

# Traffic Management User Guide for NFX Series Devices

Published  
2025-12-12

Juniper Networks, Inc.  
1133 Innovation Way  
Sunnyvale, California 94089  
USA  
408-745-2000  
[www.juniper.net](http://www.juniper.net)

Juniper Networks, the Juniper Networks logo, Juniper, and Junos are registered trademarks of Juniper Networks, Inc. in the United States and other countries. All other trademarks, service marks, registered marks, or registered service marks are the property of their respective owners.

Juniper Networks assumes no responsibility for any inaccuracies in this document. Juniper Networks reserves the right to change, modify, transfer, or otherwise revise this publication without notice.

*Traffic Management User Guide for NFX Series Devices*  
Copyright © 2025 Juniper Networks, Inc. All rights reserved.

The information in this document is current as of the date on the title page.

## YEAR 2000 NOTICE

Juniper Networks hardware and software products are Year 2000 compliant. Junos OS has no known time-related limitations through the year 2038. However, the NTP application is known to have some difficulty in the year 2036.

## END USER LICENSE AGREEMENT

The Juniper Networks product that is the subject of this technical documentation consists of (or is intended for use with) Juniper Networks software. Use of such software is subject to the terms and conditions of the End User License Agreement ("EULA") posted at <https://support.juniper.net/support/eula/>. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

# Table of Contents

**About This Guide | vi**

1

## **CoS Overview**

**Basic Concepts | 2**

Overview of Junos OS CoS | 2

Configuring CoS | 5

Understanding Junos CoS Components | 11

Assigning CoS Components to Interfaces | 16

Monitoring Interfaces That Have CoS Components | 18

Understanding CoS Packet Flow | 20

Understanding Default CoS Settings | 24

CoS Inputs and Outputs Overview | 38

Overview of Policers | 39

2

## **Classifying and Rewriting Traffic**

**Using Classifiers, Forwarding Classes, and Rewrite Rules | 51**

Understanding CoS Classifiers | 52

Defining CoS BA Classifiers (DSCP, DSCP IPv6, IEEE 802.1p) | 61

Example: Configuring Classifiers | 63

Requirements | 64

Overview | 65

Verification | 66

Monitoring CoS Classifiers | 68

Understanding Default CoS Scheduling and Classification | 69

Understanding Applying CoS Classifiers and Rewrite Rules to Interfaces | 80

Understanding CoS Code-Point Aliases | 93

Defining CoS Code-Point Aliases	96
Monitoring CoS Code-Point Value Aliases	97
Understanding CoS Forwarding Classes	99
Defining CoS Forwarding Classes	105
Example: Configuring Forwarding Classes	108

Requirements	108
Overview	109
Example 1: Configuring Forwarding Classes for Switches Except QFX10000	110
Verification	111
Example 2: Configuring Forwarding Classes for QFX10000 Switches	112
Verification	113

Monitoring CoS Forwarding Classes	115
Understanding CoS Rewrite Rules	118
Defining CoS Rewrite Rules	121
Troubleshooting an Unexpected Rewrite Value	123
Monitoring CoS Rewrite Rules	126

## 3

## Scheduling Traffic

Using Schedulers	129
Understanding CoS Scheduling Behavior and Configuration Considerations	129
Defining CoS Queue Schedulers for Port Scheduling	135
Defining CoS Queue Scheduling Priority	139
Example: Configuring Queue Scheduling Priority	141
Requirements	142
Overview	142
Verification	144
Monitoring CoS Scheduler Maps	147
Understanding CoS Traffic Control Profiles	149
Understanding CoS Priority Group Scheduling	151

Defining CoS Traffic Control Profiles (Priority Group Scheduling) | 154

Example: Configuring Traffic Control Profiles (Priority Group Scheduling) | 156

Requirements | 156

Overview | 156

Verification | 159

Understanding CoS Priority Group and Queue Guaranteed Minimum Bandwidth | 160

Example: Configuring Minimum Guaranteed Output Bandwidth | 163

Requirements | 165

Overview | 165

Verification | 167

Understanding CoS Priority Group Shaping and Queue Shaping (Maximum Bandwidth) | 170

Example: Configuring Maximum Output Bandwidth | 173

Requirements | 174

Overview | 175

Verification | 176

Understanding CoS Explicit Congestion Notification | 179

## Configuration Statements and Operational Commands

Junos CLI Reference Overview | 190

# About This Guide

Use this guide to understand and configure class of service (CoS) features in Junos OS to define service levels that provide different delay, jitter, and packet loss characteristics to particular applications served by specific traffic flows. Applying CoS features to each device in your network ensures quality of service (QoS) for traffic throughout your entire network.

# 1

PART

## CoS Overview

---

- [Basic Concepts | 2](#)
-

## CHAPTER 1

# Basic Concepts

**IN THIS CHAPTER**

- Overview of Junos OS CoS | 2
- Configuring CoS | 5
- Understanding Junos CoS Components | 11
- Assigning CoS Components to Interfaces | 16
- Monitoring Interfaces That Have CoS Components | 18
- Understanding CoS Packet Flow | 20
- Understanding Default CoS Settings | 24
- CoS Inputs and Outputs Overview | 38
- Overview of Policers | 39

## Overview of Junos OS CoS

**IN THIS SECTION**

- CoS Standards | 3
- How Junos OS CoS Works | 4
- Default CoS Behavior | 5

When a network experiences congestion and delay, some packets must be dropped. Junos OS *class of service* (CoS) enables you to divide traffic into classes and set various levels of throughput and packet loss when congestion occurs. You have greater control over packet loss because you can configure rules tailored to your needs.

You can configure CoS features to provide multiple classes of service for different applications. CoS also allows you to rewrite the Differentiated Services code point (DSCP) or IEEE 802.1p code-point bits of



packets leaving an interface, thus allowing you to tailor packets for the network requirements of the remote peers.

CoS provides multiple classes of service for different applications. You can configure multiple forwarding classes for transmitting packets, define which packets are placed into each output queue, schedule the transmission service level for each queue, and manage congestion using a weighted random early detection (WRED) algorithm.

In designing CoS applications, you must carefully consider your service needs, and you must thoroughly plan and design your CoS configuration to ensure consistency and interoperability across all platforms in a CoS domain.

Because CoS is implemented in hardware rather than in software, you can experiment with and deploy CoS features without affecting packet forwarding and switching performance.



**NOTE:** CoS policies can be enabled or disabled on each switch interface. Also, each physical and *logical interface* on the switch can have associated custom CoS rules. When you change or when you deactivate and then reactivate the class-of-service configuration, the system experiences packet drops because the system momentarily blocks traffic to change the mapping of incoming traffic to input queues.

This topic describes:

## CoS Standards

The following RFCs define the standards for CoS capabilities:

- RFC 2474, *Definition of the Differentiated Services Field in the IPv4 and IPv6 Headers*
- RFC 2597, *Assured Forwarding PHB Group*
- RFC 2598, *An Expedited Forwarding PHB*
- RFC 2698, *A Two Rate Three Color Marker*
- RFC 3168, *The Addition of Explicit Congestion Notification (ECN) to IP*

The following data center bridging (DCB) standards are also supported to provide the CoS (and other characteristics) that Fibre Channel over Ethernet (FCoE) requires for transmitting storage traffic over an Ethernet network:

- IEEE 802.1Qbb, *priority-based flow control* (PFC)
- IEEE 802.1Qaz, enhanced transmission selection (ETS)
- IEEE 802.1AB (LLDP) extension called Data Center Bridging Capability Exchange Protocol (DCBX)

## How Junos OS CoS Works

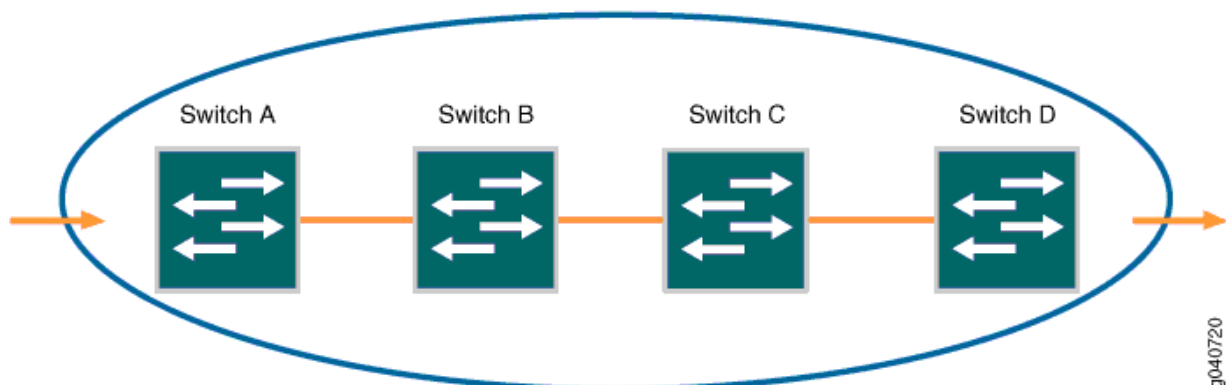
Junos OS CoS works by examining traffic entering the edge of your network. The switch classifies traffic into defined service groups to provide the special treatment of traffic across the network. For example, you can send voice traffic across certain links and data traffic across other links. In addition, the data traffic streams can be serviced differently along the network path to ensure that higher-paying customers receive better service. As the traffic leaves the network at the far edge, you can reclassify the traffic to meet the policies of the targeted peer by rewriting the DSCP or IEEE 802.1 code-point bits.

To support CoS, you must configure each switch in the network. Generally, each switch examines the packets that enter it to determine their CoS settings. These settings dictate which packets are transmitted first to the next downstream switch. Switches at the edges of the network might be required to alter the CoS settings of the packets that enter the network to classify the packets into the appropriate service groups.

In [Figure 1 on page 4](#), Switch A is receiving traffic. As each packet enters, Switch A examines the packet's current CoS settings and classifies the traffic into one of the groupings defined on the switch. This definition allows Switch A to prioritize its resources for servicing the traffic streams it receives. Switch A might alter the CoS settings (forwarding class and loss priority) of the packets to better match the defined traffic groups.

When Switch B receives the packets, it examines the CoS settings, determines the appropriate traffic groups, and processes the packet according to those settings. It then transmits the packets to Switch C, which performs the same actions. Switch D also examines the packets and determines the appropriate groups. Because Switch D sits at the far end of the network, it can reclassify (rewrite) the CoS code-point bits of the packets before transmitting them.

**Figure 1: Packet Flow Across the Network**



## Default CoS Behavior

If you do not configure CoS settings, the software performs some CoS functions to ensure that the system forwards traffic and protocol packets with minimum delay when the network is experiencing congestion. Some CoS settings, such as classifiers, are automatically applied to each logical interface that you configure. Other settings, such as *rewrite rules*, are applied only if you explicitly associate them with an interface.

### RELATED DOCUMENTATION

---

*Overview of Policers*

---

*Understanding Junos CoS Components*

---

*Understanding CoS Packet Flow*

---

*Understanding CoS Hierarchical Port Scheduling (ETS)*

## Configuring CoS

The traffic management class-of-service topics describe how to configure the Junos class-of-service (CoS) components. Junos CoS provides a flexible set of tools that enable you to fine tune control over the traffic on your network.

- Define classifiers that classify incoming traffic into forwarding classes to place traffic in groups for transmission.
- Map forwarding classes to output queues to define the type of traffic on each output queue.
- Configure schedulers for each output queue to control the service level (priority, bandwidth characteristics) of each type of traffic.
- Provide different service levels for the same forwarding classes on different interfaces.
- On switches that support data center bridging standards, configure lossless transport across the Ethernet network using priority-based flow control (PFC), Data Center Bridging Exchange protocol (DCBX), and enhanced transmission selection (ETS) hierarchical scheduling.
- Configure various CoS components individually or in combination to define CoS services.



**NOTE:** When you change the CoS configuration or when you deactivate and then reactivate the CoS configuration, the system experiences packet drops because the

system momentarily blocks traffic to change the mapping of incoming traffic to input queues. If you use a congestion notification profile for lossless behavior, you can expect the momentary generation of PFC pause frames.

[Table 1 on page 7](#) lists the primary CoS configuration tasks by platform and provides links to those tasks.



**NOTE:** Links to features that are not supported on the platform for which you are looking up information might not be functional.

**Table 1: CoS Configuration Tasks**

CoS Configuration Task	Links
<p>Basic CoS Configuration:</p> <ul style="list-style-type: none"> <li>• Configure code-point aliases to assign a name to a pattern of code-point bits that you can use instead of the bit pattern when you configure CoS components such as classifiers and rewrite rules</li> <li>• Configure classifiers and multidestination classifiers <ul style="list-style-type: none"> <li>• Set the forwarding class and loss priority of a packet based on the incoming CoS value and assign packets to output queues based on the associated forwarding class</li> <li>• Change the host default output queue and mapping of DSCP bits used in the type of service (ToS) field</li> </ul> </li> <li>• Configure forwarding classes</li> <li>• Configure rewrite rules to alter code point bit values in outgoing packets on the outbound interfaces of a switch so that the CoS treatment matches the policies of a targeted peer</li> <li>• Configure Ethernet PAUSE flow control, a congestion relief feature that provides link-level flow control for all traffic on a full-duplex Ethernet link, including those that belong to Ethernet link aggregated (LAG) interfaces. On any particular interface, symmetric and asymmetric flow control are mutually exclusive.</li> <li>• Assign the following CoS components to physical or logical interfaces: <ul style="list-style-type: none"> <li>• Classifiers</li> <li>• Congestion notification profiles</li> <li>• Forwarding classes</li> <li>• Forwarding class sets</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li>• <i>Defining CoS Code-Point Aliases</i></li> <li>• <i>Example: Configuring Classifiers</i></li> <li>• <i>Defining CoS BA Classifiers (DSCP, DSCP IPv6, IEEE 802.1p)</i></li> <li>• <i>Example: Configuring Multidestination (Multicast, Broadcast, DLF) Classifiers</i></li> <li>• <i>Changing the Host Outbound Traffic Default Queue Mapping</i></li> <li>• <i>Example: Configuring Forwarding Classes</i></li> <li>• <i>Defining CoS Rewrite Rules</i></li> <li>• <i>Enabling and Disabling CoS Symmetric Ethernet PAUSE Flow Control</i></li> <li>• <i>Configuring CoS Asymmetric Ethernet PAUSE Flow Control</i></li> <li>• <i>Assigning CoS Components to Interfaces</i></li> </ul>

Table 1: CoS Configuration Tasks *(Continued)*

CoS Configuration Task	Links
<ul style="list-style-type: none"> <li>• Output traffic control profiles</li> <li>• Port schedulers</li> <li>• Rewrite rules</li> </ul>	
<p>Configure Weighted random early detection (WRED) drop profiles that define the drop probability of packets of different packet loss probabilities (PLPs) as the output queue fills:</p> <ul style="list-style-type: none"> <li>• Configure WRED drop profiles where you associate WRED drop profiles with loss priorities in a scheduler. When you map the scheduler to a forwarding class (queue), you apply the interpolated drop profile to traffic of the specified loss priority on that queue.</li> <li>• Configure drop profile maps that map a drop profile to a packet loss priority, and associate the drop profile and packet loss priority with a scheduler</li> <li>• Configure explicit congestion notification (ECN) to enable end-to-end congestion notification between two endpoints on TCP/IP based networks. Apply WRED drop profiles to forwarding classes to control how the switch marks ECN-capable packets.</li> </ul>	<ul style="list-style-type: none"> <li>• <i>Example: Configuring WRED Drop Profiles</i></li> <li>• <i>Example: Configuring Drop Profile Maps</i></li> <li>• <i>Example: Configuring ECN</i></li> </ul>
<p>Configure queue schedulers and the bandwidth scheduling priority of individual queues. Schedulers define the CoS properties of output queues (output queues are mapped to forwarding classes, and classifiers map traffic into forwarding classes based on IEEE 802.1p or DSCP code points). Queue scheduling works with priority group scheduling to create a two-tier hierarchical scheduler. CoS scheduling properties include the amount of interface bandwidth assigned to the queue, the priority of the queue, whether explicit congestion notification (ECN) is enabled on the queue, and the WRED packet drop profiles associated with the queue.</p>	<ul style="list-style-type: none"> <li>• (Except QFX10000) <i>Example: Configuring Queue Schedulers</i></li> <li>• <i>Example: Configuring Queue Scheduling Priority</i></li> <li>• (QFX10000 only) <i>Example: Configuring Queue Schedulers for Port Scheduling</i></li> </ul>

**Table 1: CoS Configuration Tasks (Continued)**

CoS Configuration Task	Links
<p>Configure traffic control profiles to define the output bandwidth and scheduling characteristics of forwarding class sets (priority groups). The forwarding classes (queues) mapped to a forwarding class set share the bandwidth resources that you configure in the traffic control profile.</p>	<ul style="list-style-type: none"> <li>• <i>Defining CoS Traffic Control Profiles (Priority Group Scheduling)</i></li> <li>• <i>Example: Configuring Traffic Control Profiles (Priority Group Scheduling)</i></li> <li>• <i>Example: Configuring Minimum Guaranteed Output Bandwidth</i></li> <li>• <i>Example: Configuring Maximum Output Bandwidth</i></li> </ul>
<p>Configure enhanced transmission selection (ETS) and forwarding class sets, and disable the ETS recommendation TLV. Hierarchical port scheduling, the Junos OS implementation of ETS, enables you to group priorities that require similar CoS treatment into priority groups. You define the port bandwidth resources for a priority group, and you define the amount of the priority group's resources that each priority in the group can use.</p>	<ul style="list-style-type: none"> <li>• <i>Example: Configuring Forwarding Class Sets</i></li> <li>• <i>Example: Configuring CoS Hierarchical Port Scheduling (ETS)</i></li> <li>• <i>Disabling the ETS Recommendation TLV</i></li> </ul>
<p>Configure Data Center Bridging Capability Exchange protocol (DCBX), which discovers the data center bridging (DCB) capabilities of peers by exchanging feature configuration information and is an extension of the Link Layer Discovery Protocol (LLDP)</p> <ul style="list-style-type: none"> <li>• Configure the DCBX mode that an interface uses to communicate with the connected peer</li> <li>• Configure DCBX autonegotiation on a per-interface basis for each supported feature or application</li> <li>• Define each application for which you want DCBX to exchange application protocol information</li> <li>• Map applications to IEEE 802.1p code points</li> <li>• Apply an application map to a DCBX interface</li> </ul>	<ul style="list-style-type: none"> <li>• <i>Example: Configuring DCBX Application Protocol TLV Exchange</i></li> <li>• <i>Configuring the DCBX Mode</i></li> <li>• <i>Configuring DCBX Autonegotiation</i></li> <li>• <i>Defining an Application for DCBX Application Protocol TLV Exchange</i></li> <li>• <i>Configuring an Application Map for DCBX Application Protocol TLV Exchange</i></li> <li>• <i>Applying an Application Map to an Interface for DCBX Application Protocol TLV Exchange</i></li> </ul>

Table 1: CoS Configuration Tasks *(Continued)*

CoS Configuration Task	Links
<p>Configure CoS for FCoE:</p> <ul style="list-style-type: none"> <li>• Configure priority-based flow control (PFC) to divide traffic on one physical link into eight priorities</li> <li>• Configure a congestion notification profile (CNP) that enables priority-based flow control (PFC) on specified IEEE 802.1p priorities</li> <li>• Configure Multichassis link aggregation groups (MC-LAGs) to provide redundancy and load balancing between two switches</li> <li>• Configure two or more lossless forwarding classes and map them to different priorities</li> <li>• Configure lossless FCoE transport if your network uses a different priority than 3</li> <li>• Configure multiple lossless FCoE priorities on a converged Ethernet network</li> <li>• If the FCoE network uses a different priority than priority 3 for FCoE traffic, configure a rewrite value to remap incoming traffic from the FC SAN to that priority after the interface encapsulates the FC packets in Ethernet</li> <li>• Configure lossless priorities for multiple types of traffic, such as FCoE and iSCSI</li> </ul>	<ul style="list-style-type: none"> <li>• <i>Example: Configuring CoS PFC for FCoE Traffic</i></li> <li>• <a href="#">Example: Configuring CoS for FCoE Transit Switch Traffic Across an MC-LAG</a></li> <li>• <i>Configuring CoS PFC (Congestion Notification Profiles)</i></li> <li>• <i>Example: Configuring IEEE 802.1p Priority Remapping on an FCoE-FC Gateway</i></li> <li>• <i>Example: Configuring Two or More Lossless FCoE IEEE 802.1p Priorities on Different FCoE Transit Switch Interfaces</i></li> <li>• <i>Example: Configuring Lossless FCoE Traffic When the Converged Ethernet Network Does Not Use IEEE 802.1p Priority 3 for FCoE Traffic (FCoE Transit Switch)</i></li> <li>• <i>Example: Configuring Two or More Lossless FCoE Priorities on the Same FCoE Transit Switch Interface</i></li> <li>• <i>Configuring CoS Fixed Classifier Rewrite Values for Native FC Interfaces (NP_Ports)</i></li> <li>• <i>Example: Configuring Lossless IEEE 802.1p Priorities on Ethernet Interfaces for Multiple Applications (FCoE and iSCSI)</i></li> </ul>



## Understanding Junos CoS Components

### IN THIS SECTION

- [Code-Point Aliases | 11](#)
- [Policers | 11](#)
- [Classifiers | 11](#)
- [Forwarding Classes | 12](#)
- [Forwarding Class Sets | 13](#)
- [Flow Control \(Ethernet PAUSE, PFC, and ECN\) | 13](#)
- [WRED Profiles and Tail Drop | 14](#)
- [Schedulers | 15](#)
- [Rewrite Rules | 15](#)

This topic describes the Junos OS class-of-service (CoS) components:

### Code-Point Aliases

A *code-point alias* assigns a name to a pattern of code-point bits. You can use this name instead of the bit pattern when you configure other CoS components such as classifiers and *rewrite rules*.

### Policers

*Policers* limit traffic of a certain class to a specified bandwidth and burst size. Packets exceeding the policer limits can be discarded, or can be assigned to a different forwarding class, a different loss priority, or both. You define policers with filters that you can associate with input interfaces.

### Classifiers

Packet classification associates incoming packets with a particular CoS servicing level. In Junos OS, *classifiers* associate packets with a forwarding class and loss priority and assign packets to output queues based on the associated forwarding class. Junos OS supports two general types of classifiers:

- Behavior aggregate (BA) or CoS value traffic classifiers—Examine the CoS value in the packet header. The value in this single field determines the CoS settings applied to the packet. BA classifiers allow

you to set the forwarding class and loss priority of a packet based on the Differentiated Services code point (DSCP) value, IEEE 802.1p value, or MPLS EXP value.

- Multifield traffic classifiers—Examine multiple fields in the packet, such as source and destination addresses and source and destination port numbers of the packet. With multifield classifiers, you set the forwarding class and loss priority of a packet based on *firewall filter* rules.

On switches that require the separation of unicast and multideestination (multicast, broadcast, and destination lookup fail) traffic, you create separate unicast classifiers and multideestination classifiers. You cannot assign unicast traffic and multideestination traffic to the same classifier. You can apply unicast classifiers to one or more interfaces. Multideestination classifiers apply to all of the switch interfaces and cannot be applied to individual interfaces. Switches that require the separation of unicast and multideestination traffic have 12 output queues to provide 4 output queues reserved for multideestination traffic.

On switches that do not separate unicast and multideestination traffic, unicast and multideestination traffic use the same classifiers, and you do not create a separate special classifier for multideestination traffic. Switches that do not separate unicast and multideestination traffic have eight output queues because no extra queues are required to separate the traffic.

## Forwarding Classes

*Forwarding classes* group packets for transmission and CoS. You assign each packet to an output queue based on the packet's forwarding class. Forwarding classes affect the forwarding, scheduling, and rewrite marking policies applied to packets as they transit the switch.

Switches provide up to five default forwarding classes:

- best-effort—Best-effort traffic
- fcoe—Fibre Channel over Ethernet traffic
- no-loss—Lossless traffic
- network-control—Network control traffic
- mcast—Multicast traffic



**NOTE:** The default `mcast` forwarding class applies only to switches that require the separation of unicast and multideestination (multicast, broadcast, and destination lookup fail) traffic. On these switches, you create separate forwarding classes for the two types of traffic. The default `mcast` forwarding class transports only multideestination traffic, and the default `best-effort`, `fcoe`, `no-loss`, and `network-control` forwarding classes transport only unicast traffic. Unicast forwarding classes map to unicast output queues, and

multidestination forwarding classes map to multidestination output queues. You cannot assign unicast traffic and multidestination traffic to the same forwarding class or to the same output queue. Switches that require the separation of unicast and multidestination traffic have 12 output queues, 8 for unicast traffic and 4 for multidestination traffic.

On switches that do not separate unicast and multidestination traffic, unicast and multidestination traffic use the same forwarding classes and output queues, so the mcast forwarding class is not valid. You do not create separate forwarding classes for multidestination traffic. Switches that do not separate unicast and multidestination traffic have eight output queues because no extra queues are required to separate the traffic.

Switches support a total of either 12 forwarding classes (8 unicast forwarding classes and 4 multicast forwarding classes), or 8 forwarding classes (unicast and multidestination traffic use the same forwarding classes), which provides flexibility in classifying traffic.

## Forwarding Class Sets

You can group forwarding classes (output queues) into *forwarding class sets* to apply CoS to groups of traffic that require similar treatment. Forwarding class sets map traffic into priority groups to support enhanced transmission selection (ETS), which is described in IEEE 802.1Qaz.

You can configure up to three unicast forwarding class sets and one multicast forwarding class set. For example, you can configure different forwarding class sets to apply CoS to unicast groups of local area network (LAN) traffic, storage area network (SAN) traffic, and high-performance computing (HPC) traffic, and configure another group for multicast traffic.

Within each forwarding class set, you can configure special CoS treatment for the traffic mapped to each individual queue. This provides the ability to configure CoS in a two-tier hierarchical manner. At the forwarding class set tier, you configure CoS for groups of traffic using a *traffic control profile*. At the queue tier, you configure CoS for individual output queues within a forwarding class set using a *scheduler* that you map to a queue (forwarding class) using a *scheduler map*.

## Flow Control (Ethernet PAUSE, PFC, and ECN)

*Ethernet PAUSE* (described in IEEE 802.3X) is a link-level flow control mechanism. During periods of network congestion, Ethernet PAUSE stops all traffic on a full-duplex Ethernet link for a period of time specified in the PAUSE message.



**NOTE:** QFX10000 switches do not support Ethernet PAUSE.

*Priority-based flow control* (PFC) is described in IEEE 802.1Qbb as part of the IEEE data center bridging (DCB) specifications for creating a lossless Ethernet environment to transport loss-sensitive flows such as Fibre Channel over Ethernet (FCoE) traffic.

PFC is a link-level flow control mechanism similar to Ethernet PAUSE. However, Ethernet PAUSE stops all traffic on a link for a period of time. PFC decouples the pause function from the physical link and divides the traffic on the link into eight priorities (3-bit IEEE 802.1p code points). You can think of the eight priorities as eight “lanes” of traffic. You can apply pause selectively to the traffic on any priority without pausing the traffic on other priorities on the same link.

The granularity that PFC provides allows you to configure different levels of CoS for different types of traffic on the link. You can create lossless lanes for traffic such as FCoE, LAN backup, or management, while using standard frame-drop methods of congestion management for IP traffic on the same link.



**NOTE:** If you transport FCoE traffic, you must enable PFC on the priority assigned to FCoE traffic (usually IEEE 802.1p code point 011 on interfaces that carry FCoE traffic).

*Explicit congestion notification* (ECN) enables end-to-end congestion notification between two endpoints on TCP/IP based networks. ECN must be enabled on both endpoints and on all of the intermediate devices between the endpoints for ECN to work properly. Any device in the transmission path that does not support ECN breaks the end-to-end ECN functionality. ECN notifies networks about congestion with the goal of reducing packet loss and delay by making the sending device decrease the transmission rate until the congestion clears, without dropping packets. RFC 3168, *The Addition of Explicit Congestion Notification (ECN) to IP*, defines ECN.

## WRED Profiles and Tail Drop

A weighted random early detection (WRED) profile (drop profile) defines parameters that enable the network to drop packets during periods of congestion. A *drop profile* defines the conditions under which packets of different loss priorities drop, by determining the probability of dropping a packet for each loss priority when output queues become congested. Drop profiles essentially set a value for a level of queue fullness—when the queue fills to the level of the queue fullness value, packets drop. The combination of queue fill level, the probability of dropping a packet at that fill level, and loss priority of the packet, determine whether a packet is dropped or forwarded. Each pairing of a fill level with a drop probability creates a point on a drop profile curve.

You can associate different drop profiles with different loss priorities to set the probability of dropping packets. You can apply a drop profile for each loss priority to a forwarding class (output queue) by applying a drop profile to a scheduler, and then mapping the scheduler to a forwarding class using a scheduler map. When the queue mapped to the forwarding class experiences congestion, the drop profile determines the level of packet drop for traffic of each loss priority in that queue.

Loss priority affects the scheduling of a packet without affecting the packet's relative ordering. Typically you mark packets exceeding a particular service level with a high loss priority.

Tail drop is a simple drop mechanism that drops all packets indiscriminately during periods of congestion, without differentiating among the packet loss priorities of traffic flows. Tail drop requires only one curve point that corresponds to the maximum depth of the output queue, and drop probability when traffic exceeds the buffer depth is 100 percent (all packets that cannot be stored in the queue are dropped). WRED is superior to tail-drop because WRED enables you to treat traffic of different priorities in a differentiated manner, so that higher priority traffic receives preference, and because of the ability to set multiple points on the drop curve.

## Schedulers

Each switch interface has multiple queues assigned to store packets. The switch determines which queue to service based on a particular method of scheduling. This process often involves determining the sequence in which different types of packets should be transmitted.

You can define the scheduling priority (`priority`), minimum guaranteed bandwidth (`transmit-rate`), maximum bandwidth (`shaping-rate`), and WRED profiles to be applied to a particular queue (forwarding class) for packet transmission. By default, extra bandwidth is shared among queues in proportion to the minimum guaranteed bandwidth of each queue. On switches that support the `excess-rate` statement, you can configure the percentage of shared extra bandwidth an output queue receives independently from the minimum guaranteed bandwidth transmit rate, or you can use default bandwidth sharing based on the transmit rate.

A scheduler map associates a specified forwarding class with a scheduler configuration. You can associate up to four user-defined scheduler maps with the interfaces.

## Rewrite Rules

A *rewrite rule* sets the appropriate CoS bits in the outgoing packet. This allows the next downstream device to classify the packet into the appropriate service group. Rewriting (marking) outbound packets is useful when the switch is at the border of a network and must change the CoS values to meet the policies of the targeted peer.



**NOTE:** Ingress firewall filters can also rewrite forwarding class and loss priority values.

## RELATED DOCUMENTATION

| *Understanding CoS Packet Flow*

## Assigning CoS Components to Interfaces

After you define the following CoS components, you assign them to physical or logical interfaces. Components that you assign to physical interfaces are valid for all of the logical interfaces configured on the physical interface. Components that you assign to a logical interface are valid only for that logical interface.

- Classifiers—Assign to logical interfaces; on some devices, you apply classifiers to physical Layer 3 interfaces and the classifiers are applied to all logical interfaces on the physical interface.
- Congestion notification profiles—Assign only to physical interfaces.
- Forwarding classes—Assign to interfaces by mapping to forwarding class sets.
- Forwarding class sets—Assign only to physical interfaces.
- Output traffic control profiles—Assign only to physical interfaces (with a forwarding class set).
- Port schedulers—Assign only to physical interfaces on devices that support port scheduling. Associate the scheduler with a forwarding class in a scheduler map and apply the scheduler map to the physical interface.
- Rewrite rules—Assign to logical interfaces; on some devices, you apply classifiers to physical Layer 3 interfaces and the classifiers are applied to all logical interfaces on the physical interface.

You can assign a CoS component to a single interface or to multiple interfaces using wildcards. You can also assign a congestion notification profile or a forwarding class set globally to all interfaces.

To assign CoS components to interfaces:

Assign a CoS component to a physical interface by associating a CoS component (for example, a forwarding class set named `be-priority-group`) with an interface:

```
[edit class-of-service interfaces]
user@switch# set xe-0/0/7 forwarding-class-set be-priority-group
```

Assign a CoS component to a logical interface by associating a CoS component (for example, a classifier named `be_classifier`) with a logical interface:

```
[edit class-of-service interfaces]
user@switch# set xe-0/0/7 unit 0 classifiers dscp be_classifier
```

Assign a CoS component to multiple interfaces by associating a CoS component (for example, a rewrite rule named `customup-rw`) to all 10-Gigabit Ethernet interfaces on the switch, use wildcard characters for the interface name and logical interface (unit) number:

```
[edit class-of-service interfaces]
user@switch# set xe-* unit * rewrite-rules ieee-802.1 customup-rw
```

Assign a congestion notification profile or a forwarding class set globally to all interfaces using the `set class-of-service interfaces all` statement. For example, to assign a forwarding class set named `be-priority-group` to all interfaces:

```
[edit class-of-service interfaces]
user@switch# set all forwarding-class-set be-priority-group
```



**NOTE:** If there is an existing CoS configuration of any type on an interface, the global configuration is not applied to that particular interface. The global configuration is applied to all interfaces that do not have an existing CoS configuration.

For example, if you configure a rewrite rule, assign it to interfaces `xe-0/0/20.0` and `xe-0/0/22.0`, and then configure a forwarding class set and apply it to all interfaces, the forwarding class set is applied to every interface except `xe-0/0/20` and `xe-0/0/22`.



**NOTE:** Wild card configuration takes precedence over interfaces `all` configuration under the `[edit class-of-service]` hierarchy. For example, in the following configuration:

```
[edit class-of-service interfaces]
user@switch# set xe-* scheduler-map sch0
user@switch# set all unit 0 classifiers dscp cls
```

the wildcard configuration (`xe-*`) prevails meaning `classifiers dscp cls` is not applied to any logical interface at all. The logical interfaces will apply default classifiers only. If you need to apply the classifier to logical interfaces as well, you must explicitly apply the classifier to specific logical interfaces. For example:

```
[edit class-of-service interfaces]
user@switch# set xe-0/1/1:0 unit 0 classifiers dscp cls
user@switch# set xe-0/1/1:2 unit 0 classifiers dscp cls
```

RELATED DOCUMENTATION

<a href="#">Monitoring Interfaces That Have CoS Components</a>	<a href="#">18</a>
<i>Understanding Junos CoS Components</i>	

## Monitoring Interfaces That Have CoS Components

IN THIS SECTION

- [Purpose](#) | 18
- [Action](#) | 18
- [Meaning](#) | 18

### Purpose

Use the monitoring functionality to display details about the physical and logical interfaces and the CoS components assigned to them.

### Action

To monitor interfaces that have CoS components in the CLI, enter the command:

```
user@switch> show class-of-service interface
```

To monitor a specific interface in the CLI, enter the command:

```
user@switch> show class-of-service interface interface-name
```

### Meaning

[Table 2 on page 19](#) summarizes key output fields for CoS interfaces.



**Table 2: Summary of Key CoS Interfaces Output Fields**

Field	Values
Physical interface	Name of a physical interface to which CoS components are assigned.
Index	Index of this interface or the internal index of a specific object.
Queues supported	Number of queues you can configure on the interface.
Queues in use	Number of queues currently configured.
Scheduler map	Name of the scheduler map associated with this interface.
Congestion-notification	Status of congestion notification (enabled or disabled).
Rewrite Input IEEE Code-point	(Fibre Channel NP_Port interfaces only) IEEE 802.1p code point (priority) the interface assigns to incoming Fibre Channel (FC) traffic when the interface encapsulates the FC traffic in Ethernet before forwarding it onto the FCoE network.
Logical Interface	Name of a logical interface on the physical interface to which CoS components are assigned.
Object	Category of an object—for example, classifier, scheduler-map, or rewrite.
Name	Name of the object—for example, ba-classifier.
Type	Type of the object—for example, ieee8021p for a classifier.

## RELATED DOCUMENTATION

*Assigning CoS Components to Interfaces*

## Understanding CoS Packet Flow

When a packet traverses a switch, the switch provides the appropriate level of service to the packet using either default *class-of-service* (CoS) settings or CoS settings that you configure. On ingress ports, the switch classifies packets into appropriate forwarding classes and assigns a loss priority to the packets. On egress ports, the switch applies packet scheduling and (if you have configured them) *rewrite rules* to re-mark packets.

You can configure CoS on Layer 2 logical interfaces, and you can configure CoS on Layer 3 physical interfaces if you have defined at least one *logical interface* on the Layer 3 physical interface. You cannot configure CoS on Layer 2 physical interfaces and Layer 3 logical interfaces.

For Layer 2 traffic, either use the default CoS settings or configure CoS on each logical interface. You can apply different CoS settings to different Layer 2 logical interfaces.

For Layer 3 traffic, either use the default CoS settings or configure CoS on the physical interface (not on the logical unit). The switch uses the CoS applied on the physical Layer 3 interface for all logical Layer 3 interfaces configured on the physical Layer 3 interface.

The switch applies CoS to packets as they flow through the system:

- An interface has one or more classifiers of different types applied to it (configure this at the [edit class-of-service interfaces] hierarchy level). The classifier types are based on the portion of the incoming packet that the classifier examines (IEEE 802.1p code point bits or DSCP code point bits).
- When a packet enters an ingress port, the classifier assigns the packet to a forwarding class and a loss priority based on the code point bits of the packet (configure this at the [edit class-of-service classifiers] hierarchy level).
- The switch assigns each forwarding class to an output queue (configure this at the [edit class-of-service forwarding-classes] hierarchy level).
- Input (and output) policers meter traffic and can change the forwarding class and loss priority if a traffic flow exceeds its service level.
- A scheduler map is applied to each interface. When a packet exits an egress port, the scheduler map controls how it is treated (configure this at the [edit class-of-service interfaces] hierarchy level). A scheduler map assigns schedulers to forwarding classes (configure this at the [edit class-of-service scheduler-maps] hierarchy level).
- A scheduler defines how traffic is treated at the egress interface output queue (configure this at the [edit class-of-service schedulers] hierarchy level). You control the transmit rate, shaping rate, priority, and drop profile of each forwarding class by mapping schedulers to forwarding classes in scheduler maps, then applying scheduler maps to interfaces.

- A drop-profile defines how aggressively to drop packets that are mapped to a particular scheduler (configure this at the [edit class-of-service drop-profiles] hierarchy level).
- A rewrite rule takes effect as the packet leaves an interface that has a rewrite rule configured (configure this at the [edit class-of-service rewrite-rules] hierarchy level). The rewrite rule writes information to the packet (for example, a rewrite rule can re-mark the code point bits of outgoing traffic) according to the forwarding class and loss priority of the packet.

[Figure 2 on page 22](#) is a high-level flow diagram of how packets from various sources enter switch interfaces, are classified at the ingress, and then scheduled (provided bandwidth) at the egress queues.

Figure 2: CoS Classifier, Queues, and Scheduler

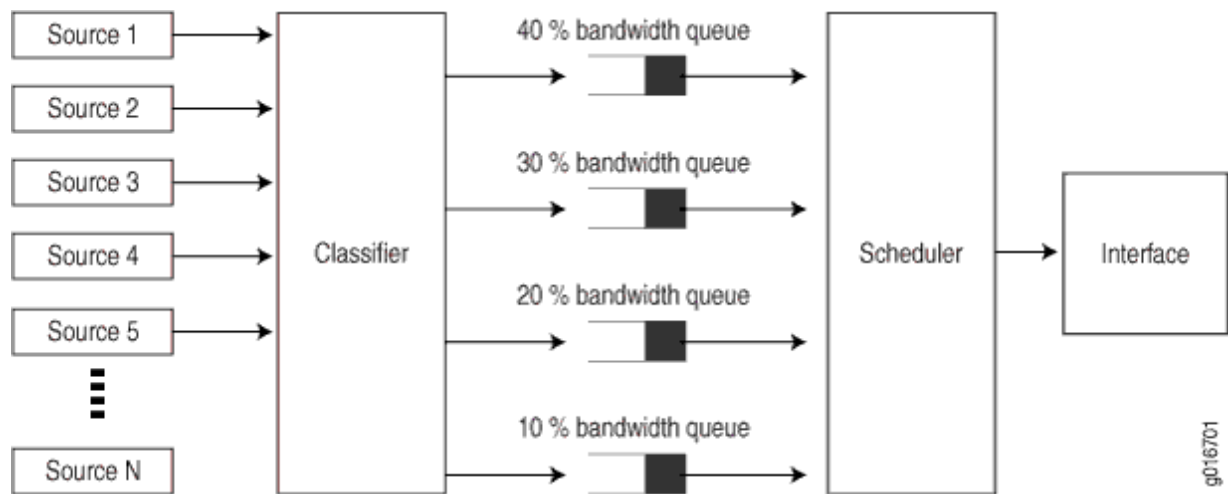
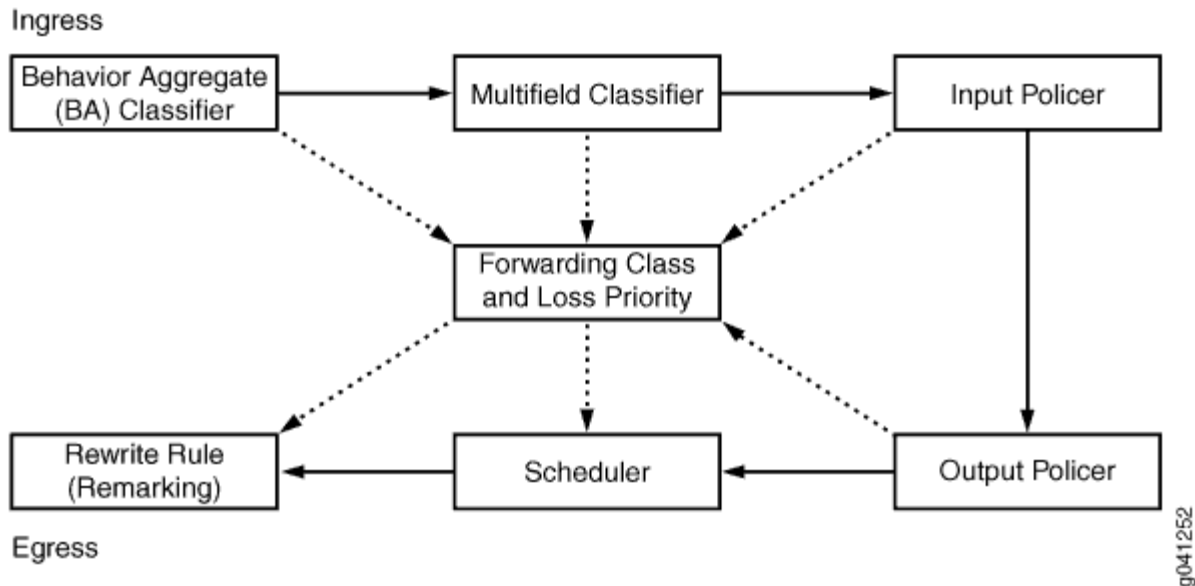


Figure 3 on page 23 shows the packet flow through the CoS components that you can configure.

Figure 3: Packet Flow Through Configurable CoS Components



The middle box (Forwarding Class and Loss Priority) represents two values that you can use on ingress and egress interfaces. The system uses these values for classifying traffic on ingress interfaces and for rewrite rule re-marking on egress interfaces. Each outer box represents a process component. The components in the top row apply to incoming packets. The components in the bottom row apply to outgoing packets.

The solid-line arrows show the direction of packet flow from ingress to egress. The dotted-line arrows that point to the forwarding class and loss priority box indicate processes that configure (set) the forwarding class and loss priority. The dotted-line arrows that point away from the forwarding class and loss priority box indicate processes that use forwarding class and loss priority as input values on which to base actions.

For example, the BA classifier sets the forwarding class and loss priority of incoming packets, so the forwarding class and loss priority are outputs of the classifier and the arrow points away from the classifier. The scheduler receives the forwarding class and loss priority settings, and queues the outgoing packets based on those settings, so the arrow points toward the scheduler.

# Understanding Default CoS Settings

IN THIS SECTION

- [Default Forwarding Classes and Queue Mapping | 24](#)
- [Default Forwarding Class Sets \(Priority Groups\) | 25](#)
- [Default Code-Point Aliases | 26](#)
- [Default Classifiers | 28](#)
- [Default Rewrite Rules | 33](#)
- [Default Drop Profile | 33](#)
- [Default Schedulers | 33](#)
- [Default Scheduler Maps | 36](#)
- [Default Shared Buffer Configuration | 37](#)

If you do not configure CoS settings, Junos OS performs some CoS functions to ensure that traffic and protocol packets are forwarded with minimum delay when the network experiences congestion. Some default mappings are automatically applied to each *logical interface* that you configure.

You can display default CoS settings by issuing the `show class-of-service` *operational mode command*.

This topic describes the default configurations for the following CoS components:

## Default Forwarding Classes and Queue Mapping

[Table 3 on page 24](#) shows the default mapping of the default forwarding classes to queues and packet drop attribute.

**Table 3: Default Forwarding Classes and Queue Mapping**

Default Forwarding Class	Description	Default Queue Mapping	Packet Drop Attribute
best-effort (be)	Best-effort traffic class (priority 0, IEEE 802.1p code point 000)	0	drop

**Table 3: Default Forwarding Classes and Queue Mapping** *(Continued)*

Default Forwarding Class	Description	Default Queue Mapping	Packet Drop Attribute
fcoe	Guaranteed delivery for FCoE traffic (priority 3, IEEE 802.1p code point 011)	3	no-loss
no-loss	Guaranteed delivery for TCP no-loss traffic (priority 4, IEEE 802.1p code point 100)	4	no-loss
network-control (nc)	Network control traffic (priority 7, IEEE 802.1p code point 111)	7	drop
(Excluding QFX10000) mcast	Multidestination traffic	8	drop  <b>NOTE:</b> You cannot configure multidestination forwarding classes as no-loss (lossless) traffic classes.



**NOTE:** On the QFX10000 switch, unicast and multidestination (multicast, broadcast, and destination lookup fail) traffic use the same forwarding classes and output queues 0 through 7.

## Default Forwarding Class Sets (Priority Groups)

If you do not explicitly configure forwarding class sets, the system automatically creates a default forwarding class set that contains all of the forwarding classes on the switch. The system assigns 100 percent of the port output bandwidth to the default forwarding class set.

Ingress traffic is classified based on the default classifier settings. The forwarding classes (queues) in the default forwarding class set receive bandwidth based on the default scheduler settings. Forwarding classes that are not part of the default scheduler receive no bandwidth.

The default forwarding class set is transparent. It does not appear in the configuration and is used for Data Center Bridging Capability Exchange (DCBX) protocol advertisement.

## Default Code-Point Aliases

[Table 4 on page 26](#) shows the default mapping of code-point aliases to IEEE code points.

**Table 4: Default IEEE 802.1 Code-Point Aliases**

CoS Value Types	Mapping
be	000
be1	001
ef	010
ef1	011
af11	100
af12	101
nc1	110
nc2	111

[Table 5 on page 26](#) shows the default mapping of code-point aliases to DSCP and DSCP IPv6 code points.

**Table 5: Default DSCP and DCSP IPv6 Code-Point Aliases**

CoS Value Types	Mapping
ef	101110
af11	001010



Table 5: Default DSCP and DCSP IPv6 Code-Point Aliases *(Continued)*

CoS Value Types	Mapping
af12	001100
af13	001110
af21	010010
af22	010100
af23	010110
af31	011010
af32	011100
af33	011110
af41	100010
af42	100100
af43	100110
be	000000
cs1	001000
cs2	010000
cs3	011000

**Table 5: Default DSCP and DCSP IPv6 Code-Point Aliases (Continued)**

CoS Value Types	Mapping
cs4	100000
cs5	101000
nc1	110000
nc2	111000

## Default Classifiers

The switch applies default unicast IEEE 802.1, unicast DSCP, and multidestination classifiers to each interface that does not have explicitly configured classifiers. If you explicitly configure one type of classifier but not other types of classifiers, the system uses only the configured classifier and does not use default classifiers for other types of traffic.



**NOTE:** The QFX10000 switch applies the default MPLS EXP classifier to a logical interface if you enable the MPLS protocol family on that interface.

There are two different default unicast IEEE 802.1 classifiers, a trusted classifier for ports that are in trunk mode or tagged-access mode, and an untrusted classifier for ports that are in access mode. [Table 6 on page 28](#) shows the default mapping of IEEE 802.1 code-point values to forwarding classes and loss priorities for ports in trunk mode or tagged-access mode.

**Table 6: Default IEEE 802.1 Classifiers for Ports in Trunk Mode or Tagged Access Mode (Trusted Classifier)**

Code Point	Forwarding Class	Loss Priority
be (000)	best-effort	low
be1 (001)	best-effort	low
ef (010)	best-effort	low

**Table 6: Default IEEE 802.1 Classifiers for Ports in Trunk Mode or Tagged Access Mode (Trusted Classifier) (Continued)**

Code Point	Forwarding Class	Loss Priority
ef1 (011)	fcoe	low
af11 (100)	no-loss	low
af12 (101)	best-effort	low
nc1 (110)	network-control	low
nc2 (111)	network-control	low

Table 7 on page 29 shows the default mapping of IEEE 802.1p code-point values to forwarding classes and loss priorities for ports in access mode (all incoming traffic is mapped to best-effort forwarding classes).

**Table 7: Default IEEE 802.1 Classifiers for Ports in Access Mode (Untrusted Classifier)**

Code Point	Forwarding Class	Loss Priority
000	best-effort	low
001	best-effort	low
010	best-effort	low
011	best-effort	low
100	best-effort	low
101	best-effort	low
110	best-effort	low

**Table 7: Default IEEE 802.1 Classifiers for Ports in Access Mode (Untrusted Classifier) (Continued)**

Code Point	Forwarding Class	Loss Priority
111	best-effort	low

Table 8 on page 30 shows the default mapping of IEEE 802.1 code-point values to multdestination (multicast, broadcast, and destination lookup fail traffic) forwarding classes and loss priorities.

**Table 8: Default IEEE 802.1 Multidestination Classifiers**

Code Point	Forwarding Class	Loss Priority
be (000)	mcast	low
be1 (001)	mcast	low
ef (010)	mcast	low
ef1 (011)	mcast	low
af11 (100)	mcast	low
af12 (101)	mcast	low
nc1 (110)	mcast	low
nc2 (111)	mcast	low

Table 9 on page 31 shows the default mapping of DSCP code-point values to forwarding classes and loss priorities for DSCP IP and DCSP IPv6.



**NOTE:** There are no default DSCP IP classifiers for multdestination traffic. DSCP IPv6 classifiers are not supported for multdestination traffic.

**Table 9: Default DSCP IP and IPv6 Classifiers**

Code Point	Forwarding Class	Loss Priority
ef (101110)	best-effort	low
af11 (001010)	best-effort	low
af12 (001100)	best-effort	low
af13 (001110)	best-effort	low
af21 (010010)	best-effort	low
af22 (010100)	best-effort	low
af23 (010110)	best-effort	low
af31 (011010)	best-effort	low
af32 (011100)	best-effort	low
af33 (011110)	best-effort	low
af41 (100010)	best-effort	low
af42 (100100)	best-effort	low
af43 (100110)	best-effort	low
be (000000)	best-effort	low
cs1 (001000)	best-effort	low

**Table 9: Default DSCP IP and IPv6 Classifiers (Continued)**

Code Point	Forwarding Class	Loss Priority
cs2 (010000)	best-effort	low
cs3 (011000)	best-effort	low
cs4 (100000)	best-effort	low
cs5 (101000)	best-effort	low
nc1 (110000)	network-control	low
nc2 (111000)	network-control	low

On QFX10000 switches, [Table 10 on page 32](#) shows the default mapping of MPLS EXP code-point values to forwarding classes and loss priorities.

**Table 10: Default EXP Classifiers on QFX10000 Switches**

Code Point	Forwarding Class	Loss Priority
000	best-effort	low
001	best-effort	high
010	expedited-forwarding	low
011	expedited-forwarding	high
100	assured-forwarding	low
101	assured-forwarding	high
110	network-control	low

**Table 10: Default EXP Classifiers on QFX10000 Switches (Continued)**

Code Point	Forwarding Class	Loss Priority
111	network-control	high

## Default Rewrite Rules

There are no default *rewrite rules*. If you do not explicitly configure rewrite rules, the switch does not reclassify egress traffic.

## Default Drop Profile

[Table 11 on page 33](#) shows the default drop profile configuration.

**Table 11: Default Drop Profile**

Fill Level	Drop Probability
100	100

## Default Schedulers

[Table 12 on page 33](#) shows the default scheduler configuration.

**Table 12: Default Schedulers**

Default Scheduler and Queue Number	Transmit Rate (Guaranteed Minimum Bandwidth)	Shaping Rate (Maximum Bandwidth)	Excess Bandwidth Sharing	Priority	Buffer Size
best-effort forwarding class scheduler (queue 0)	5% (QFX10000 15%)	None	5% (QFX10000 15%)	low	5% (QFX10000 15%)
fcoe forwarding class scheduler (queue 3)	35%	None	35%	low	35%

**Table 12: Default Schedulers (Continued)**

Default Scheduler and Queue Number	Transmit Rate (Guaranteed Minimum Bandwidth)	Shaping Rate (Maximum Bandwidth)	Excess Bandwidth Sharing	Priority	Buffer Size
no-loss forwarding class scheduler (queue 4)	35%	None	35%	low	35%
network-control forwarding class scheduler (queue 7)	5% (QFX10000 15%)	None	5% (QFX10000 15%)	low	5% (QFX10000 15%)
(Excluding QFX10000) mcast forwarding class scheduler (queue 8)	20%	None	20%	low	20%



**NOTE:** The minimum guaranteed bandwidth (transmit rate) also determines the amount of excess (extra) bandwidth that the queue can share. Extra bandwidth is allocated to queues in proportion to the transmit rate of each queue. On QFX10000 switches, you can use the `excess-rate` statement to override the default transmit rate setting and configure the excess bandwidth percentage independently of the transmit rate.

By default, only the five default schedulers shown in [Table 12 on page 33](#), excluding the mcast scheduler on QFX10000 switches, have traffic mapped to them. Only the queues associated with the default schedulers, and forwarding classes on QFX10000 switches, receive default bandwidth, based on the default scheduler transmit rate. (You can configure schedulers and forwarding classes to allocate bandwidth to other queues or to change the default bandwidth of a default queue.) In addition, other than on QFX5200, QFX5210, and QFX10000 switches, multidestination queue 11 receives enough bandwidth from the default multidestination scheduler to handle CPU-generated multidestination traffic. If a forwarding class does not transport traffic, the bandwidth allocated to that forwarding class is available to other forwarding classes.



**NOTE:** On QFX10000 switches, unicast and multidestination (multicast, broadcast, and destination lookup fail) traffic use the same forwarding classes and output queues.

Default hierarchical scheduling, known as enhanced transmission selection (ETS, defined in IEEE 802.1Qaz), divides the total port bandwidth between two groups of traffic: unicast traffic and multidestination traffic. By default, unicast traffic consists of queue 0 (best-effort forwarding class),



queue 3 (fcoe forwarding class), queue 4 (no-loss forwarding class), and queue 7 (network-control forwarding class). Unicast traffic receives and shares a total of 80 percent of the port bandwidth. By default, multdestination traffic (mcast queue 8) receives a total of 20 percent of the port bandwidth. So on a 10-Gigabit port, default scheduling provides unicast traffic 8-Gbps of bandwidth and multdestination traffic 2-Gbps of bandwidth.



**NOTE:** Except on QFX5200, QFX5210, and QFX10000 switches, multdestination queue 11 also receives a small amount of default bandwidth from the multdestination scheduler. CPU-generated multdestination traffic uses queue 11, so you might see a small number of packets egress from queue 11. In addition, in the unlikely case that firewall filter match conditions map multdestination traffic to a unicast forwarding class, that traffic uses queue 11.

On QFX10000 switches, default scheduling is port scheduling. Default hierarchical scheduling, known as ETS, allocates the total port bandwidth to the four default forwarding classes served by the four default schedulers, as defined by the four default schedulers. The result is the same as direct port scheduling. Configuring hierarchical port scheduling, however, enables you to group forwarding classes that carry similar types of traffic into forwarding class sets (also called priority groups), and to assign port bandwidth to each forwarding class set. The port bandwidth assigned to the forwarding class set is then assigned to the forwarding classes within the forwarding class set. This hierarchy enables you to control port bandwidth allocation with greater granularity, and enables hierarchical sharing of extra bandwidth to better utilize link bandwidth.

Default scheduling for all switches uses weighted round-robin (WRR) scheduling. Each queue receives a portion (weight) of the total available interface bandwidth. The scheduling weight is based on the transmit rate of the default scheduler for that queue. For example, queue 7 receives a default scheduling weight of 5 percent, 15 percent on QFX10000 switches, of the available bandwidth, and queue 4 receives a default scheduling weight of 35 percent of the available bandwidth. Queues are mapped to forwarding classes (for example, queue 7 is mapped to the network-control forwarding class and queue 4 is mapped to the no-loss forwarding class), so forwarding classes receive the default bandwidth for the queues to which they are mapped. Unused bandwidth is shared with other default queues.

If you want non-default (unconfigured) queues to forward traffic, you should explicitly map traffic to those queues (configure the forwarding classes and queue mapping) and create schedulers to allocate bandwidth to those queues. For example, except on QFX5200, QFX5210, and QFX10000 switches, by default, queues 1, 2, 5, and 6 are unconfigured, and multdestination queues 9, 10, and 11 are unconfigured. Unconfigured queues have a default scheduling weight of 1 so that they can receive a small amount of bandwidth in case they need to forward traffic. (However, queue 11 can use more of the default multdestination scheduler bandwidth if necessary to handle CPU-generated multdestination traffic.)



**NOTE:** Except on QFX10000 switches, all four multidestination queues, or two for QFX5200 and QFX5210, switches, have a scheduling weight of 1. Because by default multidestination traffic goes to queue 8, queue 8 receives almost all of the multidestination bandwidth. (There is no default traffic on queue 9 and queue 10, and very little default traffic on queue 11, so there is almost no competition for multidestination bandwidth.)

However, if you explicitly configure queue 9, 10, or 11 (by mapping code points to the unconfigured multidestination forwarding classes using the multidestination classifier), the explicitly configured queues share the multidestination scheduler bandwidth equally with default queue 8, because all of the queues have the same scheduling weight (1). To ensure that multidestination bandwidth is allocated to each queue properly and that the bandwidth allocation to the default queue (8) is not reduced too much, we strongly recommend that you configure a scheduler if you explicitly classify traffic into queue 9, 10, or 11.

If you map traffic to an unconfigured queue, the queue receives only the amount of group bandwidth proportional to its default weight (1). The actual amount of bandwidth an unconfigured queue receives depends on how much bandwidth the other queues in the group are using.

On QFX 10000 switches, if you map traffic to an unconfigured queue and do not schedule port resources for the queue (configure a scheduler, map it to the forwarding class that is mapped to the queue, and apply the scheduler mapping to the port), the queue receives only the amount of excess bandwidth proportional to its default weight (1). The actual amount of bandwidth an unconfigured queue gets depends on how much bandwidth the other queues on the port are using.

If the other queues use less than their allocated amount of bandwidth, the unconfigured queues can share the unused bandwidth. Configured queues have higher priority for bandwidth than unconfigured queues, so if a configured queue needs more bandwidth, then less bandwidth is available for unconfigured queues. Unconfigured queues always receive a minimum amount of bandwidth based on their scheduling weight (1). If you map traffic to an unconfigured queue, to allocate bandwidth to that queue, configure a scheduler for the forwarding class that is mapped to the queue and apply it to the port.

Default Scheduler Maps

Table 13 on page 37 shows the default mapping of forwarding classes to schedulers.

**Table 13: Default Scheduler Maps**

Forwarding Class	Scheduler
best-effort	Default BE scheduler
fcoe	Default FCoE scheduler
no-loss	No-loss scheduler
network-control	Default network-control scheduler
(Excluding QFX10000) mcast-be	Default multideestination scheduler

## Default Shared Buffer Configuration

Table [Table 14 on page 37](#) and [Table 15 on page 37](#) show the default shared buffer allocations:



**NOTE:** Shared buffers do not apply to QFX10000 switches.

**Table 14: Default Ingress Shared Buffer Configuration**

Total Shared Ingress Buffer	Lossless Buffer	Lossless-Headroom Buffer	Lossy Buffer
100%	9%	45%	46%

**Table 15: Default Egress Shared Buffer Configuration**

Total Shared Egress Buffer	Lossless Buffer	Lossy Buffer	Multicast Buffer
100%	50%	31%	19%

RELATED DOCUMENTATION

<i>Overview of Junos OS CoS</i>
<i>Understanding Junos CoS Components</i>
<i>Understanding Default CoS Scheduling and Classification</i>
<i>Understanding CoS Classifiers</i>
<a href="#">Understanding CoS Classifiers</a>
<i>Understanding Applying CoS Classifiers and Rewrite Rules to Interfaces</i>
<i>Understanding CoS Code-Point Aliases</i>
<i>Understanding CoS Forwarding Classes</i>
<i>Understanding CoS Rewrite Rules</i>
<i>Understanding CoS Output Queue Schedulers</i>
<i>Understanding CoS Port Schedulers on QFX Switches</i>
<i>Understanding CoS WRED Drop Profiles</i>

## CoS Inputs and Outputs Overview

Some CoS components map one set of values to another set of values. Each mapping contains one or more inputs and one or more outputs. When you configure a mapping, you set the outputs for a given set of inputs, as shown in [Table 16 on page 38](#).

**Table 16: CoS Mappings—Inputs and Outputs**

CoS Mappings	Inputs	Outputs	Comments
classifiers	code-points	forwarding-class, loss-priority	The map sets the forwarding class and packet loss priority (PLP) for a specific set of code points.
drop-profile-map	loss-priority, protocol	drop-profile	The map sets the drop profile for a specific PLP and protocol type.
rewrite-rules	loss-priority, forwarding-class	code-points	The map sets the code points for a specific forwarding class and PLP.

Table 16: CoS Mappings—Inputs and Outputs *(Continued)*

CoS Mappings	Inputs	Outputs	Comments
rewrite-value (Fibre Channel Interfaces)	<i>forwarding-class</i>	<i>code-point</i>	(Systems that support native Fibre Channel interfaces only) The map sets the code point for the forwarding class specified in the fixed classifier attached to the native Fibre Channel (NP_Port) interface.

RELATED DOCUMENTATION

| [Understanding CoS Packet Flow](#)

Overview of Policers

IN THIS SECTION

- [Policer Overview | 40](#)
- [Policer Types | 42](#)
- [Policer Actions | 43](#)
- [Policer Colors | 43](#)
- [Filter-Specific Policers | 44](#)
- [Suggested Naming Convention for Policers | 44](#)
- [Policer Counters | 45](#)
- [Policer Algorithms | 45](#)
- [Policers Can Limit Egress Firewall Filters | 45](#)
- [Platform-Specific Policer Behavior | 46](#)

A switch polices traffic by limiting the input or output transmission rate of a class of traffic according to user-defined criteria. Policing (or rate-limiting) traffic allows you to control the maximum rate of traffic sent or received on an interface and to provide multiple priority levels or classes of service.

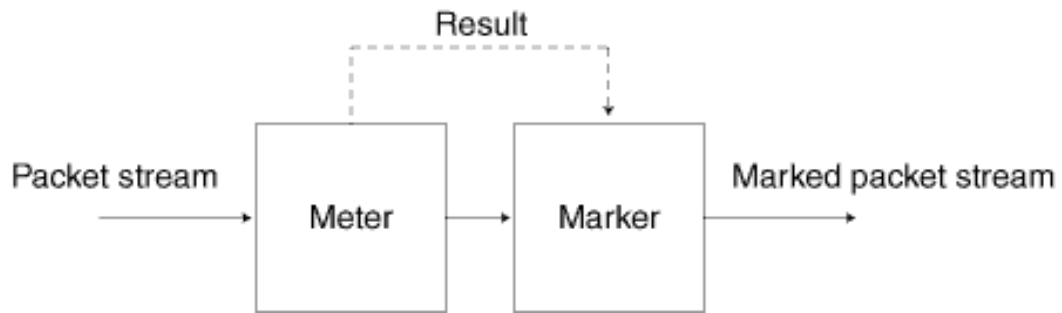
Policing is also an important component of firewall filters. You can achieve policing by including policers in *firewall filter* configurations.

## Policer Overview

You use policers to apply limits to traffic flow and set consequences for packets that exceed these limits—usually applying a higher loss priority—so that if packets encounter downstream congestion, they can be discarded first. Policers apply only to unicast packets.

Policers provide two functions: metering and marking. A policer meters (measures) each packet against traffic rates and burst sizes that you configure. It then passes the packet and the metering result to the marker, which assigns a packet loss priority that corresponds to the metering result. [Figure 4 on page 41](#) illustrates this process.

Figure 4: Flow of Tricolor Marking Policer Operation



g017049

After you name and configure a policer, you can use it by specifying it as an action in one or more firewall filters.

## Policer Types

A switch supports three types of policers:

- **Single-rate two-color marker**—A two-color policer (or “policer” when used without qualification) meters the traffic stream and classifies packets into two categories of packet loss priority (PLP) according to a configured bandwidth and burst-size limit. You can mark packets that exceed the bandwidth and burst-size limit with a specified PLP or simply discard them.

You can specify this type of policer in an ingress or egress firewall.



**NOTE:** A two-color policer is most useful for metering traffic at the port (physical interface) level.

- **Single-rate three-color marker**—This type of policer is defined in RFC 2697, *A Single Rate Three Color Marker*, as part of an assured forwarding (AF) per-hop-behavior (PHB) classification system for a Differentiated Services (DiffServ) environment. This type of policer meters traffic based on one rate—the configured committed information rate (CIR) as well as the committed burst size (CBS) and the excess burst size (EBS). The CIR specifies the average rate at which bits are admitted to the switch. The CBS specifies the usual burst size in bytes and the EBS specifies the maximum burst size in bytes. The EBS must be greater than or equal to the CBS, and neither can be 0.

You can specify this type of policer in an ingress or egress firewall.



**NOTE:** A single-rate three-color marker (TCM) is most useful when a service is structured according to packet length and not peak arrival rate.

- **Two-rate three-color marker**—This type of policer is defined in RFC 2698, *A Two Rate Three Color Marker*, as part of an assured forwarding per-hop-behavior classification system for a Differentiated Services environment. This type of policer meters traffic based on two rates—the CIR and peak information rate (PIR) along with their associated burst sizes, the CBS and peak burst size (PBS). The PIR specifies the maximum rate at which bits are admitted to the network and must be greater than or equal to the CIR.

You can specify this type of policer in an ingress or egress firewall.



**NOTE:** A two-rate three-color policer is most useful when a service is structured according to arrival rates and not necessarily packet length.



See [Table 17 on page 43](#) for information about how metering results are applied for each of these policer types.

## Policer Actions

Policer actions are implicit or explicit and vary by policer type. *Implicit* means that Junos OS assigns the loss priority automatically. [Table 17 on page 43](#) describes the policer actions.

**Table 17: Policer Actions**

Policer	Marking	Implicit Action	Configurable Action
Single-rate two-color	Green (conforming)	Assign low loss priority	None
	Red (nonconforming)	None	Discard
Single-rate three-color	Green (conforming)	Assign low loss priority	None
	Yellow (above the CIR and CBS)	Assign medium-high loss priority	None
	Red (above the EBS)	Assign high loss priority	Discard
Two-rate three-color	Green (conforming)	Assign low loss priority	None
	Yellow (above the CIR and CBS)	Assign medium-high loss priority	None
	Red (above the PIR and PBS)	Assign high loss priority	Discard



**NOTE:** If you specify a policer in an egress *firewall filter*, the only supported action is discard.

## Policer Colors

Single-rate and two-rate three-color policers can operate in two modes:

- **Color-blind**—In color-blind mode, the three-color policer assumes that all packets examined have not been previously marked or metered. In other words, the three-color policer is “blind” to any previous coloring a packet might have had.
- **Color-aware**—In color-aware mode, the three-color policer assumes that all packets examined have been previously marked or metered. In other words, the three-color policer is “aware” of the previous coloring a packet might have had. In color-aware mode, the three-color policer can increase the PLP of a packet but cannot decrease it. For example, if a color-aware three-color policer meters a packet with a medium PLP marking, it can raise the PLP level to high but cannot reduce the PLP level to low.

## Filter-Specific Policers

You can configure policers to be filter-specific, which means that Junos OS creates only one policer instance regardless of how many times the policer is referenced. When you do this on certain switches, rate limiting is applied in aggregate, so if you configure a policer to discard traffic that exceeds 1 Gbps and reference that policer in three different terms, the total bandwidth allowed by the filter is 1 Gbps. However, the behavior of a filter-specific policer is affected by how the firewall filter terms that reference the policer are stored in TCAM. If you create a filter-specific policer and reference it in multiple firewall filter terms, the policer allows more traffic than expected if the terms are stored in different TCAM slices. For example, if you configure a policer to discard traffic that exceeds 1 Gbps and reference that policer in three different terms that are stored in three separate memory slices, the total bandwidth allowed by the filter is 3 Gbps, not 1 Gbps.

To prevent this unexpected behavior from occurring, use the information about TCAM slices presented in *Planning the Number of Firewall Filters to Create* to organize your configuration file so that all the firewall filter terms that reference a given filter-specific policer are stored in the same TCAM slice.

## Suggested Naming Convention for Policers

We recommend that you use the naming convention *policertypeTCM#-color type* when configuring three-color policers and *policer#* when configuring two-color policers. TCM stands for three-color marker. Because policers can be numerous and must be applied correctly to work, a simple naming convention makes it easier to apply the policers properly. For example, the first single-rate, color-aware three-color policer configured would be named *srTCM1-ca*. The second two-rate, color-blind three-color configured would be named *trTCM2-cb*. The elements of this naming convention are explained below:

- *sr* (single-rate)
- *tr* (two-rate)
- *TCM* (tricolor marking)
- *1* or *2* (number of marker)
- *ca* (color-aware)

- cb (color-blind)

## Policer Counters

On some switches, each policer that you configure includes an implicit counter that counts the number of packets that exceed the rate limits that are specified for the policer. If you use the same policer in multiple terms—either within the same filter or in different filters—the implicit counter counts all the packets that are policed in all of these terms and provides the total amount. If you want to obtain separate packet counts for each term on an affected switch, use these options:

- Configure a unique policer for each term.
- Configure only one policer, but use a unique, explicit counter in each term.

## Policer Algorithms

Policing uses the *token-bucket algorithm*, which enforces a limit on average bandwidth while allowing bursts up to a specified maximum value. It offers more flexibility than the *leaky bucket algorithm* in allowing a certain amount of bursty traffic before it starts discarding packets.

## Policers Can Limit Egress Firewall Filters

On some switches, the number of egress policers you configure can affect the total number of allowed egress firewall filters. Every policer has two implicit counters that take up two entries in a 1024-entry TCAM. These are used for counters, including counters that are configured as action modifiers in firewall filter terms. (Policers consume two entries because one is used for green packets and one is used for nongreen packets regardless of policer type.) If the TCAM becomes full, you are unable to commit any more egress firewall filters that have terms with counters. For example, if you configure and commit 512 egress policers (two-color, three-color, or a combination of both policer types), all of the memory entries for counters get used up. If later in your configuration file you insert additional egress firewall filters with terms that also include counters, *none* of the terms in those filters are committed because there is no available memory space for the counters.

Here are some additional examples:

- Assume that you configure egress filters that include a total of 512 policers and no counters. Later in your configuration file you include another egress filter with 10 terms, 1 of which has a counter action modifier. None of the terms in this filter are committed because there is not enough TCAM space for the counter.
- Assume that you configure egress filters that include a total of 500 policers, so 1000 TCAM entries are occupied. Later in your configuration file you include the following two egress filters:

- Filter A with 20 terms and 20 counters. All the terms in this filter are committed because there is enough TCAM space for all the counters.
- Filter B comes after Filter A and has five terms and five counters. *None* of the terms in this filter are committed because there is not enough memory space for *all* the counters. (Five TCAM entries are required but only four are available.)

You can prevent this problem by ensuring that egress firewall filter terms with counter actions are placed earlier in your configuration file than terms that include policers. In this circumstance, Junos OS commits policers even if there is not enough TCAM space for the implicit counters. For example, assume the following:

- You have 1024 egress firewall filter terms with counter actions.
- Later in your configuration file you have an egress filter with 10 terms. None of the terms have counters but one has a policer action modifier.

You can successfully commit the filter with 10 terms even though there is not enough TCAM space for the implicit counters of the policer. The policer is committed without the counters.

## Platform-Specific Policer Behavior

Use the following table to review platform-specific behaviors for your platforms.

**Table 18: Platform-Specific Policer Behavior**

Platform	Difference
QFX5100	<ul style="list-style-type: none"> <li>• QFX5100 switches support 1535 ingress policers and 1024 egress policers (assuming one policer per firewall filter term).</li> </ul>
QFX5110	<ul style="list-style-type: none"> <li>• QFX5110 switches support 6144 ingress policers and 1024 egress policers (assuming one policer per firewall filter term).</li> </ul>

Table 18: Platform-Specific Policer Behavior (*Continued*)

Platform	Difference
QFX5200	<ul style="list-style-type: none"><li>• QFX5200 switches support 1535 ingress policers and 1024 egress policers (assuming one policer per firewall filter term).</li><li>• In an environment of light bursty traffic, QFX5200 switches might not replicate all multicast packets to two or more downstream interfaces. This occurs only at a line rate burst—if traffic is consistent, the issue does not occur. In addition, the issue occurs only when packet size increases beyond 6k in a one gigabit traffic flow.</li></ul>

Table 18: Platform-Specific Policer Behavior (*Continued*)

Platform	Difference
QFX10000 Series	<ul style="list-style-type: none"> <li>QFX10000 switches support 8K policers (all policer types).</li> <li>A policer restricts traffic at the configured transmission rate per PFE. In QFX10016, QFX10002, QFX10002-60C, and QFX10008 switches, when aggregated ethernet (AE) interface bundles span multiple PFEs, the overall transmission rate of the policer for the subscriber could exceed the configured transmission rate of the policer (depending on the number of PFEs involved).</li> </ul> <p>As an example:</p> <ul style="list-style-type: none"> <li>Policer with bandwidth-limit 100 mbps configured on an AE interface that has member links xe-1/0/0 (fpc1-pfe0) and xe-1/0/30 (fpc1-pfe1) . Here, the two member links belong to FPC1, but are on different PFEs. When the policer is applied to the AE interface, this will result in a total bandwidth of 200 Mbps as policer is configured for two PFEs.</li> <li>Policer with bandwidth-limit 100 mbps configured on an AE interface that has member links xe-1/0/0 (fpc1-pfe0), et-2/0/1 (fpc2-pfe1) and xe-2/0/18:0 (fpc2-pfe2) . Here, one member link belongs to FPC1 and PFE0 on this FPC. The rest two member links belong to FPC2, but different PFEs. When the policer is applied to the AE interface, this will result in a total bandwidth of 300 Mbps as policer is configured for three PFEs.</li> <li>Policer with bandwidth-limit 100 mbps configured on an AE interface that has member links xe-1/0/0 and xe-1/0/1 on a single PFE (fpc1-pfe0) . Here, the member links belong to</li> </ul>

Table 18: Platform-Specific Policer Behavior (*Continued*)

Platform	Difference
	FPC1 and to the same PFE. When the policer is applied to the AE interface, this will result in a total bandwidth of 100 Mbps as policer is configured on a per PFE basis.

## RELATED DOCUMENTATION

*Understanding Color-Blind Mode for Single-Rate Tricolor Marking*

*Understanding Color-Blind Mode for Two-Rate Tricolor Marking*

*Understanding Color-Aware Mode for Single-Rate Tricolor Marking*

*Understanding Color-Aware Mode for Two-Rate Tricolor Marking*

*Configuring Two-Color and Three-Color Policers to Control Traffic Rates*

# 2

PART

## Classifying and Rewriting Traffic

---

- [Using Classifiers, Forwarding Classes, and Rewrite Rules | 51](#)
-



# Using Classifiers, Forwarding Classes, and Rewrite Rules

## IN THIS CHAPTER

- Understanding CoS Classifiers | 52
- Defining CoS BA Classifiers (DSCP, DSCP IPv6, IEEE 802.1p) | 61
- Example: Configuring Classifiers | 63
- Monitoring CoS Classifiers | 68
- Understanding Default CoS Scheduling and Classification | 69
- Understanding Applying CoS Classifiers and Rewrite Rules to Interfaces | 80
- Understanding CoS Code-Point Aliases | 93
- Defining CoS Code-Point Aliases | 96
- Monitoring CoS Code-Point Value Aliases | 97
- Understanding CoS Forwarding Classes | 99
- Defining CoS Forwarding Classes | 105
- Example: Configuring Forwarding Classes | 108
- Monitoring CoS Forwarding Classes | 115
- Understanding CoS Rewrite Rules | 118
- Defining CoS Rewrite Rules | 121
- Troubleshooting an Unexpected Rewrite Value | 123
- Monitoring CoS Rewrite Rules | 126

## Understanding CoS Classifiers

### IN THIS SECTION

- Interfaces and Output Queues | 53
- Output Queues for Unicast and Multidestination Traffic | 53
- Behavior Aggregate Classifiers | 54
- Fixed Classifiers on Ethernet Interfaces | 57
- Fixed Classifiers on Native Fibre Channel Interfaces (NP\_Ports) | 58
- Multifield Classifiers | 59
- MPLS EXP Classifiers | 59
- Packet Classification for IRB Interfaces and RVIs | 59
- Platform-Specific Classifier Behavior | 60

Packet classification maps incoming packets to a particular class-of-service (CoS) servicing level. Classifiers map packets to a forwarding class and a loss priority, and they assign packets to output queues based on the forwarding class. There are three general types of classifiers:

- Behavior aggregate (BA) classifiers—DSCP and DSCP IPv6 classify IP and IPv6 traffic, EXP classifies MPLS traffic, and IEEE 802.1p classifies all other traffic. (Although this topic covers EXP classifiers, for more details, see *Understanding CoS MPLS EXP Classifiers and Rewrite Rules*. EXP classifiers are applied only on family mpls interfaces.)
- Fixed classifiers—Fixed classifiers classify all ingress traffic on a physical interface into one forwarding class, regardless of the CoS bits in the packet header.
- Multifield (MF) classifiers—MF classifiers classify traffic based on more than one field in the packet header and take precedence over BA and fixed classifiers.

Classifiers assign incoming unicast and multidestination (multicast, broadcast, and destination lookup fail) traffic to forwarding classes, so that different classes of traffic can receive different treatment. Classification is based on CoS bits, DSCP bits, EXP bits, a forwarding class (fixed classifier), or packet headers (multifield classifiers). Each classifier assigns all incoming traffic that matches the classifier configuration to a particular forwarding class. Classifiers and forwarding classes handle either unicast or multidestination traffic. You cannot mix unicast and multidestination traffic in the same classifier or forwarding class.

## Interfaces and Output Queues

You can apply classifiers to Layer 2 *logical interface* unit 0 (but not to other logical interfaces), and to Layer 3 physical interfaces if the Layer 3 physical interface has at least one defined logical interface. Classifiers applied to Layer 3 physical interfaces are used on all logical interfaces on that physical interface. *Understanding Applying CoS Classifiers and Rewrite Rules to Interfaces* describes the interaction between classifiers and interfaces in greater detail.

You can configure both a BA classifier and an MF classifier on an interface. If you do this, the BA classification is performed first, and then the MF classification is performed. If the two classification results conflict, the MF classification result overrides the BA classification result.

You cannot configure a fixed classifier and a BA classifier on the same interface.

You can configure either a DSCP or DSCP IPv6 classifier and an IEEE 802.1p classifier on the same interface. IP traffic uses the DSCP or DSCP IPv6 classifier. All other traffic uses the IEEE classifier. When you configure a global EXP classifier, MPLS traffic uses the EXP classifier provided that the interface is configured as `family mpls`. You can configure only one DSCP classifier on a physical interface (either one DSCP classifier or one DSCP IPv6 classifier, but not both).

Although you can configure as many EXP classifiers as you want, the switch uses only one MPLS EXP classifier as a global classifier on all interfaces.

After you configure an MPLS EXP classifier, you can configure it as the global EXP classifier by including the EXP classifier at the `[edit class-of-service system-defaults classifiers exp]` hierarchy level. All switch interfaces that are configured as `family mpls` use the global EXP classifier specified in this configuration statement to classify MPLS traffic.

## Output Queues for Unicast and Multidestination Traffic

You can create unicast BA classifiers for unicast traffic and multicast BA classifiers for multidestination traffic, which includes multicast, broadcast, and destination lookup fail (DLF) traffic. You cannot assign unicast traffic and multidestination traffic to the same BA classifier.

On each interface, the switch has separate output queues for unicast traffic and for multidestination traffic:

- The switch supports 12 output queues, with 8 queues dedicated to unicast traffic and 4 queues dedicated to multidestination traffic.
- Queues 0 through 7 are unicast traffic queues. You can apply only unicast BA classifiers to unicast queues. A unicast BA classifier should contain only forwarding classes that are mapped to unicast queues.

- Queues 8 through 11 are multdestination traffic queues. You can apply only multdestination BA classifiers to multdestination queues. A multdestination BA classifier should contain only forwarding classes that are mapped to multdestination queues.

You can apply unicast classifiers to one or more interfaces. Multdestination classifiers and EXP classifiers apply to all of the switch interfaces and cannot be applied to individual interfaces. Use the DSCP multdestination classifier for both IP and IPv6 multdestination traffic. The DSCP IPv6 classifier is not supported for multdestination traffic.

## Behavior Aggregate Classifiers

Behavior aggregate classifiers map a class-of-service (CoS) value to a forwarding class and loss priority. The forwarding class determines the output queue. A scheduler uses the loss priority to control packet discard during periods of congestion by associating different drop profiles with different loss priorities.

The switch supports three types of BA classifiers:

- Differentiated Services code point (DSCP) for IP DiffServ (IP and IPv6)
- IEEE 802.1p CoS bits
- MPLS EXP (applies only to interfaces configured as `family mpls`)

BA classifiers are based on fixed-length fields, which makes them computationally more efficient than MF classifiers. Therefore, core devices, which handle high traffic volumes, are normally configured to perform BA classification.

Unicast and multicast traffic cannot share the same classifier. You can map unicast traffic and multicast traffic to the same classifier CoS value, but the unicast traffic must belong to a unicast classifier and the multicast traffic must belong to a multdestination classifier.

## Default Behavior Aggregate Classification

Juniper Networks Junos OS automatically assigns implicit default classifiers to all logical interfaces based on the type of interface. [Table 19 on page 54](#) lists different types of interfaces and the corresponding implicit default BA classifiers.

**Table 19: Default BA Classification**

Type of Interface	Default BA Classification
Layer 2 interface in trunk mode or tagged-access mode	ieee8021p-default

**Table 19: Default BA Classification (Continued)**

Type of Interface	Default BA Classification
Layer 2 interface in access mode	ieee8021p-untrusted
Layer 3 interface	dscp-default dscp-ipv6-default
(QFX10000 only) MPLS interface	exp-default



**NOTE:** Default BA classifiers assign traffic only to the best-effort, fcoe, no-loss, network-control, and mcast forwarding classes.



**NOTE:** Except on QFX10000 switches, there is no default MPLS EXP classifier. You must configure an EXP classifier and apply it globally to all interfaces that are configured as family mpls by including it in the [edit class-of-service system-defaults classifiers exp] hierarchy. On family mpls interfaces, if a fixed classifier is present on the interface, the EXP classifier overrides the fixed classifier.

If an EXP classifier is not configured, then if a fixed classifier is applied to the interface, the MPLS traffic uses the fixed classifier. If no EXP classifier and no fixed classifier is applied to the interface, MPLS traffic is treated as best-effort traffic. DSCP classifiers are not applied to MPLS traffic.

Because the EXP classifier is global, you cannot configure some ports to use a fixed IEEE 802.1p classifier for MPLS traffic on some interfaces and the global EXP classifier for MPLS traffic on other interfaces. When you configure a global EXP classifier, all MPLS traffic on all interfaces uses the EXP classifier, even interfaces that have a fixed classifier.

When you explicitly associate a classifier with a logical interface, you override the default classifier with the explicit classifier.



**NOTE:** You can apply only one DSCP and one IEEE 802.1p classifier to a Layer 2 interface. If both types of classifiers are present, DSCP classifiers take precedence over IEEE 802.1p classifiers. If you configure an EXP classifier, or a global EXP classifier, then MPLS traffic uses that classifier on interfaces configured as family mpls.

## Importing a Classifier

You can use any existing classifier, including the default classifiers, as the basis for defining a new classifier. You accomplish this using the `import` statement.

The imported classifier is used as a template and is not modified. The modifications you make become part of a new classifier (and a new template) identified by the name of the new classifier. Whenever you commit a configuration that assigns a new forwarding class-name and loss-priority value to a code-point alias or set of bits, it replaces the old entry in the new classifier template. As a result, you must explicitly specify every CoS value in every packet classification that requires modification.

## Multidestination Classifiers

Multidestination classifiers are applied to all interfaces and cannot be applied to individual interfaces. You can configure both a DSCP multidestination classifier and an IEEE multidestination classifier. IP and IPv6 traffic use the DSCP classifier, and all other traffic uses the IEEE classifier.

DSCP IPv6 multidestination classifiers are not supported, so IPv6 traffic uses the DSCP multidestination classifier.

The default multidestination classifier is the IEEE 802.1p multidestination classifier.

## PFC Priorities

The eight IEEE 802.1p code points correspond to the eight priorities that *priority-based flow control* (PFC) uses to differentiate traffic classes for lossless transport. When you map a forwarding class (which maps to an output queue) to an IEEE 802.1p CoS value, the IEEE 802.1p CoS value identifies the PFC priority.

Although you can map a priority to any output queue (by mapping the IEEE 802.1p code point value to a forwarding class), we recommend that the priority and the forwarding class (unicast except for QFX10000 switches) match in a one-to-one correspondence. For example, priority 0 is assigned to queue 0, priority 1 is assigned to queue 1, and so on, as shown in [Table 20 on page 57](#). A one-to-one correspondence of queue and priority numbers makes it easier to configure and maintain the mapping of forwarding classes to priorities and queues.

**Table 20: Default IEEE 802.1p Code Point to PFC Priority, Output Queue, and Forwarding Class Mapping**

IEEE 802.1p Code Point	PFC Priority	Output Queue (Unicast except for QFX10000)	Forwarding Class and Packet Drop Attribute
000	0	0	best-effort (drop)
001	1	1	best-effort (drop)
010	2	2	best-effort (drop)
011	3	3	fcoe (no-loss)
100	4	4	no-loss (no-loss)
101	5	5	best-effort (drop)
110	6	6	network-control (drop)
111	7	7	network-control (drop)



**NOTE:** By convention, deployments with converged server access typically use IEEE 802.1p priority 3 (011) for FCoE traffic. The default mapping of the fcoe forwarding class is to queue 3. Apply priority-based flow control (PFC) to the entire FCoE data path to configure the end-to-end lossless behavior that FCoE requires. We recommend that you use priority 3 for FCoE traffic unless your network architecture requires that you use a different priority.

## Fixed Classifiers on Ethernet Interfaces

Fixed classifiers map all traffic on a physical interface to a forwarding class and a loss priority, unlike BA classifiers, which map traffic into multiple different forwarding classes based on the IEEE 802.1p CoS bits field value in the VLAN header or the DSCP field value in the type-of-service bits in the packet IP header. Each forwarding class maps to an output queue. However, when you use a fixed classifier,

regardless of the CoS or DSCP bits, all Incoming traffic is classified into the forwarding class specified in the fixed classifier. A scheduler uses the loss priority to control packet discard during periods of congestion by associating different drop profiles with different loss priorities.

You cannot configure a fixed classifier and a DSCP or IEEE 802.1p BA classifier on the same interface. If you configure a fixed classifier on an interface, you cannot configure a DSCP or an IEEE classifier on that interface. If you configure a DSCP classifier, an IEEE classifier, or both classifiers on an interface, you cannot configure a fixed classifier on that interface.



**NOTE:** For MPLS traffic on the same interface, you can configure both a fixed classifier and an EXP classifier on QFX10000, or a global EXP classifier on other switches. When both an EXP classifier or global EXP classifier and a fixed classifier are applied to an interface, MPLS traffic on interfaces configured as `family mpls` uses the EXP classifier, and all other traffic uses the fixed classifier.

To switch from a fixed classifier to a BA classifier, or to switch from a BA classifier to a fixed classifier, deactivate the existing classifier attachment on the interface, and then attach the new classifier to the interface.



**NOTE:** If you configure a fixed classifier that classifies all incoming traffic into the `fcoe` forwarding class (or any forwarding class designed to handle FCoE traffic), you must ensure that all traffic that enters the interface is FCoE traffic and is tagged with the FCoE IEEE 802.1p code point (priority).

## Fixed Classifiers on Native Fibre Channel Interfaces (NP\_Ports)

Applying a fixed classifier to a native Fibre Channel (FC) interface (NP\_Port) is a special case. By default, native FC interfaces classify incoming traffic from the FC SAN into the `fcoe` forwarding class and map the traffic to IEEE 802.1p priority 3 (code point 011). When you apply a fixed classifier to an FC interface, you also configure a priority rewrite value for the interface. The FC interface uses the priority rewrite value as the IEEE 802.1p tag value for all incoming packets instead of the default value of 3.

For example, if you specify a priority rewrite value of 5 (code point 101) for an FC interface, the interface tags all incoming traffic from the FC SAN with priority 5 and classifies the traffic into the forwarding class specified in the fixed classifier.



**NOTE:** The forwarding class specified in the fixed classifier on FC interfaces must be a lossless forwarding class.



## Multifield Classifiers

Multifield classifiers examine multiple fields in a packet such as source and destination addresses and source and destination port numbers of the packet. With MF classifiers, you set the forwarding class and loss priority of a packet based on *firewall filter* rules.

MF classification is normally performed at the network edge because of the general lack of DiffServ code point (DSCP) support in end-user applications. On a switch at the edge of a network, an MF classifier provides the filtering functionality that scans through a variety of packet fields to determine the forwarding class for a packet. Typically, a classifier performs matching operations on the selected fields against a configured value.

## MPLS EXP Classifiers

You can configure up to 64 EXP classifiers for MPLS traffic and apply them to `family mpls` interfaces. On QFX10000 switches you can use the default MPLS EXP, but on other switches there is no default MPLS classifier. You can configure an EXP classifier and apply it globally to all interfaces that are configured as `family mpls` by including it in the `[edit class-of-service system-defaults classifiers exp]` hierarchy level. On `family mpls` interfaces, if a fixed classifier is present on the interface, the EXP classifier overrides the fixed classifier for MPLS traffic only.

Except on QFX10000 switches, if an EXP classifier is not configured, then if a fixed classifier is applied to the interface, the MPLS traffic uses the fixed classifier. If no EXP classifier and no fixed classifier is applied to the interface, MPLS traffic is treated as best-effort traffic. DSCP classifiers are not applied to MPLS traffic.

Because the EXP classifier is global, you cannot configure some ports to use a fixed IEEE 802.1p classifier for MPLS traffic on some interfaces and the global EXP classifier for MPLS traffic on other interfaces. When you configure a global EXP classifier, all MPLS traffic on all interfaces uses the EXP classifier, even interfaces that have a fixed classifier.

For details about EXP classifiers, see *Understanding CoS MPLS EXP Classifiers and Rewrite Rules*. EXP classifiers are applied only on `family mpls` interfaces.

## Packet Classification for IRB Interfaces and RVIs

On QFX10000 switches, you cannot apply classifiers directly to integrated routing and bridging (*IRB*) interfaces. Similarly, on other switches you cannot apply classifiers directly to routed VLAN interfaces (*RVIs*). This results because the members of IRBs and RVIs are VLANs, not ports. However, you can apply classifiers to the VLAN port members of an IRB interface. You can also apply MF classifiers to IRBs and RVIs.

## Platform-Specific Classifier Behavior

Use the following table to review platform-specific behaviors for your platforms.

**Table 21: Platform-Specific Classifier Behavior**

Platform	Difference
QFX5000 Series	<ul style="list-style-type: none"> <li>QFX5000 Series switches do not have a default EXP classifier. You must explicitly define a global EXP classifier.</li> <li>QFX5200 switches support 10 output queues, with 8 queues dedicated to unicast traffic and 2 queues dedicated to multidestination traffic.</li> </ul>
QFX10000 Series	<ul style="list-style-type: none"> <li>On QFX10000 switches you can apply different classifiers to different Layer 3 logical interfaces. You cannot apply classifiers to physical interfaces.</li> <li>QFX10000 switches do not support multidestination (mcast) classifiers. On QFX10000 switches, a classifier can assign both unicast and multidestination traffic to the same forwarding class.</li> <li>QFX10000 switches do not support fixed classifiers on Native Fibre Channel Interfaces (NP_Ports)</li> <li>On QFX10000 switches you can use the default MPLS EXP, but on other switches there is no default MPLS classifier.</li> <li>On QFX10000 switches, you cannot apply classifiers directly to IRB interfaces.</li> </ul>

## RELATED DOCUMENTATION

*Understanding CoS MPLS EXP Classifiers and Rewrite Rules*

*Understanding CoS Packet Flow*

*Understanding Default CoS Settings*

*Understanding Applying CoS Classifiers and Rewrite Rules to Interfaces*

*Defining CoS BA Classifiers (DSCP, DSCP IPv6, IEEE 802.1p)*

*Example: Configuring Unicast Classifiers*

*Defining CoS BA Classifiers (DSCP, DSCP IPv6, IEEE 802.1p)*

*Example: Configuring Multidestination (Multicast, Broadcast, DLF) Classifiers*

*Configuring a Global MPLS EXP Classifier*

*Configuring Rewrite Rules for MPLS EXP Classifiers*

## Defining CoS BA Classifiers (DSCP, DSCP IPv6, IEEE 802.1p)

### Overview

Packet classification associates incoming packets with a particular CoS servicing level. Behavior aggregate (BA) classifiers examine the Differentiated Services code point (DSCP or DSCP IPv6) value, the IEEE 802.1p CoS value, or the MPLS EXP value in the packet header to determine the CoS settings applied to the packet. (See *Configuring a Global MPLS EXP Classifier* to learn how to define EXP classifiers for MPLS traffic.) BA classifiers allow you to set the forwarding class and loss priority of a packet based on the incoming CoS value.

On most devices, unicast traffic uses different classifiers than multidestination (multicast, broadcast, and destination lookup fail) traffic. You use the `multi-destination` statement at the [edit class-of-service] hierarchy level to configure a multidestination BA classifier.

Multidestination classifiers apply to all of the switch interfaces and handle multicast, broadcast, and destination lookup fail (DLF) traffic. You cannot apply a multidestination classifier to a single interface or to a range of interfaces.

### Platform-specific Information

- On QFX10000 switches and NFX Series devices, unicast and multidestination traffic use the same classifiers and forwarding classes.
- QFX5130, QFX5700 & QFX5220 switches do not support DSCP IPv6 classifiers and rewrite rules. However, you can apply DSCP classifiers and rewrite rules for IPV6 traffic as well.

### Configuring BA Classifiers

To configure a DSCP, DSCP IPv6, or IEEE 802.1p BA classifier using the CLI:

1. Create a BA classifier:

- To create a DSCP, DSCP IPv6, or IEEE 802.1p BA classifier based on the default classifier, import the default DSCP, DSCP IPv6, or IEEE 802.1p classifier and associate it with a forwarding class, a loss priority, and a code point:

```
[edit class-of-service classifiers]
user@switch# set (dscp | dscp-ipv6 | ieee-802.1) classifier-name import default forwarding-
class forwarding-class-name loss-priority level code-points [aliases] [bit-patterns]
```

- To create a BA classifier that is not based on the default classifier, create a DSCP, DSCP IPv6, or IEEE 802.1p classifier and associate it with a forwarding class, a loss priority, and a code point:

```
[edit class-of-service classifiers]
user@switch# set (dscp | dscp-ipv6 | ieee-802.1) classifier-name forwarding-class
forwarding-class-name loss-priority level code-points [aliases] [bit-patterns]
```

2. For multidestination traffic, except on QFX10000 switches or NFX Series devices, configure the classifier as a multidestination classifier:

```
[edit class-of-service]
user@switch# set multi-destination classifiers (dscp | dscp-ipv6 | ieee-802.1 | inet-
precedence) classifier-name
```

3. Apply the classifier to a specific Ethernet interface or to all Ethernet interfaces, or to all Fibre Channel interfaces on the device.

- To apply the classifier to a specific interface:

```
[edit class-of-service interfaces]
user@switch# set interface-name unit unit classifiers (dscp | dscp-ipv6 | ieee-802.1)
classifier-name
```

- To apply the classifier to all Ethernet interfaces on the switch, use wildcards for the interface name and the logical interface (unit) number:

```
[edit class-of-service interfaces]
user@switch# set xe-* unit * classifiers (dscp | dscp-ipv6 | ieee-802.1) classifier-name
```

## RELATED DOCUMENTATION

*Example: Configuring CoS Hierarchical Port Scheduling (ETS)*

*Example: Configuring Unicast Classifiers*

*Configuring a Global MPLS EXP Classifier*

*Configuring Rewrite Rules for MPLS EXP Classifiers*

*Monitoring CoS Classifiers*

*Understanding CoS Classifiers*

[Understanding CoS Classifiers](#)

*Understanding CoS MPLS EXP Classifiers and Rewrite Rules*

*Understanding Applying CoS Classifiers and Rewrite Rules to Interfaces*

[Understanding Applying CoS Classifiers and Rewrite Rules to Interfaces](#)

## Example: Configuring Classifiers

### IN THIS SECTION

- [Requirements | 64](#)
- [Overview | 65](#)
- [Verification | 66](#)

Packet classification associates incoming packets with a particular CoS servicing level. Classifiers associate packets with a forwarding class and loss priority and assign packets to output queues based on the associated forwarding class. You apply classifiers to ingress interfaces.

## Configuring Classifiers

### Step-by-Step Procedure

To configure an IEEE 802.1 BA classifier named `ba-classifier` as the default IEEE 802.1 classifier:

1. Associate code point 000 with forwarding class `be` and loss priority `low`:

```
[edit class-of-service classifiers]
user@switch# set ieee-802.1 ba-classifier import default forwarding-class be loss-priority
low code-points 000
```

2. Associate code point 011 with forwarding class `fcoe` and loss priority `low`:

```
[edit class-of-service classifiers]
user@switch# set ieee-802.1 ba-classifier forwarding-class fcoe loss-priority low code-points
011
```

3. Associate code point 100 with forwarding class `no-loss` and loss priority `low`:

```
[edit class-of-service classifiers]
user@switch# set ieee-802.1 ba-classifier forwarding-class no-loss loss-priority low code-
points 100
```

4. Associate code point 110 with forwarding class `nc` and loss priority `low`:

```
[edit class-of-service classifiers]
user@switch# set ieee-802.1 ba-classifier forwarding-class nc loss-priority low code-points
110
```

5. Apply the classifier to ingress interface `xe-0/0/10`:

```
[edit class-of-service interfaces]
user@switch# set xe-0/0/10 unit 0 classifiers ieee-802.1 ba-classifier
```

## Requirements

This example uses the following hardware and software components:

- One switch
- Any supported Junos release

# Overview

Junos supports three general types of classifiers:

- Behavior aggregate or CoS value traffic classifiers—Examine the CoS value in the packet header. The value in this single field determines the CoS settings applied to the packet. BA classifiers allow you to set the forwarding class and loss priority of a packet based on the Differentiated Services code point (DSCP or DSCP IPv6) value, IEEE 802.1p value, or MPLS EXP value. (EXP classifiers can be applied only to family mpls interfaces.)
- Fixed classifiers. Fixed classifiers classify all ingress traffic on a physical interface into one forwarding class, regardless of the CoS bits in the VLAN header or the DSCP bits in the IP packet header.
- Multifield traffic classifiers—Examine multiple fields in the packet, such as source and destination addresses and source and destination port numbers of the packet. With multifield classifiers, you set the forwarding class and loss priority of a packet based on firewall filter rules.

This example describes how to configure a BA classifier called `ba-classifier` as the default IEEE 802.1 mapping of incoming traffic to forwarding classes, and apply it to ingress interface `xe-0/0/10`. The BA classifier assigns loss priorities, as shown in [Table 22 on page 65](#), to incoming packets in the four default forwarding classes. You can adapt the example to DSCP traffic by specifying a DSCP classifier instead of an IEEE classifier, and by applying DSCP bits instead of CoS bits.

To set multifield classifiers, use firewall filter rules.

**Table 22: ba-classifier Loss Priority Assignments**

Forwarding Class	CoS Traffic Type	ba-classifier Loss Priority to IEEE 802.1p Code Point Mapping	Packet Drop Attribute
be	Best-effort traffic	Low loss priority code point: 000	drop
fcoe	Guaranteed delivery for Fibre Channel over Ethernet (FCoE) traffic	Low loss priority code point: 011	no-loss
no-loss	Guaranteed delivery for TCP traffic	Low loss priority code point: 100	no-loss

**Table 22: ba-classifier Loss Priority Assignments (Continued)**

Forwarding Class	CoS Traffic Type	ba-classifier Loss Priority to IEEE 802.1p Code Point Mapping	Packet Drop Attribute
nc	Network-control traffic	Low loss priority code point: 110	drop

## Verification

### IN THIS SECTION

- [Verifying the Classifier Configuration | 66](#)
- [Verifying the Ingress Interface Configuration | 67](#)

To verify the classifier configuration, perform these tasks:

### Verifying the Classifier Configuration

#### Purpose

Verify that you configured the classifier with the correct forwarding classes, loss priorities, and code points.

#### Action

List the classifier configuration using the operational mode command `show configuration class-of-service classifiers ieee-802.1 ba-classifier`:

```
user@switch> show configuration class-of-service classifiers ieee-802.1 ba-classifier
  forwarding-class be {
    loss-priority low code-points 000;
  }
  forwarding-class fcoe {
    loss-priority low code-points 011;
  }
```



```

forwarding-class no-loss {
    loss-priority low code-points 100;
}
forwarding-class nc
    loss-priority low code-points 110;
}

```

## Verifying the Ingress Interface Configuration

### Purpose

Verify that the classifier ba-classifier is attached to ingress interface xe-0/0/10.

### Action

List the ingress interface using the operational mode command `show configuration class-of-service interfaces xe-0/0/10`:

```

user@switch> show configuration class-of-service interfaces xe-0/0/10
congestion-notification-profile fcoe-cnp;
unit 0 {
    classifiers {
        ieee-802.1 ba-classifier;
    }
}

```

## RELATED DOCUMENTATION

*Example: Configuring CoS Hierarchical Port Scheduling (ETS)*

*Defining CoS BA Classifiers (DSCP, DSCP IPv6, IEEE 802.1p)*

*Configuring a Global MPLS EXP Classifier*

*Configuring Rewrite Rules for MPLS EXP Classifiers*

*Monitoring CoS Classifiers*

*Understanding CoS Classifiers*

*Understanding Applying CoS Classifiers and Rewrite Rules to Interfaces*

# Monitoring CoS Classifiers

IN THIS SECTION

- Purpose | 68
- Action | 68
- Meaning | 68

## Purpose

Display the mapping of incoming CoS values to forwarding class and loss priority for each classifier.

## Action

To monitor CoS classifiers in the CLI, enter the CLI command:

```
user@switch> show class-of-service classifier
```

To monitor a particular classifier in the CLI, enter the CLI command:

```
user@switch> show class-of-service classifier name classifier-name
```

To monitor a particular type of classifier in the CLI, enter the CLI command:

```
user@switch> show class-of-service classifier type classifier-type
```

## Meaning

[Table 23 on page 68](#) summarizes key output fields for CoS classifiers.

Table 23: Summary of Key CoS Classifier Output Fields

Field	Values
Classifier	Name of a classifier.

**Table 23: Summary of Key CoS Classifier Output Fields (Continued)**

Field	Values
Code point type	Type of classifier: <ul style="list-style-type: none"> <li>• dscp—All classifiers of the DSCP type.</li> <li>• ieee-802.1—All classifiers of the IEEE 802.1 type.</li> <li>• ieee-mcast—All classifiers of the IEEE 802.1 multicast type.</li> <li>• exp—All classifiers of the MPLS exp type.</li> </ul>
Index	Internal index of the classifier.
Code point	DSCP or IEEE 802.1 code point value of the incoming packets, in bits. These values are used for classification.
Forwarding Class	Name of the forwarding class that the classifier assigns to an incoming packet. This class affects the forwarding and scheduling policies that are applied to the packet as it transits the switch.
Loss Priority	Loss priority value that the classifier assigns to the incoming packet based on its code point value.

## Understanding Default CoS Scheduling and Classification

### IN THIS SECTION

- [Default Classification | 70](#)
- [Default Scheduling | 75](#)
- [Default DCBX Advertisement | 79](#)
- [Default Scheduling and Classification Summary | 79](#)

If you do not explicitly configure classifiers and apply them to interfaces, the switch uses the default classifier to group ingress traffic into forwarding classes. If you do not configure scheduling on an interface, the switch uses the default schedulers to provide egress port resources for traffic. Default classification maps all traffic into default forwarding classes (best-effort, fcoe, no-loss, network-control, and mcast). Each default forwarding class has a default scheduler, so that the traffic mapped to each default forwarding class receives port bandwidth, prioritization, and packet drop characteristics.

The switch supports direct port scheduling and enhanced transmission selection (ETS), also known as hierarchical port scheduling, except on QFX5200 and QFX5210 switches.

Hierarchical scheduling groups IEEE 802.1p priorities (IEEE 802.1p code points, which classifiers map to forwarding classes, which in turn are mapped to output queues) into priority groups (forwarding class sets). If you use only the default traffic scheduling and classification, the switch automatically creates a default priority group that contains all of the priorities (which are mapped to forwarding classes and output queues), and assigns 100 percent of the port output bandwidth to that priority group. The forwarding classes (queues) in the default forwarding class set receive bandwidth based on the default classifier settings. The default priority group is transparent. It does not appear in the configuration and is used for Data Center Bridging Capability Exchange (DCBX) protocol advertisement.



**NOTE:** If you explicitly configure one or more priority groups on an interface, any forwarding class that is not assigned to a priority group on that interface receives *no bandwidth*. This means that if you configure hierarchical scheduling on an interface, every forwarding class (priority) that you want to forward traffic on that interface must belong to a forwarding class set (priority group). ETS is not supported on QFX5200 or QFX5210 switches.

The following sections describe:

## Default Classification

On switches except QFX10000 and NFX Series devices, the default classifiers assign unicast and multicast best-effort and network-control ingress traffic to default forwarding classes and loss priorities. The switch applies default unicast IEEE 802.1, unicast DSCP, and multidestination classifiers to each interface that does not have explicitly configured classifiers.

On QFX10000 switches and NFX Series devices, the default classifiers assign ingress traffic to default forwarding classes and loss priorities. The switch applies default IEEE 802.1, DSCP, and DSCP IPv6 classifiers to each interface that does not have explicitly configured classifiers. If you do not configure and apply EXP classifiers for MPLS traffic to logical interfaces, MPLS traffic on interfaces configured as family `mpls` uses the IEEE classifier.

If you explicitly configure one type of classifier but not other types of classifiers, the system uses only the configured classifier and does not use default classifiers for other types of traffic. There are two

default IEEE 802.1 classifiers: a trusted classifier for ports that are in trunk mode or tagged-access mode, and an untrusted classifier for ports that are in access mode.



**NOTE:** The default classifiers apply to unicast traffic except on QFX10000 switches and NFX Series devices. Tagged-access mode does not apply to QFX10000 switches or NFX Series devices.

[Table 24 on page 71](#) shows the default mapping of IEEE 802.1 code-point values to forwarding classes and loss priorities for ports in trunk mode or tagged-access mode.

**Table 24: Default IEEE 802.1 Classifiers for Ports in Trunk Mode or Tagged-Access Mode (Trusted Classifier)**

Code Point	Forwarding Class	Loss Priority
be (000)	best-effort	low
be1 (001)	best-effort	low
ef (010)	best-effort	low
ef1 (011)	fcoe	low
af11 (100)	no-loss	low
af12 (101)	best-effort	low
nc1 (110)	network-control	low
nc2 (111)	network-control	low

[Table 25 on page 72](#) shows the default mapping of IEEE 802.1p code-point values to forwarding classes and loss priorities for ports in access mode (all incoming traffic is mapped to best-effort forwarding classes).



**NOTE:** [Table 25 on page 72](#) applies only to unicast traffic except on QFX10000 switches and NFX Series devices.

**Table 25: Default IEEE 802.1 Classifiers for Ports in Access Mode (Untrusted Classifier)**

Code Point	Forwarding Class	Loss Priority
000	best-effort	low
001	best-effort	low
010	best-effort	low
011	best-effort	low
100	best-effort	low
101	best-effort	low
110	best-effort	low
111	best-effort	low

Table 26 on page 72 shows the default mapping of IEEE 802.1 code-point values to multideestination (multicast, broadcast, and destination lookup fail traffic) forwarding classes and loss priorities.



**NOTE:** Table 26 on page 72 does not apply to QFX10000 switches or NFX Series devices.

**Table 26: Default IEEE 802.1 Multideestination Classifiers**

Code Point	Forwarding Class	Loss Priority
be (000)	mcast	low
be1 (001)	mcast	low
ef (010)	mcast	low

**Table 26: Default IEEE 802.1 Multidestination Classifiers (Continued)**

Code Point	Forwarding Class	Loss Priority
ef1 (011)	mcast	low
af11 (100)	mcast	low
af12 (101)	mcast	low
nc1 (110)	mcast	low
nc2 (111)	mcast	low

Table 27 on page 73 shows the default mapping of DSCP code-point values to forwarding classes and loss priorities for DSCP IP and DCSP IPv6.



**NOTE:** Table 27 on page 73 applies only to unicast traffic except on QFX10000 switches and NFX Series devices.

**Table 27: Default DSCP IP and IPv6 Classifiers**

Code Point	Forwarding Class	Loss Priority
ef (101110)	best-effort	low
af11 (001010)	best-effort	low
af12 (001100)	best-effort	low
af13 (001110)	best-effort	low
af21 (010010)	best-effort	low
af22 (010100)	best-effort	low

**Table 27: Default DSCP IP and IPv6 Classifiers (Continued)**

Code Point	Forwarding Class	Loss Priority
af23 (010110)	best-effort	low
af31 (011010)	best-effort	low
af32 (011100)	best-effort	low
af33 (011110)	best-effort	low
af41 (100010)	best-effort	low
af42 (100100)	best-effort	low
af43 (100110)	best-effort	low
be (000000)	best-effort	low
cs1 (001000)	best-effort	low
cs2 (010000)	best-effort	low
cs3 (011000)	best-effort	low
cs4 (100000)	best-effort	low
cs5 (101000)	best-effort	low
nc1 (110000)	network-control	low
nc2 (111000)	network-control	low





**NOTE:** There are no default DSCP IP or IPv6 multdestination classifiers for multdestination traffic. DSCP IPv6 multdestination classifiers are not supported for multdestination traffic.

Table 28 on page 75 shows the default mapping of MPLS EXP code-point values to forwarding classes and loss priorities, which apply only on QFX10000 switches and NFX Series devices.

**Table 28: Default EXP Classifiers on QFX10000 Switches and NFX Series Devices**

Code Point	Forwarding Class	Loss Priority
000	best-effort	low
001	best-effort	high
010	expedited-forwarding	low
011	expedited-forwarding	high
100	assured-forwarding	low
101	assured-forwarding	high
110	network-control	low
111	network-control	high

## Default Scheduling

The default schedulers allocate egress bandwidth resources to egress traffic as shown in Table 29 on page 76:

**Table 29: Default Scheduler Configuration**

Default Scheduler and Queue Number	Transmit Rate (Guaranteed Minimum Bandwidth)	Shaping Rate (Maximum Bandwidth)	Excess Bandwidth Sharing	Priority	Buffer Size
best-effort forwarding class scheduler (queue 0)	5%  15% (QFX10000, NFX Series)	None	5%  15% (QFX10000, NFX Series)	low	5%  15% (QFX10000, NFX Series)
fcoe forwarding class scheduler (queue 3)	35%	None	35%	low	35%
no-loss forwarding class scheduler (queue 4)	35%	None	35%	low	35%
network-control forwarding class scheduler (queue 7)	5%  15% (QFX10000, NFX Series)	None	5%  15% (QFX10000, NFX Series)	low	5%  15% (QFX10000, NFX Series)
(Excluding QFX10000 and NFX Series)  mcast forwarding class scheduler (queue 8)	20%	None	20%	low	20%



**NOTE:** By default, the minimum guaranteed bandwidth (transmit rate) determines the amount of excess (extra) bandwidth that a queue can share. Extra bandwidth is allocated to queues in proportion to the transmit rate of each queue. On switches that support the excess-rate statement, you can override the default setting and configure the excess bandwidth percentage independently of the transmit rate on queues that are not strict-high priority queues.

By default, only the four (QFX10000 switches and NFX Series devices) or five (other switches) default schedulers shown in [Table 29 on page 76](#) have traffic mapped to them. Only the forwarding classes and

queues associated with the default schedulers receive default bandwidth, based on the default scheduler transmit rate. (You can configure schedulers and forwarding classes to allocate bandwidth to other queues or to change the bandwidth and other scheduling properties of a default queue.)

On QFX10000 switches and NFX Series devices, if a forwarding class does not transport traffic, the bandwidth allocated to that forwarding class is available to other forwarding classes. Unicast and multdestination (multicast, broadcast, and destination lookup fail) traffic use the same forwarding classes and output queues.

On switches other than QFX10000 and NFX Series devices, multdestination queue 11 receives enough bandwidth from the default multdestination scheduler to handle CPU-generated multdestination traffic.

On QFX10000 and NFX Series devices, default scheduling is port scheduling. Default hierarchical scheduling, known as enhanced transmission selection (ETS, defined in IEEE 802.1Qaz), allocates the total port bandwidth to the four default forwarding classes served by the four default schedulers, as defined by the four default schedulers. The result is the same as direct port scheduling. Configuring hierarchical port scheduling, however, enables you to group forwarding classes that carry similar types of traffic into forwarding class sets (also called priority groups), and to assign port bandwidth to each forwarding class set. The port bandwidth assigned to the forwarding class set is then assigned to the forwarding classes within the forwarding class set. This hierarchy enables you to control port bandwidth allocation with greater granularity, and enables hierarchical sharing of extra bandwidth to better utilize link bandwidth.

Except on QFX10000 switches and NFX Series devices, default hierarchical scheduling divides the total port bandwidth between two groups of traffic: unicast traffic and multdestination traffic. By default, unicast traffic consists of queue 0 (best-effort forwarding class), queue 3 (fcoe forwarding class), queue 4 (no-loss forwarding class), and queue 7 (network-control forwarding class). Unicast traffic receives and shares a total of 80 percent of the port bandwidth. By default, multdestination traffic (mcast queue 8) receives a total of 20 percent of the port bandwidth. So on a 10-Gigabit port, unicast traffic receives 8-Gbps of bandwidth and multdestination traffic receives 2-Gbps of bandwidth.



**NOTE:** Except on QFX5200, QFX5210, and QFX10000 switches and NFX Series devices, which do not support queue 11, multdestination queue 11 also receives a small amount of default bandwidth from the multdestination scheduler. CPU-generated multdestination traffic uses queue 11, so you might see a small number of packets egress from queue 11. In addition, in the unlikely case that firewall filter match conditions map multdestination traffic to a unicast forwarding class, that traffic uses queue 11.

Default scheduling uses weighted round-robin (WRR) scheduling. Each queue receives a portion (weight) of the total available interface bandwidth. The scheduling weight is based on the transmit rate of the default scheduler for that queue. For example, queue 7 receives a default scheduling weight of 5 percent, or 15 percent on QFX10000 and NFX Series devices, of the available bandwidth, and queue 4

receives a default scheduling weight of 35 percent of the available bandwidth. Queues are mapped to forwarding classes, so forwarding classes receive the default bandwidth for the queues to which they are mapped.

On QFX10000 switches and NFX Series devices, for example, queue 7 is mapped to the network-control forwarding class and queue 4 is mapped to the no-loss forwarding class. Each forwarding class receives the default bandwidth for the queue to which it is mapped. Unused bandwidth is shared with other default queues.

If you want non-default (unconfigured) queues to forward traffic, you should explicitly map traffic to those queues (configure the forwarding classes and queue mapping) and create schedulers to allocate bandwidth to those queues. By default, queues 1, 2, 5, and 6 are unconfigured.

Except on QFX5200, QFX5210, and QFX10000 switches and NFX Series devices, which do not support them, multidestination queues 9, 10, and 11 are unconfigured. Unconfigured queues have a default scheduling weight of 1 so that they can receive a small amount of bandwidth in case they need to forward traffic. However, queue 11 can use more of the default multidestination scheduler bandwidth if necessary to handle CPU-generated multidestination traffic.



**NOTE:** All four (two on QFX5200 and QFX5210 switches) multidestination queues have a scheduling weight of 1. Because by default multidestination traffic goes to queue 8, queue 8 receives almost all of the multidestination bandwidth. (There is no traffic on queue 9 and queue 10, and very little traffic on queue 11, so there is almost no competition for multidestination bandwidth.)

However, if you explicitly configure queue 9, 10, or 11 (by mapping code points to the unconfigured multidestination forwarding classes using the multidestination classifier), the explicitly configured queues share the multidestination scheduler bandwidth equally with default queue 8, because all of the queues have the same scheduling weight (1). To ensure that multidestination bandwidth is allocated to each queue properly and that the bandwidth allocation to the default queue (8) is not reduced too much, we strongly recommend that you configure a scheduler if you explicitly classify traffic into queue 9, 10, or 11.

If you map traffic to an unconfigured queue, the queue receives only the amount of excess bandwidth proportional to its default weight (1). The actual amount of bandwidth an unconfigured queue gets depends on how much bandwidth the other queues are using.

If some queues use less than their allocated amount of bandwidth, the unconfigured queues can share the unused bandwidth. Sharing unused bandwidth is one of the key advantages of hierarchical port scheduling. Configured queues have higher priority for bandwidth than unconfigured queues, so if a configured queue needs more bandwidth, then less bandwidth is available for unconfigured queues. Unconfigured queues always receive a minimum amount of bandwidth based on their scheduling weight (1). If you map traffic to an unconfigured queue, to allocate bandwidth to that queue, configure a scheduler for the forwarding class that is mapped to the queue.

## Default DCBX Advertisement

When you configure hierarchical scheduling on an interface, DCBX advertises each priority group, the priorities in each priority group, and the bandwidth properties of each priority and priority group.

If you do not configure hierarchical scheduling on an interface, DCBX advertises the automatically created default priority group and its priorities. DCBX also advertises the default bandwidth allocation of the priority group, which is 100 percent of the port bandwidth.

## Default Scheduling and Classification Summary

If you do not configure scheduling on an interface:

- Default classifiers classify ingress traffic.
- Default schedulers schedule egress traffic.
- DCBX advertises a single default priority group with 100 percent of the port bandwidth allocated to that priority group. All priorities (forwarding classes) are assigned to the default priority group and receive bandwidth based on their default schedulers. The default priority group is generated automatically and is not user-configurable.

## RELATED DOCUMENTATION

*Understanding CoS Packet Flow*

*Understanding CoS Hierarchical Port Scheduling (ETS)*

*Understanding Default CoS Settings*

*Understanding CoS Virtual Output Queues (VOQs) on QFX10000 Switches*

*Understanding Applying CoS Classifiers and Rewrite Rules to Interfaces*

*Understanding DCB Features and Requirements*

*Example: Configuring Unicast Classifiers*

*Example: Configuring Queue Schedulers*

# Understanding Applying CoS Classifiers and Rewrite Rules to Interfaces

## IN THIS SECTION

- Supported Classifier and Rewrite Rule Types | 80
- Ethernet Interfaces Supported for Classifier and Rewrite Rule Configuration | 82
- Default Classifiers | 86
- Default Rewrite Rules | 86
- Classifier Precedence | 86
- Classifier Behavior and Limitations | 88
- Rewrite Rule Precedence and Behavior | 89
- Classifier and Rewrite Rule Configuration Interaction with Ethernet Interface Configuration | 90
- Platform-Specific Behavior | 92

At ingress interfaces, classifiers group incoming traffic into classes based on the IEEE 802.1p, DSCP, or MPLS EXP *class of service* (CoS) code points in the packet header. At egress interfaces, you can use *rewrite rules* to change (re-mark) the code point bits before the interface forwards the packets.

You can apply classifiers and rewrite rules to interfaces to control the level of CoS applied to each packet as it traverses the system and the network. This topic describes:

## Supported Classifier and Rewrite Rule Types

Table 30 on page 80 shows the supported types of classifiers and rewrite rules supports:

Table 30: Supported Classifiers and Rewrite Rules

Classifier or Rewrite Rule Type	Description
Fixed classifier	Classifies all ingress traffic on a physical interface into one fixed forwarding class, regardless of the CoS bits in the packet header.
DSCP and DSCP IPv6 unicast classifiers	Classifies IP and IPv6 traffic into forwarding classes and assigns loss priorities to the traffic based on DSCP code point bits.

Table 30: Supported Classifiers and Rewrite Rules *(Continued)*

Classifier or Rewrite Rule Type	Description
IEEE 802.1p unicast classifier	Classifies Ethernet traffic into forwarding classes and assigns loss priorities to the traffic based on IEEE 802.1p code point bits.
MPLS EXP classifier	Classifies MPLS traffic into forwarding classes and assigns loss priorities to the traffic on interfaces configured as family mpls.  QFX5200, QFX5100, EX4600 switches use one global EXP classifier on all family mpls switch interfaces.
DSCP multidestination classifier (also used for IPv6 multidestination traffic)  <b>NOTE:</b> This applies only to switches that use different classifiers for unicast and multidestination traffic. It does not apply to switches that use the same classifiers for unicast and multidestination traffic.	Classifies IP and IPv6 multicast, broadcast, and destination lookup fail (DLF) traffic into multidestination forwarding classes. Multidestination classifiers are applied to all interfaces and cannot be applied to individual interfaces.
IEEE 802.1p multidestination classifier  <b>NOTE:</b> This applies only to switches that use different classifiers for unicast and multidestination traffic. It does not apply to switches that use the same classifiers for unicast and multidestination traffic.	Classifies Ethernet multicast, broadcast, and destination lookup fail (DLF) traffic into multidestination forwarding classes. Multidestination classifiers are applied to all interfaces and cannot be applied to individual interfaces.
DSCP and DSCP IPv6 rewrite rules	Re-marks the DSCP code points of IP and IPv6 packets before forwarding the packets.
IEEE 802.1p rewrite rule	Re-marks the IEEE 802.1p code points of Ethernet packets before forwarding the packets.
MPLS EXP rewrite rule	Re-marks the EXP code points of MPLS packets before forwarding the packets on interfaces configured as family mpls.



**NOTE:** On switches that support native Fibre Channel (FC) interfaces, you can specify a rewrite value on native FC interfaces (NP\_Ports) to set the IEEE 802.1p code point of incoming FC traffic when the NP\_Port encapsulates the FC packet in Ethernet before forwarding it to the FCoE network (see *Understanding CoS IEEE 802.1p Priority Remapping on an FCoE-FC Gateway*).

DSCP, IEEE 802.1p, and MPLS EXP classifiers are behavior aggregate (BA) classifiers. On QFX5100, QFX5200, and EX4600 switches, unlike DSCP and IEEE 802.1p classifiers, EXP classifiers are global and apply only to all interfaces that are configured as `family mpls`.

Unlike DSCP and IEEE 802.1p BA classifiers, there is no default EXP classifier. Also unlike DSCP and IEEE 802.1p classifiers, for MPLS traffic on `family mpls` interfaces only, EXP classifiers overwrite fixed classifiers. (An interface that has a fixed classifier uses the EXP classifier for MPLS traffic, not the fixed classifier, and the fixed classifier is used for all other traffic.)

On switches that use different classifiers for unicast and multdestination traffic, multdestination classifiers are global and apply to all interfaces; you cannot apply a multdestination classifier to individual interfaces.

Classifying packets into forwarding classes assigns packets to the output queues mapped to those forwarding classes. The traffic classified into a forwarding class receives the CoS scheduling configured for the output queue mapped to that forwarding class.



**NOTE:** In addition to BA classifiers and fixed classifiers, which classify traffic based on the CoS field in the packet header, you can use firewall filters to configure multifield (MF) classifiers. MF classifiers classify traffic based on more than one field in the packet header and take precedence over BA and fixed classifiers.

## Ethernet Interfaces Supported for Classifier and Rewrite Rule Configuration

To apply a classifier to incoming traffic or a rewrite rule to outgoing traffic, you need to apply the classifier or rewrite rule to one or more interfaces. When you apply a classifier or rewrite rule to an interface, the interface uses the classifier to group incoming traffic into forwarding classes and uses the rewrite rule to re-mark the CoS code point value of each packet before it leaves the system.

Not all interfaces types support all types of CoS configuration. This section describes:



## Interface Types That Support Classifier and Rewrite Rule Configuration

You can apply classifiers and rewrite rules to Ethernet interfaces. For Layer 3 LAGs, configure BA or fixed classifiers on the LAG (ae) interface. The classifier configured on the LAG is valid on all of the LAG member interfaces.

On switches that support native FC interfaces, you can apply fixed classifiers to native FC interfaces (NP\_Ports). You cannot apply other types of classifiers or rewrite rules to native FC interfaces. You can rewrite the value of the IEEE 802.1p code point of incoming FC traffic when the interface encapsulates it in Ethernet before forwarding it to the FCoE network as described in *Understanding CoS IEEE 802.1p Priority Remapping on an FCoE-FC Gateway*.

## Classifier and Rewrite Rule Physical and Logical Ethernet Interface Support

The Ethernet ports can function as:

- Layer 2 physical interfaces (family ethernet-switching)
- Layer 2 logical interfaces (family ethernet-switching)
- Layer 3 physical interfaces (family inet/inet6)
- Layer 3 logical interfaces (family inet/inet6)
- MPLS interfaces (family mpls)

## Ethernet Interface Support for Most QFX Series Switches

On most QFX Series switches you can apply classifiers and rewrite rules to Layer 2 logical interfaces or to Layer 3 physical interfaces. [Table 31 on page 84](#) shows on which interfaces you can configure and apply classifiers and rewrite rules.



**NOTE:** The CoS feature support listed in this table is identical on single interfaces and aggregated Ethernet interfaces.

**Table 31: Ethernet Interface Support for Classifier and Rewrite Rule Configuration (QFX5000 and EX4600 Switches)**

CoS Classifiers and Rewrite Rules	Layer 2 Physical Interfaces	Layer 2 Logical Interface ( <b>unit *</b> applies rule to all logical interfaces)	Layer 3 Physical Interfaces (If at Least One Logical Layer 3 Interface Is Defined)	Layer 3 Logical Interfaces
Fixed classifier	No	Yes	Yes	No
DSCP classifier	No	Yes	Yes	No
DSCP IPv6 classifier	No	Yes	Yes	No
IEEE 802.1p classifier	No	Yes	Yes	No
EXP classifier	Global classifier, applies only to all switch interfaces that are configured as family mpls. Cannot be configured on individual interfaces.			
DSCP rewrite rule	No	Yes	Yes	No
DSCP IPv6 rewrite rule	No	Yes	Yes	No
IEEE 802.1p rewrite rule	No	Yes	Yes	No
EXP rewrite rule	No	Yes	Yes	No



**NOTE:** IEEE 802.1p multdestination and DSCP multdestination classifiers are applied to all interfaces and cannot be applied to individual interfaces. No DSCP IPv6 multdestination classifier is supported. IPv6 multdestination traffic uses the DSCP multdestination classifier.

#### Ethernet Interface Support for QFX10000 Switches

On QFX10000 switches, you can apply classifiers and rewrite rules only to Layer 2 logical interface unit 0. You can apply different classifiers and rewrite rules to different Layer 3 logical interfaces. You cannot apply classifiers or rewrite rules to Layer 2 or Layer 3 physical interfaces. [Table 32 on page 85](#) shows on which interfaces you can configure and apply classifiers and rewrite rules.



**NOTE:** The CoS feature support listed in this table is identical on single interfaces and aggregated Ethernet interfaces.

**Table 32: Ethernet Interface Support for Classifier and Rewrite Rule Configuration (QFX10000 Switches)**

CoS Classifiers and Rewrite Rules	Layer 2 Physical Interfaces	Layer 2 Logical Interface (Unit 0 Only)	Layer 3 Physical Interfaces	Layer 3 Logical Interfaces
Fixed classifier	No	Yes	No	Yes
DSCP classifier	No	Yes	No	Yes
DSCP IPv6 classifier	No	Yes	No	Yes
IEEE 802.1p classifier	No	Yes	No	Yes
EXP classifier	No	Yes	No	Yes
DSCP rewrite rule	No	Yes	No	Yes
DSCP IPv6 rewrite rule	No	Yes	No	Yes
IEEE 802.1p rewrite rule	No	Yes	No	Yes
EXP rewrite rule	No	Yes	No	Yes

### Routed VLAN Interfaces (RVIs) and Integrated Routing and Bridging (IRB) Interfaces

You cannot apply classifiers and rewrite rules directly to routed VLAN interfaces (RVIs) or integrated routing and bridging (IRB) interfaces because the members of RVIs and IRBs are VLANs, not ports.

However, you can apply classifiers and rewrite rules to the VLAN port members of an *RV* or an *IRB*. You can also apply MF classifiers to *RVIs* and *IRBs*.

## Default Classifiers

If you do not explicitly configure classifiers on an Ethernet interface, the switch applies default classifiers so that the traffic receives basic CoS treatment. The factors that determine the default classifier applied to the interface include the interface type (Layer 2 or Layer 3), the port mode (trunk, tagged-access, or access), and whether logical interfaces have been configured.

The switch applies default classifiers using the following rules:

- If the physical interface has at least one Layer 3 *logical interface* configured, the logical interfaces use the default DSCP classifier.
- If the physical interface has a Layer 2 logical interface in trunk mode or tagged-access mode, it uses the default IEEE 802.1p trusted classifier.
- If the physical interface has a Layer 2 logical interface in access mode, it uses the default IEEE 802.1p untrusted classifier.
- If the physical interface has no logical interface configured, no default classifier is applied.
- On switches that use different classifiers for unicast and multidestination traffic, the default multdestination classifier is the IEEE 802.1p multdestination classifier.
- There is no default MPLS EXP classifier.

## Default Rewrite Rules

No default rewrite rules are applied to interfaces. If you want to re-mark packets at the egress interface, you must explicitly configure a rewrite rule.

## Classifier Precedence

You can apply multiple classifiers (MF, fixed, IEEE 802.1p, DSCP, or EXP) to an Ethernet interface to handle different types of traffic. (EXP classifiers are global and apply only to all MPLS traffic on all *family mpls* interfaces.) When you apply more than one classifier to an interface, the system uses an order of precedence to determine which classifier to use on interfaces:

### Classifier Precedence on Physical Ethernet Interfaces

The precedence of classifiers on physical interfaces, from the highest-priority classifier to the lowest-priority classifier, is:

- MF classifier on a logical interface (no classifier has a higher priority than MF classifiers)
- Fixed classifier on the physical interface
- DSCP or DSCP IPv6 classifier on the physical interface
- IEEE 802.1p classifier on the physical interface



**NOTE:** If an EXP classifier is configured, MPLS traffic uses the EXP classifier on all family `mpls` interfaces, even if an MF or fixed classifier is applied to the interface. If an EXP classifier is not configured, then if a fixed classifier is applied to the interface, the MPLS traffic uses the fixed classifier. If no EXP classifier and no fixed classifier is applied to the interface, MPLS traffic is treated as best-effort traffic. DSCP classifiers are not applied to MPLS traffic.

You can apply a DSCP classifier, an IEEE 802.1p classifier, and an EXP classifier on a physical interface. When all three classifiers are on an interface, IP traffic uses the DSCP classifier, MPLS traffic on family `mpls` interfaces uses the EXP classifier, and all other traffic uses the IEEE classifier.



**NOTE:** You cannot apply a fixed classifier and a DSCP or IEEE classifier to the same interface. If a DSCP classifier, an IEEE classifier, or both are on an interface, you cannot apply a fixed classifier to that interface unless you first delete the DSCP and IEEE classifiers. If a fixed classifier is on an interface, you cannot apply a DSCP classifier or an IEEE classifier unless you first delete the fixed classifier.

## Classifier Precedence on Logical Ethernet Interfaces

The precedence of classifiers on logical interfaces, from the highest priority classifier to the lowest priority classifier, is:

- MF classifier on a logical interface (no classifier has a higher priority than MF classifiers).
- Fixed classifier on the logical interface.
- DSCP or DSCP IPv6 classifier on the physical or logical interface..
- IEEE 802.1p classifier on the physical or logical interface.



**NOTE:** If a global EXP classifier is configured, MPLS traffic uses the EXP classifier on all family `mpls` interfaces, even if a fixed classifier is applied to the interface. If a global EXP classifier is not configured, then:

- If a fixed classifier is applied to the interface, the MPLS traffic uses the fixed classifier. If no EXP classifier and no fixed classifier is applied to the interface, MPLS traffic is treated as best-effort traffic.

You can apply both a DSCP classifier and an IEEE 802.1p classifier on a logical interface. When both a DSCP and an IEEE classifier are on an interface, IP traffic uses the DSCP classifier, and all other traffic uses the IEEE classifier. Only MPLS traffic on interfaces configured as `family mpls` uses the EXP classifier.

## Classifier Behavior and Limitations

Consider the following behaviors and constraints when you apply classifiers to Ethernet interfaces.

- You can configure only one DSCP classifier (IP or IPv6) on a physical interface. You cannot configure both types of DSCP classifier on one physical interface. Both IP and IPv6 traffic use whichever DSCP classifier is configured on the interface.
- When you configure a DSCP or a DSCP IPv6 classifier on a physical interface and the physical interface has at least one logical Layer 3 interface, all packets (IP, IPv6, and non-IP) use that classifier.
- An interface with both a DSCP classifier (IP or IPv6) and an IEEE 802.1p classifier uses the DSCP classifier for IP and IPv6 packets, and uses the IEEE classifier for all other packets.
- Fixed classifiers and BA classifiers (DSCP and IEEE classifiers) are not permitted simultaneously on an interface. If you configure a fixed classifier on an interface, you cannot configure a DSCP or an IEEE classifier on that interface. If you configure a DSCP classifier, an IEEE classifier, or both classifiers on an interface, you cannot configure a fixed classifier on that interface.
- When you configure an IEEE 802.1p classifier on a physical interface and a DSCP classifier is not explicitly configured on that interface, the interface uses the IEEE classifier for all types of packets. No default DSCP classifier is applied to the interface. (In this case, if you want a DSCP classifier on the interface, you must explicitly configure it and apply it to the interface.)
- The system does not apply a default classifier to a physical interface until you create a logical interface on that physical interface. If you configure a Layer 3 logical interface, the system uses the default DSCP classifier. If you configure a Layer 2 logical interface, the system uses the default IEEE 802.1p trusted classifier if the port is in trunk mode or tagged-access mode, or the default IEEE 802.1p untrusted classifier if the port is in access mode.
- MF classifiers configured on logical interfaces take precedence over BA and fixed classifiers, with the exception of the global EXP classifier, which is always used for MPLS traffic on `family mpls` interfaces. (Use firewall filters to configure MF classifiers.) When BA or fixed classifiers are present on an interface, you can still configure an MF classifier on that interface.
- There is no default EXP classifier for MPLS traffic.

- You can configure up to 64 EXP classifiers.

All family `mpls` switch interfaces use the EXP classifier specified using this configuration statement to classify MPLS traffic, even on interfaces that have a fixed classifier. No other traffic uses the EXP classifier.

## Rewrite Rule Precedence and Behavior

The following rules apply on Ethernet interfaces for rewrite rules:

- If you configure one DSCP (or DSCP IPv6) rewrite rule and one IEEE 802.1p rewrite rule on an interface, both rewrite rules take effect. Traffic with IP and IPv6 headers use the DSCP rewrite rule, and traffic with a VLAN tag uses the IEEE rewrite rule.
- If you do not explicitly configure a rewrite rule, there is no default rewrite rule, so the system does not apply any rewrite rule to the interface.
- You can apply a DSCP rewrite rule or a DSCP IPv6 rewrite rule to an interface, but you cannot apply both a DSCP and a DSCP IPv6 rewrite rule to the same interface. Both IP and IPv6 packets use the same DSCP rewrite rule, regardless of whether the configured rewrite rule is DSCP or DSCP IPv6.
- MPLS EXP rewrite rules apply only to logical interfaces on family `mpls` interfaces. You cannot apply to an EXP rewrite rule to a physical interface. You can configure up to 64 EXP rewrite rules, but you can only use 16 EXP rewrite rules at any time on the switch.
- A logical interface can use both DSCP (or DSCP IPv6) and EXP rewrite rules.
- DSCP and DSCP IPv6 rewrite rules are not applied to MPLS traffic.
- If the switch is performing penultimate hop popping (PHP), EXP rewrite rules do not take effect. If both an EXP classifier and an EXP rewrite rule are configured on the switch, then the EXP value from the last popped label is copied into the inner label. If either an EXP classifier or an EXP rewrite rule (but not both) is configured on the switch, then the inner label EXP value is sent unchanged.



**NOTE:** On each physical interface, either all forwarding classes that are being used on the interface must have rewrite rules configured or no forwarding classes that are being used on the interface can have rewrite rules configured. On any physical port, do not mix forwarding classes with rewrite rules and forwarding classes without rewrite rules.



**NOTE:** Rewrite rules are applied *before* the egress filter is matched to traffic. Because the code point rewrite occurs before the egress filter is matched to traffic, the egress filter match is based on the rewrite value, not on the original code point value in the packet.

## Classifier and Rewrite Rule Configuration Interaction with Ethernet Interface Configuration

This section focuses on BA classifiers, but the interaction between BA classifiers and interfaces described in this section also applies to fixed classifiers and rewrite rules.



**NOTE:** See *Defining CoS BA Classifiers (DSCP, DSCP IPv6, IEEE 802.1p)* for how to configure multdestination classifiers and see *Configuring a Global MPLS EXP Classifier* for how to configure EXP classifiers.

On switches that use different classifiers for unicast and multdestination traffic, multdestination classifiers are global and apply to all switch interfaces.

There are two components to applying classifiers or rewrite rules to interfaces:

1. Setting the interface family (inet, inet6, or ethernet-switching; ethernet-switching is the default interface family) in the [edit interfaces] configuration hierarchy.
2. Applying a classifier or rewrite rule to the interface in the [edit class-of-service] hierarchy.

These are separate operations that can be set and committed at different times. Because the type of classifier or rewrite rule you can apply to an interface depends on the interface family configuration, the system performs checks to ensure that the configuration is valid. The method the system uses to notify you of an invalid configuration depends on the set operation that causes the invalid configuration.

If applying the classifier or rewrite rule to the interface in the [edit class-of-service] hierarchy causes an invalid configuration, the system rejects the configuration and returns a commit check error.

If setting the interface family in the [edit interfaces] configuration hierarchy causes an invalid configuration, the system creates a syslog error message. If you receive the error message, you need to remove the classifier or rewrite rule configuration from the logical interface and apply it to the physical interface, or remove the classifier or rewrite rule configuration from the physical interface and apply it to the logical interface. For classifiers, if you do not take action to correct the error, the system programs the default classifier for the interface family on the interface. (There are no default rewrite rules. If the commit check fails, no rewrite rule is applied to the interface.)

Two scenarios illustrate these situations:

- Applying a classifier to an Ethernet interface causes a commit check error
- Configuring the Ethernet interface family causes a syslog error

These scenarios differ on different switches because some switches support classifiers on physical Layer 3 interfaces but not on logical Layer 3 interfaces, while other switches support classifiers on logical Layer 3 interfaces but not on physical Layer 3 interfaces.



Two scenarios illustrate these situations:



**NOTE:** Both of these scenarios also apply to fixed classifiers and rewrite rules.

## QFX5000 Series and EX4600 Switch Scenarios

### *Scenario 1: Applying a Classifier to an Ethernet Interface Causes a Commit Check Error*

In Scenario 1, we set the interface family, and then specify an invalid classifier.

1. Set and commit the interface as a Layer 3 (family `inet`) interface:

```
[edit interfaces]
user@switch# set xe-0/0/20 unit 0 family inet
user@switch# commit
```

This commit operation succeeds.

2. Set and commit a DSCP classifier on the logical interface (this example uses a DSCP classifier named `dscp1`):

```
[edit class-of-service]
user@switch# set interfaces xe-0/0/20 unit 0 classifiers dscp dscp1
user@switch# commit
```

This configuration is not valid, because it attempts to apply a classifier to a Layer 3 logical interface. Because the failure is caused by the class-of-service configuration and not by the interface configuration, the system rejects the commit operation and issues a commit error, not a syslog message.

Note that the commit operation succeeds if you apply the classifier to the physical Layer 3 interface as follows:

```
[edit class-of-service]
user@switch# set interfaces xe-0/0/20 classifiers dscp dscp1
user@switch# commit
```

Because the logical unit is not specified, the classifier is applied to the physical Layer 3 interface in a valid configuration, and the commit check succeeds.

### *Scenario 2: Configuring the Ethernet Interface Family Causes a Syslog Error*

In Scenario 2, we set the classifier first, and then set an invalid interface type.

1. Set and commit a DSCP classifier on a logical interface that has no existing configuration:

```
[edit class-of-service]
user@switch# set interfaces xe-0/0/20 unit 0 classifiers dscp dscp1
user@switch# commit
```

This commit succeeds. Because no explicit configuration existed on the interface, it is by default a Layer 2 (family ethernet-switching) interface. Layer 2 logical interfaces support BA classifiers, so applying the classifier is a valid configuration.

2. Set and commit the interface as a Layer 3 interface (family inet) interface:

```
[edit interfaces]
user@switch# set xe-0/0/20 unit 0 family inet
user@switch# commit
```

This configuration is not valid because it attempts to change an interface from Layer 2 (family ethernet-switching) to Layer 3 (family inet) when a classifier has already been applied to a logical interface. Layer 3 logical interfaces do not support classifiers. Because the failure is caused by the interface configuration and not by the class-of-service configuration, the system does not issue a commit error, but instead issues a syslog message.

When the system issues the syslog message, it programs the default classifier for the interface type on the interface. In this scenario, the interface has been configured as a Layer 3 interface, so the system applies the default DSCP profile to the physical Layer 3 interface.

In this scenario, to install a configured DSCP classifier, remove the misconfigured classifier from the Layer 3 logical interface and apply it to the Layer 3 physical interface. For example:

```
[edit]
user@switch# delete class-of-service interfaces xe-0/0/20 unit 0 classifiers dscp dscp1
user@switch# commit
user@switch# set class-of-service interfaces xe-0/0/20 classifiers dscp dscp1
user@switch# commit
```

## Platform-Specific Behavior

Use the following table to review platform-specific behaviors for your platforms.

Table 33: Platform-Specific Behavior

Platform	Difference
QFX5000 Series and EX4600 switches	<ul style="list-style-type: none"> <li>On QFX5100, QFX5200, and EX4600 switches, you can apply classifier and rewrite rules to Layer 3 physical interfaces if at least one logical Layer 3 interface is configured on the physical interface. The CoS you configure on a Layer 3 physical interface is applied to all of the Layer 3 logical interfaces on that physical interface. This means that each Layer 3 interface uses the same classifiers and rewrite rules for all of the Layer 3 traffic on that interface.</li> <li>On QFX5100, QFX5200, and EX4600 switches, configure an EXP classifier and configure it as the global system default EXP classifier.</li> </ul>
QFX 10000 Series	<ul style="list-style-type: none"> <li>QFX10000 switches do not support global EXP classifiers. You can apply the same EXP classifier or different EXP classifiers to different family mpls interfaces.</li> </ul> <p>If you want to classify MPLS traffic using EXP bits on QFX10000 switches, configure an EXP classifier and apply it to a logical interface that is configured as family mpls.</p>

## RELATED DOCUMENTATION

*Understanding CoS Packet Flow*

*Configuring CoS*

## Understanding CoS Code-Point Aliases

A code-point alias assigns a name to a pattern of code-point bits. You can use this name instead of the bit pattern when you configure other CoS components such as classifiers and *rewrite rules*.



**NOTE:** This topic applies to all EX Series switches except the EX4600. Because the EX4600 uses a different chipset than other EX Series switches, the code-point aliases on EX4600 match those on QFX Series switches. For EX4600 code-point aliases, see *Understanding CoS Code-Point Aliases*.

Behavior aggregate classifiers use class-of-service (CoS) values such as Differentiated Services Code Points (DSCPs) or IEEE 802.1 bits to associate incoming packets with a particular forwarding class and the CoS servicing level associated with that forwarding class. You can assign a meaningful name or alias to the CoS values and use that alias instead of bits when configuring CoS components. These aliases are not part of the specifications but are well known through usage. For example, the alias for DSCP 101110 is widely accepted as ef (expedited forwarding).

When you configure forwarding classes and define classifiers, you can refer to the markers by alias names. You can configure code point alias names for user-defined classifiers. If the value of an alias changes, it alters the behavior of any classifier that references it.

You can configure code-point aliases for the following type of CoS markers:

- dscp or dscp-ipv6—Handles incoming IP and IPv6 packets.
- ieee-802.1—Handles Layer 2 frames.

[Table 34 on page 94](#) shows the default mapping of code-point aliases to IEEE code points.

**Table 34: Default IEEE 802.1 Code-Point Aliases**

CoS Value Types	Mapping
be	000
be1	001
ef	010
ef1	011
af11	100
af12	101

**Table 34: Default IEEE 802.1 Code-Point Aliases (Continued)**

CoS Value Types	Mapping
nc1	110
nc2	111

[Table 35 on page 95](#) shows the default mapping of code-point aliases to DSCP and DSCP IPv6 code points.

**Table 35: Default DSCP and DSCP IPv6 Code-Point Aliases**

CoS Value Types	Mapping
ef	101110
af11	001010
af12	001100
af13	001110
af21	010010
af22	010100
af23	010110
af31	011010
af32	011100
af33	011110
af41	100010

Table 35: Default DSCP and DSCP IPv6 Code-Point Aliases (*Continued*)

CoS Value Types	Mapping
af42	100100
af43	100110
be	000000
cs1	001000
cs2	010000
cs3	011000
cs4	100000
cs5	101000
nc1	110000
nc2	111000

## RELATED DOCUMENTATION

*Understanding Junos CoS Components*

*Defining CoS Code-Point Aliases*

## Defining CoS Code-Point Aliases

You can use code-point aliases to streamline the process of configuring CoS features on your switch. A code-point alias assigns a name to a pattern of code-point bits. You can use this name instead of the bit pattern when you configure other CoS components such as classifiers and rewrite rules.

You can configure code-point aliases for the following CoS marker types:

- DSCP or DSCP IPv6—Handles incoming IPv4 or IPv6 packets.
- IEEE 802.1p—Handles Layer 2 frames.

To configure a code-point alias:

1. Specify a CoS marker type (IEEE 802.1 or DSCP).
2. Assign an alias.
3. Specify the code point that corresponds to the alias.

```
[edit class-of-service code-point-aliases]
user@switch# set (dscp | dscp-ipv6 | ieee-802.1) alias-name code-point-bits
```

For example, to configure a code-point alias for an IEEE 802.1 CoS marker type that has the alias name be2 and maps to the code-point bits 001:

```
[edit class-of-service code-point-aliases]
user@switch# set ieee-802.1 be2 001
```

## RELATED DOCUMENTATION

*Monitoring CoS Code-Point Value Aliases*

*Understanding CoS Code-Point Aliases*

## Monitoring CoS Code-Point Value Aliases

### IN THIS SECTION

- Purpose | 98
- Action | 98
- Meaning | 98

Purpose

Use the monitoring functionality to display information about the CoS code-point value aliases that the system is currently using to represent DSCP and IEEE 802.1p code point bits.

Action

To monitor CoS value aliases in the CLI, enter the CLI command:

```
user@switch> show class-of-service code-point-aliases
```

To monitor a specific type of code-point alias (DSCP, DSCP IPv6, IEEE 802.1, or MPLS EXP) in the CLI, enter the CLI command:

```
user@switch> show class-of-service code-point-aliases ieee-802.1
```

Meaning

[Table 36 on page 98](#) summarizes key output fields for CoS value aliases.

Table 36: Summary of Key CoS Value Alias Output Fields

Field	Values
Code point type	Type of the CoS value: <ul style="list-style-type: none"><li>dscp—Examines Layer 3 packet headers for IP packet classification.</li><li>dscp-ipv6—Examines Layer 3 packet headers for IPv6 packet classification.</li><li>ieee-802.1—Examines Layer 2 packet headers for packet classification.</li><li>exp—Examines MPLS packet headers for packet classification.</li></ul>
Alias	Name given to a set of bits—for example, af11 is a name for bits 001010.
Bit pattern	Set of bits associated with the alias.



## RELATED DOCUMENTATION

| *Defining CoS Code-Point Aliases*

## Understanding CoS Forwarding Classes

### IN THIS SECTION

- [Default Forwarding Classes | 100](#)
- [Forwarding Class Configuration Rules | 101](#)
- [Lossless Transport Support | 103](#)
- [Platform-Specific Forwarding Class Behavior | 104](#)

Forwarding classes group traffic and assign the traffic to output queues. Each forwarding class is mapped to an output queue. Classification maps incoming traffic to forwarding classes based on the code point bits in the packet or frame header. Forwarding class to queue mapping defines the output queue used for the traffic classified into a forwarding class.

A classifier must associate each packet with one of the following default forwarding classes or with a user-configured forwarding class to assign an output queue to the packet:

- **fcoe**—Guaranteed delivery for Fibre Channel over Ethernet (FCoE) traffic.
- **no-loss**—Guaranteed delivery for TCP lossless traffic.
- **best-effort**—Provides best-effort delivery without a service profile. Loss priority is typically not carried in a class-of-service (CoS) value.
- **network-control**—Supports protocol control and is typically high priority.
- **mcast**—Delivery of multdestination (multicast, broadcast, and destination lookup fail) packets.

Depending on your platform, the switch supports up to 12 forwarding classes, thus enabling flexible, differentiated, packet classification. For example, you can configure multiple classes of best-effort traffic such as **best-effort**, **best-effort1**, and **best-effort2**.

Most QFX devices support 8 queues for unicast traffic (queues 0 through 7) and 4 output queues for multdestination traffic (queues 8 through 11). Forwarding classes mapped to unicast queues are associated with unicast traffic, and forwarding classes mapped to multdestination queues are associated with multdestination traffic. You cannot map unicast and multdestination traffic to the same queue.

You cannot map a strict-high priority queue to a multidestination forwarding class because queues 8 through 11 do not support strict-high priority configuration.

## Default Forwarding Classes

Table 37 on page 100 shows the four default forwarding classes that apply to all QFX devices. You can rename the forwarding classes. Assigning a new forwarding class name does not alter the default classification or scheduling applied to the queue that is mapped to that forwarding class. CoS configurations can be complex, so unless it is required by your scenario, we recommend that you use the default class names and queue number associations.

**Table 37: Default Forwarding Classes**

Forwarding Class Name	Default Queue Mapping	Comments
best-effort	0	<p>The software does not apply any special CoS handling to best-effort traffic. This is a backward compatibility feature. Best-effort traffic is usually the first traffic to be dropped during periods of network congestion.</p> <p>By default, this is a lossy forwarding class with a packet drop attribute of drop.</p>
fcoe	3	<p>By default, the fcoe forwarding class is a lossless forwarding class designed to handle Fibre Channel over Ethernet (FCoE) traffic. The no-loss packet drop attribute is applied by default.</p> <p><b>NOTE:</b> By convention, deployments with converged server access typically use IEEE 802.1p priority 3 (011) for FCoE traffic. The default mapping of the fcoe forwarding class is to queue 3. Apply <i>priority-based flow control</i> (PFC) to the entire FCoE data path to configure the end-to-end lossless behavior that FCoE requires.</p> <p>We recommend that you use priority 3 for FCoE traffic unless your network architecture requires that you use a different priority.</p>
no-loss	4	<p>By default, this is a lossless forwarding class with a packet drop attribute of no-loss.</p>

**Table 37: Default Forwarding Classes (Continued)**

Forwarding Class Name	Default Queue Mapping	Comments
network-control	7	<p>The software delivers packets in this service class with a high priority. (These packets are not delay-sensitive.)</p> <p>Typically, these packets represent routing protocol hello or keepalive messages. Because loss of these packets jeopardizes proper network operation, packet delay is preferable to packet discard.</p> <p>By default, this is a lossy forwarding class with a packet drop attribute of drop.</p>

**Table 38: Default Forwarding Classes for Multidestination Packets**

Forwarding Class Name	Default Queue Mapping	Comments
mcast	8	<p>The software does not apply any special CoS handling to the multidestination packets. These packets are usually dropped under congested network conditions.</p> <p>By default, this is a lossy forwarding class with a packet drop attribute of drop.</p>



**NOTE:** Mirrored traffic is always sent to the queue that corresponds to the multidestination forwarding class. The switched copy of the mirrored traffic is forwarded with the priority determined by the behavior aggregate classification process.

## Forwarding Class Configuration Rules

Take the following rules into account when you configure forwarding classes:

### Queue Assignment Rules

The following rules govern queue assignment:

- CoS configurations that specify more queues than the switch can support are not accepted. The commit operation fails with a detailed message that states the total number of queues available.

- All default CoS configurations are based on queue number. The name of the forwarding class that appears in the default configuration is the forwarding class currently mapped to that queue.
- Only unicast forwarding classes can be mapped to unicast queues (0 through 7), and only multdestination forwarding classes can be mapped to multdestination queues (8 through 11).
- Strict-high priority queues cannot be mapped to multdestination forwarding classes. (Strict-high priority traffic cannot be mapped to queues 8 through 11).
- If you map more than one forwarding class to a queue, all of the forwarding classes mapped to the same queue must have the same packet drop attribute: either all of the forwarding classes must be lossy or all of the forwarding classes must be lossless.

You can limit the amount of traffic that receives strict-high priority treatment on a strict-high priority queue by configuring a transmit rate. The transmit rate sets the amount of traffic on the queue that receives strict-high priority treatment. The switch treats traffic that exceeds the transmit rate as low priority traffic that receives the queue excess rate bandwidth. Limiting the amount of traffic that receives strict-high priority treatment prevents other queues from being starved while also ensuring that the amount of traffic specified in the transmit rate receives strict-high priority treatment.



**NOTE:** You can use the `shaping-rate` statement to throttle the rate of packet transmission by setting a maximum bandwidth. On QFX10000 and NFX Series devices, you can use the `transmit-rate` statement to set a limit on the amount of bandwidth that receives strict-high priority treatment on a strict-high priority queue.

On QFX10000 and NFX Series devices, if you configure more than one strict-high priority queue on a port, you must configure a transmit rate on each of the strict-high priority queues. If you configure more than one strict-high priority queue on a port and you do not configure a transmit rate on the strict-high priority queues, the switch treats only the first queue you configure as a strict-high priority queue. The switch treats the other queues as low priority queues. If you configure a transmit rate on some strict-high priority queues but not on other strict-high priority queues on a port, the switch treats the queues that have a transmit rate as strict-high priority queues, and treats the queues that do not have a transmit rate as low priority queues.

## Scheduling Rules

When you configure a forwarding class and map traffic to it (that is, you are not using a default classifier and forwarding class), you must also define a scheduling policy for the forwarding class.

Defining a scheduling policy means:

- Mapping a scheduler to the forwarding class in a scheduler map
- Including the forwarding class in a forwarding class set

- Associating the scheduler map with a traffic control profile
- Attaching the traffic control profile to a forwarding class set and applying the traffic control profile to an interface

You can define a scheduling policy using port scheduling as follows:

- Mapping a scheduler to the forwarding class in a scheduler map
- Applying the scheduler map to one or more interfaces

## Rewrite Rules

On each physical interface, either all forwarding classes that are being used on the interface must have rewrite rules configured, or no forwarding classes that are being used on the interface can have rewrite rules configured. On any physical port, do not mix forwarding classes with rewrite rules and forwarding classes without rewrite rules.

## Lossless Transport Support

The switch supports up to six lossless forwarding classes. For lossless transport, you must enable PFC on the IEEE 802.1p code point of lossless forwarding classes. The following limitations apply to support lossless transport:

- The external cable length from the switch to other devices cannot exceed 300 meters.
- For FCoE traffic, the interface maximum transmission unit (MTU) must be at least 2180 bytes to accommodate the packet payload, headers, and checks.
- Changing any portion of a PFC configuration on a port blocks the entire port until the change is completed. After a PFC change is completed, the port is unblocked and traffic resumes. Changing the PFC configuration means any change to a congestion notification profile that is configured on a port (enabling or disabling PFC on a code point, changing the MRU or cable-length value, or specifying an output flow control queue). Blocking the port stops ingress and egress traffic, and causes packet loss on all queues on the port until the port is unblocked.



**NOTE:** If you explicitly configure the `fcoe` or the `no-loss` forwarding class, that forwarding class is no longer treated as a lossless forwarding class. Traffic mapped to these forwarding classes is treated as lossy (best-effort) traffic. This is true even if the explicit configuration is exactly the same as the default configuration.

You can configure up to six lossless forwarding classes. All explicitly configured lossless forwarding classes must include the `no-loss` packet drop attribute or the forwarding class is lossy.

## Platform-Specific Forwarding Class Behavior

Use the following table to review platform-specific behaviors for your platforms.

**Table 39: Platform-Specific Forwarding Class Behavior**

Platform	Difference
NFX Series	<ul style="list-style-type: none"> <li>• NFX Series devices support up to 8 forwarding classes.</li> <li>• NFX Series devices support the following four default forwarding classes: <ul style="list-style-type: none"> <li>• best-effort (be)</li> <li>• expedited-forwarding (ef)</li> <li>• assured-forwarding (af)</li> <li>• network-control (nc)</li> </ul> </li> <li>• On NFX Series devices, unicast and multdestination (multicast, broadcast, and destination lookup fail) traffic use the same forwarding classes and output queues.</li> </ul>
QFX5000 Series	<ul style="list-style-type: none"> <li>• QFX5200 switches support up to 10 forwarding classes, 8 unicast and 2 multicast.</li> </ul>
QFX10000 Series	<ul style="list-style-type: none"> <li>• QFX10000 switches support up to 8 forwarding classes.</li> <li>• On QFX10000 devices, unicast and multdestination (multicast, broadcast, and destination lookup fail) traffic use the same forwarding classes and output queues.</li> <li>• QFX10002-60C does not support PFC and lossless queues; that is, default lossless queues (fcoe and no-loss) will be lossy queues.</li> </ul>

## RELATED DOCUMENTATION

*Overview of CoS Changes Introduced in Junos OS Release 12.2*

*Understanding Junos CoS Components*

*Understanding CoS Packet Flow*

*Understanding CoS Flow Control (Ethernet PAUSE and PFC)*

*Example: Configuring Forwarding Classes*

*Defining CoS Forwarding Classes*

## Defining CoS Forwarding Classes

### IN THIS SECTION

- [Platform-Specific Forwarding Class Behavior | 106](#)

Forwarding classes allow you to group packets for transmission. To forward traffic, you map (assign) the forwarding classes to output queues.

Queues 0 through 7 are for unicast traffic and queues 8 through 11 are for multicast traffic. Forwarding classes mapped to unicast queues must carry unicast traffic, and forwarding classes mapped to multidestination queues must carry multidestination traffic. There are four default unicast forwarding classes and one default multidestination forwarding class.

The default forwarding classes, except on NFX Series devices, are:

- `best-effort`—Best-effort traffic
- `fcoe`—Guaranteed delivery for Fibre Channel over Ethernet traffic (do not use on OCX Series switches)
- `no-loss`—Guaranteed delivery for TCP no-loss traffic (do not use on OCX Series switches)
- `network-control`—Network control traffic

The default multidestination forwarding class, except on QFX10000 switches and NFX Series devices, is:

- `mcast`—Multidestination traffic

The NFX Series devices have the following default forwarding classes:

- best-effort (be)—Provides no service profile. Loss priority is typically not carried in a CoS value.
- expedited-forwarding (ef)—Provides a low loss, low latency, low jitter, assured bandwidth, end-to-end service.
- assured-forwarding (af)—Provides a group of values you can define and includes four subclasses: AF1, AF2, AF3, and AF4, each with two drop probabilities: low and high.
- network-control (nc)—Supports protocol control and thus is typically high priority.

You can map forwarding classes to queues using the `class` statement. You can map more than one forwarding class to a single queue. You cannot mix unicast and multicast forwarding classes on the same queue.

All of the forwarding classes mapped to the same queue must have the same packet drop attribute: either all of the forwarding classes must be lossy or all of the forwarding classes must be lossless. This is important because the default `fcoe` and `no-loss` forwarding classes have the `no-loss` drop attribute.

```
[edit class-of-service forwarding-classes]
user@switch# set class class-name queue-num queue-number <no-loss>
```

One example is to create a forwarding class named `be2` and map it to queue 1:

```
[edit class-of-service forwarding-classes]
user@switch# set class be2 queue-num 1
```

Another example is to create a lossless forwarding class named `fcoe2` and map it to queue 5:

```
[edit class-of-service forwarding-classes]
user@switch# set class fcoe2 queue-num 5 no-loss
```



**NOTE:** On switches that do not run ELS software, use the default forwarding-class-to-queue mapping for the lossless `fcoe` and `no-loss` forwarding classes. If you explicitly configure the lossless forwarding classes, the traffic mapped to those forwarding classes is treated as lossy (best-effort) traffic and does *not* receive lossless treatment unless you include the optional `no-loss` packet drop attribute.

## Platform-Specific Forwarding Class Behavior

Use the following table to review platform-specific behaviors for your platforms.



**Table 40: Platform-Specific Forwarding Class Behavior**

Platform	Difference
NFX Series	<ul style="list-style-type: none"> <li>NFX devices support up to 8 forwarding classes.</li> <li>NFX device queues 0 through 7 support both unicast and multicast traffic.</li> </ul>
QFX5000 Series	<ul style="list-style-type: none"> <li>QFX5200 switches support up to 10 (8 unicast and 2 multicast) forwarding classes. All other QFX5000 switches support up to 12 (8 unicast and 4 multicast) forwarding classes.</li> <li>QFX5000 switch queues 0 through 7 support unicast traffic. Queues 8 through 11 support multicast traffic.</li> </ul>
QFX10000 Series	<ul style="list-style-type: none"> <li>Prior to Junos OS Release 22.1R1, QFX10000 switches support up to 8 forwarding classes.</li> <li>Starting with Junos OS Release 22.1R1, QFX10000 switches support up to 16 forwarding classes.</li> <li>QFX10000 switch queues 0 through 7 support both unicast and multicast traffic.</li> <li>QFX10002-60C does not support PFC and lossless queues; that is, default lossless queues (fcoe and no-loss) will be lossy queues.</li> </ul>

**Change History Table**

Feature support is determined by the platform and release you are using. Use [Feature Explorer](#) to determine if a feature is supported on your platform.

Release	Description
22.1R1	Starting in Junos OS Release 22.1R1, QFX10000 Series devices support 16 forwarding classes.

## RELATED DOCUMENTATION

*Example: Configuring CoS Hierarchical Port Scheduling (ETS)*

*Example: Configuring Forwarding Classes*

*Monitoring CoS Forwarding Classes*

*Understanding CoS Forwarding Classes*

*Understanding CoS Port Schedulers on QFX Switches*

## Example: Configuring Forwarding Classes

### IN THIS SECTION

- [Requirements | 108](#)
- [Overview | 109](#)
- [Example 1: Configuring Forwarding Classes for Switches Except QFX10000 | 110](#)
- [Example 2: Configuring Forwarding Classes for QFX10000 Switches | 112](#)

Forwarding classes group packets for transmission. Forwarding classes map to output queues, so the packets assigned to a forwarding class use the output queue mapped to that forwarding class.

### Requirements

This example uses the following hardware and software components for two configuration examples:

Configuring forwarding classes for switches except QFX10000

- One switch except QFX10000
- Any supported Junos release

Configuring forwarding classes for QFX10000 switches

- One QFX10000 switch
- Any supported Junos release

## Overview

The QFX10000 switch supports eight forwarding classes. Other switches support up to 12 forwarding classes. To forward traffic, you must map (assign) the forwarding classes to output queues. On the QFX10000 switch, queues 0 through 7 are for both unicast and multdestination traffic. On other switches, queues 0 through 7 are for unicast traffic, and queues 8 through 9 (QFX5200 switch) or 8 through 11 (other switches) are for multdestination traffic.

The switch provides four default forwarding classes, and except on QFX10000 switches, these four forwarding classes are unicast, plus one default multdestination forwarding class. You can define the remaining forwarding classes and configure them as unicast or multdestination forwarding classes by mapping them to unicast or multdestination queues. The type of queue, unicast or multdestination, determines the type of forwarding class.

The four default forwarding classes (unicast except on QFX10000) are:

- `be`—Best-effort traffic
- `fcoe`—Guaranteed delivery for Fibre Channel over Ethernet traffic (do not use on OCX Series switches)
- `no-loss`—Guaranteed delivery for TCP no-loss traffic (do not use on OCX Series switches)
- `nc`—Network control traffic

Except on QFX10000 switches, the default multdestination forwarding class is:

- `mcast`—Multdestination traffic

Map forwarding classes to queues using the `class` statement. You can map more than one forwarding class to a single queue, but all forwarding classes mapped to a particular queue must be of the same type:

- Except on QFX10000 switches, all forwarding classes mapped to a particular queue must be either unicast or multicast. You cannot mix unicast and multicast forwarding classes on the same queue.
- On QFX10000 switches, all forwarding classes mapped to a particular queue must have the same packet drop attribute: all of the forwarding classes must be lossy, or all of the forwarding classes mapped to a queue must be lossless.

```
[edit class-of-service forwarding-classes]
user@switch# set class class-name queue-num queue-number;
```



**NOTE:** You can include the *no-loss* packet drop attribute in explicit forwarding class configurations to configure a lossless forwarding class.



**NOTE:** Hierarchical scheduling controls output queue forwarding. When you define a forwarding class and classify traffic into it, you must also define a scheduling policy for the forwarding class. Defining a scheduling policy means:

- Mapping a scheduler to the forwarding class in a scheduler map
- Including the forwarding class in a forwarding class set
- Associating the scheduler map with a traffic control profile
- Attaching the traffic control profile to a forwarding class set and applying the traffic control profile to an interface

On QFX10000 switches, you can define a scheduling policy using port scheduling:

- Mapping a scheduler to the forwarding class in a scheduler map.
- Applying the scheduler map to one or more interfaces.

## Example 1: Configuring Forwarding Classes for Switches Except QFX10000

### IN THIS SECTION

- [Verification](#) | 111

### Configuration

### Step-by-Step Procedure

[Table 41 on page 111](#) shows the configuration forwarding-class-to-queue mapping for this example:

**Table 41: Forwarding-Class-to-Queue Example Configuration Except on QFX10000**

Forwarding Class	Queue
best-effort	0
nc	7
mcast	8

To configure CoS forwarding classes for switches except QFX10000:

1. Map the best-effort forwarding class to queue 0:

```
[edit class-of-service forwarding-classes]
user@switch# set class best-effort queue-num 0
```

2. Map the nc forwarding class to queue 7:

```
[edit class-of-service forwarding-classes]
user@switch# set class nc queue-num 7
```

3. Map the mcast-be forwarding class to queue 8:

```
[edit class-of-service forwarding-classes]
user@switch# set class mcast-be queue-num 8
```

## Verification

### IN THIS SECTION

- [Verifying the Forwarding-Class-to-Queue Mapping | 112](#)

*Verifying the Forwarding-Class-to-Queue Mapping*

**Purpose**

Verify the forwarding-class-to-queue mapping. (The system shows only the explicitly configured forwarding classes; it does not show default forwarding classes such as fcoe and no-loss.)

**Action**

Verify the results of the forwarding class configuration using the operational mode command `show configuration class-of-service forwarding-classes`:

```
user@switch> show configuration class-of-service forwarding-classes
class best-effort queue-num 0;
class network-control queue-num 7;
class mcast queue-num 8;
```

**Example 2: Configuring Forwarding Classes for QFX10000 Switches**

IN THIS SECTION

Verification | 113

**Configuration**

**Step-by-Step Procedure**

[Table 42 on page 112](#) shows the configuration forwarding-class-to-queue mapping for this example:

**Table 42: Forwarding-Class-to-Queue Example Configuration on QFX10000**

Forwarding Class	Queue
best-effort	0
be1	1

**Table 42: Forwarding-Class-to-Queue Example Configuration on QFX10000 (Continued)**

Forwarding Class	Queue
nc	7

To configure CoS forwarding classes for QFX10000 switches:

1. Map the best-effort forwarding class to queue 0:

```
[edit class-of-service forwarding-classes]
user@switch# set class best-effort queue-num 0
```

2. Map the be1 forwarding class to queue 1:

```
[edit class-of-service forwarding-classes]
user@switch# set class be1 queue-num 1
```

3. Map the nc forwarding class to queue 7:

```
[edit class-of-service forwarding-classes]
user@switch# set class nc queue-num 7
```

## Verification

### IN THIS SECTION

- [Verifying the Forwarding-Class-to-Queue Mapping | 114](#)
- [Platform-Specific Behavior | 114](#)

*Verifying the Forwarding-Class-to-Queue Mapping*

**Purpose**

Verify the forwarding-class-to-queue mapping. (The system shows only the explicitly configured forwarding classes; it does not show default forwarding classes such as fcoe and no-loss.)

**Action**

Verify the results of the forwarding class configuration using the operational mode command `show configuration class-of-service forwarding-classes`:

```
user@switch> show configuration class-of-service forwarding-classes
class best-effort queue-num 0;
class be1 queue-num 1;
class network-control queue-num 7;
```

*Platform-Specific Behavior*

Use the following table to review platform-specific behaviors for your platforms.

**Table 43: Platform-Specific Behavior**

Platform	Difference
QFX10000 Series	<ul style="list-style-type: none"><li>Unlike other QFX switches, QFX10000 Series switches apply a forwarding class to both unicast traffic and multideestination (multicast, broadcast, and destination lookup fail) traffic.</li></ul>

**RELATED DOCUMENTATION**

<i>Example: Configuring CoS Hierarchical Port Scheduling (ETS)</i>
<i>Defining CoS Forwarding Classes</i>
<i>Monitoring CoS Forwarding Classes</i>
<i>Overview of CoS Changes Introduced in Junos OS Release 11.3</i>
<i>Overview of CoS Changes Introduced in Junos OS Release 12.2</i>



## Monitoring CoS Forwarding Classes

### IN THIS SECTION

- Purpose | 115
- Action | 115
- Meaning | 115

### Purpose

Use the monitoring functionality to view the current assignment of CoS forwarding classes to queue numbers on the system.

### Action

To monitor CoS forwarding classes in the CLI, enter the following CLI command:

```
user@switch> show class-of-service forwarding-class
```

### Meaning

Some switches use different forwarding classes, output queues, and classifiers for unicast and multdestination (multicast, broadcast, destination lookup fail) traffic. These switches support 12 forwarding classes and output queues, eight for unicast traffic and four for multdestination traffic.

Some switches use the same forwarding classes, output queues, and classifiers for unicast and multdestination traffic. These switches support eight forwarding classes and eight output queues.

[Table 44 on page 116](#) summarizes key output fields on switches that use different forwarding classes and output queues for unicast and multdestination traffic.

**Table 44: Summary of Key CoS Forwarding Class Output Fields on Switches that Separate Unicast and Multidestination Traffic**

Field	Values
Forwarding Class	<p>Names of forwarding classes assigned to queue numbers. By default, the following unicast forwarding classes are assigned to queues 0, 3, 4, and 7, respectively:</p> <ul style="list-style-type: none"> <li>• best-effort—Provides no special CoS handling of packets. Loss priority is typically not carried in a CoS value.</li> <li>• fcoe—Provides guaranteed delivery for Fibre Channel over Ethernet (FCoE) traffic.</li> <li>• no-loss—Provides guaranteed delivery for TCP lossless traffic</li> <li>• network-control—Packets can be delayed but not dropped.</li> </ul> <p>By default, the following multidestination forwarding class is assigned to queue 8:</p> <ul style="list-style-type: none"> <li>• mcast—Provides no special CoS handling of packets.</li> </ul>
Queue	<p>Queue number corresponding to (mapped to) the forwarding class name.</p> <p>By default, four queues (0, 3, 4, and 7) are assigned to unicast forwarding classes and one queue (8) is assigned to a multidestination forwarding class:</p> <ul style="list-style-type: none"> <li>• Queue 0—best-effort</li> <li>• Queue 3—fcoe</li> <li>• Queue 4—no-loss</li> <li>• Queue 7—network-control</li> <li>• Queue 8—mcast</li> </ul>

**Table 44: Summary of Key CoS Forwarding Class Output Fields on Switches that Separate Unicast and Multidestination Traffic (*Continued*)**

Field	Values
No-Loss	<p>Packet drop attribute associated with each forwarding class:</p> <ul style="list-style-type: none"> <li>• Disabled—The forwarding class is configured for lossy transport (packets might drop during periods of congestion)</li> <li>• Enabled—The forwarding class is configured for lossless transport</li> </ul> <p><b>NOTE:</b> To achieve lossless transport, you must ensure that priority-based flow control (PFC) and DCBX are properly configured on the lossless priority (IEEE 802.1p code point), and that sufficient port bandwidth is reserved for the lossless traffic flows.</p>

[Table 45 on page 117](#) summarizes key output fields on switches that use the same forwarding classes and output queues for unicast and multidestination traffic.

**Table 45: Summary of Key CoS Forwarding Class Output Fields on Switches That Do Not Separate Unicast and Multidestination Traffic**

Field	Values
Forwarding Class	<p>Names of forwarding classes assigned to queue numbers. By default, the following forwarding classes are assigned to queues 0, 3, 4, and 7, respectively:</p> <ul style="list-style-type: none"> <li>• best-effort—Provides no special CoS handling of packets. Loss priority is typically not carried in a CoS value.</li> <li>• fcoe—Provides guaranteed delivery for Fibre Channel over Ethernet (FCoE) traffic.</li> <li>• no-loss—Provides guaranteed delivery for TCP lossless traffic</li> <li>• network-control—Packets can be delayed but not dropped.</li> </ul>

**Table 45: Summary of Key CoS Forwarding Class Output Fields on Switches That Do Not Separate Unicast and Multidestination Traffic (*Continued*)**

Field	Values
Queue	<p>Queue number corresponding to (mapped to) the forwarding class name.</p> <p>By default, four queues (0, 3, 4, and 7) are assigned to forwarding classes:</p> <ul style="list-style-type: none"> <li>• Queue 0—best-effort</li> <li>• Queue 3—fcoe</li> <li>• Queue 4—no-loss</li> <li>• Queue 7—network-control</li> </ul>
No-Loss	<p>Packet drop attribute associated with each forwarding class:</p> <ul style="list-style-type: none"> <li>• Disabled—The forwarding class is configured for lossy transport (packets might drop during periods of congestion).</li> <li>• Enabled—The forwarding class is configured for lossless transport.</li> </ul> <p><b>NOTE:</b> To achieve lossless transport, you must ensure that priority-based flow control (PFC) and DCBX are properly configured on the lossless priority (IEEE 802.1p code point), and that sufficient port bandwidth is reserved for the lossless traffic flows.</p>

## Understanding CoS Rewrite Rules

As packets enter or exit a network, edge switches might be required to alter the class-of-service (CoS) settings of the packets. *Rewrite rules* set the value of the code point bits (Layer 3 DSCP bits, Layer 2 CoS bits, or MPLS EXP bits) within the header of the outgoing packet. Each rewrite rule:

1. Reads the current forwarding class and loss priority associated with the packet.
2. Locates the new (rewrite) code point value from a table.

3. Writes that code point value into the packet header, replacing the old code point value.

Rewrite rules must be assigned to an interface for rewrites to take effect.

You can apply (bind) one DSCP or DSCP IPv6 rewrite rule and one IEEE 802.1p rewrite rule to each interface. You can also bind EXP rewrite rules to `family mpls` logical interfaces to rewrite the CoS bits of MPLS traffic.

You cannot apply both a DSCP and a DSCP IPv6 rewrite rule to the same physical interface. Each physical interface supports only one DSCP rewrite rule. Both IP and IPv6 packets use the same DSCP rewrite rule, regardless if the configured rewrite rule is DSCP or DSCP IPv6. You can apply an EXP rewrite rule on an interface that has DSCP or IEEE rewrite rules. Only MPLS traffic on `family mpls` interfaces uses the EXP rewrite rule.

You *can* apply both a DSCP rewrite rule and a DSCP IPv6 rewrite rule to a logical interface. IPv6 packets are rewritten with DSCP-IPv6 rewrite-rules and IPv4 packets are remarked with DSCP rewrite-rules.



**NOTE:** There are no default rewrite rules. If you want to apply a rewrite rule to outgoing packets, you must explicitly configure the rewrite rule.

You can look at behavior aggregate (BA) classifiers and rewrite rules as two sides of the same coin. A BA classifier reads the code point bits of incoming packets and classifies the packets into forwarding classes, then the system applies the CoS configured for the forwarding class to those packets. Rewrite rules change (rewrite) the code point bits just before the packets leave the system so that the next switch or router can apply the appropriate level of CoS to the packets. When you apply a rewrite rule to an interface, the rewrite rule is the last CoS action performed on the packet before it is forwarded.

Rewrite rules alter CoS values in outgoing packets on the outbound interfaces of an edge switch to accommodate the policies of a targeted peer. This allows the downstream switch in a neighboring network to classify each packet into the appropriate service group.



**NOTE:** On each physical interface, either all forwarding classes that are being used on the interface must have rewrite rules configured or no forwarding classes that are being used on the interface can have rewrite rules configured. On any physical port, do not mix forwarding classes with rewrite rules and forwarding classes without rewrite rules.



**NOTE:** Rewrite rules are applied *before* the egress filter is matched to traffic. Because the code point rewrite occurs before the egress filter is matched to traffic, the egress filter match is based on the rewrite value, not on the original code point value in the packet.

For packets that carry both an inner VLAN tag and an outer VLAN tag, the rewrite rule rewrites only the outer VLAN tag.

MPLS EXP rewrite rules apply only to family mpls logical interfaces. You cannot apply to an EXP rewrite rule to a physical interface. You can configure up to 64 EXP rewrite rules, but you can only use 16 EXP rewrite rules at any time on the switch. On a given logical interface, all pushed MPLS labels have the same EXP rewrite rule applied to them. You can apply different EXP rewrite rules to different logical interfaces on the same physical interface.



**NOTE:** If the switch is performing penultimate hop popping (PHP), EXP rewrite rules do not take effect. If both an EXP classifier and an EXP rewrite rule are configured on the switch, then the EXP value from the last popped label is copied into the inner label. If either an EXP classifier or an EXP rewrite rule (but not both) is configured on the switch, then the inner label EXP value is sent unchanged.

You can configure enough rewrite rules to handle most, if not all, network scenarios. [Table 46 on page 120](#) shows how many of each type of rewrite rules you can configure, and how many entries you can configure per rewrite rule.

**Table 46: Configuring Rewrite Rules**

Rewrite Rule Type	Maximum Number of Rewrite Rules	Maximum Number of Entries per Rewrite Rule
IEEE 802.1p	64	128
DSCP	32	128
DSCP IPv6	32	128
MPLS EXP	64	128

You cannot apply rewrite rules directly to integrated routing and bridging (IRB), also known as routed VLAN interfaces (RVIs), because the members of IRBs/RVIs are VLANs, not ports. However, you can apply rewrite rules to the VLAN port members of an IRB/RVI.

## RELATED DOCUMENTATION

*Understanding Junos CoS Components*

*Defining CoS Rewrite Rules*

## Defining CoS Rewrite Rules

### IN THIS SECTION

- [Platform-Specific Rewrite Rule Behavior | 123](#)

### Overview

Edge switches might need to change the class-of-service (CoS) settings of the packets. You can configure rewrite rules to alter code point bit values in outgoing packets on the outbound interfaces of a switch so that the CoS treatment matches the policies of a targeted peer. Policy matching allows the downstream routing platform or switch in a neighboring network to classify each packet into the appropriate service group.

To configure a CoS rewrite rule, create the rule by giving it a name and associating it with a forwarding class, loss priority, and code point. This creates a rewrite table. After the rewrite rule is created, enable it on an interface (EXP rewrite rules can only be enabled on `family mpls` logical interfaces, not on physical interfaces). You can also apply an existing rewrite rule on an interface.



**NOTE:** On each physical interface, either all forwarding classes that are being used on the interface must have rewrite rules configured, or no forwarding classes that are being used on the interface can have rewrite rules configured. On any physical port, do not mix forwarding classes with rewrite rules and forwarding classes without rewrite rules.



**NOTE:** To replace an existing rewrite rule on the interface with a new rewrite rule of the same type, first explicitly remove the existing rewrite rule and then apply the new rule.



**NOTE:** For packets that carry both an inner VLAN tag and an outer VLAN tag, the rewrite rule rewrites only the outer VLAN tag.

### Configuring Rewrite Rules

To create rewrite rules and enable them on interfaces:

- To create an 802.1p rewrite rule named `customup-rw` in the rewrite table for all Layer 2 interfaces:

```
[edit class-of-service rewrite-rules]
user@switch# set ieee-802.1 customup-rw forwarding-class be loss-priority low code-point 000
user@switch# set ieee-802.1 customup-rw forwarding-class be loss-priority high code-point 001
user@switch# set ieee-802.1 customup-rw forwarding-class be loss-priority low code-point 010
user@switch# set ieee-802.1 customup-rw forwarding-class fcoe loss-priority low code-point 011
user@switch# set ieee-802.1 customup-rw forwarding-class ef-no-loss loss-priority low code-point 100
user@switch# set ieee-802.1 customup-rw forwarding-class ef-no-loss loss-priority high code-point 101
user@switch# set ieee-802.1 customup-rw forwarding-class nc loss-priority low code-point 110
user@switch# set ieee-802.1 customup-rw forwarding-class nc loss-priority high code-point 111
```

- To enable an 802.1p rewrite rule named `customup-rw` on a Layer 2 interface:

```
[edit]
user@switch# set class-of-service interfaces xe-0/0/7 unit 0 rewrite-rules ieee-802.1
customup-rw
```



**NOTE:** All forwarding classes assigned to port `xe-0/0/7` must have rewrite rules. Do not mix forwarding classes that have rewrite rules with forwarding classes that do not have rewrite rules on the same physical interface.

- To enable an 802.1p rewrite rule named `customup-rw` on all 10-Gigabit Ethernet interfaces on the switch, use wildcards for the interface name and logical interface (unit) number:

```
[edit]
user@switch# set class-of-service interfaces xe-* unit * rewrite-rules customup-rw
```



**NOTE:** In this case, *all* forwarding classes assigned to *all* 10-Gigabit Ethernet ports must have rewrite rules. Do not mix forwarding classes that have rewrite rules with forwarding classes that do not have rewrite rules on the same physical interface.



## Platform-Specific Rewrite Rule Behavior

Use the following table to review platform-specific behaviors for your platforms.

Table 47: Platform-Specific Rewrite Rule Behavior

Platform	Difference
QFX5000 Series	<ul style="list-style-type: none"><li>QFX5130, QFX5220, and QFX5700 switches do not support DSCP IPv6 classifiers and rewrite rules. However, you can apply DSCP classifiers and rewrite rules for IPV6 traffic as well.</li></ul>

### RELATED DOCUMENTATION

<i>Monitoring CoS Rewrite Rules</i>
<i>Configuring Rewrite Rules for MPLS EXP Classifiers</i>
<i>Understanding CoS Rewrite Rules</i>
<i>Understanding CoS MPLS EXP Classifiers and Rewrite Rules</i>

## Troubleshooting an Unexpected Rewrite Value

### IN THIS SECTION

- Problem | 123
- Cause | 124
- Solution | 124

### Problem

### Description

Traffic from one or more forwarding classes on an egress port is assigned an unexpected rewrite value.



**NOTE:** For packets that carry both an inner VLAN tag and an outer VLAN tag, the rewrite rules rewrite only the outer VLAN tag.

## Cause

If you configure a rewrite rule for a forwarding class on an egress port, but you do not configure a rewrite rule for every forwarding class on that egress port, then the forwarding classes that do not have a configured rewrite rule are assigned random rewrite values.

For example:

1. Configure forwarding classes fc1, fc2, and fc3.
2. Configure rewrite rules for forwarding classes fc1 and fc2, but not for forwarding class fc3.
3. Assign forwarding classes fc1, fc2, and fc3 to a port.

When traffic for these forwarding classes flows through the port, traffic for forwarding classes fc1 and fc2 is rewritten correctly. However, traffic for forwarding class fc3 is assigned a random rewrite value.

## Solution

If any forwarding class on an egress port has a configured rewrite rule, then all forwarding classes on that egress port must have a configured rewrite rule. Configuring a rewrite rule for any forwarding class that is assigned a random rewrite value solves the problem.



**TIP:** If you want the forwarding class to use the same code point value assigned to it by the ingress classifier, specify that value as the rewrite rule value. For example, if a forwarding class has the IEEE 802.1 ingress classifier code point value 011, configure a rewrite rule for that forwarding class that uses the IEEE 802.1p code point value 011.



**NOTE:** There are no default rewrite rules. You can bind one rewrite rule for DSCP traffic and one rewrite rule for IEEE 802.1p traffic to an interface. A rewrite rule can contain multiple forwarding-class-to-rewrite-value mappings.

1. To assign a rewrite value to a forwarding class, add the new rewrite value to the same rewrite rule as the other forwarding classes on the port:

```
[edit class-of-service rewrite-rules]
user@switch# set (dscp | ieee-802.1) rewrite-name forwarding-class class-name loss-priority
priority code-point (alias | bits)
```

For example, if the other forwarding classes on the port use rewrite values defined in the rewrite rule `custom-rw`, the forwarding class `be2` is being randomly rewritten, and you want to use IEEE 802.1 code point `002` for the `be2` forwarding class:

```
[edit class-of-service rewrite-rules]
user@switch# set ieee-802.1 custom-rw forwarding-class be2 loss-priority low code-point 002
```

2. Enable the rewrite rule on an interface if it is not already enabled on the desired interface:

```
[edit]
user@switch# set class-of-service interfaces interface-name unit unit rewrite-rules (dscp |
ieee-802.1) rewrite-rule-name
```

For example, to enable the rewrite rule `custom-rw` on interface `xe-0/0/24.0`:

```
[edit]
user@switch# set class-of-service interfaces xe-0/0/24 unit 0 rewrite-rules ieee-802.1 custom-
rw
```

## RELATED DOCUMENTATION

[interfaces](#)

[rewrite-rules](#)

[Defining CoS Rewrite Rules](#)

[Monitoring CoS Rewrite Rules](#)

# Monitoring CoS Rewrite Rules

IN THIS SECTION

- Purpose | 126
- Action | 126
- Meaning | 126

## Purpose

Use the monitoring functionality to display information about CoS value rewrite rules, which are based on the forwarding class and loss priority.

## Action

To monitor CoS rewrite rules in the CLI, enter the CLI command:

```
user@switch> show class-of-service rewrite-rule
```

To monitor a particular rewrite rule in the CLI, enter the CLI command:

```
user@switch> show class-of-service rewrite-rule name rewrite-rule-name
```

To monitor a particular type of rewrite rule (for example, DSCP, DSCP IPv6, IEEE-802.1, or MPLS EXP) in the CLI, enter the CLI command:

```
user@switch> show class-of-service rewrite-rule type rewrite-rule-type
```

## Meaning

[Table 48 on page 126](#) summarizes key output fields for CoS rewrite rules.

**Table 48: Summary of Key CoS Rewrite Rule Output Fields**

Field	Values
Rewrite rule	Name of the rewrite rule.

**Table 48: Summary of Key CoS Rewrite Rule Output Fields** *(Continued)*

Field	Values
Code point type	Rewrite rule type: <ul style="list-style-type: none"> <li>• dscp—For IPv4 DiffServ traffic.</li> <li>• dscp-ipv6—For IPv6 Diffserv traffic.</li> <li>• ieee-802.1—For Layer 2 traffic.</li> <li>• exp—For MPLS traffic.</li> </ul>
Index	Internal index for the rewrite rule.
Forwarding class	Name of the forwarding class that is used to determine CoS values for rewriting in combination with loss priority.  Rewrite rules are applied to CoS values in outgoing packets based on forwarding class and loss priority setting.
Loss priority	Level of loss priority that is used to determine CoS values for rewriting in combination with forwarding class.
Code point	Rewrite code point value.

**RELATED DOCUMENTATION**

| *Defining CoS Rewrite Rules*

# 3

PART

## Scheduling Traffic

---

- [Using Schedulers | 129](#)
-

## CHAPTER 3

# Using Schedulers

**IN THIS CHAPTER**

- Understanding CoS Scheduling Behavior and Configuration Considerations | 129
- Defining CoS Queue Schedulers for Port Scheduling | 135
- Defining CoS Queue Scheduling Priority | 139
- Example: Configuring Queue Scheduling Priority | 141
- Monitoring CoS Scheduler Maps | 147
- Understanding CoS Traffic Control Profiles | 149
- Understanding CoS Priority Group Scheduling | 151
- Defining CoS Traffic Control Profiles (Priority Group Scheduling) | 154
- Example: Configuring Traffic Control Profiles (Priority Group Scheduling) | 156
- Understanding CoS Priority Group and Queue Guaranteed Minimum Bandwidth | 160
- Example: Configuring Minimum Guaranteed Output Bandwidth | 163
- Understanding CoS Priority Group Shaping and Queue Shaping (Maximum Bandwidth) | 170
- Example: Configuring Maximum Output Bandwidth | 173
- Understanding CoS Explicit Congestion Notification | 179

## Understanding CoS Scheduling Behavior and Configuration Considerations

Many factors affect scheduling configuration and bandwidth requirements, including:

- When you configure bandwidth for a forwarding class (each forwarding class is mapped to a queue) or a forwarding class set (priority group), the switch considers only the data as the configured bandwidth. The switch does not account for the bandwidth consumed by the preamble and the interframe gap (IFG). Therefore, when you calculate and configure the bandwidth requirements for a forwarding class or for a forwarding class set, consider the preamble and the IFG as well as the data in the calculations.

- When you configure a forwarding class to carry traffic on the switch (instead of using only default forwarding classes), you must also define a scheduling policy for the user-configured forwarding class. Some switches support enhanced transmission selection (ETS) hierarchical port scheduling, some switches support port scheduling, and some switches support both methods of scheduling.



**NOTE:** Use [Feature Explorer](#) to confirm platform and release support for [ETS](#) and [port scheduling](#).

For ETS hierarchical port scheduling, defining a hierarchical scheduling policy using ETS means:

- Mapping a scheduler to the forwarding class in a scheduler map
- Including the forwarding class in a forwarding class set
- Associating the scheduler map with a traffic control profile
- Attaching the traffic control profile to a forwarding class set and an interface

On switches that support port scheduling, defining a scheduling policy means:

- Mapping a scheduler to the forwarding class in a scheduler map.
- Applying the scheduler map to one or more interfaces.
- On each physical interface, either all forwarding classes that are being used on the interface must have rewrite rules configured, or no forwarding classes that are being used on the interface can have rewrite rules configured. On any physical port, do not mix forwarding classes with rewrite rules and forwarding classes without rewrite rules.
- For packets that carry both an inner VLAN tag and an outer VLAN tag, rewrite rules rewrite only the outer VLAN tag.
- For ETS hierarchical port scheduling, configuring the minimum guaranteed bandwidth (`transmit-rate`) for a forwarding class does not work unless you also configure the minimum guaranteed bandwidth (`guaranteed-rate`) for the forwarding class set in the traffic control profile.

Additionally, the sum of the transmit rates of the forwarding classes in a forwarding class set should not exceed the guaranteed rate for the forwarding class set. (You cannot guarantee a minimum bandwidth for the queues that is greater than the minimum bandwidth guaranteed for the entire set of queues.) If you configure transmit rates whose sum exceeds the guaranteed rate of the forwarding class set, the commit check fails and the system rejects the configuration.

- For ETS hierarchical port scheduling, the sum of the forwarding class set guaranteed rates cannot exceed the total port bandwidth. If you configure guaranteed rates whose sum exceeds the port bandwidth, the system sends a syslog message to notify you that the configuration is not valid. However, the system does not perform a commit check. If you commit a configuration in which the



sum of the guaranteed rates exceeds the port bandwidth, the hierarchical scheduler behaves unpredictably.

- For ETS hierarchical port scheduling, if you configure the `guaranteed-rate` of a forwarding class set as a percentage, configure all of the transmit rates associated with that forwarding class set as percentages. In this case, if any of the transmit rates are configured as absolute values instead of percentages, the configuration is not valid and the system sends a syslog message.
- There are several factors to consider if you want to configure a strict-high priority queue (forwarding class):

- On QFX5200 switches you can configure only one strict-high priority queue (forwarding class).

On QFX5100 and EX4600 switches, you can configure only one forwarding-class-set (priority group) as strict-high priority. All queues which are part of that strict-high forwarding class set then act as strict-high queues.

On QFX10000 switches, there is no limit to the number of strict-high priority queues you can configure.

- You cannot configure a minimum guaranteed bandwidth (`transmit-rate`) for a strict-high priority queue on QFX5200, QFX5100, EX4600 switches.

On QFX5200 and QFX10000 switches, you can set the `transmit-rate` on strict-high priority queues to set a limit on the amount of traffic that the queue treats as strict-high priority traffic. Traffic in excess of the `transmit-rate` is treated as best-effort traffic, and receives an excess bandwidth sharing weight of “1”, which is the proportion of extra bandwidth the strict-high priority queue can share on the port. Queues that are not strict-high priority queues use the transmit rate (default) or the configured excess rate to determine the proportion (weight) of extra port bandwidth the queue can share. However, you cannot configure an excess rate on a strict-high priority queue, and you cannot change the excess bandwidth sharing weight of “1” on a strict-high priority queue.

For ETS hierarchical port scheduling, you cannot configure a minimum guaranteed bandwidth (`guaranteed-rate`) for a forwarding class set that includes a strict-high priority queue.

- Except on QFX10000 switches, for ETS hierarchical port scheduling only, you must create a separate forwarding class set for a strict-high priority queue. On QFX10000 switches, you can mix strict-high priority and low priority queues in the same forwarding class set.
- Except on QFX10000 switches, for ETS hierarchical port scheduling, only one forwarding class set can contain a strict-high priority queue. On QFX10000 switches, this restriction does not apply.
- Except on QFX10000 switches, for ETS hierarchical port scheduling, a strict-high priority queue cannot belong to the same forwarding class set as queues that are not strict-high priority. (You cannot mix a strict-high priority forwarding class with forwarding classes that are not strict-high

priority in one forwarding class set.) On QFX10000 switches, you can mix strict-high priority and low priority queues in the same forwarding class set.

- For ETS hierarchical port scheduling on switches that use different forwarding class sets for unicast and multdestination (multicast, broadcast, and destination lookup fail) traffic, a strict-high priority queue cannot belong to a multdestination forwarding class set.
- On QFX10000 systems, we recommend that you always configure a transmit rate on strict-high priority queues to prevent them from starving other queues. If you do not apply a transmit rate to limit the amount of bandwidth strict-high priority queues can use, then strict-high priority queues can use all of the available port bandwidth and starve other queues on the port.

On QFX5200, QFX5100, EX4600 switches, we recommend that you always apply a shaping rate to the strict-high priority queue to prevent it from starving other queues. If you do not apply a shaping rate to limit the amount of bandwidth a strict-high priority queue can use, then the strict-high priority queue can use all of the available port bandwidth and starve other queues on the port.

- For transmit rates below 1 Gbps, we recommend that you configure the transmit rate as a percentage instead of as a fixed rate. This is because the system converts fixed rates into percentages and might round small fixed rates to a lower percentage. For example, a fixed rate of 350 Mbps is rounded down to 3 percent instead of 3.5 percent.
- When you set the maximum bandwidth for a queue or for a priority group (shaping-rate) at 100 Kbps or lower, the traffic shaping behavior is accurate only within +/- 20 percent of the configured shaping-rate.
- On QFX10000 switches, configuring rate shaping ([set class-of-service schedulers *scheduler-name* transmit-rate (*rate / percentage*) exact) on a LAG interface using the [edit class-of-service interfaces *lag-interface-name* scheduler-map *scheduler-map-name*] statement can result in scheduled traffic streams receiving more LAG link bandwidth than expected.

You configure rate shaping in a scheduler to set the maximum bandwidth for traffic assigned to a forwarding class on a particular output queue on a port. For example, you can use a scheduler to configure rate shaping on traffic assigned to the best-effort forwarding class mapped to queue 0, and then apply the scheduler to an interface using a scheduler map, to set the maximum bandwidth for best-effort traffic mapped to queue 0 on that port. Traffic in the best-effort forwarding can use no more than the amount of port bandwidth specified by the transmit rate when you use the exact option.

LAG interfaces are composed of two or more Ethernet links bundled together to function as a single interface. The switch can hash traffic entering a LAG interface onto any member link in the LAG interface. When you configure rate shaping and apply it to a LAG interface, the way that the switch applies the rate shaping to traffic depends on how the switch hashes the traffic onto the LAG links.

To illustrate how link hashing affects the way the switch applies a shaping rate to LAG traffic, let's look at a LAG interface (ae0) that has two member links (xe-0/0/20 and xe-0/0/21). On LAG ae0, we configure rate shaping of 2g for traffic assigned to the best-effort forwarding class, which is mapped to output queue 0. When traffic in the best-effort forwarding class reaches the LAG interface, the switch hashes the traffic onto one of the two member links.

If the switch hashes all of the best-effort traffic onto the same LAG link, the traffic receives a maximum of 2g bandwidth on that link. In this case, the intended cumulative limit of 2g for best-effort traffic on the LAG is enforced.

However, if the switch hashes the best-effort traffic onto both of the LAG links, the traffic receives a maximum of 2g bandwidth on *each* LAG link, not 2g as a cumulative total for the entire LAG, so the best-effort traffic receives a maximum of 4g on the LAG, not the 2g set by the rate shaping configuration. When hashing spreads the traffic assigned to an output queue (which is mapped to a forwarding class) across multiple LAG links, the effective rate shaping (cumulative maximum bandwidth) on the LAG is:

(number of LAG member interfaces) x (rate shaping for the output queue) = cumulative LAG rate shaping

- On switches that do not use virtual output queues (VOQs), ingress port congestion can occur during periods of egress port congestion if an ingress port forwards traffic to more than one egress port, and at least one of those egress ports experiences congestion. If this occurs, the congested egress port can cause the ingress port to exceed its fair allocation of ingress buffer resources. When the ingress port exceeds its buffer resource allocation, frames are dropped at the ingress. Ingress port frame drop affects not only the congested egress ports, but also all of the egress ports to which the congested ingress port forwards traffic.

If a congested ingress port drops traffic that is destined for one or more uncongested egress ports, configure a weighted random early detection (WRED) drop profile and apply it to the egress queue that is causing the congestion. The drop profile prevents the congested egress queue from affecting egress queues on other ports by dropping frames at the egress instead of causing congestion at the ingress port.



**NOTE:** On systems that support lossless transport, do not configure drop profiles for lossless forwarding classes such as the default fcoe and no-loss forwarding classes. FCoE and other lossless traffic queues require lossless behavior. Use priority-based flow control (PFC) to prevent frame drop on lossless priorities.

- On systems that use different classifiers for unicast and multidestination traffic and that support lossless transport, on an ingress port, do not configure classifiers that map the same IEEE 802.1p code point to both a multidestination traffic flow and a lossless unicast traffic flow (such as the

default lossless fcoe or no-loss forwarding classes). Any code point used for multidestination traffic on a port should not be used to classify unicast traffic into a lossless forwarding class on the same port.

If a multidestination traffic flow and a lossless unicast traffic flow use the same code point on a port, the multidestination traffic is treated the same way as the lossless traffic. For example, if priority-based flow control (PFC) is applied to the lossless traffic, the multidestination traffic of the same code point is also paused. During periods of congestion, treating multidestination traffic the same as lossless unicast traffic can create ingress port congestion for the multidestination traffic and affect the multidestination traffic on all of the egress ports the multidestination traffic uses.

For example, the following configuration can cause ingress port congestion for the multidestination flow:

1. For unicast traffic, IEEE 802.1p code point 011 is classified into the fcoe forwarding class:

```
user@switch# set class-of-service classifiers ieee-802.1 ucast-cl forwarding-class fcoe
loss-priority low code-points 011
```

2. For multidestination traffic, IEEE 802.1p code point 011 is classified into the mcast forwarding class:

```
user@switch# set class-of-service classifiers ieee-802.1 mcast-cl forwarding-class mcast
loss-priority low code-points 011
```

3. The unicast classifier that maps traffic with code point 011 to the fcoe forwarding class is mapped to interface xe-0/0/1:

```
user@switch# set class-of-service interfaces xe-0/0/1 unit 0 classifiers ieee-802.1
ucast-cl
```

4. The multidestination classifier that maps traffic with code point 011 to the mcast forwarding class is mapped to all interfaces (multidestination traffic maps to all interfaces and cannot be mapped to individual interfaces):

```
user@switch# set class-of-service multi-destination classifiers ieee-802.1 mcast-cl
```

Because the same code point (011) maps unicast traffic to a lossless traffic flow and also maps multidestination traffic to a multidestination traffic flow, the multidestination traffic flow might experience ingress port congestion during periods of congestion.

To avoid ingress port congestion, do not map the code point used by the multdestination traffic to lossless unicast traffic. For example:

1. Instead of classifying code point 011 into the fcoe forwarding class, classify code point 011 into the best-effort forwarding class:

```
user@switch# set class-of-service classifiers ieee-802.1 ucast_cl forwarding-class best-effort loss-priority low code-points 011
```

2. 

```
user@switch# set class-of-service classifiers ieee-802.1 mcast-cl forwarding-class mcast loss-priority low code-points 011
```

3. 

```
user@switch# set class-of-service interfaces xe-0/0/1 unit 0 classifiers ieee-802.1 ucast_cl
```

4. 

```
user@switch# set class-of-service multi-destination classifiers ieee-802.1 mcast-cl
```

Because the code point 011 does not map unicast traffic to a lossless traffic flow, the multdestination traffic flow does not experience ingress port congestion during periods of congestion.

The best practice is to classify unicast traffic with IEEE 802.1p code points that are also used for multdestination traffic into best-effort forwarding classes.

## Defining CoS Queue Schedulers for Port Scheduling

Schedulers define the CoS properties of output queues. You configure CoS properties in a scheduler, then map the scheduler to a forwarding class. Forwarding classes are in turn mapped to output queues. Classifiers map incoming traffic into forwarding classes based on IEEE 802.1p, DSCP, or EXP code points. CoS scheduling properties include the amount of interface bandwidth assigned to the queue, the priority of the queue, whether explicit congestion notification (ECN) is enabled on the queue, and the WRED packet drop profiles associated with the queue.

The parameters you configure in a scheduler define the following characteristics for the queues mapped to the scheduler:

- **priority**—One of three bandwidth priorities that queues associated with a scheduler can receive:
  - **low**—The scheduler has low priority.

- **high**—The scheduler has high priority. High priority traffic takes precedence over low priority traffic.
- **strict-high**—The scheduler has strict-high priority. Strict-high priority queues receive preferential treatment over low-priority queues and receive all of their configured bandwidth before low-priority queues are serviced. Low-priority queues do not transmit traffic until strict-high priority queues are empty.



**NOTE:** We strongly recommend that you configure a transmit rate on all strict-high priority queues to limit the amount of traffic the switch treats as strict-high priority traffic and prevent strict-high priority queues from starving other queues on the port. This is especially important if you configure more than one strict-high priority queue on a port. If you do not configure a transmit rate to limit the amount of bandwidth strict-high priority queues can use, then the strict-high priority queues can use all of the available port bandwidth and starve other queues on the port. The switch treats traffic in excess of the transmit rate as best-effort traffic that receives bandwidth from the leftover (excess) port bandwidth pool. On strict-high priority queues, all traffic that exceeds the transmit rate shares in the port excess bandwidth pool based on the strict-high priority excess bandwidth sharing weight of “1”, which is not configurable. The actual amount of extra bandwidth that traffic exceeding the transmit rate receives depends on how many other queues consume excess bandwidth and the excess rates of those queues.

- **transmit-rate**—Minimum guaranteed bandwidth, also known as the *committed information rate (CIR)*, set as a percentage rate or as an absolute value in bits per second. By default, the transmit rate also determines the amount of excess (extra) port bandwidth the queue can share if you do not explicitly configure an excess rate. Extra bandwidth is allocated among the queues on the port in proportion to the transmit rate of each queue. Except on QFX10000 switches, you can configure *shaping-rate* to throttle the rate of packet transmission. On QFX10000 switches, on queues that are not strict-high priority queues, you can configure a transmit rate as *exact*, which shapes the transmission by setting the transmit rate as the maximum bandwidth the queue can consume on the port.



**NOTE:** On QFX10000 switches, oversubscribing all 8 queues configured with the transmit rate *exact* (shaping) statement at the [edit class-of-service schedulers *scheduler-name*] hierarchy level might result in less than 100 percent utilization of port bandwidth.

On strict-high priority queues, the transmit rate sets the amount of bandwidth used for strict-high priority forwarding; traffic in excess of the transmit rate is treated as best-effort traffic that receives the queue excess rate.



**NOTE:** Include the preamble bytes and interframe gap (IFG) bytes as well as the data bytes in your bandwidth calculations.

- **excess-rate**—Percentage of extra bandwidth (bandwidth that is not used by other queues) a low-priority queue can receive. If not set, the switch uses the transmit rate to determine extra bandwidth sharing. You cannot set an excess rate on a strict-high priority queue.
- **drop-profile-map**—Drop profile mapping to a packet loss priority to apply WRED to the scheduler and control packet drop for different packet loss priorities during periods of congestion.
- **buffer-size**—Size of the queue buffer as a percentage of the dedicated buffer space on the port, or as a proportional share of the dedicated buffer space on the port that remains after the explicitly configured queues are served.
- **explicit-congestion-notification**—ECN enable on a best-effort queue. ECN enables end-to-end congestion notification between two ECN-enabled endpoints on TCP/IP based networks. ECN must be enabled on both endpoints and on all of the intermediate devices between the endpoints for ECN to work properly. ECN is disabled by default.



**NOTE:** Do not configure drop profiles for the fcoe and no-loss forwarding classes. FCoE and other lossless traffic queues require lossless behavior. Use priority-based flow control (PFC) to prevent frame drop on lossless priorities.

To apply scheduling properties to traffic, map schedulers to forwarding classes using a scheduler map, and then apply the scheduler map to interfaces. Using different scheduler maps, you can map different schedulers to the same forwarding class on different interfaces, to apply different scheduling to that traffic on different interfaces.

To configure a scheduler using the CLI:

1. Name the scheduler and set the minimum guaranteed bandwidth for the queue; optionally, set a maximum bandwidth limit (shaping rate) on a low priority queue by configuring either *shaping-rate* (except on QFX10000 switches) or the *exact* option (only on QFX10000 switches):

```
[edit class-of-service]
user@switch# set schedulers scheduler-name transmit-rate (rate | percent percentage) <exact>
```

2. Set the amount of excess bandwidth a low-priority queue can share:

```
[edit class-of-service]
user@switch# set schedulers scheduler-name excess-rate percent percentage
```

3. Set the queue priority:

```
[edit class-of-service schedulers scheduler-name]
user@switch# set priority level
```

4. Specify drop profiles for packet loss priorities using a drop profile map:

```
[edit class-of-service schedulers scheduler-name]
user@switch# set drop-profile-map loss-priority (low | medium-high | high) drop-profile drop-profile-name
```

5. Configure the size of the buffer space for the queue:

```
[edit class-of-service schedulers scheduler-name]
user@switch# set buffer-size (percent percent | remainder)
```

6. Enable ECN, if desired (on best-effort traffic only):

```
[edit class-of-service schedulers scheduler-name]
user@switch# set explicit-congestion-notification
```

7. Configure a scheduler map to map the scheduler to a forwarding class, which applies the scheduler's properties to the traffic in that forwarding class:

```
[edit class-of-service]
user@switch# set scheduler-maps scheduler-map-name forwarding-class forwarding-class-name
scheduler scheduler-name
```

8. Assign the scheduler map and its associated schedulers to one or more interfaces.

```
[edit class-of-service]
user@switch# set interfaces interface-name scheduler-map scheduler-map-name
```

## RELATED DOCUMENTATION

*Example: Configuring Queue Schedulers for Port Scheduling*

*Example: Configuring ECN*



*Defining CoS Queue Scheduling Priority*

*Configuring CoS WRED Drop Profiles*

*Monitoring CoS Scheduler Maps*

*Understanding CoS Port Schedulers on QFX Switches*

*CoS Explicit Congestion Notification (ECN)*

## Defining CoS Queue Scheduling Priority

### IN THIS SECTION

- [Platform-Specific Behavior | 139](#)

You can configure the scheduling priority of individual queues by specifying the priority in a scheduler, and then associating the scheduler with a queue by using a scheduler map.



**NOTE:** By default, all queues are low priority queues.

The switch services low priority queues after servicing any queue that has strict-high priority traffic or high priority traffic. Strict-high priority queues receive preferential treatment over all other queues and receive all of their configured bandwidth before other queues are serviced. Low-priority queues do not transmit traffic until strict-high priority queues are empty, and receive the bandwidth that remains after the strict-high queues have been serviced. High priority queues receive preference over low priority queues.

- To configure queue priority using the CLI:

```
[edit class-of-service]
user@host# set schedulers scheduler-name priority level
```

### Platform-Specific Behavior

Use the following table to review platform-specific behaviors for your platforms.

Table 49: Platform-Specific Behavior

Platform	Difference
QFX5000 Series and EX4600	<ul style="list-style-type: none"> <li>On QFX5100 and EX4600 switches, you can configure only one forwarding-class-set (priority group) as strict-high priority. All queues which are part of that strict-high forwarding class set then act as strict-high queues.</li> <li>On QFX5200 switches, it is not possible to support multiple queues with strict-high priority because QFX5200 doesn't support flexible hierarchical scheduling. When multiple strict-high priority queues are configured, all of those queues are treated as strict-high priority but the higher number queue among them is given highest priority.</li> <li>On QFX5200, QFX5100, and EX4600 switches, apply a shaping rate to strict-high priority queues to prevent those queues from starving other queues on the port.</li> </ul>
QFX10000 Series	<ul style="list-style-type: none"> <li>On QFX10000 switches, you can configure as many queues as you want as strict-high priority queues.</li> <li>On QFX10000 switches, apply a transmit rate to strict-high priority queues to prevent those queues from starving other queues on the port.</li> <li>QFX10000 switches treat traffic in excess of the transmit rate as best-effort traffic that receives bandwidth from the leftover (excess) port bandwidth pool.</li> </ul>

## RELATED DOCUMENTATION

*Example: Configuring Queue Scheduling Priority*

*Monitoring CoS Scheduler Maps*

## Example: Configuring Queue Scheduling Priority

### IN THIS SECTION

- [Requirements | 142](#)
- [Overview | 142](#)
- [Verification | 144](#)

You can configure the bandwidth scheduling priority of individual queues by specifying the priority in a scheduler, and then using a scheduler map to associate the scheduler with a queue.

## Configuring Queue Scheduling Priority

### CLI Quick Configuration

To quickly configure queue scheduling priority, copy the following commands, paste them in a text file, remove line breaks, change variables and details to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level:

```
[edit class-of-service]
set schedulers fcoe-sched priority low
set schedulers nl-sched priority low
set scheduler-maps schedmap1 forwarding-class fcoe scheduler fcoe-sched
set scheduler-maps schedmap1 forwarding-class no-loss scheduler nl-sched
```

### Step-by-Step Procedure

To configure queue priority using the CLI:

1. Create the FCoE scheduler with `low` priority:

```
[edit class-of-service]
user@switch# set schedulers fcoe-sched priority low
```

2. Create the no-loss scheduler with low priority:

```
[edit class-of-service]
user@switch# set schedulers nl-sched priority low
```

3. Associate the schedulers with the desired queues in the scheduler map:

```
[edit class-of-service]
user@switch# set scheduler-maps schedmap1 forwarding-class fcoe scheduler fcoe-sched
user@switch# set scheduler-maps schedmap1 forwarding-class no-loss scheduler nl-sched
```

## Requirements

This example uses the following hardware and software components:

- One QFX switch
- Any supported Junos release

## Overview

Queues can have one of several bandwidth priorities:

- **strict-high**—Strict-high priority allocates bandwidth to the queue before any other queue receives bandwidth. Other queues receive the bandwidth that remains after the strict-high queue has been serviced.



### NOTE:

If you configure strict-high priority queues on a port, we strongly recommend that you configure a transmit rate (for QFX10000 Series switches) or a shaping rate (non-QFX10000 Series switches) on those queues. The transmit rate or shaping rate sets the amount of traffic that the switch forwards as strict-high priority; traffic in excess of the transmit rate or shaping rate is treated as best-effort traffic that receives the queue excess rate. Even if you configure only one strict-high priority queue, we strongly recommend that you configure a transmit rate or shaping rate on the queue to prevent it from starving other queues. If you do not configure a transmit rate or shaping rate to limit the amount of bandwidth a strict-high priority queue can use, then the strict-high priority queue can use all of the available port bandwidth and starve other queues on the port.



**NOTE:** On switches that support enhanced transmission selection (ETS) hierarchical scheduling, if you use ETS and you configure a strict-high priority queue, you must create a forwarding class set that is dedicated only to strict-high priority traffic. Only one forwarding class set can contain a strict-high priority queue. Queues that are not strict-high priority cannot belong to the same forwarding class set as strict-high priority queues.

On switches that use different output queues for unicast and multideestination traffic, the multideestination forwarding class set cannot contain strict-high priority queues.

- **high**—High priority. Traffic with high priority is serviced after any queue that has a strict-high priority, and before queues with low priority.
- **low**—Low priority. Traffic with low priority is serviced after any queue that has a strict-high priority.



**NOTE:** By default, all queues are low priority queues.

Table 50 on page 143 shows the configuration components for this example.

This example describes how to set the queue priority for two forwarding classes (queues) named `fcoe` and `no-loss`. Both queues have a priority of `low`. The scheduler for the `fcoe` queue is named `fcoe-sched` and the scheduler for the `no-loss` queue is named `n1-sched`. One scheduler map, `schedmap1`, associates the schedulers to the queues.

**Table 50: Components of the Queue Scheduler Priority Configuration Example**

Component	Settings
Hardware	One switch
Schedulers	fcoe-sched for FCoE traffic n1-sched for no-loss traffic
Priority	low for FCoE traffic low for no-loss traffic

Table 50: Components of the Queue Scheduler Priority Configuration Example *(Continued)*

Component	Settings
Scheduler map	<div>schedmap1:</div> <div>FCoE mapping: scheduler fcoe-sched to forwarding class fcoe</div> <div>No-loss mapping: scheduler nl-sched to forwarding class no-loss</div>

Verification

IN THIS SECTION

- [Verifying the Queue Scheduling Priority | 144](#)
- [Verifying the Scheduler-to-Forwarding-Class Mapping | 145](#)
- [Platform-Specific Behavior | 145](#)

To verify that you configured the queue scheduling priority for bandwidth and mapped the schedulers to the correct forwarding classes, perform these tasks:

Verifying the Queue Scheduling Priority

Purpose

Verify that you configured the queue schedulers fcoe-sched and nl-sched with low queue scheduling priority.

## Action

Display the fcoe-sched scheduler priority configuration using the operational mode command `show configuration class-of-service schedulers fcoe-sched priority`:

```
user@switch> show configuration class-of-service schedulers fcoe-sched priority
priority low;
```

Display the nl-sched scheduler priority configuration using the operational mode command `show configuration class-of-service schedulers nl-sched priority`:

```
user@switch> show configuration class-of-service schedulers nl-sched priority
priority low;
```

## Verifying the Scheduler-to-Forwarding-Class Mapping

### Purpose

Verify that you configured the scheduler map `schedmap1` to map scheduler `fcoe-sched` to forwarding class `fcoe` and schedule `nl-sched` to forwarding class `no-loss`.

## Action

Display the scheduler map `schedmap1` using the operational mode command `show configuration class-of-service scheduler-maps schedmap1`:

```
user@switch> show configuration class-of-service scheduler-maps schedmap1
forwarding-class fcoe scheduler fcoe-sched;
forwarding-class no-loss scheduler nl-sched;
```

## Platform-Specific Behavior

Use the following table to review platform-specific behaviors for your platforms.

Table 51: Platform-Specific Behavior

Platform	Difference
QFX5000 Series and EX4600	<ul style="list-style-type: none"> <li>On QFX5100 and EX4600 switches, you can configure only one forwarding-class-set (priority group) as strict-high priority. All queues which are part of that strict-high forwarding class set then act as strict-high queues.</li> <li>On QFX5200 switches, it is not possible to support multiple queues with strict-high priority because QFX5200 doesn't support flexible hierarchical scheduling. When multiple strict-high priority queues are configured, all of those queues are treated as strict-high priority but the higher number queue among them is given highest priority.</li> <li>On QFX5200, QFX5100, and EX4600 switches, apply a shaping rate to strict-high priority queues to prevent those queues from starving other queues on the port.</li> </ul>
QFX10000 Series	<ul style="list-style-type: none"> <li>On QFX10000 switches, you can configure as many queues as you want as strict-high priority queues.</li> <li>On QFX10000 switches, apply a transmit rate to strict-high priority queues to prevent those queues from starving other queues on the port.</li> </ul>

## RELATED DOCUMENTATION

---

*Defining CoS Queue Scheduling Priority*


---

*Monitoring CoS Scheduler Maps*



# Monitoring CoS Scheduler Maps

IN THIS SECTION

- Purpose | 147
- Action | 147
- Meaning | 147
- Platform-Specific Behavior | 148

## Purpose

Use the monitoring functionality to display assignments of CoS forwarding classes to schedulers.

## Action

To monitor CoS scheduler maps in the CLI, enter the CLI command:

```
user@switch> show class-of-service scheduler-map
```

To monitor a specific scheduler map in the CLI, enter the CLI command:

```
user@switch> show class-of-service scheduler-map scheduler-map-name
```

## Meaning

[Table 52 on page 147](#) summarizes key output fields for CoS scheduler maps.

Table 52: Summary of Key CoS Scheduler Maps Output Fields

Field	Values
Scheduler map	Name of a scheduler map that maps forwarding classes to schedulers.
Index	Index of a specific object—scheduler maps, schedulers, or drop profiles.

**Table 52: Summary of Key CoS Scheduler Maps Output Fields** *(Continued)*

Field	Values
Scheduler	Name of a scheduler that controls queue properties such as bandwidth and scheduling priority.
Forwarding class	Name(s) of the forwarding class(es) to which the scheduler is mapped.
Transmit rate	Guaranteed minimum bandwidth configured on the queue mapped to the scheduler.
Priority	<p>Scheduling priority of traffic on a queue:</p> <ul style="list-style-type: none"> <li>• strict-high or high—Packets on a strict-high priority queue are transmitted first, before all other traffic, up to the configured maximum bandwidth (shaping rate).</li> <li>• low—Packets in this queue are transmitted after packets in the strict-high queue.</li> </ul>
Drop Profiles	Name and index of a drop profile that is mapped to a specific loss priority and protocol pair. The drop profile determines the way best effort queues drop packets during periods of congestion.
Loss Priority	Packet loss priority mapped to the drop profile. You can configure different drop profiles for low, medium-high, and high loss priority traffic.
Protocol	Transport protocol of the drop profile for the particular priority.
Name	Name of the drop profile.

## Platform-Specific Behavior

Use the following table to review platform-specific behaviors for your platforms.

**Table 53: Platform-Specific Behavior**

Platform	Difference
EX4600	<ul style="list-style-type: none"> <li>On EX4600 switches, only one queue can be configured as strict-high or high priority.</li> </ul>
QFX10000	<ul style="list-style-type: none"> <li>On strict-high priority queues on QFX10000 switches, transmit-rate defines the maximum amount of traffic on the queue that is treated as strict-high priority traffic.</li> <li>On QFX10000 switches, you can configure more than one strict-high priority queue.</li> </ul>

## Understanding CoS Traffic Control Profiles

A traffic control profile defines the output bandwidth and scheduling characteristics of forwarding class sets (priority groups). The forwarding classes (which are mapped to output queues) that belong to a forwarding class set (fc-set) share the bandwidth that you assign to the fc-set in the traffic control profile.

This two-tier hierarchical scheduling architecture provides flexibility in allocating resources among forwarding classes, and also:

- Assigns a portion of port bandwidth to an fc-set. You define the port resources for the fc-set in a traffic control profile.
- Allocates fc-set bandwidth among the forwarding classes (queues) that belong to the fc-set. A scheduler map attached to the traffic control profile defines the amount of the fc-set's resources that each forwarding class can use.

Attaching an fc-set and a traffic control profile to a port defines the hierarchical scheduling properties of the group and the forwarding classes that belong to the group.

The ability to create fc-sets supports enhanced transmission selection (ETS), which is described in IEEE 802.1Qaz. When an fc-set does not use its allocated port bandwidth, ETS shares the excess port bandwidth among other fc-sets on the port in proportion to their guaranteed minimum bandwidth (guaranteed rate). This utilizes the port bandwidth better than scheduling schemes that reserve bandwidth for groups even if that bandwidth is not used. ETS shares unused port bandwidth, so traffic

groups that need extra bandwidth can use it if the bandwidth is available, while preserving the ability to specify the minimum guaranteed bandwidth for traffic groups.

Traffic control profiles define the following CoS properties for fc-sets:

- Minimum guaranteed bandwidth—Also known as the *committed information rate (CIR)*. This is the minimum amount of port bandwidth the priority group receives. Priorities in the priority group receive their minimum guaranteed bandwidth as a portion of the priority group's minimum guaranteed bandwidth. The `guaranteed-rate` statement defines the minimum guaranteed bandwidth.



**NOTE:** You cannot apply a traffic control profile with a minimum guaranteed bandwidth to a priority group that includes strict-high priority queues.

- Shared excess (extra) bandwidth—When the priority groups on a port do not consume the full amount of bandwidth allocated to them or there is unallocated link bandwidth available, priority groups can contend for that extra bandwidth if they need it. Priorities in the priority group contend for extra bandwidth as a portion of the priority group's extra bandwidth. The amount of extra bandwidth for which a priority group can contend is proportional to the priority group's guaranteed minimum bandwidth (guaranteed rate).
- Maximum bandwidth—Also known as *peak information rate (PIR)*. This is the maximum amount of port bandwidth the priority group receives. Priorities in the priority group receive their maximum bandwidth as a portion of the priority group's maximum bandwidth. The `shaping-rate` statement defines the maximum bandwidth.
- Queue scheduling—Each traffic control profile includes a scheduler map. The scheduler map maps forwarding classes (priorities) to schedulers to define the scheduling characteristics of the individual forwarding classes in the fc-set. The resources scheduled for each forwarding class represent portions of the resources that the traffic control profile schedules for the entire fc-set, not portions of the total link bandwidth. The `scheduler-maps` statement defines the mapping of forwarding classes to schedulers.

## RELATED DOCUMENTATION

*Understanding CoS Hierarchical Port Scheduling (ETS)*

*Example: Configuring CoS Hierarchical Port Scheduling (ETS)*

*Example: Configuring Traffic Control Profiles (Priority Group Scheduling)*

*Defining CoS Traffic Control Profiles (Priority Group Scheduling)*

## Understanding CoS Priority Group Scheduling

### IN THIS SECTION

- [Priority Group Scheduling Components | 151](#)
- [Default Traffic Control Profile | 152](#)
- [Guaranteed Rate \(Minimum Guaranteed Bandwidth\) | 152](#)
- [Sharing Extra Bandwidth | 153](#)
- [Shaping Rate \(Maximum Bandwidth\) | 153](#)
- [Scheduler Maps | 153](#)

Priority group scheduling defines the class-of-service (CoS) properties of a group of output queues (priorities). Priority group scheduling works with output queue scheduling to create a two-tier hierarchical scheduler. The hierarchical scheduler allocates bandwidth to a group of queues (a priority group, called a forwarding class set in Junos OS configuration). Queue scheduling determines the portion of the priority group bandwidth that the particular queue can use.

You configure priority group scheduling in a traffic control profile and then associate the traffic control profile with a forwarding class set and an interface. You attach a scheduler map to the traffic control profile to specify the queue scheduling characteristics.



**NOTE:** When you configure bandwidth for a queue or a priority group, the switch considers only the data as the configured bandwidth. The switch does not account for the bandwidth consumed by the preamble and the interframe gap (IFG). Therefore, when you calculate and configure the bandwidth requirements for a queue or for a priority group, consider the preamble and the IFG as well as the data in the calculations.

### Priority Group Scheduling Components

[Table 54 on page 152](#) provides a quick reference to the traffic control profile components you can configure to determine the bandwidth properties of priority groups, and [Table 55 on page 152](#) provides a quick reference to some related scheduling configuration components.

**Table 54: Priority Group Scheduler Components**

Traffic Control Profile Component	Description
Guaranteed rate	Sets the minimum guaranteed port bandwidth for the priority group. Extra port bandwidth is shared among priority groups in proportion to the guaranteed rate of each priority group on the port.
Shaping rate	Sets the maximum port bandwidth the priority group can consume.
Scheduler map	Maps schedulers to queues (forwarding classes, also called priorities). This determines the portion of the priority group bandwidth that a queue receives.

**Table 55: Other Scheduling Components**

Other Scheduling Components	Description
Forwarding class	Maps traffic to a queue (priority).
Forwarding class set	Name of a priority group. You map forwarding classes to priority groups. A forwarding class set consists of one or more forwarding classes.
Scheduler	Sets the bandwidth and scheduling priority of individual queues (forwarding classes).

## Default Traffic Control Profile

There is no default traffic control profile.

## Guaranteed Rate (Minimum Guaranteed Bandwidth)

The guaranteed rate determines the minimum guaranteed bandwidth for each priority group. It also determines how much excess (extra) port bandwidth the priority group can share; each priority group shares extra port bandwidth in proportion to its guaranteed rate. You specify the rate in bits per second as a fixed value such as 3 Mbps or as a percentage of the total port bandwidth.

The minimum transmission bandwidth can exceed the configured rate if additional bandwidth is available from other priority groups on the port. In case of congestion, the configured guaranteed rate is guaranteed for the priority group. This property enables you to ensure that each priority group receives the amount of bandwidth appropriate to its level of service.



**NOTE:** Configuring the minimum guaranteed bandwidth (transmit rate) for a forwarding class does not work unless you also configure the minimum guaranteed bandwidth (guaranteed rate) for the forwarding class set in the traffic control profile.

Additionally, the sum of the transmit rates of the queues in a forwarding class set should not exceed the guaranteed rate for the forwarding class set. (You cannot guarantee a minimum bandwidth for the queues that is greater than the minimum bandwidth guaranteed for the entire set of queues.)

You cannot configure a guaranteed rate for forwarding class sets that include strict-high priority queues.

## Sharing Extra Bandwidth

Extra bandwidth is available to priority groups when the priority groups do not use the full amount of available port bandwidth. This extra port bandwidth is shared among the priority groups based on the minimum guaranteed bandwidth of each priority group.

For example, Port A has three priority groups: fc-set-1, fc-set-2, and fc-set-3. Fc-set-1 has a guaranteed rate of 2 Gbps, fc-set-2 has a guaranteed rate of 2 Gbps, and fc-set-3 has a guaranteed rate of 4 Gbps. After servicing the minimum guaranteed bandwidth of these priority groups, the port has an extra 2 Gbps of available bandwidth, and all three priority groups have still have packets to forward. The priority groups receive the extra bandwidth in proportion to their guaranteed rates, so fc-set-1 receives an extra 500 Mbps, fc-set-2 receives an extra 500 Mbps, and fc-set-3 receives an extra 1 Gbps.

## Shaping Rate (Maximum Bandwidth)

The shaping rate determines the maximum bandwidth the priority group can consume. You specify the rate in bits per second as a fixed value such as 5 Mbps or as a percentage of the total port bandwidth.

The maximum bandwidth for a priority group depends on the total bandwidth available on the port and how much bandwidth the other priority groups on the port consume.

## Scheduler Maps

A scheduler map maps schedulers to queues. When you associate a scheduler map with a traffic control profile, then associate the traffic control profile with an interface and a forwarding class set, the

scheduling defined by the scheduler map determines the portion of the priority group resources that each individual queue can use.

You can associate up to four user-defined scheduler maps with traffic control profiles.

## RELATED DOCUMENTATION

*Understanding Junos CoS Components*

*Understanding CoS Output Queue Schedulers*

*Understanding CoS Hierarchical Port Scheduling (ETS)*

*Understanding CoS Scheduling Behavior and Configuration Considerations*

*Example: Configuring CoS Hierarchical Port Scheduling (ETS)*

*Example: Configuring Minimum Guaranteed Output Bandwidth*

*Example: Configuring Maximum Output Bandwidth*

*Example: Configuring Queue Schedulers*

*Example: Configuring Traffic Control Profiles (Priority Group Scheduling)*

*Example: Configuring WRED Drop Profiles*

*Example: Configuring Drop Profile Maps*

## Defining CoS Traffic Control Profiles (Priority Group Scheduling)

A traffic control profile defines the output bandwidth and scheduling characteristics of forwarding class sets (priority groups). The forwarding classes (which are mapped to output queues) contained in a forwarding class set (fc-set) share the bandwidth resources that you configure in the traffic control profile. A scheduler map associates forwarding classes with schedulers to define how the individual forwarding classes that belong to an fc-set share the bandwidth allocated to that fc-set.

The parameters you configure in a traffic control profile define the following characteristics for the fc-set:

- **guaranteed-rate**—Minimum bandwidth, also known as the *committed information rate (CIR)*. The guaranteed rate also determines the amount of excess (extra) port bandwidth that the fc-set can share. Extra port bandwidth is allocated among the fc-sets on a port in proportion to the guaranteed rate of each fc-set.





**NOTE:** You cannot configure a guaranteed rate for a fc-set that includes strict-high priority queues. If the traffic control profile is for an fc-set that contains strict-high priority queues, do not configure a guaranteed rate.

- **shaping-rate**—Maximum bandwidth, also known as the *peak information rate (PIR)*.
- **scheduler-map**—Bandwidth and scheduling characteristics for the queues, defined by mapping forwarding classes to schedulers. (The queue scheduling characteristics represent amounts or percentages of the fc-set bandwidth, not the amounts or percentages of total link bandwidth.)



**NOTE:** Because a port can have more than one fc-set, when you assign resources to an fc-set, keep in mind that the total port bandwidth must serve all of the queues associated with that port.

To configure a traffic control profile using the CLI:

1. Name the traffic control profile and define the minimum guaranteed bandwidth for the fc-set:

```
[edit class-of-service]
user@switch# set traffic-control-profiles traffic-control-profile-name guaranteed-rate (rate
| percent percentage)
```

2. Define the maximum bandwidth for the fc-set:

```
[edit class-of-service traffic-control-profiles traffic-control-profile-name]
user@switch# set shaping-rate (rate | percent percentage)
```

3. Attach a scheduler map to the traffic control profile:

```
[edit class-of-service traffic-control-profiles ]
user@switch# set scheduler-map scheduler-map-name
```

## RELATED DOCUMENTATION

*Example: Configuring CoS Hierarchical Port Scheduling (ETS)*

*Example: Configuring Traffic Control Profiles (Priority Group Scheduling)*

*Example: Configuring Minimum Guaranteed Output Bandwidth*

---

*Example: Configuring Maximum Output Bandwidth*

---

*Defining CoS Queue Schedulers*

---

*Understanding CoS Traffic Control Profiles*

## Example: Configuring Traffic Control Profiles (Priority Group Scheduling)

### IN THIS SECTION

- [Requirements | 156](#)
- [Overview | 156](#)
- [Verification | 159](#)

A traffic control profile defines the output bandwidth and scheduling characteristics of forwarding class sets (priority groups). The forwarding classes (queues) mapped to a forwarding class set share the bandwidth resources that you configure in the traffic control profile. A scheduler map associates forwarding classes with schedulers to define how the individual queues in a forwarding class set share the bandwidth allocated to that forwarding class set.

### Requirements

This example uses the following hardware and software components:

- QFX Series switch
- Any supported Junos Release

Use [Feature Explorer](#) to confirm platform and release support for ETS.

### Overview

#### IN THIS SECTION

- [Configuring a Traffic Control Profile | 158](#)

The parameters you configure in a traffic control profile define the following characteristics for the priority group:

- **guaranteed-rate**—Minimum bandwidth, also known as the *committed information rate (CIR)*. Each fc-set receives a minimum of either the configured amount of absolute bandwidth or the configured percentage of bandwidth. The guaranteed rate also determines the amount of excess (extra) port bandwidth that the fc-set can share. Extra port bandwidth is allocated among the fc-sets on a port in proportion to the guaranteed rate of each fc-set.



**NOTE:** In order for the *transmit-rate* option (minimum bandwidth for a queue that you set using scheduler configuration) to work properly, you must configure the *guaranteed-rate* for the fc-set. If an fc-set does not have a guaranteed minimum bandwidth, the forwarding classes that belong to the fc-set cannot have a guaranteed minimum bandwidth.



**NOTE:** Include the preamble bytes and interframe gap bytes as well as the data bytes in your bandwidth calculations.

- **shaping-rate**—Maximum bandwidth, also known as the *peak information rate (PIR)*. Each fc-set receives a maximum of the configured amount of absolute bandwidth or the configured percentage of bandwidth, even if more bandwidth is available.



**NOTE:** Include the preamble bytes and interframe gap bytes as well as the data bytes in your bandwidth calculations.

- **scheduler-map**—Bandwidth and scheduling characteristics for the queues, defined by mapping forwarding classes to schedulers. (The queue scheduling characteristics represent amounts or percentages of the fc-set bandwidth, not the amounts or percentages of total link bandwidth.)



**NOTE:** Because a port can have more than one fc-set, when you assign resources to an fc-set, keep in mind that the total port bandwidth must serve all of the queues associated with that port.

For example, if you map three fc-sets to a 10-Gigabit Ethernet port, the queues associated with all three of the fc-sets share the 10-Gbps bandwidth as defined by the traffic control profiles. Therefore, the total combined *guaranteed-rate* value of the three fc-sets should not exceed 10 Gbps. If you configure guaranteed rates whose sum exceeds the port bandwidth, the system sends a syslog message to notify you that the

configuration is not valid. However, the system does not perform a commit check. If you commit a configuration in which the sum of the guaranteed rates exceeds the port bandwidth, the hierarchical scheduler behaves unpredictably.

The sum of the forwarding class (queue) transmit rates cannot exceed the total guaranteed-rate of the fc-set to which the forwarding classes belong. If you configure transmit rates whose sum exceeds the fc-set guaranteed rate, the commit check fails and the system rejects the configuration.

If you configure the guaranteed-rate of an fc-set as a percentage, configure all of the transmit rates associated with that fc-set as percentages. In this case, if any of the transmit rates are configured as absolute values instead of percentages, the configuration is not valid and the system sends a syslog message.

## Configuring a Traffic Control Profile

### Step-by-Step Procedure

This example describes how to configure a traffic control profile named `san-tcp` with a scheduler map named `san-map1` and allocate to it a minimum bandwidth of 4 Gbps and a maximum bandwidth of 8 Gbps:

1. Create the traffic control profile and set the guaranteed-rate (minimum guaranteed bandwidth) to 4g:

```
[edit class-of-service]
user@switch# set traffic-control-profiles san-tcp guaranteed-rate 4g
```

2. Set the shaping-rate (maximum guaranteed bandwidth) to 8g:

```
[edit class-of-service]
user@switch# set traffic-control-profiles san-tcp shaping-rate 8g
```

3. Associate the scheduler map `san-map1` with the traffic control profile:

```
[edit class-of-service]
user@switch# set traffic-control-profiles san-tcp scheduler-map san-map1
```

## Verification

### IN THIS SECTION

- [Verifying the Traffic Control Profile Configuration | 159](#)

## Verifying the Traffic Control Profile Configuration

### Purpose

Verify that you created the traffic control profile `san-tcp` with a minimum guaranteed bandwidth of 4 Gbps, a maximum bandwidth of 8 Gbps, and the scheduler map `san-map1`.

### Action

List the traffic control profile using the operational mode command `show configuration class-of-service traffic-control-profiles san-tcp`:

```
user@switch> show configuration class-of-service traffic-control-profiles san-tcp
scheduler-map san-map1;
shaping-rate percent 8g;
guaranteed-rate 4g;
```

## RELATED DOCUMENTATION

---

*Example: Configuring CoS Hierarchical Port Scheduling (ETS)*

---

*Example: Configuring Minimum Guaranteed Output Bandwidth*

---

*Example: Configuring Maximum Output Bandwidth*

---

*Example: Configuring Queue Schedulers*

---

*Defining CoS Traffic Control Profiles (Priority Group Scheduling)*

---

*Understanding CoS Traffic Control Profiles*

---

*Understanding CoS Hierarchical Port Scheduling (ETS)*

## Understanding CoS Priority Group and Queue Guaranteed Minimum Bandwidth

### IN THIS SECTION

- [Guaranteeing Bandwidth Using Hierarchical Scheduling | 160](#)
- [Priority Group Guaranteed Rate \(Guaranteed Minimum Bandwidth\) | 162](#)
- [Queue Transmit Rate \(Guaranteed Minimum Bandwidth\) | 162](#)

You can set a guaranteed minimum bandwidth for individual forwarding classes (queues) and for groups of forwarding classes called *forwarding class sets* (priority groups). Setting a minimum guaranteed bandwidth ensures that priority groups and queues receive the bandwidth required to support the expected traffic.

### Guaranteeing Bandwidth Using Hierarchical Scheduling

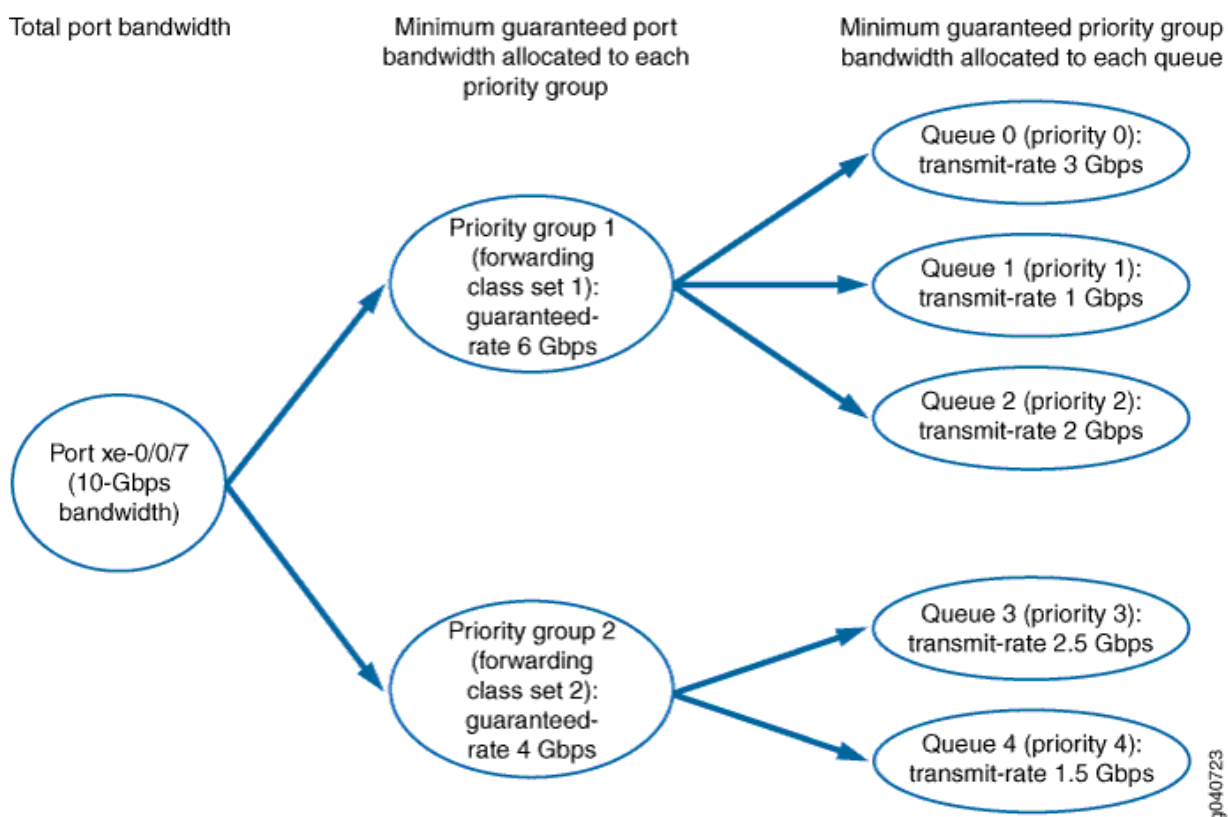
The *guaranteed-rate* value for the priority group (configured in a traffic control profile) defines the minimum amount of bandwidth allocated to a forwarding class set on a port, whereas the *transmit-rate* value of the queue (configured in a scheduler) defines the minimum amount of bandwidth allocated to a particular queue in a priority group. The queue bandwidth is a portion of the priority group bandwidth.



**NOTE:** You cannot configure a minimum guaranteed bandwidth (transmit rate) for a forwarding class that is mapped to a strict-high priority queue, and you cannot configure a minimum guaranteed bandwidth (guaranteed rate) for a priority group that includes strict-high priority queues.

[Figure 5 on page 161](#) shows how the total port bandwidth is allocated to priority groups (forwarding class sets) based on the guaranteed rate of each priority group. It also shows how the guaranteed bandwidth of each priority group is allocated to the queues in the priority group based on the transmit rate of each queue.

Figure 5: Allocating Guaranteed Bandwidth Using Hierarchical Scheduling



The sum of the priority group guaranteed rates cannot exceed the total port bandwidth. If you configure guaranteed rates whose sum exceeds the port bandwidth, the system sends a syslog message to notify you that the configuration is not valid. However, the system does not perform a commit check. If you commit a configuration in which the sum of the guaranteed rates exceeds the port bandwidth, the hierarchical scheduler behaves unpredictably.

The sum of the queue transmit rates cannot exceed the total guaranteed rate of the priority group to which the queues belong. If you configure transmit rates whose sum exceeds the priority group guaranteed rate, the commit check fails and the system rejects the configuration.



**NOTE:** You must set both the priority group guaranteed-rate value and the queue transmit-rate value in order to configure the minimum bandwidth for individual queues. If you set the transmit-rate value but do not set the guaranteed-rate value, the configuration fails.

You can set the guaranteed-rate value for a priority group without setting the transmit-rate value for individual queues in the priority group. However, queues that do not have a configured transmit-rate value can become starved for bandwidth if other higher-priority

queues need the priority group's bandwidth. To avoid starving a queue, it is a good practice to configure a `transmit-rate` value for most queues.

If you configure the guaranteed rate of a priority group as a percentage, configure all of the transmit rates associated with that priority group as percentages. In this case, if any of the transmit rates are configured as absolute values instead of percentages, the configuration is not valid and the system sends a syslog message.

## Priority Group Guaranteed Rate (Guaranteed Minimum Bandwidth)

Setting a priority group (forwarding class set) `guaranteed-rate` enables you to reserve a portion of the port bandwidth for the forwarding classes (queues) in that forwarding class set. The minimum bandwidth (`guaranteed-rate`) that you configure for a priority group sets the minimum bandwidth available to all of the forwarding classes in the forwarding class set.

The combined `guaranteed-rate` value of all of the forwarding class sets associated with an interface cannot exceed the amount of bandwidth available on that interface.

You configure the priority group `guaranteed-rate` in the traffic control profile. You cannot apply a traffic control profile that has a guaranteed rate to a priority group that includes a strict-high priority queue.

## Queue Transmit Rate (Guaranteed Minimum Bandwidth)

Setting a queue (forwarding class) `transmit-rate` enables you to reserve a portion of the priority group bandwidth for the individual queue. For example, a queue that handles Fibre Channel over Ethernet (FCoE) traffic might require a minimum rate of 4 Gbps to ensure the *class of service* that storage area network (SAN) traffic requires.

The priority group `guaranteed-rate` sets the aggregate minimum amount of bandwidth available to the queues that belong to the priority group. The cumulative total minimum bandwidth the queues consume cannot exceed the minimum bandwidth allocated to the priority group to which they belong. (The combined transmit rates of the queues in a priority group cannot exceed the priority group's guaranteed rate.)

You must configure the `guaranteed-rate` value of the priority group in order to set a `transmit-rate` value for individual queues that belong to the priority group. The reason is that if there is no guaranteed bandwidth for a priority group, there is no way to guarantee bandwidth for queues in that priority group.

You configure the queue `transmit-rate` in the scheduler configuration. You cannot configure a transmit rate for a strict-high priority queue.



## RELATED DOCUMENTATION

*Understanding CoS Output Queue Schedulers*

*Understanding CoS Traffic Control Profiles*

*Example: Configuring CoS Hierarchical Port Scheduling (ETS)*

*Example: Configuring Queue Schedulers*

*Example: Configuring Traffic Control Profiles (Priority Group Scheduling)*

*Defining CoS Queue Schedulers*

*Defining CoS Traffic Control Profiles (Priority Group Scheduling)*

## Example: Configuring Minimum Guaranteed Output Bandwidth

### IN THIS SECTION

- [Requirements | 165](#)
- [Overview | 165](#)
- [Verification | 167](#)

Scheduling the minimum guaranteed output bandwidth for a queue (forwarding class) requires configuring both tiers of the two-tier hierarchical scheduler. One tier is scheduling the resources for the individual queue. The other tier is scheduling the resources for the priority group (forwarding class set) to which the queue belongs. You set a minimum guaranteed bandwidth to ensure that priority groups and queues receive the bandwidth required to support the expected traffic.

### Configuring Guaranteed Minimum Bandwidth

#### CLI Quick Configuration

To quickly configure the minimum guaranteed bandwidth for a priority group and a queue, copy the following commands, paste them in a text file, remove line breaks, change variables and details to match

your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level:

```
[edit class-of-service]
set schedulers be-sched transmit-rate 2g
set traffic-control-profiles be-tcp guaranteed-rate 4g
set scheduler-maps be-map forwarding-class best-effort scheduler be-sched
set traffic-control-profiles be-tcp scheduler-map be-map
set forwarding-class-sets be-pg class best-effort
set interfaces xe-0/0/7 forwarding-class-set be-pg output-traffic-control-profile be-tcp
```

## Step-by-Step Procedure

To configure the minimum guaranteed bandwidth hierarchical scheduling for a queue and a priority group:

1. Configure the minimum guaranteed queue bandwidth of 2 Gbps for scheduler be-sched:

```
[edit class-of-service schedulers]
user@switch# set be-sched transmit-rate 2g
```

2. Configure the minimum guaranteed priority group bandwidth of 4 Gbps for traffic control profile be-tcp:

```
[edit class-of-service traffic-control-profiles]
user@switch# set be-tcp guaranteed-rate 4g
```

3. Associate the scheduler be-sched with the best-effort queue in the scheduler map be-map:

```
[edit class-of-service scheduler-maps]
user@switch# set be-map forwarding-class best-effort scheduler be-sched
```

4. Associate the scheduler map with the traffic control profile:

```
[edit class-of-service traffic-control-profiles]
user@switch# set be-tcp scheduler-map be-map
```

## 5. Assign the best-effort queue to the priority group be-pg:

```
[edit class-of-service forwarding-class-sets]
user@switch# set be-pg class best-effort
```

## 6. Apply the configuration to interface xe-0/0/7:

```
[edit class-of-service interfaces]
user@switch# set xe-0/0/7 forwarding-class-set be-pg output-traffic-control-profile be-tcp
```

## Requirements

This example uses the following hardware and software components:

- A Juniper Networks QFX switch
- Any supported Junos release

## Overview

The priority group minimum guaranteed bandwidth defines the minimum total amount of bandwidth available for all of the queues in the priority group to meet their minimum bandwidth requirements.

The transmit-rate setting in the scheduler configuration determines the minimum guaranteed bandwidth for an individual queue. The transmit rate also determines the amount of excess (extra) priority group bandwidth that the queue can share. Extra priority group bandwidth is allocated among the queues in the priority group in proportion to the transmit rate of each queue.

The guaranteed-rate setting in the traffic control profile configuration determines the minimum guaranteed bandwidth for a priority group. The guaranteed rate also determines the amount of excess (extra) port bandwidth that the priority group can share. Extra port bandwidth is allocated among the priority groups on a port in proportion to the guaranteed rate of each priority group.



**NOTE:** You must configure both the transmit-rate value for the queue and the guaranteed-rate value for the priority group to set a valid minimum bandwidth guarantee for a queue. (If the priority group does not have a guaranteed minimum bandwidth, there is no guaranteed bandwidth pool from which the queue can take its guaranteed minimum bandwidth.)

The sum of the queue transmit rates in a priority group should not exceed the guaranteed rate for the priority group. (You cannot guarantee a minimum bandwidth for

the queues that is greater than the minimum bandwidth guaranteed for the entire set of queues.)



**NOTE:** When you configure bandwidth for a queue or a priority group, the switch considers only the data as the configured bandwidth. The switch does not account for the bandwidth consumed by the preamble and the interframe gap (IFG). Therefore, when you calculate and configure the bandwidth requirements for a queue or for a priority group, consider the preamble and the IFG as well as the data in the calculations.



**NOTE:** You cannot configure minimum guaranteed bandwidth on strict-high priority queues or on a priority group that contains strict-high priority queues.

This example describes how to:

- Configure a transmit rate (minimum guaranteed queue bandwidth) of 2 Gbps for queues in a scheduler named `be-sched`.
- Configure a guaranteed rate (minimum guaranteed priority group bandwidth) of 4 Gbps for a priority group in a traffic control profile named `be-tcp`.
- Assign the scheduler to a queue named `best-effort` by using a scheduler map named `be-map`.
- Associate the scheduler map `be-map` with the traffic control profile `be-tcp`.
- Assign the queue `best-effort` to a priority group named `be-pg`.
- Assign the priority group and the minimum guaranteed bandwidth scheduling to the egress interface `xe-0/0/7`.

[Table 56 on page 166](#) shows the configuration components for this example:

**Table 56: Components of the Minimum Guaranteed Output Bandwidth Configuration Example**

Component	Settings
Hardware	QFX switch
Minimum guaranteed queue bandwidth	Transmit rate: 2g
Minimum guaranteed priority group bandwidth	Guaranteed rate: 4g

**Table 56: Components of the Minimum Guaranteed Output Bandwidth Configuration Example**  
*(Continued)*

Component	Settings
Scheduler	be-sched
Scheduler map	be-map
Traffic control profile	be-tcp
Forwarding class set (priority group)	be-pg
Queue (forwarding class)	best-effort
Egress interface	xe-0/0/7

## Verification

### IN THIS SECTION

- [Verifying the Minimum Guaranteed Queue Bandwidth | 167](#)
- [Verifying the Priority Group Minimum Guaranteed Bandwidth and Scheduler Map Association | 168](#)
- [Verifying the Scheduler Map Configuration | 168](#)
- [Verifying Queue \(Forwarding Class\) Membership in the Priority Group | 169](#)
- [Verifying the Egress Interface Configuration | 169](#)

To verify the minimum guaranteed output bandwidth configuration, perform these tasks:

### Verifying the Minimum Guaranteed Queue Bandwidth

#### Purpose

Verify that you configured the minimum guaranteed queue bandwidth as 2g in the scheduler be-sched.

## Action

Display the minimum guaranteed bandwidth in the be-sched scheduler configuration using the operational mode command `show configuration class-of-service schedulers be-sched transmit-rate`:

```
user@switch> show configuration class-of-service schedulers be-sched transmit-rate
2g;
```

## Verifying the Priority Group Minimum Guaranteed Bandwidth and Scheduler Map Association

### Purpose

Verify that the minimum guaranteed priority group bandwidth is 4g and the attached scheduler map is be-map in the traffic control profile be-tcp.

## Action

Display the minimum guaranteed bandwidth in the be-tcp traffic control profile configuration using the operational mode command `show configuration class-of-service traffic-control-profiles be-tcp guaranteed-rate`:

```
user@switch> show configuration class-of-service traffic-control-profiles be-tcp guaranteed-rate
4g;
```

Display the scheduler map in the be-tcp traffic control profile configuration using the operational mode command `show configuration class-of-service traffic-control-profiles be-tcp scheduler-map`:

```
user@switch> show configuration class-of-service traffic-control-profiles be-tcp scheduler-map
scheduler-map be-map;
```

## Verifying the Scheduler Map Configuration

### Purpose

Verify that the scheduler map be-map maps the forwarding class best-effort to the scheduler be-sched.

## Action

Display the be-map scheduler map configuration using the operational mode command `show configuration class-of-service schedulers maps be-map`:

```
user@switch> show configuration class-of-service scheduler-maps be-map
forwarding-class best-effort scheduler be-sched;
```

## Verifying Queue (Forwarding Class) Membership in the Priority Group

### Purpose

Verify that the forwarding class set be-pg includes the forwarding class best-effort.

## Action

Display the be-pg forwarding class set configuration using the operational mode command `show configuration class-of-service forwarding-class-sets be-pg`:

```
user@switch> show configuration class-of-service forwarding-class-sets be-pg
class best-effort;
```

## Verifying the Egress Interface Configuration

### Purpose

Verify that the forwarding class set be-pg and the traffic control profile be-tcp are attached to egress interface xe-0/0/7.

## Action

Display the egress interface using the operational mode command `show configuration class-of-service interfaces xe-0/0/7`:

```
user@switch> show configuration class-of-service interfaces xe-0/0/7
forwarding-class-set {
    be-pg {
        output-traffic-control-profile be-tcp;
```

```
}
}
```

## RELATED DOCUMENTATION

*Example: Configuring CoS Hierarchical Port Scheduling (ETS)*

*Example: Configuring Queue Schedulers*

*Example: Configuring Traffic Control Profiles (Priority Group Scheduling)*

*Example: Configuring Queue Scheduling Priority*

*Example: Configuring Forwarding Class Sets*

*Understanding CoS Traffic Control Profiles*

*Understanding CoS Hierarchical Port Scheduling (ETS)*

## Understanding CoS Priority Group Shaping and Queue Shaping (Maximum Bandwidth)

### IN THIS SECTION

- [Priority Group Shaping | 171](#)
- [Queue Shaping | 171](#)
- [Shaping Maximum Bandwidth Using Hierarchical Scheduling | 172](#)

If the amount of traffic on an interface exceeds the maximum bandwidth available on the interface, it leads to congestion. You can use priority group (forwarding class set) shaping and queue (forwarding class) shaping to manage traffic and avoid congestion.

Configuring a maximum bandwidth sets the most bandwidth a priority group or a queue can use after all of the priority group and queue minimum bandwidth requirements are met, even if more bandwidth is available.



## Priority Group Shaping

Priority group shaping enables you to shape the aggregate traffic of a forwarding class set on a port to a maximum rate that is less than the line or port rate. The maximum bandwidth (*shaping-rate*) that you configure for a priority group sets the maximum bandwidth available to all of the forwarding classes (queues) in the forwarding class set.

If a port has more than one priority group and the combined *shaping-rate* value of the priority groups is greater than the amount of port bandwidth available, the bandwidth is shared proportionally among the priority groups.

You configure the priority group *shaping-rate* in the traffic control profile.

## Queue Shaping

Queue shaping throttles the rate at which queues transmit packets. For example, using queue shaping, you can rate-limit a strict-high priority queue so that the strict-priority queue does not lock out (or starve) low-priority queues.



**NOTE:** We recommend that you always apply a shaping rate to strict-high priority queues to prevent them from starving other queues. If you do not apply a shaping rate to limit the amount of bandwidth a strict-high priority queue can use, then the strict-high priority queue can use all of the available port bandwidth and starve other queues on the port.

Similarly, for any queue, you can configure queue shaping (*shaping-rate*) to set the maximum bandwidth for a particular queue.

The *shaping-rate* value of the priority group sets the aggregate maximum amount of bandwidth available to the queues that belong to the priority group. On a port, the cumulative total bandwidth the queues consume cannot exceed the maximum bandwidth of the priority group to which they belong.

If a priority group has more than one queue, and the combined *shaping-rate* of the queues is greater than the amount of bandwidth available to the priority group, the bandwidth is shared proportionally among the queues.

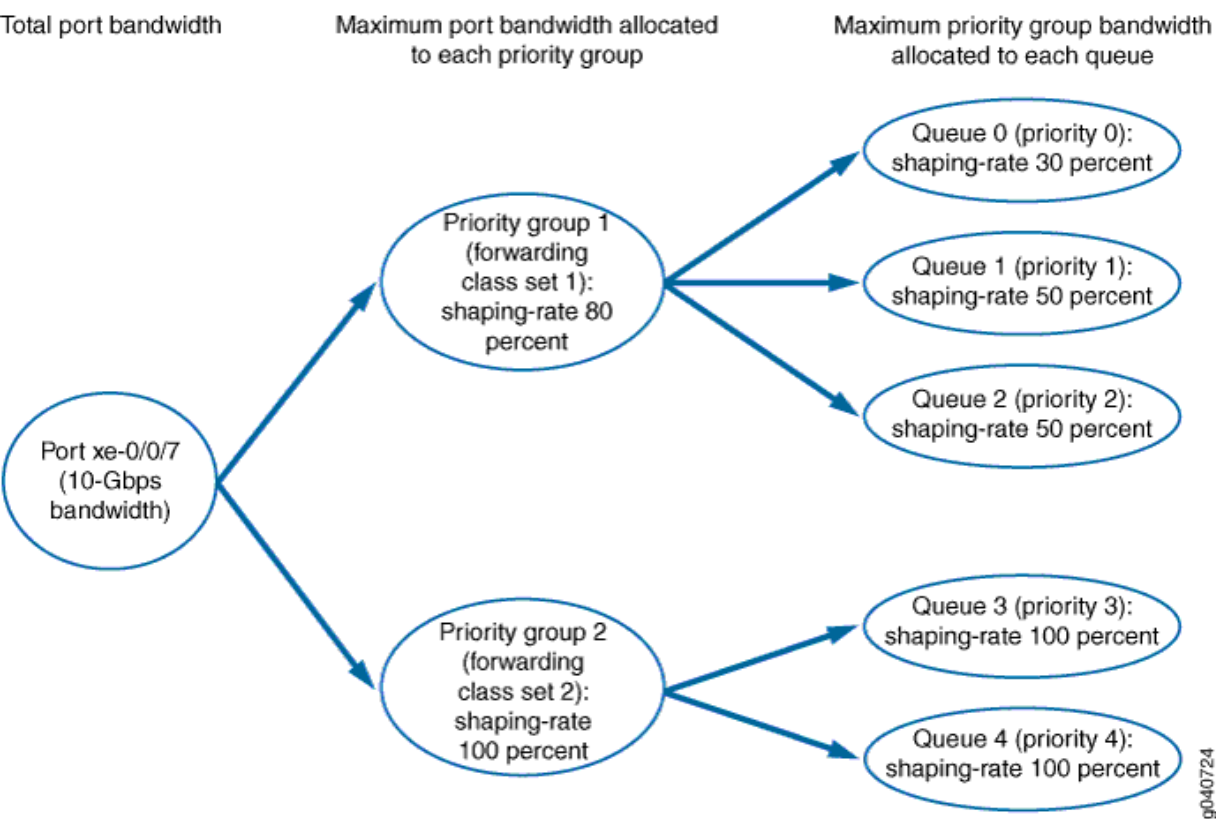
You configure the queue *shaping-rate* in the scheduler configuration, and you set the *shaping-rate* for priority groups in the traffic control profile configuration.

## Shaping Maximum Bandwidth Using Hierarchical Scheduling

Priority group shaping defines the maximum bandwidth allocated to a forwarding class set on a port, whereas queue shaping defines a limit on maximum bandwidth usage per queue. The queue bandwidth is a portion of the priority group bandwidth.

Figure 6 on page 172 shows how the port bandwidth is allocated to priority groups (forwarding class sets) based on the shaping rate of each priority group, and how the bandwidth of each priority group is allocated to the queues in the priority group based on the shaping rate of each queue.

Figure 6: Setting Maximum Bandwidth Using Hierarchical Scheduling



### RELATED DOCUMENTATION

*Understanding CoS Output Queue Schedulers*

*Understanding CoS Traffic Control Profiles*

*Example: Configuring CoS Hierarchical Port Scheduling (ETS)*

*Example: Configuring Queue Schedulers*

*Example: Configuring Traffic Control Profiles (Priority Group Scheduling)*

*Defining CoS Queue Schedulers*

*Defining CoS Traffic Control Profiles (Priority Group Scheduling)*

## Example: Configuring Maximum Output Bandwidth

### IN THIS SECTION

- [Requirements | 174](#)
- [Overview | 175](#)
- [Verification | 176](#)

Scheduling the maximum output bandwidth for a queue (forwarding class) requires configuring both tiers of the hierarchical scheduler. One tier is scheduling the resources for the individual queue. The other tier is scheduling the resources for the priority group (forwarding class set) to which the queue belongs. You can use priority group and queue shaping to prevent traffic from using more bandwidth than you want the traffic to receive.

## Configuring Maximum Bandwidth

### CLI Quick Configuration

To quickly configure the maximum bandwidth for a priority group and a queue, copy the following commands, paste them in a text file, remove line breaks, change variables and details to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level:

```
[edit class-of-service]
set schedulers be-sched shaping-rate percent 4g
set traffic-control-profiles be-tcp shaping-rate 6g
set scheduler-maps be-map forwarding-class best-effort scheduler be-sched
set traffic-control-profiles be-tcp scheduler-map be-map
set forwarding-class-sets be-pg class best-effort
set interfaces xe-0/0/7 forwarding-class-set be-pg output-traffic-control-profile be-tcp
```

## Step-by-Step Procedure

To configure the maximum bandwidth hierarchical scheduling for a queue and a priority group:

1. Configure the maximum queue bandwidth of 4 Gbps for scheduler `be-sched`:

```
[edit class-of-service schedulers]
user@switch# set be-sched shaping-rate 4g
```

2. Configure the maximum priority group bandwidth of 6 Gbps for traffic control profile `be-tcp`:

```
[edit class-of-service traffic-control-profiles]
user@switch# set be-tcp shaping-rate 6g
```

3. Associate the scheduler `be-sched` with the best-effort queue in the scheduler map `be-map`:

```
[edit class-of-service scheduler-maps]
user@switch# set be-map forwarding-class best-effort scheduler be-sched
```

4. Associate the scheduler map with the traffic control profile:

```
[edit class-of-service traffic-control-profiles]
user@switch# set be-tcp scheduler-map be-map
```

5. Assign the best-effort queue to the priority group `be-pg`:

```
[edit class-of-service forwarding-class-sets]
user@switch# set be-pg class best-effort
```

6. Apply the configuration to interface `xe-0/0/7`:

```
[edit class-of-service interfaces]
user@switch# set xe-0/0/7 forwarding-class-set be-pg output-traffic-control-profile be-tcp
```

## Requirements

This example uses the following hardware and software components:

- One QFX switch
- Any supported Junos release

## Overview

The priority group maximum bandwidth defines the maximum total amount of bandwidth available for all of the queues in the priority group.

The `shaping-rate` setting in the scheduler configuration determines the maximum bandwidth for an individual queue.

The `shaping-rate` setting in the traffic control profile configuration determines the maximum bandwidth for a priority group.



**NOTE:** When you configure bandwidth for a queue or a priority group, the switch considers only the data as the configured bandwidth. The switch does not account for the bandwidth consumed by the preamble and the interframe gap (IFG). Therefore, when you calculate and configure the bandwidth requirements for a queue or for a priority group, consider the preamble and the IFG as well as the data in the calculations.



**NOTE:** When you set the maximum bandwidth (`shaping-rate`) for a queue or for a priority group at 100 Kbps or less, the traffic shaping behavior is accurate only within  $\pm 20$  percent of the configured `shaping-rate` value.

This example describes how to:

- Configure a maximum rate of 4 Gbps for queues in a scheduler named `be-sched`.
- Configure a maximum rate of 6 Gbps for a priority group in a traffic control profile named `be-tcp`.
- Assign the scheduler to a queue named `best-effort` by using a scheduler map named `be-map`.
- Associate the scheduler map `be-map` with the traffic control profile `be-tcp`.
- Assign the queue `best-effort` to a priority group named `be-pg`.
- Assign the priority group and the bandwidth scheduling to the interface `xe-0/0/7`.

Table 57 on page 176 shows the configuration components for this example:

**Table 57: Components of the Maximum Output Bandwidth Configuration Example**

Component	Settings
Hardware	QFX switch
Maximum queue bandwidth	Shaping rate: 4g
Maximum priority group bandwidth	Shaping rate: 6g
Scheduler	be-sched
Scheduler map	be-map
Traffic control profile	be-tcp
Forwarding class set (priority group)	be-pg
Queue (forwarding class)	best-effort
Egress interface	xe-0/0/7

## Verification

### IN THIS SECTION

- [Verifying the Maximum Queue Bandwidth | 177](#)
- [Verifying the Priority Group Maximum Bandwidth and Scheduler Map Association | 177](#)
- [Verifying the Scheduler Map Configuration | 178](#)
- [Verifying Queue \(Forwarding Class\) Membership in the Priority Group | 178](#)
- [Verifying the Egress Interface Configuration | 178](#)

To verify the maximum output bandwidth configuration, perform these tasks:

## Verifying the Maximum Queue Bandwidth

### Purpose

Verify that you configured the maximum queue bandwidth as 4g in the scheduler be-sched.

### Action

List the maximum bandwidth in the be-sched scheduler configuration using the operational mode command `show configuration class-of-service schedulers be-sched shaping-rate`:

```
user@switch> show configuration class-of-service schedulers be-sched shaping-rate
4g;
```

## Verifying the Priority Group Maximum Bandwidth and Scheduler Map Association

### Purpose

Verify that the maximum priority group bandwidth is 6g and the attached scheduler map is be-map in the traffic control profile be-tcp.

### Action

List the maximum bandwidth in the be-tcp traffic control profile configuration using the operational mode command `show configuration class-of-service traffic-control-profiles be-tcp shaping-rate`:

```
user@switch> show configuration class-of-service traffic-control-profiles be-tcp shaping-rate
6g;
```

List the scheduler map in the be-tcp traffic control profile configuration using the operational mode command `show configuration class-of-service traffic-control-profiles be-tcp scheduler-map`:

```
user@switch> show configuration class-of-service traffic-control-profiles be-tcp scheduler-map
scheduler-map be-map;
```

## Verifying the Scheduler Map Configuration

### Purpose

Verify that the scheduler map `be-map` maps the forwarding class `best-effort` to the scheduler `be-sched`.

### Action

List the `be-map` scheduler map configuration using the operational mode command `show configuration class-of-service schedulers maps be-map`:

```
user@switch> show configuration class-of-service scheduler-maps be-map
forwarding-class best-effort scheduler be-sched;
```

## Verifying Queue (Forwarding Class) Membership in the Priority Group

### Purpose

Verify that the forwarding class set `be-pg` includes the forwarding class `best-effort`.

### Action

List the `be-pg` forwarding class set configuration using the operational mode command `show configuration class-of-service forwarding-class-sets be-pg`:

```
user@switch> show configuration class-of-service forwarding-class-sets be-pg
class best-effort;
```

## Verifying the Egress Interface Configuration

### Purpose

Verify that the forwarding class set `be-pg` and the traffic control profile `be-tcp` are attached to egress interface `xe-0/0/7`.



## Action

List the egress interface using the operational mode command `show configuration class-of-service interfaces xe-0/0/7`:

```
user@switch> show configuration class-of-service interfaces xe-0/0/7
forwarding-class-set {
    be-pg {
        output-traffic-control-profile be-tcp;
    }
}
```

## RELATED DOCUMENTATION

*Example: Configuring CoS Hierarchical Port Scheduling (ETS)*

*Example: Configuring Queue Schedulers*

*Example: Configuring Traffic Control Profiles (Priority Group Scheduling)*

*Example: Configuring Forwarding Class Sets*

*Understanding CoS Traffic Control Profiles*

*Understanding CoS Hierarchical Port Scheduling (ETS)*

## Understanding CoS Explicit Congestion Notification

### IN THIS SECTION

- [How ECN Works | 180](#)
- [WRED Drop Profile Control of ECN Thresholds | 186](#)
- [Support, Limitations, and Notes | 188](#)

Explicit congestion notification (ECN) enables end-to-end congestion notification between two endpoints on TCP/IP based networks. The two endpoints are an ECN-enabled sender and an ECN-enabled receiver. ECN must be enabled on both endpoints. However, in the case of an unsupported

peer, a NFX device that supports ECN bootstraps the incoming packets from the unsupported peer and marks the packets to signal network congestion when it occurs.

ECN notifies networks about congestion with the goal of reducing packet loss and delay by making the sending device decrease the transmission rate until the congestion clears, without dropping packets. RFC 3168, *The Addition of Explicit Congestion Notification (ECN) to IP*, defines ECN.

ECN is disabled by default. Normally, you enable ECN only on queues that handle best-effort traffic because other traffic types use different methods of congestion notification—lossless traffic uses priority-based flow control (PFC) and strict-high priority traffic receives all of the port bandwidth it requires up to the point of a configured maximum rate.

You enable ECN on individual output queues (as represented by forwarding classes) by enabling ECN in the queue scheduler configuration, mapping the scheduler to forwarding classes (queues), and then applying the scheduler to interfaces.



**NOTE:** For ECN to work on a queue, you must also apply a weighted random early detection (WRED) packet drop profile to the queue.

## How ECN Works

Without ECN, devices respond to network congestion by dropping TCP/IP packets. Dropped packets signal the network that congestion is occurring. Devices on the IP network respond to TCP packet drops by reducing the packet transmission rate to allow the congestion to clear. However, the packet drop method of congestion notification and management has some disadvantages. For example, packets are dropped and must be retransmitted. Also, bursty traffic can cause the network to reduce the transmission rate too much, resulting in inefficient bandwidth utilization.

Instead of dropping packets to signal network congestion, ECN marks packets to signal network congestion, without dropping the packets. For ECN to work, all of the devices in the path between two ECN-enabled endpoints must have ECN enabled. ECN is negotiated during the establishment of the TCP connection between the endpoints.

ECN-enabled devices determine the queue congestion state based on the WRED packet drop profile configuration applied to the queue, so each ECN-enabled queue must also have a WRED drop profile. If a queue fills to the level at which the WRED drop profile has a packet drop probability greater than zero (0), the device marks the packet as experiencing congestion. Whether or not a device marks a packet as experiencing congestion is the same probability as the drop probability of the queue at that fill level.

ECN communicates whether or not congestion is experienced by marking the two least-significant bits in the differentiated services (DiffServ) field in the IP header. The most significant six bits in the DiffServ field contain the Differentiated Services Code Point (DSCP) bits. The state of the two ECN bits signals whether or not the packet is an ECN-capable packet and whether or not congestion has been experienced.

ECN-capable senders mark packets as ECN-capable. If a sender is not ECN-capable, it marks packets as not ECN-capable. If an ECN-capable packet experiences congestion at the egress queue of a device, then the device marks the packet as experiencing congestion. When the packet reaches the ECN-capable receiver (destination endpoint), the receiver echoes the congestion indicator to the sender (source endpoint) by sending a packet marked to indicate congestion.

After receiving the congestion indicator from the receiver, the source endpoint reduces the transmission rate to relieve the congestion. This is similar to the result of TCP congestion notification and management, but instead of dropping the packet to signal network congestion, ECN marks the packet and the receiver echoes the congestion notification to the sender. Because the packet is not dropped, the packet does not need to be retransmitted.

### ECN Bits in the DiffServ Field

The two ECN bits in the DiffServ field provide four codes that determine if a packet is marked as an ECN-capable transport (ECT) packet, meaning that both endpoints of the transport protocol are ECN-capable, and if there is congestion experienced (CE), as shown in [Table 58 on page 181](#):

**Table 58: ECN Bit Codes**

ECN Bits (Code)	Meaning
00	Non-ECT—Packet is marked as not ECN-capable
01	ECT(1)—Endpoints of the transport protocol are ECN-capable
10	ECT(0)—Endpoints of the transport protocol are ECN-capable
11	CE—Congestion experienced

Codes 01 and 10 have the same meaning: the sending and receiving endpoints of the transport protocol are ECN-capable. There is no difference between these codes.

### End-to-End ECN Behavior

After the sending and receiving endpoints negotiate ECN, the sending endpoint marks packets as ECN-capable by setting the DiffServ ECN field to ECT(1) (01) or ECT(0) (10).

When a packet traverses a device and experiences congestion at an output queue that uses the WRED packet drop mechanism, the device marks the packet as experiencing congestion by setting the DiffServ

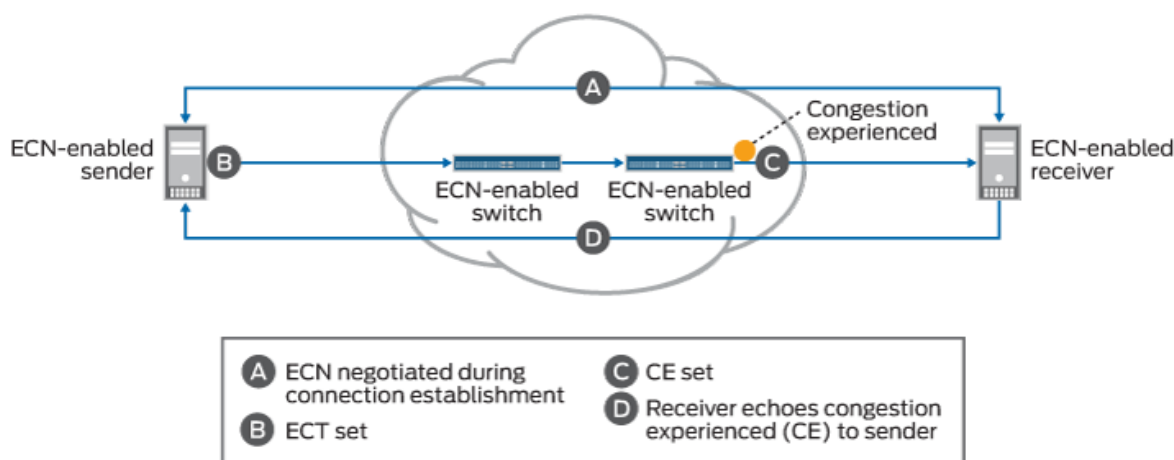
ECN field to CE (11). Instead of dropping the packet (as with TCP congestion notification), the device forwards the packet.



**NOTE:** At the egress queue, the WRED algorithm determines whether or not a packet is drop eligible based on the queue fill level (how full the queue is). If a packet is drop eligible and marked as ECN-capable, the packet can be marked CE and forwarded. If a packet is drop eligible and is not marked as ECN-capable, it is dropped. See ["WRED Drop Profile Control of ECN Thresholds" on page 186](#) for more information about the WRED algorithm.

When the packet reaches the receiver endpoint, the CE mark tells the receiver that there is network congestion. The receiver then sends (echoes) a message to the sender that indicates there is congestion on the network. The sender acknowledges the congestion notification message and reduces its transmission rate. [Figure 7 on page 182](#) summarizes how ECN works to mitigate network congestion:

**Figure 7: Explicit Congestion Notification**



End-to-end ECN behavior includes:

1. The ECN-capable sender and receiver negotiate ECN capability during the establishment of their connection.



**NOTE:** If the client is not ECN capable, then the NFX device negotiates ECN on behalf of client

during the connection establishment. The NFX device sets the ECE and CWR bits in the TCP

header of the SYN packet.

2. After successful negotiation of ECN capability, the ECN-capable sender sends IP packets with the ECT field set to the receiver.
3. If the WRED algorithm on a device egress queue determines that the queue is experiencing congestion and the packet is drop eligible, the device can mark the packet as “congestion experienced” (CE) to indicate to the receiver that there is congestion on the network. If the packet has already been marked CE (congestion has already been experienced at the egress of another device), then the device forwards the packet with CE marked.

If there is no congestion at the device egress queue, then the device forwards the packet and does not change the ECT-enabled marking of the ECN bits, so the packet is still marked as ECN-capable but not as experiencing congestion.

4. The receiver receives a packet marked CE to indicate that congestion was experienced along the congestion path.
5. The receiver echoes (sends) a packet back to the sender with the ECE bit (bit 9) marked in the flag field of the TCP header. The ECE bit is the ECN echo flag bit, which notifies the sender that there is congestion on the network.
6. The sender reduces the data transmission rate and sends a packet to the receiver with the CWR bit (bit 8) marked in the flag field of the TCP header. The CWR bit is the congestion window reduced flag bit, which acknowledges to the receiver that the congestion experienced notification was received.
7. When the receiver receives the CWR flag, the receiver stops setting the ECE bit in replies to the sender.

[Table 59 on page 184](#) summarizes the behavior of traffic on ECN-enabled queues.

**Table 59: Traffic Behavior on ECN-Enabled Queues**

Incoming IP Packet Marking of ECN Bits	ECN Configuration on the Output Queue	Action if WRED Algorithm Determines Packet is Drop Eligible	Outgoing Packet Marking of ECN Bits	Log Format
Non-ECT (00) SYN	WRED enabled—both scenarios where threshold is crossed and within the threshold limit	Bootstrap to provide ECN support	Set ECE and CWR in TCP header and ECT in IP header	ECT-BIT: 00 WRED-MET: true
Non-ECT (00) Data	WRED enabled	Do not drop. Mark ECN bit to 01/10.	Packet marked ECT 01/10	Not applicable
Non-ECT (00) Data	WRED enabled—threshold met	Do not drop. Mark ECN bit 11.	Packet marked ECT (CE)	ECT-BIT: 00 WRED-MET: true
Non-ECT (00)	WRED disabled	No change	No change	Not applicable
ECT (10 or 01)	WRED enabled	No change	No change	Not applicable
ECT (10 or 01)	WRED enabled—threshold met	Do no drop. Mark ECN bit to 11 and drop according to drop profile.	Packet marked ECT (CE)	ECT-BIT: 10 WRED-MET: true
ECT(10 or 01)	WRED disabled	No change	No change	Not applicable
ECT(11)	WRED enabled	Do not drop. As packet is already marked with CE, send the packet without any change	Packet marked ECT (11) to indicate congestion	ECT-BIT: 11 WRED-MET: false
ECT (11)	WRED disabled	Drop packet	Drop packet	Not applicable

**Table 59: Traffic Behavior on ECN-Enabled Queues (Continued)**

Incoming IP Packet Marking of ECN Bits	ECN Configuration on the Output Queue	Action if WRED Algorithm Determines Packet is Drop Eligible	Outgoing Packet Marking of ECN Bits	Log Format
ECT (11)	WRED enabled—threshold met	Do not drop. Packet is already marked as experiencing congestion, forward the packet without changing the ECN marking.	Packet marked ECT (11) to indicate congestion	ECT-BIT: 11  WRED-MET: true

When an output queue is not experiencing congestion as defined by the WRED drop profile mapped to the queue, all packets are forwarded, and no packets are dropped.

## ECN Compared to PFC and Ethernet PAUSE

ECN is an end-to-end network congestion notification mechanism for IP traffic. Priority-based flow control (PFC) (IEEE 802.1Qbb) and Ethernet PAUSE (IEEE 802.3X) are different types of congestion management mechanisms.

ECN requires that an output queue must also have an associated WRED packet drop profile. Output queues used for traffic on which PFC is enabled should not have an associated WRED drop profile. Interfaces on which Ethernet PAUSE is enabled should not have an associated WRED drop profile.

PFC is a peer-to-peer flow control mechanism to support lossless traffic. PFC enables connected peer devices to pause flow transmission during periods of congestion. PFC enables you to pause traffic on a specified type of flow on a link instead of on all traffic on a link. For example, you can (and should) enable PFC on lossless traffic classes such as the `fcoe` forwarding class. Ethernet PAUSE is also a peer-to-peer flow control mechanism, but instead of pausing only specified traffic flows, Ethernet PAUSE pauses all traffic on a physical link.

With PFC and Ethernet PAUSE, the sending and receiving endpoints of a flow do not communicate congestion information to each other across the intermediate devices. Instead, PFC controls flows between two PFC-enabled peer devices that support data center bridging (DCB) standards. PFC works by sending a pause message to the connected peer when the flow output queue becomes congested. Ethernet PAUSE simply pauses all traffic on a link during periods of congestion and does not require DCB.

## WRED Drop Profile Control of ECN Thresholds

You apply WRED drop profiles to forwarding classes (which are mapped to output queues) to control how the device marks ECN-capable packets. A scheduler map associates a drop profile with a scheduler and a forwarding class, and then you apply the scheduler map to interfaces to implement the scheduling properties for the forwarding class on those interfaces.

Drop profiles define queue fill level (the percentage of queue fullness) and drop probability (the percentage probability that a packet is dropped) pairs. When a queue fills to a specified level, traffic that matches the drop profile has the drop probability paired with that fill level. When you configure a drop profile, you configure pairs of fill levels and drop probabilities to control how packets drop at different levels of queue fullness.

The first fill level and drop probability pair is the drop start point. Until the queue reaches the first fill level, packets are not dropped. When the queue reaches the first fill level, packets that exceed the fill level have a probability of being dropped that equals the drop probability paired with the fill level.

The last fill level and drop probability pair is the drop end point. When the queue reaches the last fill level, all packets are dropped unless they are configured for ECN.



**NOTE:** Lossless queues (forwarding class configured with the `no-loss` packet drop attribute) and strict-high priority queues do not use drop profiles. Lossless queues use PFC to control the flow of traffic.

The drop profile configuration affects ECN packets as follows:

- Drop start point—ECN-capable packets might be marked as congestion experienced (CE).
- Drop end point—ECN-capable packets are always marked CE.

As a queue fills from the drop start point to the drop end point, the probability that an ECN packet is marked CE is the same as the probability that a non-ECN packet is dropped if you apply the drop profile to best-effort traffic. As the queue fills, the probability of an ECN packet being marked CE increases, just as the probability of a non-ECN packet being dropped increases when you apply the drop profile to best-effort traffic.

At the drop end point, all ECN packets are marked CE, but the ECN packets are not dropped. When the queue fill level exceeds the drop end point, all ECN packets are marked CE. ECN packets (and all other packets) are tail-dropped if the queue fills completely.

To configure a WRED packet drop profile and apply it to an output queue (using hierarchical scheduling on devices that support ETS):

1. Configure a drop profile using the statement `set class-of-service drop-profiles profile-name interpolate fill-level drop-start-point fill-level drop-end-point drop-probability 0 drop-probability percentage`.



2. Map the drop profile to a queue scheduler using the statement `set class-of-service schedulers scheduler-name drop-profile-map loss-priority (low | medium-high | high) protocol any drop-profile profile-name`. The name of the drop-profile is the name of the WRED profile configured in Step 1.
3. Map the scheduler, which Step 2 associates with the drop profile, to the output queue using the statement `set class-of-service scheduler-maps map-name forwarding-class forwarding-class-name scheduler scheduler-name`. The forwarding class identifies the output queue. Forwarding classes are mapped to output queues by default, and can be remapped to different queues by explicit user configuration. The scheduler name is the scheduler configured in Step 2.
4. Associate the scheduler map with a traffic control profile using the statement `set class-of-service traffic-control-profiles tcp-name scheduler-map map-name`. The scheduler map name is the name configured in Step 3.
5. Associate the traffic control profile with an interface using the statement `set class-of-service interface interface-name forwarding-class-set forwarding-class-set-name output-traffic-control-profile tcp-name`. The output traffic control profile name is the name of the traffic control profile configured in Step 4.

The interface uses the scheduler map in the traffic control profile to apply the drop profile (and other attributes, including the enable ECN attribute) to the output queue (forwarding class) on that interface. Because you can use different traffic control profiles to map different schedulers to different interfaces, the same queue number on different interfaces can handle traffic in different ways.

You can configure a WRED packet drop profile and apply it to an output queue on devices that support port scheduling (ETS hierarchical scheduling is either not supported or not used). To configure a WRED packet drop profile and apply it to an output queue on devices that support port scheduling (ETS hierarchical scheduling is either not supported or not used):

1. Configure a drop profile using the statement `set class-of-service drop-profiles profile-name interpolate fill-level level1 level2 ... level32 drop-probability probability1 probability2 ... probability32`. You can specify as few as two fill level/drop probability pairs or as many as 32 pairs.
2. Map the drop profile to a queue scheduler using the statement `set class-of-service schedulers scheduler-name drop-profile-map loss-priority (low | medium-high | high) drop-profile profile-name`. The name of the drop-profile is the name of the WRED profile configured in Step 1.
3. Map the scheduler, which Step 2 associates with the drop profile, to the output queue using the statement `set class-of-service scheduler-maps map-name forwarding-class forwarding-class-name scheduler scheduler-name`. The forwarding class identifies the output queue. Forwarding classes are mapped to output queues by default, and can be remapped to different queues by explicit user configuration. The scheduler name is the scheduler configured in Step 2.
4. Associate the scheduler map with an interface using the statement `set class-of-service interfaces interface-name scheduler-map scheduler-map-name`.

The interface uses the scheduler map to apply the drop profile (and other attributes) to the output queue mapped to the forwarding class on that interface. Because you can use different scheduler maps on different interfaces, the same queue number on different interfaces can handle traffic in different ways.

## Support, Limitations, and Notes

If the WRED algorithm that is mapped to a queue does not find a packet drop eligible, then the ECN configuration and ECN bits marking does not matter. The packet transport behavior is the same as when ECN is not enabled.

ECN is disabled by default. Normally, you enable ECN only on queues that handle best-effort traffic, and you do not enable ECN on queues that handle lossless traffic or strict-high priority traffic.

ECN supports the following:

- IPv4 and IPv6 packets
- Untagged, single-tagged, and double-tagged packets
- The outer IP header of IP tunneled packets (but not the inner IP header)

ECN does not support the following:

- IP packets with MPLS encapsulation
- The inner IP header of IP tunneled packets (however, ECN works on the outer IP header)
- Multicast, broadcast, and destination lookup fail (DLF) traffic
- Non-IP traffic

# 4

PART

## Configuration Statements and Operational Commands

---

- [Junos CLI Reference Overview | 190](#)
-

# Junos CLI Reference Overview

We've consolidated all Junos CLI commands and configuration statements in one place. Read this guide to learn about the syntax and options that make up the statements and commands. Also understand the contexts in which you'll use these CLI elements in your network configurations and operations.

- [Junos CLI Reference](#)

Click the links to access Junos OS and Junos OS Evolved configuration statement and command summary topics.

- [Configuration Statements](#)
- [Operational Commands](#)