include physical interfaces.
Step 3	**mls qos trust cos pass-through dscp**	Enable pass-through mode. The interface is configured to trust the CoS value of the incoming packets. The switch does not modify the DSCP value.
	or	or
	mls qos trust dscp pass-through cos	Enable pass-through mode. The interface is configured to trust the DSCP value of the incoming packets. The switch does not modify the CoS value.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show mls qos interface** [*interface-id*]	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable pass-through mode, use the **no mls qos trust cos pass-through dscp** or the **no mls qos trust dscp pass-through cos** interface configuration command.

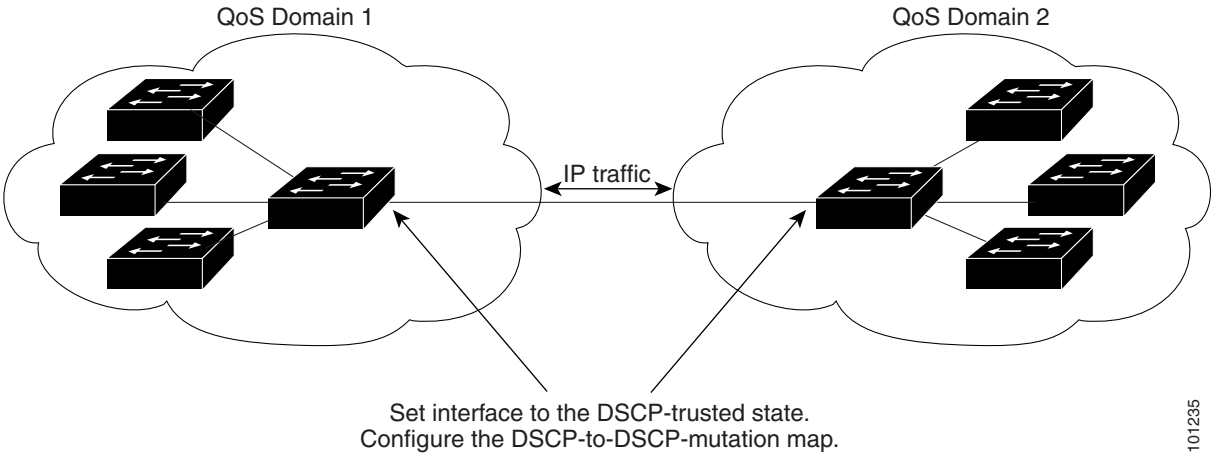
If you configure the **mls qos trust** [**cos pass-through dscp** | **dscp pass-through cos**] interface configuration command and then configure the **mls qos trust** [**cos** | **dscp**] interface configuration command, pass-through mode is disabled.

Note If you configure an interface for DSCP pass-through mode by using the **mls qos trust cos pass-through dscp** interface configuration command and apply the DSCP-to-DSCP mutation map to the same interface, the DSCP value changes according to the mutation map.

Configuring the DSCP Trust State on a Port Bordering Another QoS Domain

If you are administering two separate QoS domains between which you want to implement QoS features for IP traffic, you can configure the switch ports bordering the domains to a DSCP-trusted state as shown in Figure 30-10. Then the receiving port accepts the DSCP-trusted value and avoids the classification stage of QoS. If the two domains use different DSCP values, you can configure the DSCP-to-DSCP-mutation map to translate a set of DSCP values to match the definition in the other domain.

Figure 30-10 DSCP-Trusted State on a Port Bordering Another QoS Domain



Beginning in privileged EXEC mode, follow these steps to configure the DSCP-trusted state on a port and modify the DSCP-to-DSCP-mutation map. To ensure a consistent mapping strategy across both QoS domains, you must perform this procedure on the ports in both domains:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.

	Command	Purpose
Step 3	**mls qos map dscp-mutation** *dscp-mutation-name in-dscp* **to** *out-dscp*	Modify the DSCP-to-DSCP-mutation map. The default DSCP-to-DSCP-mutation map is a null map, which maps an incoming DSCP value to the same DSCP value. • For *dscp-mutation-name*, enter the mutation map name. You can create more than one map by specifying a new name. • For *in-dscp*, enter up to eight DSCP values separated by spaces. Then enter the **to** keyword. • For *out-dscp*, enter a single DSCP value. The DSCP range is 0 to 63.
Step 4	**interface** *interface-id*	Specify the interface to be trusted, and enter interface configuration mode. Valid interfaces include physical interfaces.
Step 5	**mls qos trust dscp**	Configure the ingress port as a DSCP-trusted port.
Step 6	**mls qos dscp-mutation** *dscp-mutation-name*	Apply the map to the specified ingress DSCP-trusted port. You can apply the map to different Gigabit-capable Ethernet ports. However, on 10/100 Ethernet ports, you can attach only one DSCP-to-DSCP-mutation map to a group of twelve ports. For example, Fast Ethernet ports 1 to 12 are a group, Fast Ethernet ports 13 to 24 are a group, Gigabit Ethernet 1 is a group, and Gigabit Ethernet 2 is a group. When applying a mutation map to any port in a group, all ports in the same group are automatically configured with the same map.
Step 7	**end**	Return to privileged EXEC mode.
Step 8	**show mls qos maps dscp-mutation**	Verify your entries.
Step 9	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return a port to its non-trusted state, use the **no mls qos trust** interface configuration command. To return to the default DSCP-to-DSCP-mutation map values, use the **no mls qos map dscp-mutation** *dscp-mutation-map-name* global configuration command.

This example shows how to configure an interface to the DSCP-trusted state and to modify the DSCP-to-DSCP-mutation map (named *gi0/2-mutation*) so that incoming DSCP values 10 to 13 are mapped to DSCP values 30:

```
Switch# configure terminal
Switch(config)# mls qos map dscp-mutation gi0/2-mutation 10 11 12 13 to 30
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# mls qos trust dscp
Switch(config-if)# mls qos dscp-mutation gi0/2-mutation
Switch(config-if)# end
```

Configuring a QoS Policy

Configuring a QoS policy typically requires classifying traffic into classes, configuring policies applied to those traffic classes, and attaching policies to interfaces.

For background information, see the "Classification" section on page 30-5 and the "Policing and Marking" section on page 30-8.

These sections how to configure a QoS policy:

- Classifying Traffic by Using ACLs, page 30-37
- Classifying Traffic on a Physical-Port Basis by Using Class Maps, page 30-40
- Classifying Traffic on a Per-Port Per-VLAN Basis by Using Class Maps, page 30-42
- Classifying, Policing, and Marking Traffic by Using Policy Maps, page 30-44
- Classifying, Policing, and Marking Traffic by Using Aggregate Policers, page 30-50

Classifying Traffic by Using ACLs

You can classify IP traffic by using IP standard or IP extended ACLs; you can classify non-IP traffic by using Layer 2 MAC ACLs.

Beginning in privileged EXEC mode, follow these steps to create an IP standard ACL for IP traffic:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]	Create an IP standard ACL, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number. The range is 1 to 99 and 1300 to 1999. • Use the **permit** keyword to permit a certain type of traffic if the conditions are matched. Use the **deny** keyword to deny a certain type of traffic if conditions are matched. • For *source*, enter the network or host from which the packet is being sent. You can use the **any** keyword as an abbreviation for 0.0.0.0 255.255.255.255. • (Optional) For *source-wildcard*, enter the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. **Note** When creating an access list, remember that, by default, the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show access-lists**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete an access list, use the **no access-list** *access-list-number* global configuration command.

This example shows how to allow access for only those hosts on the three specified networks. The wildcard bits apply to the host portions of the network addresses. Any host with a source address that does not match the access list statements is rejected.

```
Switch(config)# access-list 1 permit 192.5.255.0 0.0.0.255
Switch(config)# access-list 1 permit 128.88.0.0 0.0.255.255
Switch(config)# access-list 1 permit 36.0.0.0 0.0.0.255
! (Note: all other access implicitly denied)
```

Beginning in privileged EXEC mode, follow these steps to create an IP extended ACL for IP traffic:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard*	Create an IP extended ACL, repeating the command as many times as necessary. • For *access-list-number*, enter the access list number. The range is 100 to 199 and 2000 to 2699. • Use the **permit** keyword to permit a certain type of traffic if the conditions are matched. Use the **deny** keyword to deny a certain type of traffic if conditions are matched. • For *protocol*, enter the name or number of an IP protocol. Use the question mark (?) to see a list of available protocol keywords. • For *source*, enter the network or host from which the packet is being sent. You specify this by using dotted decimal notation, by using the **any** keyword as an abbreviation for *source* 0.0.0.0 *source-wildcard* 255.255.255.255, or by using the **host** keyword for *source* 0.0.0.0. • For *source-wildcard*, enter the wildcard bits by placing ones in the bit positions that you want to ignore. You specify the wildcard by using dotted decimal notation, by using the **any** keyword as an abbreviation for *source* 0.0.0.0 *source-wildcard* 255.255.255.255, or by using the **host** keyword for *source* 0.0.0.0. • For *destination*, enter the network or host to which the packet is being sent. You have the same options for specifying the *destination* and *destination-wildcard* as those described by *source* and *source-wildcard*. **Note** When creating an access list, remember that, by default, the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.
Step 4	**end**	Return to privileged EXEC mode.
Step 5	**show access-lists**	Verify your entries.
Step 6	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete an access list, use the **no access-list** *access-list-number* global configuration command.

This example shows how to create an ACL that permits IP traffic with a DSCP value set to 32 from any source to any destination:

```
Switch(config)# access-list 100 permit ip any any dscp 32
```

This example shows how to create an ACL that permits IP traffic with a precedence value of 5 from a source host at 10.1.1.1 to a destination host at 10.1.1.2:

```
Switch(config)# access-list 100 permit ip host 10.1.1.1 host 10.1.1.2 precedence 5
```

This example shows how to create an ACL that permits PIM traffic with a DSCP set to 32 from any source to a destination group address of 224.0.0.2:

```
Switch(config)# access-list 102 permit pim any 224.0.0.2 dscp 32
```

Beginning in privileged EXEC mode, follow these steps to create a Layer 2 MAC ACL for non-IP traffic:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**mac access-list extended** *name*	Create a Layer 2 MAC ACL by specifying the name of the list. After entering this command, the mode changes to extended MAC ACL configuration.
Step 4	{**permit** | **deny**} {**host** *src-MAC-addr mask* | **any** | **host** *dst-MAC-addr* | *dst-MAC-addr mask*} [*type mask*]	Specify the type of traffic to permit or deny if the conditions are matched, entering the command as many times as necessary. • For *src-MAC-addr*, enter the MAC address of the host from which the packet is being sent. You specify this by using the hexadecimal format (H.H.H), by using the **any** keyword as an abbreviation for *source* 0.0.0, *source-wildcard* ffff.ffff.ffff, or by using the **host** keyword for *source* 0.0.0. • For *mask,* enter the wildcard bits by placing ones in the bit positions that you want to ignore. • For *dst-MAC-addr*, enter the MAC address of the host to which the packet is being sent. You specify this by using the hexadecimal format (H.H.H), by using the **any** keyword as an abbreviation for *source* 0.0.0, *source-wildcard* ffff.ffff.ffff, or by using the **host** keyword for *source* 0.0.0. • (Optional) For *type mask*, specify the Ethertype number of a packet with Ethernet II or SNAP encapsulation to identify the protocol of the packet. For *type*, the range is from 0 to 65535, typically specified in hexadecimal. For *mask*, enter the *don't care* bits applied to the Ethertype before testing for a match. **Note** When creating an access list, remember that, by default, the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show access-lists** [*access-list-number* | *access-list-name*]	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete an access list, use the **no mac access-list extended** *access-list-name* global configuration command.

This example shows how to create a Layer 2 MAC ACL with two permit statements. The first statement allows traffic from the host with MAC address 0001.0000.0001 to the host with MAC address 0002.0000.0001. The second statement allows only Ethertype XNS-IDP traffic from the host with MAC address 0001.0000.0002 to the host with MAC address 0002.0000.0002.

```
Switch(config)# mac access-list extended maclist1
Switch(config-ext-macl)# permit 0001.0000.0001 0.0.0 0002.0000.0001 0.0.0
Switch(config-ext-macl)# permit 0001.0000.0002 0.0.0 0002.0000.0002 0.0.0 xns-idp
! (Note: all other access implicitly denied)
```

Classifying Traffic on a Physical-Port Basis by Using Class Maps

You use the **class-map** global configuration command to name and to isolate a specific traffic flow (or class) from all other traffic. The class map defines the criteria to use to match against a specific traffic flow to further classify it. Match statements can include criterion such as an ACL, IP precedence values, or DSCP values. The match criterion is defined with one match statement entered within the class-map configuration mode.

You cannot configure both port-based classification and VLAN-based classification at the same time.

Note You can also create class-maps during policy map creation by using the **class** policy-map configuration command. For more information, see the "Classifying, Policing, and Marking Traffic by Using Policy Maps" section on page 30-44.

Beginning in privileged EXEC mode, follow these steps to create a class map and to define the match criterion to classify traffic on a physical-port basis:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] or **access-list** *access-list-number* {**deny** | **permit**} *protocol source* [*source-wildcard*] *destination* [*destination-wildcard*] or **mac access-list extended** *name* {**permit** | **deny**} {**host** *src-MAC-addr mask* | **any** | **host** *dst-MAC-addr* | *dst-MAC-addr mask*} [*type mask*]	Create an IP standard or extended ACL for IP traffic or a Layer 2 MAC ACL for non-IP traffic, repeating the command as many times as necessary. For more information, see the "Classifying Traffic by Using ACLs" section on page 30-37. **Note** When creating an access list, remember that, by default, the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.

	Command	Purpose
Step 4	**class-map** [**match-all** | **match-any**] *class-map-name*	Create a class map, and enter class-map configuration mode. By default, no class maps are defined. • (Optional) Use the **match-all** keyword to perform a logical-AND of all matching statements under this class map. All match criteria in the class map must be matched. • (Optional) Use the **match-any** keyword to perform a logical-OR of all matching statements under this class map. One or more match criteria must be matched. • For *class-map-name*, specify the name of the class map. If neither the **match-all** or **match-any** keyword is specified, the default is **match-all**. **Note** Because only one **match** command per class map is supported, the **match-all** and **match-any** keywords function the same.
Step 5	**match** {**access-group** *acl-index-or-name* | **ip dscp** *dscp-list* | **ip precedence** *ip-precedence-list*}	Define the match criterion to classify traffic. By default, no match criterion is defined. Only one match criterion per class map is supported, and only one ACL per class map is supported. • For **access-group** *acl-index-or-name*, specify the number or name of the ACL created in Step 3. • For **ip dscp** *dscp-list*, enter a list of up to eight IP DSCP values to match against incoming packets. Separate each value with a space. The range is 0 to 63. • For **ip precedence** *ip-precedence-list*, enter a list of up to eight IP-precedence values to match against incoming packets. Separate each value with a space. The range is 0 to 7. **Note** The only match criterion in a policy map that can be attached to an egress interface is the **match ip dscp** *dscp-list* class-map configuration command.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show class-map**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete an existing class map, use the **no class-map** [**match-all** | **match-any**] *class-map-name* global configuration command. To remove a match criterion, use the **no match** {**access-group** *acl-index-or-name* | **ip dscp** | **ip precedence**} class-map configuration command.

This example shows how to configure the class map called *class1*. The *class1* has one match criterion, which is access list 103. It permits traffic that matches a DSCP value of 10 from any host to any destination.

```
Switch(config)# access-list 103 permit any any dscp 10
Switch(config)# class-map class1
Switch(config-cmap)# match access-group 103
Switch(config-cmap)# end
```

This example shows how to create a class map called *class2*, which matches incoming traffic with DSCP values of 10, 11, and 12.

```
Switch(config)# class-map class2
Switch(config-cmap)# match ip dscp 10 11 12
Switch(config-cmap)# end
```

This example shows how to create a class map called *class3*, which matches incoming traffic with IP-precedence values of 5, 6, and 7:

```
Switch(config)# class-map class3
Switch(config-cmap)# match ip precedence 5 6 7
Switch(config-cmap)# end
```

Classifying Traffic on a Per-Port Per-VLAN Basis by Using Class Maps

You use the **class-map** global configuration command to name and to isolate a specific traffic flow (or class) from all other traffic. To further classify the traffic flow, the class map defines the matching criteria to use.

To define packet classification on a per-port per-VLAN basis, follow these guidelines:

- You must use the **match-all** keyword with the **class-map** global configuration command.
- Per-port per-VLAN classification is a per-port feature and does not work on redundant links. It is supported only on an ingress port configured as a trunk or as a static-access port.
- The class map must have two **match** commands in this order: one **match vlan** *vlan-list* class-map configuration command and one **match class-map** *class-map-name* class-map configuration command. The class map specified in the **match class-map** *class-map-name* command must be predefined and cannot contain the **match vlan** *vlan-list* and the **match class-map** *class-map-name* commands.
- You cannot configure both port-based classification and VLAN-based classification at the same time. When you configure the **match vlan** *vlan-list* command, the class map becomes per-port per-VLAN based. If you configure a policy map that contains both port-based and VLAN-based class maps, the switch rejects the policy map when you attach it to an interface.
- With per-port per-VLAN classification, unmatched VLANs are treated similarly to the default class, which means that the unmatched VLANs share the remaining bandwidth from those used by the matched VLAN classes. You cannot modify this default-class behavior. If necessary, you can use VLAN map filters to block these VLANs.
- Within a policy map, when you use the **match vlan** *vlan-list* command, all other class maps must use the **match vlan** *vlan-list* command.
- If you want to modify the VLAN list, first remove the previous configuration in the class map by using the **no match vlan** *vlan-list* command and the **no match class-map** *class-map-name* command. Then reconfigure the class map, and specify the new VLAN list. If the policy map is attached to an interface and you modify the class map by using any other method, the policy map detaches from the interface.

Note When you use the **match vlan** *vlan-list* class-map configuration command, you can enter up to 30 VLAN IDs. When you enter a range of VLANs, such as *10-15*, the VLAN range is counted as two VLAN IDs.

Note You can also create class-maps during policy map creation by using the **class** policy-map configuration command. For more information, see the "Classifying, Policing, and Marking Traffic by Using Policy Maps" section on page 30-44.

Beginning in privileged EXEC mode, follow these steps to create a class map and to define the match criterion to classify traffic on a per-port per-VLAN basis:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**class-map match-any** *class-map-name*	Create a class map, and enter class-map configuration mode. By default, no class maps are defined. • Use the **match-any** keyword to perform a logical-OR of all matching statements under this class map. One or more match criteria must be matched. • For *class-map-name*, specify the name of the class map.
Step 4	**match** {**access-group** *acl-index-or-name* | **ip dscp** *dscp-list* | **ip precedence** *ip-precedence-list*}	Define the match criterion to classify traffic. By default, no match criterion is defined. • For **access-group** *acl-index-or-name*, specify the number or name of the ACL. • For **ip dscp** *dscp-list*, enter a list of up to eight IP DSCP values to match against incoming packets. Separate each value with a space. The range is 0 to 63. • For **ip precedence** *ip-precedence-list*, enter a list of up to eight IP-precedence values to match against incoming packets. Separate each value with a space. The range is 0 to 7.
Step 5	**exit**	Return to global configuration mode.
Step 6	**class-map match-all** *class-map-name*	Create a class map, and enter class-map configuration mode. By default, no class maps are defined. • Use the **match-all** keyword to perform a logical-AND of all matching statements under this class map. All match criteria in the class map must be matched. • For *class-map-name*, specify the name of the class map created in Step 3.
Step 7	**match vlan** *vlan-list*	Define the match criterion to classify traffic. By default, no match criterion is defined. For *vlan-list*, specify a list of VLANs to match against incoming packets. You can enter up to 30 VLAN IDs. Use a hyphen for a range of VLANs; the VLAN range is counted as two VLAN IDs. Use a space to separate individual VLANs. The range is 1 to 4094. You can enter only one **match vlan** command, and you must enter it before the **match class-map** command.

	Command	Purpose
Step 8	**match class-map** *class-map-name*	Specify the name of the class map created in Step 3.
Step 9	**end**	Return to privileged EXEC mode.
Step 10	**show class-map**	Verify your entries.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete an existing class map, use the **no class-map** [**match-all** | **match-any**] *class-map-name* global configuration command. To remove a match criterion, use the **no match** {**access-group** *acl-index-or-name* | **ip dscp** | **ip precedence**} class-map configuration command.

This example shows how to configure a class map called *dscp_class* whose match criterion is to match IP DSCP 9. A second class map, called *vlan_class*, matches traffic on VLANs 10, 20 to 30, and 40 to class map *dscp_class*:

```
Switch(config)# class-map match-any dscp_class
Switch(config-cmap)# match ip dscp 9
Switch(config-cmap)# exit
Switch(config)# class-map match-all vlan_class
Switch(config-cmap)# match vlan 10 20-30 40
Switch(config-cmap)# match class-map dscp_class
Switch(config-cmap)# exit
```

Classifying, Policing, and Marking Traffic by Using Policy Maps

A policy map specifies which traffic class to act on. Actions can include trusting the CoS, DSCP, or IP precedence values in the traffic class; setting a specific CoS, DSCP, or IP precedence value in the traffic class; and specifying the traffic bandwidth limitations for each matched traffic class (policer) and the action to take (marking) when the traffic is out of profile.

A policy map also has these characteristics:

- A policy map can contain multiple class statements, each with different match criteria and policers.
- A separate policy-map class can exist for each type of traffic received through an interface.
- A policy-map trust state supersedes an interface trust state.

Follow these guidelines when configuring policy maps:

- Only one policy map per interface per direction is supported. You can apply the same policy map to multiple interfaces and directions.
- If you configure the IP-precedence-to-DSCP map by using the **mls qos map ip-prec-dscp** *dscp1...dscp8* global configuration command, the settings only affect packets on ingress interfaces that are configured to trust the IP precedence value. In a policy map, if you set the packet IP precedence value to a new value by using the **set ip precedence** *new-precedence* policy-map class configuration command, the egress DSCP value is not affected by the IP-precedence-to-DSCP map. If you want to egress DSCP value to be different than the ingress value, use the **set dscp** *new-dscp* policy-map class configuration command.
- When you apply a policy map defined by the **policy-map** global configuration command to the output of an interface or remove the policy map and interface association, the interface goes down. To re-enable the interface, use the **shutdown** and then the **no shutdown** interface configuration commands.

Beginning in privileged EXEC mode, follow these steps to create a policy map:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] or **access-list** *access-list-number* {**deny** | **permit**} *protocol source* [*source-wildcard*] *destination* [*destination-wildcard*] or **mac access-list extended** *access-list name* {**permit** | **deny**} {*source-MAC-addr mask* | **any** | **host**} {*destination-MAC-addr mask* | **any** | **host**} [*ethertype*]	Create an IP standard or extended ACL for IP traffic or a Layer 2 MAC ACL for non-IP traffic, repeating the command as many times as necessary. For more information, see the "Classifying Traffic by Using ACLs" section on page 30-37. **Note** When creating an access list, remember that, by default, the end of the access list contains an implicit deny statement for everything if it did not find a match before reaching the end.
Step 4	**class-map** [**match-all** | **match-any**] *class-map-name*	Create a class map to classify traffic as necessary. For more information, see the "Classifying Traffic on a Physical-Port Basis by Using Class Maps" section on page 30-40 and the "Classifying Traffic on a Per-Port Per-VLAN Basis by Using Class Maps" section on page 30-42.
Step 5	**mls qos cos policy-map**	(Optional) Define the CoS value of a port in a policy map. When you enter this command, you must also enter the **trust dscp** policy-map configuration command in Step 8 and the **set cos** *new-cos* policy-map configuration command in Step 9.
Step 6	**policy-map** *policy-map-name*	Create a policy map by entering the policy map name, and enter policy-map configuration mode. By default, no policy maps are defined. The default behavior of a policy map is to set the DSCP to 0 if the packet is an IP packet and to set the CoS to 0 if the packet is tagged. No policing is performed.
Step 7	**class** *class-map-name*	Define a traffic classification, and enter policy-map class configuration mode. By default, no policy map class-maps are defined. If a traffic class has already been defined by using the **class-map** global configuration command, specify its name for *class-map-name* in this command.

	Command	Purpose
Step 8	**trust** [**cos** | **dscp** | **ip-precedence**]	Configure the trust state, which selects the value that QoS uses as the source of the internal DSCP value. **Note** This command is mutually exclusive with the **set** command within the same policy map. If you enter the **trust** command, then skip Step 7. By default, the port is not trusted. If no keyword is specified when the command is entered, the default is **dscp**. The keywords have these meanings: • **cos**—QoS derives the internal DSCP value by using the received or default port CoS value and the CoS-to-DSCP map. • **dscp**—QoS derives the internal DSCP value by using the DSCP value from the ingress packet. For non-IP packets that are tagged, QoS derives the internal DSCP value by using the received CoS value; for non-IP packets that are untagged, QoS derives the internal DSCP value by using the default port CoS value. In either case, the internal DSCP value is derived from the CoS-to-DSCP map. **Note** If you use the **mls qos cos policy-map** global configuration command, you must use the **dscp** keyword. • **ip-precedence**—QoS derives the internal DSCP value by using the IP precedence value from the ingress packet and the IP-precedence-to-DSCP map. For non-IP packets that are tagged, QoS derives the internal DSCP value by using the received CoS value; for non-IP packets that are untagged, QoS derives the internal DSCP value by using the default port CoS value. In either case, the internal DSCP value is derived from the CoS-to-DSCP map. For more information, see the "Configuring the CoS-to-DSCP Map" section on page 30-54.
Step 9	**set** {**cos** *new-cos* | **dscp** *new-dscp* | **ip precedence** *new-precedence*}	Classify IP traffic by setting a new value in the packet. • For **cos** *new-cos*, enter a new CoS value to be assigned to the classified traffic. The range is 0 to 7. **Note** If you use the **mls qos cos policy-map** global configuration command, you must use the **cos** *new-cos* keyword. • For **dscp** *new-dscp*, enter a new DSCP value to be assigned to the classified traffic. The range is 0 to 63. • For **ip precedence** *new-precedence*, enter a new IP-precedence value to be assigned to the classified traffic. The range is 0 to 7.

	Command	Purpose
Step 10	**police** *rate-bps burst-byte* [**exceed-action** {**drop** | **policed-dscp-transmit**}]	Define a policer for the classified traffic. You can configure up to 128 policers on ingress Gigabit-capable Ethernet ports, up to 8 policers on ingress 10/100 Ethernet ports, and up to 8 policers on egress ports. • For *rate-bps*, specify average traffic rate in bits per second (bps). The range is 8000 to 1000000000. • For *burst-byte*, specify the normal burst size in bytes. The range is 8000 to 20000000. **Note** Although the command-line help strings show a large range of values, the *rate-bps* option cannot exceed the configured port speed, and the *burst-byte* option cannot exceed 2000000 bytes. If you enter a larger value, the switch rejects the policy map when you attach it to an interface. • (Optional) Specify the action to take when the rates are exceeded. Use the **exceed-action drop** keywords to drop the packet. Use the **exceed-action policed-dscp-transmit** keywords to mark down the DSCP value (by using the policed-DSCP map) and send the packet. For more information, see the "Configuring the Policed-DSCP Map" section on page 30-56.
Step 11	**exit**	Return to policy map configuration mode.
Step 12	**exit**	Return to global configuration mode.
Step 13	**interface** *interface-id*	Specify the interface to attach to the policy map, and enter interface configuration mode. Valid interfaces include physical interfaces.
Step 14	**service-policy** {**input** *policy-map-name* | **output** *policy-map-name*}	Apply a policy map to the input or output of a particular interface. Only one policy map per interface per direction is supported. • Use **input** *policy-map-name* to apply the specified policy-map to the input of an interface. • Use **output** *policy-map-name* to apply the specified policy-map to the output of an interface. You cannot use the **service-policy** interface configuration command to attach policy maps that contain these elements to an egress interface: • **set** or **trust** policy-map class configuration commands. Instead, you can use the **police** policy-map class configuration command to mark down (reduce) the DSCP value at the egress interface. • Access control list (ACL) classification. • Per-port per-VLAN classification. The only match criterion in a policy map that can be attached to an egress interface is the **match ip dscp** *dscp-list* class-map configuration command. Per-port per-VLAN policing is not supported on routed ports or on virtual (logical) interfaces.

	Command	Purpose
Step 15	**end**	Return to privileged EXEC mode.
Step 16	**show policy-map** [*policy-map-name* [**class** *class-name*]]	Verify your entries.
Step 17	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To delete an existing policy map, use the **no policy-map** *policy-map-name* global configuration command. To delete an existing class map, use the **no class** *class-map-name* policy-map configuration command. To use the DSCP-to-CoS map to define the CoS value, use the **no mls qos cos policy-map** global configuration command. To return to the default trust state, use the **no trust** [**cos** | **dscp** | **ip-precedence**] policy-map configuration command. To remove an assigned CoS, DSCP, or IP precedence value, use the **no set** {**cos** *new-cos* | **dscp** *new-dscp* | **ip precedence** *new-precedence*} policy-map configuration command. To remove an existing policer, use the **no police** *rate-bps burst-byte* [**exceed-action** {**drop** | **policed-dscp-transmit**}] policy-map configuration command. To remove the policy map and interface association, use the **no service-policy** {**input** *policy-map-name* | **output** *policy-map-name*} interface configuration command.

This example shows how to create a policy map and attach it to an ingress interface. In the configuration, the IP standard ACL permits traffic from network 10.1.0.0. For traffic matching this classification, the DSCP value in the incoming packet is trusted. If the matched traffic exceeds an average traffic rate of 48000 bps and a normal burst size of 8000 bytes, its DSCP is marked down (based on the policed-DSCP map) and sent:

```
Switch(config)# access-list 1 permit 10.1.0.0 0.0.255.255
Switch(config)# class-map ipclass1
Switch(config-cmap)# match access-group 1
Switch(config-cmap)# exit
Switch(config)# policy-map flow1t
Switch(config-pmap)# class ipclass1
Switch(config-pmap-c)# trust dscp
Switch(config-pmap-c)# police 48000 8000 exceed-action policed-dscp-transmit
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# service-policy input flow1t
```

This example shows how to create a policy map and attach it to an ingress interface. In the configuration, the IP extended ACL permits TCP traffic with an IP precedence of 4 from any host destined for the host at 224.0.0.5. For traffic matching this classification, the DSCP value in the incoming packet is set to 63.

```
Switch(config)# access-list 104 permit tcp any host 224.0.0.5 precedence 4
Switch(config)# class-map ipclass1
Switch(config-cmap)# match access-group 104
Switch(config-cmap)# exit
Switch(config)# policy-map ip104
Switch(config-pmap)# class ipclass1
Switch(config-pmap-c)# set dscp 63
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# service-policy input ip104
```

This example shows how to create a Layer 2 MAC ACL with two permit statements and attach it to an ingress interface. The first permit statement allows traffic from the host with MAC address 0001.0000.0001 destined for the host with MAC address 0002.0000.0001. The second permit statement allows only Ethertype XNS-IDP traffic from the host with MAC address 0001.0000.0002 destined for the host with MAC address 0002.0000.0002.

```
Switch(config)# mac access-list extended maclist1
Switch(config-ext-mac)# permit 0001.0000.0001 0.0.0 0002.0000.0001 0.0.0
Switch(config-ext-mac)# permit 0001.0000.0002 0.0.0 0002.0000.0002 0.0.0 xns-idp
Switch(config-ext-mac)# exit
Switch(config)# mac access-list extended maclist2
Switch(config-ext-mac)# permit 0001.0000.0003 0.0.0 0002.0000.0003 0.0.0
Switch(config-ext-mac)# permit 0001.0000.0004 0.0.0 0002.0000.0004 0.0.0 aarp
Switch(config-ext-mac)# exit
Switch(config)# class-map macclass1
Switch(config-cmap)# match access-group maclist1
Switch(config-cmap)# exit
Switch(config)# class-map macclass2
Switch(config-cmap)# match access-group maclist2
Switch(config-cmap)# exit
Switch(config)# policy-map macpolicy1
Switch(config-pmap)# class macclass1
Switch(config-pmap-c)# set dscp 63
Switch(config-pmap-c)# exit
Switch(config-pmap)# class macclass2
Switch(config-pmap-c)# set dscp 45
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# mls qos trust cos
Switch(config-if)# service-policy input macpolicy1
```

This example shows how to create a policy map that contains per-port per-VLAN classification and attach it to an ingress interface. A class map, called *vlan_class*, matches traffic received on VLANs 10, 20 to 30, and 40 that contains IP DSCP 9 (defined in class map *dscp_class*). If the specified average traffic rates and the burst sizes are exceeded, the switch drops the packet.

```
Switch(config)# class-map match-any dscp_class
Switch(config-cmap)# match ip dscp 9
Switch(config-cmap)# exit
Switch(config)# class-map match-all vlan_class
Switch(config-cmap)# match vlan 10 20-30 40
Switch(config-cmap)# match class-map dscp_class
Switch(config-cmap)# exit
Switch(config)# policy-map policymap2
Switch(config-pmap)# class vlan_class
Switch(config-pmap-c)# police 80000 8000 exceed-action drop
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# service-policy input policymap2
```

This example shows how to create a policy map that defines the CoS value for a port and how to attach it to an ingress interface. A class map, called *class1*, matches traffic received on VLANs 10, 20 to 30, and 40.

```
Switch (config)# mls qos cos policy-map
Switch (config)# class-map match-all class1
Switch (config-cmap)# match vlan 10 20-30 40
Switch (config-cmap)# match class-map some_class
Switch (config-cmap)# exit
Switch (config)# policy-map policymap1
Switch (config-pmap)# class class1
Switch (config-pmap-c)# trust dscp
Switch (config-pmap-c)# set cos 3
Switch (config-pmap-c)# exit
Switch (config-pmap)# exit
Switch (config)# interface gigabitethernet0/1
Switch(config-if)# service-policy input policymap1
```

Classifying, Policing, and Marking Traffic by Using Aggregate Policers

By using an aggregate policer, you can create a policer that is shared by multiple traffic classes within the same policy map. However, you cannot use the aggregate policer across different policy maps or interfaces.

Beginning in privileged EXEC mode, follow these steps to create an aggregate policer:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.

	Command	Purpose
Step 3	**mls qos aggregate-police** *aggregate-policer-name rate-bps burst-byte* **exceed-action** {**drop** | **policed-dscp-transmit**}	Define the policer parameters that can be applied to multiple traffic classes within the same policy map. By default, no aggregate policer is defined. You can configure up to 128 policers on ingress Gigabit-capable Ethernet ports, up to 8 policers on ingress 10/100 Ethernet ports, and up to 8 policers on egress ports. • For *aggregate-policer-name*, specify the name of the aggregate policer. • For *rate-bps*, specify average traffic rate in bits per second (bps). The range is 8000 to 1000000000. • For *burst-byte*, specify the normal burst size in bytes. The range is 8000 to 2000000. **Note** Although the command-line help strings show a large range of values, the *rate-bps* option cannot exceed the configured port speed, and the *burst-byte* option cannot exceed 2000000 bytes. If you enter a larger value, the switch rejects the policy map when you attach it to an interface. • (Optional) Specify the action to take when the rates are exceeded. Use the **exceed-action drop** keywords to drop the packet. Use the **exceed-action policed-dscp-transmit** keywords to mark down the DSCP value (by using the policed-DSCP map) and to send the packet. For more information, see the "Configuring the Policed-DSCP Map" section on page 30-56.
Step 4	**class-map** [**match-all** | **match-any**] *class-map-name*	Create a class map to classify traffic as necessary. For more information, see the "Classifying Traffic on a Physical-Port Basis by Using Class Maps" section on page 30-40 and the "Classifying Traffic on a Per-Port Per-VLAN Basis by Using Class Maps" section on page 30-42.
Step 5	**policy-map** *policy-map-name*	Create a policy map by entering the policy map name, and enter policy-map configuration mode. For more information, see the "Classifying, Policing, and Marking Traffic by Using Policy Maps" section on page 30-44.
Step 6	**class** *class-map-name*	Define a traffic classification, and enter policy-map class configuration mode. By default, no policy map class-maps are defined. If a traffic class has already been defined by using the **class-map** global configuration command, specify its name for *class-map-name* in this command.
Step 7	**police aggregate** *aggregate-policer-name*	Apply an aggregate policer to multiple classes in the same policy map. For *aggregate-policer-name*, enter the name specified in Step 3.
Step 8	**exit**	Return to global configuration mode.

	Command	Purpose
Step 9	**interface** *interface-id*	Specify the interface to attach to the policy map, and enter interface configuration mode. Valid interfaces include physical interfaces.
Step 10	**service-policy** {**input** *policy-map-name* | **output** *policy-map-name*}	Apply a policy map to the input or output of a particular interface. Only one policy map per interface per direction is supported. • Use **input** *policy-map-name* to apply the specified policy-map to the input of an interface. • Use **output** *policy-map-name* to apply the specified policy-map to the output of an interface. You cannot use the **service-policy** interface configuration command to attach policy maps that contain these elements to an egress interface: • **set** or **trust** policy-map class configuration commands. Instead, you can use the **police** policy-map class configuration command to mark down (reduce) the DSCP value at the egress interface. • Access control list (ACL) classification. • Per-port per-VLAN classification. The only match criterion in a policy map that can be attached to an egress interface is the **match ip dscp** *dscp-list* class-map configuration command. Per-port per-VLAN policing is not supported on routed ports or on virtual (logical) interfaces.
Step 11	**end**	Return to privileged EXEC mode.
Step 12	**show mls qos aggregate-policer** [*aggregate-policer-name*]	Verify your entries.
Step 13	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To remove the specified aggregate policer from a policy map, use the **no police aggregate** *aggregate-policer-name* policy map configuration mode. To delete an aggregate policer and its parameters, use the **no mls qos aggregate-policer** *aggregate-policer-name* global configuration command.

This example shows how to create an aggregate policer and attach it to multiple classes within a policy map. In the configuration, the IP ACLs permit traffic from network 10.1.0.0 and from host 11.3.1.1. For traffic coming from network 10.1.0.0, the DSCP in the incoming packets is trusted. For traffic coming from host 11.3.1.1, the DSCP in the packet is changed to 56. The traffic rate from the 10.1.0.0 network and from host 11.3.1.1 is policed. If the traffic exceeds an average rate of 48000 bps and a normal burst size of 8000 bytes, its DSCP is marked down (based on the policed-DSCP map) and sent. The policy map is attached to an ingress interface.

```
Switch(config)# access-list 1 permit 10.1.0.0 0.0.255.255
Switch(config)# access-list 2 permit 11.3.1.1
Switch(config)# mls qos aggregate-police transmit1 48000 8000 exceed-action
policed-dscp-transmit
Switch(config)# class-map ipclass1
Switch(config-cmap)# match access-group 1
Switch(config-cmap)# exit
Switch(config)# class-map ipclass2
Switch(config-cmap)# match access-group 2
Switch(config-cmap)# exit
Switch(config)# policy-map aggflow1
Switch(config-pmap)# class ipclass1
Switch(config-pmap-c)# trust dscp
Switch(config-pmap-c)# police aggregate transmit1
Switch(config-pmap-c)# exit
Switch(config-pmap)# class ipclass2
Switch(config-pmap-c)# set dscp 56
Switch(config-pmap-c)# police aggregate transmit1
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# service-policy input aggflow1
Switch(config-if)# exit
```

Configuring DSCP Maps

These sections describe how to configure the DSCP maps:

- Configuring the CoS-to-DSCP Map, page 30-54
- Configuring the IP-Precedence-to-DSCP Map, page 30-55
- Configuring the Policed-DSCP Map, page 30-56
- Configuring the DSCP-to-CoS Map, page 30-56
- Configuring the DSCP-to-DSCP-Mutation Map, page 30-58

All the maps, except the DSCP-to-DSCP-mutation map, are globally defined and are applied to all ports.You can have multiple DSCP-to-DSCP-mutation maps and apply them to different Gigabit-capable Ethernet ports. However, on 10/100 Ethernet ports, you can attach only one DSCP-to-DSCP-mutation map to a group of twelve ports.

Configuring the CoS-to-DSCP Map

You use the CoS-to-DSCP map to map CoS values in incoming packets to a DSCP value that QoS uses internally to represent the priority of the traffic.

Table 30-6 shows the default CoS-to-DSCP map.

Table 30-6 Default CoS-to-DSCP Map

CoS Value	DSCP Value
0	0
1	8
2	16
3	24
4	32
5	40
6	48
7	56

If these values are not appropriate for your network, you need to modify them.

Beginning in privileged EXEC mode, follow these steps to modify the CoS-to-DSCP map:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos map cos-dscp** *dscp1...dscp8*	Modify the CoS-to-DSCP map. For *dscp1...dscp8*, enter eight DSCP values that correspond to CoS values 0 to 7. Separate each DSCP value with a space. The range is 0 to 63.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show mls qos maps cos-dscp**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default map, use the **no mls qos map cos-dscp** global configuration command.

This example shows how to modify and display the CoS-to-DSCP map:

```
Switch# configure terminal
Switch(config)# mls qos map cos-dscp 10 15 20 25 30 35 40 45
Switch(config)# end
Switch# show mls qos maps cos-dscp

Cos-dscp map:
        cos:   0  1  2  3  4  5  6  7
     --------------------------------
       dscp:   10 15 20 25 30 35 40 45
```

Configuring the IP-Precedence-to-DSCP Map

You use the IP-precedence-to-DSCP map to map IP precedence values in incoming packets to a DSCP value that QoS uses internally to represent the priority of the traffic.

Table 30-7 shows the default IP-precedence-to-DSCP map:

Table 30-7 Default IP-Precedence-to-DSCP Map

IP Precedence Value	DSCP Value
0	0
1	8
2	16
3	24
4	32
5	40
6	48
7	56

If these values are not appropriate for your network, you need to modify them.

Beginning in privileged EXEC mode, follow these steps to modify the IP-precedence-to-DSCP map:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos map ip-prec-dscp** *dscp1...dscp8*	Modify the IP-precedence-to-DSCP map. For *dscp1...dscp8*, enter eight DSCP values that correspond to the IP precedence values 0 to 7. Separate each DSCP value with a space. The range is 0 to 63.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show mls qos maps ip-prec-dscp**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default map, use the **no mls qos map ip-prec-dscp** global configuration command.

This example shows how to modify and display the IP-precedence-to-DSCP map:

```
Switch# configure terminal
Switch(config)# mls qos map ip-prec-dscp 10 15 20 25 30 35 40 45
Switch(config)# end
Switch# show mls qos maps ip-prec-dscp

IpPrecedence-dscp map:
     ipprec:   0  1  2  3  4  5  6  7
     --------------------------------
       dscp:   10 15 20 25 30 35 40 45
```

Configuring the Policed-DSCP Map

You use the policed-DSCP map to mark down a DSCP value to a new value as the result of a policing and marking action.

The default policed-DSCP map is a null map, which maps an incoming DSCP value to the same DSCP value.

Beginning in privileged EXEC mode, follow these steps to modify the policed-DSCP map:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos map policed-dscp** *dscp-list* **to** *mark-down-dscp*	Modify the policed-DSCP map. • For *dscp-list*, enter up to eight DSCP values, with each value separated by a space. Then enter the **to** keyword. • For *mark-down-dscp*, enter the corresponding policed (marked down) DSCP value. The range is 0 to 63.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show mls qos maps policed-dscp**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default map, use the **no mls qos map policed-dscp** global configuration command.

This example shows how to map DSCP values 50 to 57 to a marked-down DSCP value of 0:

```
Switch# configure terminal
Switch(config)# mls qos map policed-dscp 50 51 52 53 54 55 56 57 to 0
Switch(config)# end
Switch# show mls qos maps policed-dscp
Policed-dscp map:
     d1 :  d2 0  1  2  3  4  5  6  7  8  9
     ---------------------------------------
      0 :    00 01 02 03 04 05 06 07 08 09
      1 :    10 11 12 13 14 15 16 17 18 19
      2 :    20 21 22 23 24 25 26 27 28 29
      3 :    30 31 32 33 34 35 36 37 38 39
      4 :    40 41 42 43 44 45 46 47 48 49
      5 :    00 00 00 00 00 00 00 00 58 59
      6 :    60 61 62 63
```

Note In the policed-DSCP map, the marked-down DSCP values are shown in the body of the matrix. The d1 column specifies the most-significant digit of the original DSCP; the d2 row specifies the least-significant digit of the original DSCP. The intersection of the d1 and d2 values gives the marked-down value. For example, an original DSCP value of 53 corresponds to a marked-down DSCP value of 0.

Configuring the DSCP-to-CoS Map

You use the DSCP-to-CoS map to generate a CoS value, which is used to select one of the four egress queues.

Table 30-8 shows the default DSCP-to-CoS map.

Table 30-8 Default DSCP-to-CoS Map

DSCP Value	CoS Value
0–7	0
8–15	1
16–23	2
24–31	3
32–39	4
40–47	5
48–55	6
56–63	7

If these values are not appropriate for your network, you need to modify them.

Beginning in privileged EXEC mode, follow these steps to modify the DSCP-to-CoS map:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos map dscp-cos** *dscp-list* **to** *cos*	Modify the DSCP-to-CoS map. • For *dscp-list*, enter up to eight DSCP values, with each value separated by a space. Then enter the **to** keyword. The range is 0 to 63. • For *cos*, enter the CoS value to which the DSCP values correspond. The range is 0 to 7.
Step 3	**end**	Return to privileged EXEC mode.
Step 4	**show mls qos maps dscp-to-cos**	Verify your entries.
Step 5	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default map, use the **no mls qos map dscp-cos** global configuration command.

This example shows how to map DSCP values 0, 8, 16, 24, 32, 40, 48, and 50 to CoS value 0:

```
Switch# configure terminal
Switch(config)# mls qos map dscp-cos 0 8 16 24 32 40 48 50 to 0
Switch(config)# end
Switch# show mls qos maps dscp-cos
Dscp-cos map:
     d1 :  d2 0  1  2  3  4  5  6  7  8  9
     ---------------------------------------
      0 :    00 00 00 00 00 00 00 00 00 01
      1 :    01 01 01 01 01 01 00 02 02 02
      2 :    02 02 02 02 00 03 03 03 03 03
      3 :    03 03 00 04 04 04 04 04 04 04
      4 :    00 05 05 05 05 05 05 05 00 06
      5 :    00 06 06 06 06 06 07 07 07 07
      6 :    07 07 07 07
```

Note In the DSCP-to-CoS map, the CoS values are shown in the body of the matrix. The d1 column specifies the most-significant digit of the DSCP; the d2 row specifies the least-significant digit of the DSCP. The intersection of the d1 and d2 values gives the CoS value. For example, in the DSCP-to-CoS map, a DSCP value of 08 corresponds to a CoS value of 0.

Configuring the DSCP-to-DSCP-Mutation Map

You apply the DSCP-to-DSCP-mutation map to a port at the boundary of a QoS administrative domain. If the two domains have different DSCP definitions between them, use the DSCP-to-DSCP-mutation map to translate a set of DSCP values to match the definition of the other domain.

The default DSCP-to-DSCP-mutation map is a null map, which maps an incoming DSCP value to the same DSCP value.

Beginning in privileged EXEC mode, follow these steps to modify the DSCP-to-DSCP-mutation map:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos map dscp-mutation** *dscp-mutation-name in-dscp* **to** *out-dscp*	Modify the DSCP-to-DSCP-mutation map. • For *dscp-mutation-name*, enter the mutation map name. You can create more than one map by specifying a new name. • For *in-dscp*, enter up to eight DSCP values, with each value separated by a space. Then enter the **to** keyword. • For *out-dscp*, enter a single DSCP value. The range is 0 to 63.
Step 3	**interface** *interface-id*	Specify the interface to which to attach the map, and enter interface configuration mode. Valid interfaces include physical interfaces.
Step 4	**mls qos trust dscp**	Configure the ingress port as a DSCP-trusted port.
Step 5	**mls qos dscp-mutation** *dscp-mutation-name*	Apply the map to the specified ingress DSCP-trusted port. For *dscp-mutation-name*, enter the mutation map name specified in Step 2. You can apply the map to different Gigabit-capable Ethernet ports. However, on 10/100 Ethernet ports, you can attach only one DSCP-to-DSCP-mutation map to a group of twelve ports. For example, Fast Ethernet ports 1 to 12 are a group, Fast Ethernet ports 13 to 24 are a group, Gigabit Ethernet port 1 is a group, and Gigabit Ethernet port 2 is a group. When applying a mutation map to any port in a group, all ports in the same group are automatically configured with the same map.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show mls qos maps dscp-mutation**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default map, use the **no mls qos map dscp-mutation** *dscp-mutation-name* global configuration command.

This example shows how to define the DSCP-to-DSCP-mutation map. All the entries that are not explicitly configured are not modified (remain as specified in the null map):

```
Switch# configure terminal
Switch(config)# mls qos map dscp-mutation mutation1 1 2 3 4 5 6 7 to 0
Switch(config)# mls qos map dscp-mutation mutation1 8 9 10 11 12 13 to 10
Switch(config)# mls qos map dscp-mutation mutation1 20 21 22 to 20
Switch(config)# mls qos map dscp-mutation mutation1 30 31 32 33 34 to 30
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# mls qos trust dscp
Switch(config-if)# mls qos dscp-mutation mutation1
Switch(config-if)# end
Switch# show mls qos maps dscp-mutation mutation1
Dscp-dscp mutation map:
   mutation1:
     d1 :  d2 0  1  2  3  4  5  6  7  8  9
     ---------------------------------------
      0 :    00 00 00 00 00 00 00 00 10 10
      1 :    10 10 10 10 14 15 16 17 18 19
      2 :    20 20 20 23 24 25 26 27 28 29
      3 :    30 30 30 30 30 35 36 37 38 39
      4 :    40 41 42 43 44 45 46 47 48 49
      5 :    50 51 52 53 54 55 56 57 58 59
      6 :    60 61 62 63
```

Note In the DSCP-to-DSCP-mutation map, the mutated values are shown in the body of the matrix. The d1 column specifies the most-significant digit of the original DSCP; the d2 row specifies the least-significant digit of the original DSCP. The intersection of the d1 and d2 values gives the mutated value. For example, a DSCP value of 12 corresponds to a mutated value of 10.

Configuring Egress Queues on Gigabit-Capable Ethernet Ports

This section describes how to configure the egress queues on Gigabit-capable Ethernet ports. For information on configuring 10/100 Ethernet ports, see "Configuring Egress Queues on 10/100 Ethernet Ports" section on page 30-66.

Depending on the complexity of your network and your QoS solution, you might need to perform all of the tasks in the next sections. You will need to make decisions about these characteristics:

- Which packets are assigned (by CoS value) to each queue?
- How much of the available buffer space (limit) is allotted to each queue?
- What drop percentage thresholds apply to each queue and which DSCP values map to each threshold?
- Is one of the queues the expedite (high-priority) egress queue?
- How much of the available bandwidth is allotted to each queue?

These sections contain this configuration information:

- Mapping CoS Values to Select Egress Queues, page 30-60
- Configuring the Egress Queue Size Ratios, page 30-61
- Configuring Tail-Drop Threshold Percentages, page 30-61
- Configuring WRED Drop Thresholds Percentages, page 30-63

- Configuring the Egress Expedite Queue, page 30-65
- Allocating Bandwidth among Egress Queues, page 30-65

Mapping CoS Values to Select Egress Queues

Beginning in privileged EXEC mode, follow these steps to map CoS ingress values to select one of the egress queues:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**interface** *interface-id*	Specify the egress Gigabit-capable Ethernet interface, and enter interface configuration mode.
Step 4	**wrr-queue cos-map** *queue-id cos1 ... cos8*	Map assigned CoS values to select one of the egress queues. The default map has these values: CoS value 0, 1 selects queue 1. CoS value 2, 3 selects queue 2. CoS value 4, 5 selects queue 3. CoS value 6, 7 selects queue 4. • For *queue-id*, specify the ID of the egress queue. The range is 1 to 4, where 4 can be configured as the expedite queue. For more information, see the "Configuring the Egress Expedite Queue" section on page 30-65. • For *cos1 ... cos8*, specify the CoS values that select a queue. Enter up to eight CoS values. Separate each value with a space. The range is 0 to 7.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show mls qos interface queueing**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return the default CoS-to-egress-queue map, use the **no wrr-queue cos-map** interface configuration command.

This example shows how to map CoS values 6 and 7 to queue 1, 4 and 5 to queue 2, 2 and 3 to queue 3, 0 and 1 to queue 4.

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# wrr-queue cos-map 1 6 7
Switch(config-if)# wrr-queue cos-map 2 4 5
Switch(config-if)# wrr-queue cos-map 3 2 3
Switch(config-if)# wrr-queue cos-map 4 0 1
```

Configuring the Egress Queue Size Ratios

Beginning in privileged EXEC mode, follow these steps to configure the egress queue size ratios:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**interface** *interface-id*	Specify the egress Gigabit-capable Ethernet interface, and enter interface configuration mode.
Step 4	**wrr-queue queue-limit** *weight1 weight2 weight3 weight4*	Configure the egress queue size ratios. The defaults weights are 25 (1/4 of the buffer size is allocated to each queue). For *weight1*, *weight2*, *weight3*, and *weight4*, specify a weight from 1 to 100. Separate each value with a space. The relative size difference in the numbers show the relative differences in the queue sizes. When you enter this command, the queue is temporarily shutdown during the hardware reconfiguration, and the switch drops newly arrived packets to this queue.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show mls qos interface buffers**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default weights, use the **no wrr-queue queue-limit** interface configuration command.

This example shows how to configure the size ratio of the four queues. The ratio of the size allocated for each queue is 1/10, 2/10, 3/10, and 4/10 for queues 1, 2, 3, and 4. (Queue 4 is four times larger than queue 1, twice as large as queue 2, and 1.33 times as large as queue 3.)

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# wrr-queue queue-limit 1 2 3 4
```

Configuring Tail-Drop Threshold Percentages

Tail drop is the default congestion-avoidance technique on Gigabit-capable Ethernet ports. With tail drop, packets are queued until the thresholds are exceeded. For example, all packets with DSCPs assigned to the first threshold are dropped until the threshold is no longer exceeded. However, packets assigned to a second threshold continue to be queued and sent as long as the second threshold is not exceeded.

You modify the DSCP-to-threshold map to determine which DSCPs are mapped to which threshold ID by using the **wrr-queue dscp-map** interface configuration command. By default, all DSCPs are mapped to threshold 1, and when this threshold is exceeded, all the packets are dropped.

If you use tail-drop thresholds, you cannot use WRED, and vice versa.

Beginning in privileged EXEC mode, follow these steps to configure the tail-drop threshold percentage values on Gigabit-capable Ethernet ports:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**interface** *interface-id*	Specify the egress Gigabit-capable Ethernet interface, and enter interface configuration mode.
Step 4	**wrr-queue threshold** *queue-id threshold-percentage1 threshold-percentage2*	Configure tail-drop threshold percentages on each egress queue. The default threshold is 100 percent for thresholds 1 and 2. • For *queue-id*, specify the ID of the egress queue. The range is 1 to 4. • For *threshold-percentage1 threshold-percentage2*, specify the tail-drop threshold percentage values. Separate each value with a space. The range is 1 to 100.
Step 5	**exit**	Return to global configuration mode.
Step 6	**interface** *interface-id*	Specify the ingress Gigabit-capable Ethernet interface, and enter interface configuration mode.
Step 7	**wrr-queue dscp-map** *threshold-id dscp1 ... dscp8*	Map DSCP values to the tail-drop thresholds of the egress queues. By default, all DSCP values are mapped to threshold 1. • For *threshold-id*, specify the threshold ID of the queue. The range is 1 to 2. • For *dscp1 ... dscp8*, specify the DSCP values that are mapped to the threshold ID. Enter up to eight DSCP values per command. Separate each value with a space. The range is 0 to 63.
Step 8	**end**	Return to privileged EXEC mode.
Step 9	**show running-config** or **show mls qos interface** *interface-id* **queueing**	Verify the DSCP-to-threshold map.
Step 10	**show mls qos interface buffers**	Verify the thresholds.
Step 11	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default thresholds, use the **no wrr-queue threshold** *queue-id* interface configuration command. To return to the default DSCP-to-threshold map, use the **no wrr-queue dscp-map** [*threshold-id*] interface configuration command.

This example shows how to configure the tail-drop queue threshold values for queue 1 to 10 percent and 100 percent, for queue 2 to 40 percent and 100 percent, for queue 3 to 60 percent and 100 percent, and for queue 4 to 80 percent and 100 percent on the egress interface (Gigabit Ethernet port 1). The ingress interface (Gigabit Ethernet port 2) is configured to trust the DSCP in the incoming packets, to map DSCPs 0, 8, 16, 24, 32, 40, 48, and 56 to threshold 1, and to map DSCPs 10, 20, 30, 40, 50, and 60 to threshold 2:

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# switchport mode access
Switch(config-if)# wrr-queue threshold 1 10 100
Switch(config-if)# wrr-queue threshold 2 40 100
Switch(config-if)# wrr-queue threshold 3 60 100
Switch(config-if)# wrr-queue threshold 4 80 100
Switch(config-if)# exit
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# mls qos trust dscp
Switch(config-if)# wrr-queue dscp-map 1 0 8 16 24 32 40 48 56
Switch(config-if)# wrr-queue dscp-map 2 10 20 30 40 50 60
```

As a result of this configuration, when queue 1 is filled above 10 percent, packets with DSCPs 0, 8, 16, 24, 32, 40, 48, and 56 are dropped. The same packets are dropped when queue 2 is filled above 40 percent, queue 3 above 60 percent, and queue 4 above 80 percent. When the second threshold (100 percent) is exceeded, all queues drop packets with DSCPs 10, 20, 30, 40, 50, and 60.

Configuring WRED Drop Thresholds Percentages

WRED reduces the chances of tail drop by selectively dropping packets when the output interface begins to show signs of congestion. By dropping some packets early rather than waiting until the queue is full, WRED avoids dropping large numbers of packets at once.

All packets with DSCPs assigned to the first threshold are randomly dropped when the first threshold is exceeded. However, packets with DSCPs assigned to the second threshold continue to be queued and sent as long as the second threshold is not exceeded. Each threshold percentage represents where WRED starts to randomly drop packets. By default, WRED is disabled.

If you use WRED, you cannot use tail-drop thresholds, and vice versa.

Beginning in privileged EXEC mode, follow these steps to configure the WRED drop threshold percentage values on Gigabit-capable Ethernet ports:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**interface** *interface-id*	Specify the egress Gigabit-capable Ethernet interface, and enter interface configuration mode.

	Command	Purpose
Step 4	**wrr-queue random-detect max-threshold** *queue-id threshold-percentage1 threshold-percentage2*	Configure WRED drop threshold percentages on each egress queue. The default, WRED is disabled, and no thresholds are configured. • For *queue-id*, specify the ID of the egress queue. The range is 1 to 4, where queue 4 can be configured as the expedite queue. For more information, see the "Configuring the Egress Expedite Queue" section on page 30-65. • For *threshold-percentage1 threshold-percentage2*, specify the threshold percentage values. Separate each value with a space. The range is 1 to 100.
Step 5	**exit**	Return to global configuration mode.
Step 6	**interface** *interface-id*	Specify the ingress Gigabit-capable Ethernet interface, and enter interface configuration mode.
Step 7	**wrr-queue dscp-map** *threshold-id dscp1 ... dscp8*	Map DSCP values to the WRED drop thresholds of the egress queues. By default, all DSCP values are mapped to threshold 1. • For *threshold-id*, specify the threshold ID of the queue. The range is 1 to 2. • For *dscp1 ... dscp8*, specify the DSCP values that are mapped to the threshold ID. Enter up to eight DSCP values per command. Separate each value with a space. The range is 0 to 63.
Step 8	**show running-config** or **show mls qos interface** *interface-id* **queueing**	Verify the DSCP-to-threshold map.
Step 9	**show mls qos interface buffers**	Verify the thresholds.
Step 10	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable WRED, use the **no wrr-queue random-detect max-threshold** *queue-id* interface configuration command. To return to the default DSCP-to-threshold map, use the **no wrr-queue dscp-map** [*threshold-id*] interface configuration command.

This example shows how to configure the WRED queue threshold values for queue 1 to 50 percent and 100 percent, for queue 2 to 70 percent and 100 percent, for queue 3 to 50 percent and 100 percent, and for queue 4 to 70 percent and 100 percent on the egress interface (Gigabit Ethernet port 1). The ingress interface (Gigabit Ethernet port 2) is configured to trust the DSCP in the incoming packets, to map DSCPs 0, 8, 16, 24, 32, 40, 48, and 56 to threshold 1, and to map DSCPs 10, 20, 30, 40, 50, and 60 to threshold 2.

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# wrr-queue random-detect max-threshold 1 50 100
Switch(config-if)# wrr-queue random-detect max-threshold 2 70 100
Switch(config-if)# wrr-queue random-detect max-threshold 3 50 100
Switch(config-if)# wrr-queue random-detect max-threshold 4 70 100
Switch(config-if)# exit
Switch(config)# interface gigabitethernet0/2
Switch(config-if)# mls qos trust dscp
Switch(config-if)# wrr-queue dscp-map 1 0 8 16 24 32 40 48 56
Switch(config-if)# wrr-queue dscp-map 2 10 20 30 40 50 60
```

As a result of this configuration, when the queues 1 and 3 are filled above 50 percent, packets with DSCPs 0, 8, 16, 24, 32, 40, 48, and 56 are randomly dropped. The same packets are randomly dropped when queues 2 and 4 are filled above 70 percent. When the second threshold (100 percent) is exceeded, all queues randomly drop packets with DSCPs 10, 20, 30, 40, 50, and 60.

Configuring the Egress Expedite Queue

You can ensure that certain packets have priority over all others by queuing them in the egress expedite queue. WRR services this queue until it is empty before servicing the other queues.

Beginning in privileged EXEC mode, follow these steps to enable the egress expedite queue:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**interface** *interface-id*	Specify the egress Gigabit-capable Ethernet interface, and enter interface configuration mode.
Step 4	**priority-queue out**	Enable the egress expedite queue, which is disabled by default. When you configure this command, the WRR weight and queue size ratios are affected because there is one fewer queue participating in WRR. This means that *weight4* in the **wrr-queue bandwidth** command is ignored (not used in the ratio calculation).
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the egress expedite queue, use the **no priority-queue out** interface configuration command.

Allocating Bandwidth among Egress Queues

You need to specify how much of the available bandwidth is allocated to each queue. The ratio of the weights is the ratio of frequency in which the WRR scheduler dequeues packets from each queue.

Beginning in privileged EXEC mode, follow these steps to allocate bandwidth to each queue:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**interface** *interface-id*	Specify the egress Gigabit-capable Ethernet interface, and enter interface configuration mode.

	Command	Purpose
Step 4	**wrr-queue bandwidth** *weight1 weight2 weight3 weight4*	Assign WRR weights to the egress queues. By default, all the weights are set to 25 (1/4 of the bandwidth is allocated to each queue). For *weight1 weight2 weight3 weight4*, enter the ratio, which determines the ratio of the frequency in which the WRR scheduler dequeues packets. Separate each value with a space. The range is 1 to 65536. All four queues participate in the WRR unless the expedite queue (queue 4) is enabled, in which case *weight4* is ignored (not used in the ratio calculation). The expedite queue is a priority queue, and it is serviced until empty before the other queues are serviced. A weight of 1 means that the minimum bandwidth is allocated for that queue. To allocate no bandwidth for a queue, use the **wrr-queue cos-map** interface configuration command so that the available bandwidth is shared among the remaining queues.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show mls qos interface queueing**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default bandwidth setting, use the **no wrr-queue bandwidth** interface configuration command.

This example shows how to configure the weight ratio of the WRR scheduler running on the egress queues. In this example, four queues are used (no expedite queue), and the ratio of the bandwidth allocated for each queue is 1/(1+2+3+4), 2/(1+2+3+4), 3/(1+2+3+4), and 4/(1+2+3+4), which is 1/10, 1/5, 3/10, and 2/5 for queues 1, 2, 3, and 4.

```
Switch(config)# interface gigabitethernet0/1
Switch(config-if)# wrr-queue bandwidth 1 2 3 4
```

Configuring Egress Queues on 10/100 Ethernet Ports

This section describes how to configure the egress queues on 10/100 Ethernet ports. For information on configuring Gigabit-capable Ethernet ports, see the "Configuring Egress Queues on Gigabit-Capable Ethernet Ports" section on page 30-59.

Depending on the complexity of your network and your QoS solution, you might need to perform all of the tasks in the next sections. You will need to make decisions about these characteristics:

- Which packets are assigned (by CoS value) to each queue?
- How much of the available buffer space is allotted to each queue?
- Is one of the queues the expedite (high-priority) egress queue?
- How much of the available bandwidth is allotted to each queue?

These sections contain this configuration information:

- Mapping CoS Values to Select Egress Queues, page 30-67
- Configuring the Minimum-Reserve Levels, page 30-68

- Configuring the Egress Expedite Queue, page 30-69
- Allocating Bandwidth among Egress Queues, page 30-69

Mapping CoS Values to Select Egress Queues

Beginning in privileged EXEC mode, follow these steps to map CoS ingress values to select one of the egress queues:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**interface** *interface-id*	Specify the egress 10/100 Ethernet interface, and enter interface configuration mode.
Step 4	**wrr-queue cos-map** *queue-id cos1 ... cos8*	Map assigned CoS values to select one of the egress queues. Theses are the default map values: CoS value 0, 1 selects queue 1. CoS value 2, 3 selects queue 2. CoS value 4, 5 selects queue 3. CoS value 6, 7 selects queue 4. • For *queue-id*, specify the ID of the egress queue. The range is 1 to 4, where 4 can be configured as the expedite queue. For more information, see the "Configuring the Egress Expedite Queue" section on page 30-69. • For *cos1 ... cos8*, specify the CoS values that select a queue. Enter up to eight CoS values. Separate each value with a space. The range is 0 to 7.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show mls qos interface queueing**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default CoS-to-egress-queue map, use the **no wrr-queue cos-map** interface configuration command.

This example shows how to map CoS values 6 and 7 to queue 1, 4 and 5 to queue 2, 2 and 3 to queue 3, and 0 and 1 to queue 4.

```
Switch(config)# interface fastethernet0/1
Switch(config-if)# wrr-queue cos-map 1 6 7
Switch(config-if)# wrr-queue cos-map 2 4 5
Switch(config-if)# wrr-queue cos-map 3 2 3
Switch(config-if)# wrr-queue cos-map 4 0 1
```

Configuring the Minimum-Reserve Levels

You can configure the buffer size of the minimum-reserve levels on all 10/100 ports and assign the minimum-reserve level to an egress queue on a 10/100 Ethernet port.

Beginning in privileged EXEC mode, follow these steps to configure the egress queue sizes:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**mls qos min-reserve** *min-reserve-level min-reserve-buffersize*	Configure the buffer size of the minimum-reserve level, if necessary, for all the 10/100 Ethernet ports. By default, the buffer size for all eight minimum-reserve levels is 100 packets. • For *min-reserve-level*, specify the minimum-reserve level number. The range is 1 to 8. • For *min-reserve-buffersize*, specify the buffer size. The range is 10 to 170 packets. When you enter this command, the queue is temporarily shutdown during the hardware reconfiguration, and the switch drops newly arrived packets to this queue.
Step 4	**interface** *interface-id*	Specify the egress 10/100 Ethernet interface, and enter interface configuration mode.
Step 5	**wrr-queue min-reserve** *queue-id min-reserve-level*	Assign a minimum-reserve level number to a particular egress queue. By default, queue 1 selects minimum-reserve level 1, queue 2 selects minimum-reserve level 2, queue 3 selects minimum-reserve level 3, and queue 4 selects minimum-reserve level 4. • For *queue-id*, specify the ID of the egress queue. The range is 1 to 4, where 4 can be configured as the expedite queue. For more information, see the "Configuring the Egress Expedite Queue" section on page 30-69. • For *min-reserve-level*, specify the minimum-reserve level configured in Step 3.
Step 6	**end**	Return to privileged EXEC mode.
Step 7	**show mls qos interface buffers**	Verify your entries.
Step 8	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default minimum-reserve buffer size, use the **no mls qos min-reserve** *min-reserve-level* global configuration command. To return to the default queue selection of the minimum-reserve level, use the **no wrr-queue min-reserve** *queue-id* interface configuration command.

This example shows how to configure minimum-reserve level 5 to 20 packets and to assign minimum-reserve level 5 to egress queue 1 on an interface:

```
Switch(config)# mls qos min-reserve 5 20
Switch(config)# interface fastethernet0/1
Switch(config-if)# wrr-queue min-reserve 1 5
```

Configuring the Egress Expedite Queue

You can ensure that certain packets have priority over all others by queuing them in the egress expedite queue. WRR services this queue until it is empty before servicing the other queues.

Beginning in privileged EXEC mode, follow these steps to enable the egress expedite queue:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**interface** *interface-id*	Specify the egress 10/100 Ethernet interface, and enter interface configuration mode.
Step 4	**priority-queue out**	Enable the egress expedite queue, which is disabled by default. When you configure this command, the WRR weight is affected because there is one fewer queue participating in WRR. This means that *weight4* in the **wrr-queue bandwidth** command is ignored (not used in the ratio calculation).
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show running-config**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To disable the egress expedite queue, use the **no priority-queue out** interface configuration command.

Allocating Bandwidth among Egress Queues

You need to specify how much of the available bandwidth is allocated to each queue. The ratio of the weights is the ratio of frequency in which the WRR scheduler dequeues packets from each queue.

Beginning in privileged EXEC mode, follow these steps to allocate bandwidth to each queue:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**interface** *interface-id*	Specify the egress 10/100 Ethernet interface, and enter interface configuration mode.

	Command	Purpose
Step 4	**wrr-queue bandwidth** *weight1 weight2 weight3 weight4*	Assign WRR weights to the egress queues. By default, all the weights are set to 25 (1/4 of the bandwidth is allocated to each queue). For *weight1 weight2 weight3 weight4*, enter the ratio, which determines the ratio of the frequency in which the WRR scheduler dequeues packets. Separate each value with a space. The range is 1 to 65536. All four queues participate in the WRR unless the expedite queue (queue 4) is enabled, in which case *weight4* is ignored (not used in the ratio calculation). The expedite queue is a priority queue, and it is serviced until empty before the other queues are serviced. A weight of 1 means that the minimum bandwidth is allocated for that queue. To allocate no bandwidth for a queue, use the **wrr-queue cos-map** interface configuration command so that the available bandwidth is shared among the remaining queues.
Step 5	**end**	Return to privileged EXEC mode.
Step 6	**show mls qos interface queueing**	Verify your entries.
Step 7	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

To return to the default bandwidth setting, use the **no wrr-queue bandwidth** interface configuration command.

This example shows how to configure the weight ratio of the WRR scheduler running on the egress queues. In this example, four queues are used (no expedite queue), and the ratio of the bandwidth allocated for each queue is 1/(1+2+3+4), 2/(1+2+3+4), 3/(1+2+3+4), and 4/(1+2+3+4), which is 1/10, 2/10, 3/10, and 4/10 for queues 1, 2, 3, and 4.

```
Switch(config)# interface fastethernet0/1
Switch(config-if)# wrr-queue bandwidth 1 2 3 4
```

Displaying Standard QoS Information

To display standard QoS information, use one or more of the privileged EXEC commands in Table 30-9:

Table 30-9 Commands for Displaying Standard QoS Information

Command	Purpose
show class-map [*class-map-name*]	Display QoS class maps, which define the match criteria to classify traffic.
show mls qos aggregate-policer [*aggregate-policer-name*]	Display the aggregate policer configuration.
show mls qos interface [*interface-id*] [**buffers** | **policers** | **queueing** | **statistics**]	Display QoS information at the interface level, including the configuration of the egress queues and the CoS-to-egress-queue map, which interfaces have configured policers, and ingress and egress statistics (including the number of bytes dropped).[1]
show mls qos maps [**cos-dscp** | **dscp-cos** | **dscp-mutation** | **ip-prec-dscp** | **policed-dscp**]	Display QoS mapping information. Maps are used to generate an internal DSCP value, which represents the priority of the traffic.
show policy-map [*policy-map-name* [**class** *class-map-name*]	Display QoS policy maps, which define classification criteria for incoming traffic.

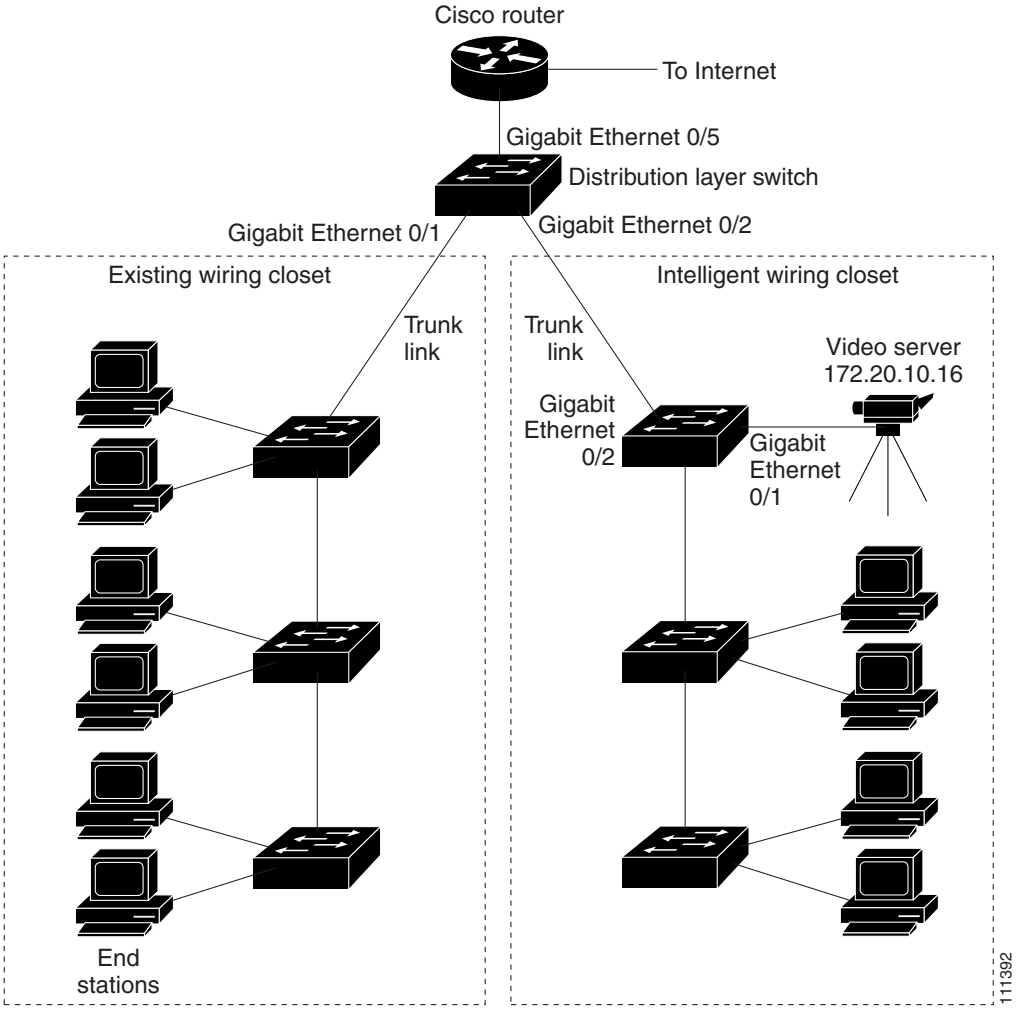
1. You can define up to 16 DSCP values for which byte or packet statistics are gathered by hardware by using the **mls qos monitor** {**bytes** \| **dscp** *dscp1 ... dscp8* \| **packets**} interface configuration command and the **show mls qos interface statistics** privileged EXEC command.

Standard QoS Configuration Examples

This section shows a QoS migration path to help you quickly implement QoS features based on your existing network and planned changes to your network, as shown in Figure 30-11. It contains this information:

- QoS Configuration for the Existing Wiring Closet, page 30-72
- QoS Configuration for the Intelligent Wiring Closet, page 30-73
- QoS Configuration for the Distribution Layer, page 30-74

Figure 30-11 QoS Configuration Example Network



QoS Configuration for the Existing Wiring Closet

Figure 30-11 shows an existing wiring closet with Catalyst 3500 XL and 2900 XL switches, for example. These switches are running Cisco IOS Release 12.0(5)XP or later, which supports the QoS-based IEEE 802.1p CoS values. QoS classifies frames by assigning priority-indexed CoS values to them and gives preference to higher-priority traffic.

Recall that on the Catalyst 3500 XL and 2900 XL switches, you can classify untagged (native) Ethernet frames at the ingress ports by setting a default CoS priority (**switchport priority default** *default-priority-id* interface configuration command) for each port. For ISL or IEEE 802.1Q frames with tag information, the priority value from the header frame is used. On the Catalyst 3524-PWR XL and 3548 XL switches, you can override this priority with the default value by using the **switchport priority default override** interface configuration command. For Catalyst 3500 XL, 2950, other 2900 XL models that do not have the override feature, the Catalyst 3550-12T switch at the distribution layer can override the IEEE 802.1p CoS value by using the **mls qos cos override** interface configuration command.

For the Catalyst 3500 XL and 2900 XL switches, CoS configures each egress port with a normal-priority transmit queue and a high-priority transmit queue, depending on the frame tag or the port information. Frames in the normal-priority queue are forwarded only after frames in the high-priority queue are forwarded. Frames that have IEEE 802.1p CoS values of 0 to 3 are placed in the normal-priority transmit queue whereas frames with CoS values of 4 to 7 are placed in the expedite (high-priority) queue.

QoS Configuration for the Intelligent Wiring Closet

Figure 30-11 shows an intelligent wiring closet with Catalyst 3550 multilayer switches, for example. One of the switches is connected to a video server, which has an IP address of 172.20.10.16.

The object of this example is to prioritize the video traffic over all other traffic. To do so, a DSCP of 56 is assigned to the video traffic. This traffic is stored in the expedite queue (queue 4), which is serviced until empty before the other queues are serviced. The appropriate CoS value selects queue 4 in the CoS-to-egress-queue map.

Beginning in privileged EXEC mode, follow these steps to configure the switch to prioritize video packets over all other traffic:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**access-list 1 permit 172.20.10.16**	Define an IP standard ACL, and permit traffic from the video server at 172.20.10.16.
Step 3	**class-map videoclass**	Create a class map called *videoclass*, and enter class-map configuration mode.
Step 4	**match access-group 1**	Define the match criterion by matching the traffic specified by access list 1.
Step 5	**exit**	Return to global configuration mode.
Step 6	**policy-map videopolicy**	Create a policy map called *videopolicy*, and enter policy-map configuration mode.
Step 7	**class videoclass**	Specify the class on which to act, and enter policy-map class configuration mode.
Step 8	**set dscp 56**	For traffic matching ACL 1, set the DSCP of incoming packets to 56.
Step 9	**police 5000000 2000000 exceed-action drop**	Define a policer for the classified video traffic to drop traffic that exceeds 5-Mbps average traffic rate with a 2-MB burst size.
Step 10	**exit**	Return to policy-map configuration mode.
Step 11	**exit**	Return to global configuration mode.
Step 12	**interface** *interface-id*	Specify the switch ingress interface that is connected to the video server, and enter interface configuration mode.
Step 13	**service-policy input videopolicy**	Apply the policy to the ingress interface.
Step 14	**exit**	Return to global configuration mode.
Step 15	**interface gigabitethernet0/2**	Enter interface configuration mode, and specify the egress interface (to configure the queues).
Step 16	**priority-queue out**	Enable the expedite queue.

	Command	Purpose
Step 17	**wrr-queue cos-map 4 6 7**	Configure the CoS-to-egress-queue map so that CoS values 6 and 7 select queue 4 (this is the default setting). Because the default DSCP-to-CoS map has DSCP values 56 to 63 mapped to CoS value 7, the matched traffic that is set to DSCP 56 goes to the queue 4, the priority queue.
Step 18	**end**	Return to privileged EXEC mode.
Step 19	**show class-map videoclass** **show policy-map videopolicy** **show mls qos maps** [**cos-dscp** | **dscp-cos**] **show mls qos interface queueing**	Verify your entries.
Step 20	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

QoS Configuration for the Distribution Layer

Figure 30-11 shows a distribution layer switch, for example, a Catalyst 3550 switch. This example focuses on the configuration steps for the distribution layer switch. Because the classification was performed by the switches at the edge of the network, fewer classification steps are needed at the distribution layer switch.

For the connection to the existing wiring closet, Gigabit Ethernet interface 1 on the distribution layer switch is configured to trust the received CoS value. In this situation, the default CoS-to-DSCP map on the multilayer switch is sufficient. For information on the default map settings, see the "Configuring the CoS-to-DSCP Map" section on page 30-54.

For the connection to the intelligent wiring closet, Gigabit Ethernet interface 2 on the distribution layer switch is configured to trust the received DSCP value. The DSCP-to-threshold map also needs to be configured on this ingress interface so that on the egress interface, WRED can provide congestion avoidance control. By default, all DSCP values are mapped to threshold 1.

You need to configure several of the switch maps from their default settings. The object of the configuration is to have only DSCP value 56 sent to the expedite queue (queue 4). The default CoS-to-egress-queue map is sufficient; however, you need to configure the DSCP-to-CoS map so that DSCP values 57 to 63 map to CoS 5.

For the egress interface, Gigabit Ethernet interface 5, WRR weights need to be configured by using the **wrr-queue bandwidth** interface configuration command. WRED needs to be enabled and the threshold percentages configured for each queue. The bandwidth allocated to each queue must be configured to determine the ratio of the frequency at which packets are dequeued.

Beginning in privileged EXEC mode, follow these steps to configure the switch at the distribution layer:

	Command	Purpose
Step 1	**configure terminal**	Enter global configuration mode.
Step 2	**mls qos**	Enable QoS on the switch.
Step 3	**interface** *interface-id*	Specify the ingress interface that is connected to the existing wiring closet, and enter interface configuration mode.
Step 4	**mls qos trust cos**	Classify incoming packets on this port by using the packet CoS value.

	Command	Purpose
Step 5	**switchport mode trunk**	Configure this port as a trunk port.
Step 6	**exit**	Return to global configuration mode.
Step 7	**interface** *interface-id*	Specify the ingress interface connected to the intelligent wiring closet, and enter interface configuration mode.
Step 8	**mls qos trust dscp**	Classify incoming packets on this port by trusting the packet DSCP value.
Step 9	**wrr-queue dscp-map** *threshold-id dscp1 ... dscp8*	Map the ingress DSCP values to the WRED thresholds of the egress queues. In the default DSCP-to-threshold map, all DSCP values are mapped to threshold 1. • For *threshold-id*, specify the threshold ID of the queue. The range is 1 to 2. • For *dscp1 ... dscp8*, specify the DSCP values that are mapped to a threshold ID. Enter up to eight DSCP values per command. Separate each value with a space. The DSCP range is 0 to 63.
Step 10	**switchport mode trunk**	Configure this port as a trunk port.
Step 11	**exit**	Return to global configuration mode.
Step 12	**mls qos map dscp-cos** *dscp-list* **to** *cos*	Modify the DSCP-to-CoS map. You can enter up to eight DSCP values separated by spaces in the DSCP-to-CoS map. For example, to map DSCP values 57 to 63 to CoS 5, enter: **mls qos map dscp-cos 57 58 59 60 61 62 63 to 5**
Step 13	**interface** *interface-id*	Specify the egress interface connected to the upstream router, and enter interface configuration mode.
Step 14	**priority-queue out**	Enable the expedite queue.
Step 15	**wrr-queue bandwidth** *weight1 weight2 weight3 weight4*	Configure WRR weights to the egress queues to determine the ratio of the frequency at which packets are dequeued. Separate each value with a space. The weight range is 0 to 65536. In this example, to configure the weights so that queue 4 is serviced more frequently than the other queues, enter: **wrr-queue bandwidth 1 2 3 4** Because the expedite queue is enabled, only the first three weights are used in the ratio calculation.
Step 16	**wrr-queue random-detect max-threshold** *queue-id threshold-percentage1 threshold-percentage2*	Enable WRED and assign two WRED threshold values to an egress queue of a Gigabit-capable Ethernet port. • For *queue-id*, the range is 1 to 4. • For *threshold-percentage1 threshold-percentage2*, the range is 1 to 100 percent. In this example, to configure the thresholds, enter: **wrr-queue random-detect max-threshold 1 20 100** **wrr-queue random-detect max-threshold 2 40 100** **wrr-queue random-detect max-threshold 3 60 100** **wrr-queue random-detect max-threshold 4 80 100**

	Command	Purpose
Step 17	**end**	Return to privileged EXEC mode.
Step 18	**show mls qos interface** and **show interfaces**	Verify your entries.
Step 19	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.