Juniper Networks | Engineering Simplicity

Traffic Management User Guide (QFX Series and EX4600 Switches)

Published
2020-09-23

Junos

Juniper Networks, Inc.
1133 Innovation Way
Sunnyvale, California 94089
USA
408-745-2000
www.juniper.net

Traffic Management User Guide (QFX Series and EX4600 Switches)

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

2

Weighted Random Early Detection (WRED) and Explicit Congestion Notification (ECN)

3

CoS Queue Schedulers, Traffic Control Profiles, and Hierarchical Port Scheduling (ETS)

About the Documentation

IN THIS SECTION

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. This guide applies to all QFX Series and the EX4600 line of switches.

Documentation and Release Notes

To obtain the most current version of all Juniper Networks® technical documentation, see the product documentation page on the Juniper Networks website at https://www.juniper.net/documentation/.

If the information in the latest release notes differs from the information in the documentation, follow the product Release Notes.

Juniper Networks Books publishes books by Juniper Networks engineers and subject matter experts. These books go beyond the technical documentation to explore the nuances of network architecture, deployment, and administration. The current list can be viewed at https://www.juniper.net/books.

Using the Examples in This Manual

If you want to use the examples in this manual, you can use the **load merge** or the **load merge relative** command. These commands cause the software to merge the incoming configuration into the current candidate configuration. The example does not become active until you commit the candidate configuration.

If the example configuration contains the top level of the hierarchy (or multiple hierarchies), the example is a *full example*. In this case, use the **load merge** command.

If the example configuration does not start at the top level of the hierarchy, the example is a *snippet*. In this case, use the **load merge relative** command. These procedures are described in the following sections.

Merging a Full Example

To merge a full example, follow these steps:

1. From the HTML or PDF version of the manual, copy a configuration example into a text file, save the file with a name, and copy the file to a directory on your routing platform.

   For example, copy the following configuration to a file and name the file **ex-script.conf**. Copy the **ex-script.conf** file to the **/var/tmp** directory on your routing platform.

   ```
   system {
       scripts {
           commit {
               file ex-script.xsl;
           }
       }
   }
   interfaces {
       fxp0 {
           disable;
           unit 0 {
               family inet {
                   address 10.0.0.1/24;
               }
           }
       }
   }
   ```

2. Merge the contents of the file into your routing platform configuration by issuing the **load merge** configuration mode command:

   ```
   [edit]
   user@host# load merge /var/tmp/ex-script.conf
   load complete
   ```

Merging a Snippet

To merge a snippet, follow these steps:

1. From the HTML or PDF version of the manual, copy a configuration snippet into a text file, save the file with a name, and copy the file to a directory on your routing platform.

   For example, copy the following snippet to a file and name the file **ex-script-snippet.conf**. Copy the **ex-script-snippet.conf** file to the **/var/tmp** directory on your routing platform.

   ```
   commit {
     file ex-script-snippet.xsl; }
   ```

2. Move to the hierarchy level that is relevant for this snippet by issuing the following configuration mode command:

   ```
   [edit]
   user@host# edit system scripts
   [edit system scripts]
   ```

3. Merge the contents of the file into your routing platform configuration by issuing the **load merge relative** configuration mode command:

   ```
   [edit system scripts]
   user@host# load merge relative /var/tmp/ex-script-snippet.conf
   load complete
   ```

For more information about the **load** command, see CLI Explorer.

Documentation Conventions

Table 1 on page xxii defines notice icons used in this guide.

Table 1: Notice Icons

Icon	Meaning	Description
	Informational note	Indicates important features or instructions.
	Caution	Indicates a situation that might result in loss of data or hardware damage.
	Warning	Alerts you to the risk of personal injury or death.
	Laser warning	Alerts you to the risk of personal injury from a laser.
	Tip	Indicates helpful information.
	Best practice	Alerts you to a recommended use or implementation.

Table 2 on page xxii defines the text and syntax conventions used in this guide.

Table 2: Text and Syntax Conventions

Convention	Description	Examples
Bold text like this	Represents text that you type.	To enter configuration mode, type the **configure** command: user@host> **configure**
`Fixed-width text like this`	Represents output that appears on the terminal screen.	`user@host>` **show chassis alarms** `No alarms currently active`
Italic text like this	• Introduces or emphasizes important new terms. • Identifies guide names. • Identifies RFC and Internet draft titles.	• A policy *term* is a named structure that defines match conditions and actions. • *Junos OS CLI User Guide* • RFC 1997, *BGP Communities Attribute*

Table 2: Text and Syntax Conventions *(continued)*

<table>
<tr><th>Convention</th><th>Description</th><th>Examples</th></tr>
<tr><td>Italic text like this</td><td>Represents variables (options for which you substitute a value) in commands or configuration statements.</td><td>Configure the machine’s domain name:

[edit]
root@# set system domain-name
domain-name</td></tr>
<tr><td>Text like this</td><td>Represents names of configuration statements, commands, files, and directories; configuration hierarchy levels; or labels on routing platform components.</td><td>• To configure a stub area, include the stub statement at the [edit protocols ospf area area-id] hierarchy level.
• The console port is labeled CONSOLE.</td></tr>
<tr><td>< > (angle brackets)</td><td>Encloses optional keywords or variables.</td><td>stub <default-metric <i>metric</i>>;</td></tr>
<tr><td>| (pipe symbol)</td><td>Indicates a choice between the mutually exclusive keywords or variables on either side of the symbol. The set of choices is often enclosed in parentheses for clarity.</td><td>broadcast | multicast

(string1 | string2 | string3)</td></tr>
<tr><td># (pound sign)</td><td>Indicates a comment specified on the same line as the configuration statement to which it applies.</td><td>rsvp { # Required for dynamic MPLS only</td></tr>
<tr><td>[ ] (square brackets)</td><td>Encloses a variable for which you can substitute one or more values.</td><td>community name members [community-ids]</td></tr>
<tr><td>Indention and braces ({ })</td><td>Identifies a level in the configuration hierarchy.</td><td rowspan="2">[edit]
routing-options {
  static {
    route default {
      nexthop address;
      retain;
    }
  }
}</td></tr>
<tr><td>; (semicolon)</td><td>Identifies a leaf statement at a configuration hierarchy level.</td></tr>
<tr><td colspan="3">GUI Conventions</td></tr>
</table>

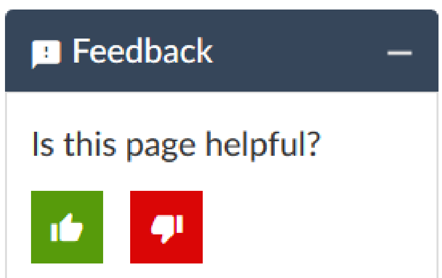
Table 2: Text and Syntax Conventions *(continued)*

Convention	Description	Examples
Bold text like this	Represents graphical user interface (GUI) items you click or select.	• In the Logical Interfaces box, select **All Interfaces**. • To cancel the configuration, click **Cancel**.
> (bold right angle bracket)	Separates levels in a hierarchy of menu selections.	In the configuration editor hierarchy, select **Protocols>Ospf**.

Documentation Feedback

We encourage you to provide feedback so that we can improve our documentation. You can use either of the following methods:

- Online feedback system—Click TechLibrary Feedback, on the lower right of any page on the Juniper Networks TechLibrary site, and do one of the following:



  - Click the thumbs-up icon if the information on the page was helpful to you.
  - Click the thumbs-down icon if the information on the page was not helpful to you or if you have suggestions for improvement, and use the pop-up form to provide feedback.

- E-mail—Send your comments to techpubs-comments@juniper.net. Include the document or topic name, URL or page number, and software version (if applicable).

Requesting Technical Support

Technical product support is available through the Juniper Networks Technical Assistance Center (JTAC). If you are a customer with an active Juniper Care or Partner Support Services support contract, or are covered under warranty, and need post-sales technical support, you can access our tools and resources online or open a case with JTAC.

- JTAC policies—For a complete understanding of our JTAC procedures and policies, review the *JTAC User Guide* located at https://www.juniper.net/us/en/local/pdf/resource-guides/7100059-en.pdf.
- Product warranties—For product warranty information, visit https://www.juniper.net/support/warranty/.
- JTAC hours of operation—The JTAC centers have resources available 24 hours a day, 7 days a week, 365 days a year.

Self-Help Online Tools and Resources

For quick and easy problem resolution, Juniper Networks has designed an online self-service portal called the Customer Support Center (CSC) that provides you with the following features:

- Find CSC offerings: https://www.juniper.net/customers/support/
- Search for known bugs: https://prsearch.juniper.net/
- Find product documentation: https://www.juniper.net/documentation/
- Find solutions and answer questions using our Knowledge Base: https://kb.juniper.net/
- Download the latest versions of software and review release notes: https://www.juniper.net/customers/csc/software/
- Search technical bulletins for relevant hardware and software notifications: https://kb.juniper.net/InfoCenter/
- Join and participate in the Juniper Networks Community Forum: https://www.juniper.net/company/communities/
- Create a service request online: https://myjuniper.juniper.net

To verify service entitlement by product serial number, use our Serial Number Entitlement (SNE) Tool: https://entitlementsearch.juniper.net/entitlementsearch/

Creating a Service Request with JTAC

You can create a service request with JTAC on the Web or by telephone.

- Visit https://myjuniper.juniper.net.
- Call 1-888-314-JTAC (1-888-314-5822 toll-free in the USA, Canada, and Mexico).

For international or direct-dial options in countries without toll-free numbers, see https://support.juniper.net/support/requesting-support/.

1

PART

Basic CoS Configuration

CHAPTER 1

CoS Overview

IN THIS CHAPTER

Overview of Junos OS CoS

IN THIS SECTION

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)

> **NOTE:** OCX Series switches and NFX250 Network Services platforms do not support PFC and DCBX.
>
> Juniper Networks QFX10000 switches support both enhanced transmission selection (ETS) hierarchical port scheduling and direct port scheduling.

How Junos OS CoS Works

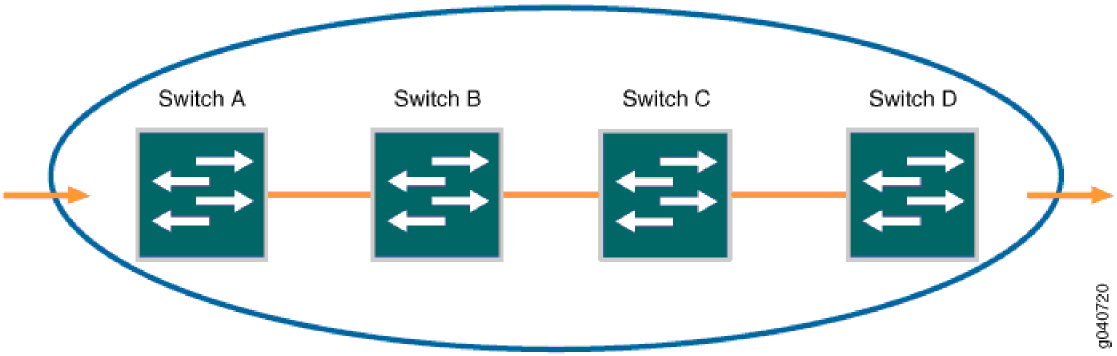
Junos OS CoS works by examining traffic entering the edge of your network. The switch classifies traffic into defined service groups to provide the special treatment of traffic across the network. For example, you can send voice traffic across certain links and data traffic across other links. In addition, 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 5, 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 | 5
Understanding Junos CoS Components | 17
Understanding CoS Packet Flow | 23
Understanding CoS Hierarchical Port Scheduling (ETS) | 407

Overview of Policers

IN THIS SECTION

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.

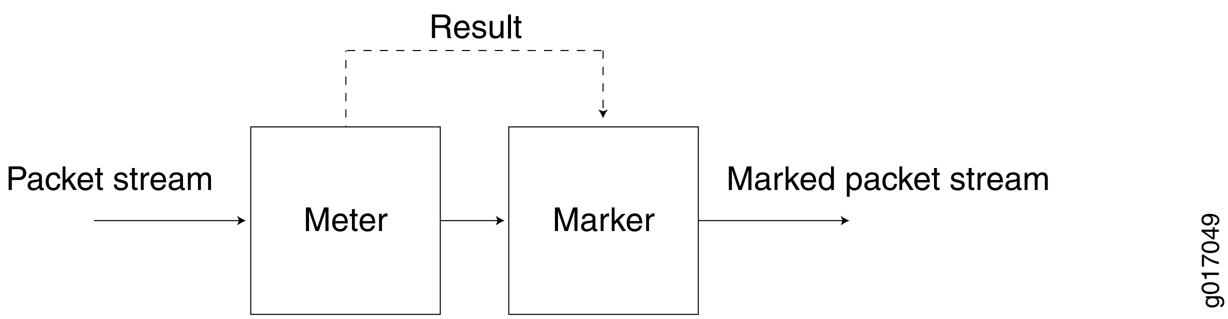
Policing is also an important component of firewall filters. You can achieve policing by including policers in firewall filter configurations.

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 2 on page 6 illustrates this process.

Figure 2: Flow of Tricolor Marking Policer Operation



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 3 on page 8 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 3 on page 8 describes the policer actions.

Table 3: 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 some QFX 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. (This behavior does not occur in QFX10000 switches.)

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 QFX 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. (This does not apply to QFX10000 switches.) 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.

> **NOTE:** In an environment of light bursty traffic, QFX5200 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.

How Many Policers Are Supported?

QFX10000 switches support 8K policers (all policer types). QFX5100 and QFX5200 switches support 1535 ingress policers and 1024 egress policers (assuming one policer per firewall filter term). QFX5110 switches support 6144 ingress policers and 1024 egress policers (assuming one policer per firewall filter term).

QFX3500 and QFX3600 standalone switches and QFabric Node devices support the following numbers of policers (assuming one policer per firewall filter term):

- Two-color policers used in ingress firewall filters: 767
- Three-color policers used in ingress firewall filters: 767
- Two-color policers used in egress firewall filters: 1022
- Three-color policers used in egress firewall filters: 512

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.

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

Configuring CoS

The traffic management class-of-service topics describe how to configure the Junos OS 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 (OCX Series switches and NFX250 Network Services platform do not support lossless transport, PFC, and DCBX).
- 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.

Table 4 on page 13 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 4: CoS Configuration Tasks

CoS Configuration Task	Platforms Supported	Links

Basic CoS Configuration:
 	
• 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
• Configure classifiers and multidestination classifiers
  ◦ 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
  ◦ Change the host default output queue and mapping of DSCP bits used in the type of service (ToS) field
• Configure forwarding classes
• 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
• 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.
• Assign the following CoS components to physical or logical interfaces:
  ◦ Classifiers
  ◦ Congestion notification profiles
  ◦ Forwarding classes
  ◦ Forwarding class sets
  ◦ Output traffic control profiles
  ◦ Port schedulers
  ◦ Rewrite rules
	• QFX3500
	• QFX3600
	• EX4600
	• NFX250
	• QFX5100
	• QFX5200
	• QFX5210
	• QFX10000
	• OCX1100 switches
	• QFabric systems
		• Defining CoS Code-Point Aliases on page 82
		• (QFX10000 only) "Example: Configuring Classifiers" on page 96
		• (Except QFX10000) "Defining CoS BA Classifiers (DSCP, DSCP IPv6, IEEE 802.1p)" on page 94
		• (Except NFX250 and QFX10000) "Example: Configuring Multidestination (Multicast, Broadcast, DLF) Classifiers" on page 104
		• Changing the Host Outbound Traffic Default Queue Mapping on page 250
		• Example: Configuring Forwarding Classes on page 157
		• Defining CoS Rewrite Rules on page 114
		• (Except NFX250) "Enabling and Disabling CoS Symmetric Ethernet PAUSE Flow Control" on page 211
		• (Except NFX250 and OCX1100) "Configuring CoS Asymmetric Ethernet PAUSE Flow Control" on page 212
		• Assigning CoS Components to Interfaces on page 77

Table 4: CoS Configuration Tasks *(continued)*

CoS Configuration Task	Platforms Supported	Links
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: • 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. • 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 • 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.	• QFX3500 • QFX3600 • EX4600 • QFX5100 • QFX5200 • QFX5210 • QFX10000 • OCX1100 switches • QFabric systems	• Example: Configuring WRED Drop Profiles on page 264 • Example: Configuring Drop Profile Maps on page 271 • Example: Configuring ECN on page 284
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.	• QFX3500 • QFX3600 • EX4600 • NFX250 • QFX5100 • QFX5200 • QFX5210 • QFX10000 • OCX1100 switches • QFabric systems	• (Except QFX10000) "Example: Configuring Queue Schedulers" on page 325 • Example: Configuring Queue Scheduling Priority on page 336 • (QFX10000 only) "Example: Configuring Queue Schedulers for Port Scheduling" on page 361

Table 4: CoS Configuration Tasks *(continued)*

CoS Configuration Task	Platforms Supported	Links
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.	• QFX3500 • QFX3600 • EX4600 • NFX250 • QFX5100 • QFX5200 • QFX5210 • QFX10000 • OCX1100 switches • QFabric systems	• (Except NFX250) "Defining CoS Traffic Control Profiles (Priority Group Scheduling)" on page 384 • (Except NFX250) "Example: Configuring Traffic Control Profiles (Priority Group Scheduling)" on page 385 • Example: Configuring Minimum Guaranteed Output Bandwidth on page 392 • (Except NFX250) "Example: Configuring Maximum Output Bandwidth" on page 400
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.	• QFX3500 • QFX3600 • EX4600 • QFX5100 • OCX1100 switches • QFX10000 • QFabric systems	• Example: Configuring Forwarding Class Sets on page 165 • Example: Configuring CoS Hierarchical Port Scheduling (ETS) on page 415 • (Except OCX1100)"Disabling the ETS Recommendation TLV" on page 447
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) • Configure the DCBX mode that an interface uses to communicate with the connected peer • Configure DCBX autonegotiation on a per-interface basis for each supported feature or application • Define each application for which you want DCBX to exchange application protocol information • Map applications to IEEE 802.1p code points • Apply an application map to a DCBX interface	• QFX3500 • QFX3600 • EX4600 • QFX5100 • QFX5200 • QFX5210 • QFX10000 • QFabric systems	• Example: Configuring DCBX Application Protocol TLV Exchange on page 476 • Configuring the DCBX Mode on page 464 • Configuring DCBX Autonegotiation on page 465 • Defining an Application for DCBX Application Protocol TLV Exchange on page 473 • Configuring an Application Map for DCBX Application Protocol TLV Exchange on page 474 • Applying an Application Map to an Interface for DCBX Application Protocol TLV Exchange on page 475

Table 4: CoS Configuration Tasks *(continued)*

CoS Configuration Task	Platforms Supported	Links
Configure CoS for FCoE: • Configure priority-based flow control (PFC) to divide traffic on one physical link into eight priorities • Configure a congestion notification profile (CNP) that enables priority-based flow control (PFC) on specified IEEE 802.1p priorities • Configure Multichassis link aggregation groups (MC-LAGs) to provide redundancy and load balancing between two switches • Configure two or more lossless forwarding classes and map them to different priorities • Configure lossless FCoE transport if your network uses a different priority than 3 • Configure multiple lossless FCoE priorities on a converged Ethernet network • 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 • Configure lossless priorities for multiple types of traffic, such as FCoE and iSCSI	• QFX3500 • QFX3600 • EX4600 • QFX5100 • QFX5200 • QFX5210 • QFX10000 • QFabric systems	• Example: Configuring CoS PFC for FCoE Traffic on page 490 • *Example: Configuring CoS for FCoE Transit Switch Traffic Across an MC-LAG* • Configuring CoS PFC (Congestion Notification Profiles) on page 194 • (QFX3500 and QFabric only) *Example: Configuring IEEE 802.1p Priority Remapping on an FCoE-FC Gateway* • Example: Configuring Two or More Lossless FCoE IEEE 802.1p Priorities on Different FCoE Transit Switch Interfaces on page 587 • Example: Configuring Lossless FCoE Traffic When the Converged Ethernet Network Does Not Use IEEE 802.1p Priority 3 for FCoE Traffic (FCoE Transit Switch) on page 565 • Example: Configuring Two or More Lossless FCoE Priorities on the Same FCoE Transit Switch Interface on page 576 • (QFX3500, NFX250, and QFabric only) *Configuring CoS Fixed Classifier Rewrite Values for Native FC Interfaces (NP_Ports)* • Example: Configuring Lossless IEEE 802.1p Priorities on Ethernet Interfaces for Multiple Applications (FCoE and iSCSI) on page 605

Understanding Junos CoS Components

IN THIS SECTION

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.

NOTE: OCX Series switches and NFX250 Network Services platform do not support MPLS.

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

On switches that do not separate unicast and multidestination traffic, unicast and multidestination traffic use the same classifiers, and you do not create a separate special classifier 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.

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 multidestination (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 multidestination 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.

NOTE: On OCX Series switches only, do not map traffic to the default fcoe and no-loss forwarding classes. By default, the DSCP default classifier does not map traffic to the fcoe and no-loss forwarding classes, so by default, OCX Series switches do not classify traffic into those forwarding classes. (On other switches, the fcoe and no-loss forwarding classes provide lossless transport for Layer 2 traffic. OCX Series switches do not support lossless Layer 2 transport.)

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.

NFX250 Network Services platform provide the following 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.

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.

> **NOTE:** OCX Series switches do not support PFC.

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 | 23

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.

> **NOTE:** OCX Series switches do not support Layer 2 interfaces (family ethernet-switching).

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 3 on page 24 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 3: CoS Classifier, Queues, and Scheduler

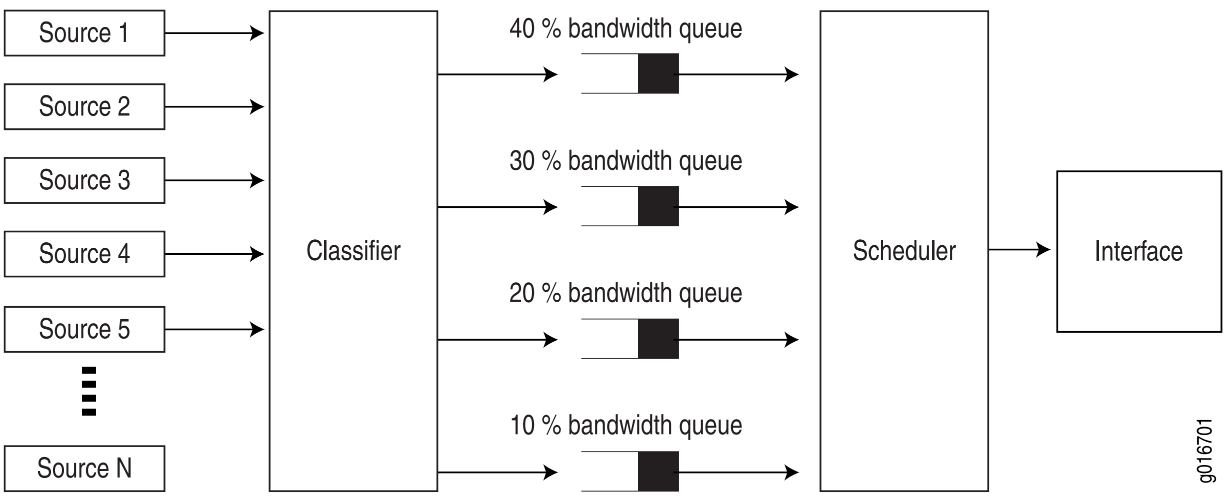
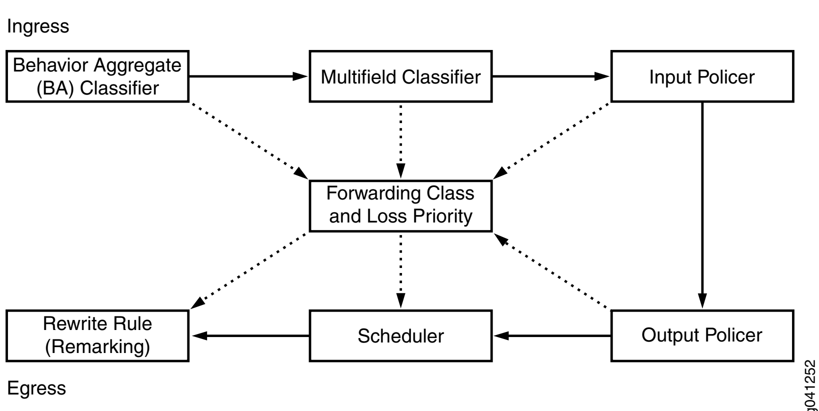


Figure 4 on page 24 shows the packet flow through the CoS components that you can configure.

Figure 4: 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

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 5 on page 26 shows the default mapping of the default forwarding classes to queues and packet drop attribute.

Table 5: 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
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 **NOTE:** 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 6 on page 27 shows the default mapping of code-point aliases to IEEE code points.

Table 6: 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 7 on page 27 shows the default mapping of code-point aliases to DSCP and DSCP IPv6 code points.

Table 7: Default DSCP and DCSP IPv6 Code-Point Aliases

CoS Value Types	Mapping
ef	101110
af11	001010

Table 7: 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
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 8 on page 29 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 8: 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 9 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 9: Default IEEE 802.1 Classifiers for Ports in Access Mode (Untrusted Classifier)

Code Point	Forwarding Class	Loss Priority
000	best-effort	low

Table 9: Default IEEE 802.1 Classifiers for Ports in Access Mode (Untrusted Classifier) *(continued)*

Code Point	Forwarding Class	Loss Priority
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 10 on page 30 shows the default mapping of IEEE 802.1 code-point values to multidestination (multicast, broadcast, and destination lookup fail traffic) forwarding classes and loss priorities.

Table 10: 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 11 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 multidestination traffic. DSCP IPv6 classifiers are not supported for multidestination traffic.

Table 11: 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
cs2 (010000)	best-effort	low
cs3 (011000)	best-effort	low
cs4 (100000)	best-effort	low

Table 11: Default DSCP IP and IPv6 Classifiers *(continued)*

Code Point	Forwarding Class	Loss Priority
cs5 (101000)	best-effort	low
nc1 (110000)	network-control	low
nc2 (111000)	network-control	low

On QFX10000 switches, Table 12 on page 32 shows the default mapping of MPLS EXP code-point values to forwarding classes and loss priorities.

Table 12: 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
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 13 on page 33 shows the default drop profile configuration.

Table 13: Default Drop Profile

Fill Level	Drop Probability
100	100

Default Schedulers

Table 14 on page 33 shows the default scheduler configuration.

Table 14: 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%
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 14 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, multidestination 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 multidestination traffic 2-Gbps of bandwidth.

NOTE: Except on QFX5200, QFX5210, and QFX10000 switches, multidestination queue 11 also receives a small amount of default bandwidth from the multidestination scheduler. CPU-generated multidestination 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 multidestination 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 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: 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 15 on page 36 shows the default mapping of forwarding classes to schedulers.

Table 15: 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 multidestination scheduler

Default Shared Buffer Configuration

Table Table 16 on page 36 and Table 17 on page 36 show the default shared buffer allocations:

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

Table 16: Default Ingress Shared Buffer Configuration

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

Table 17: Default Egress Shared Buffer Configuration

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

RELATED DOCUMENTATION

CoS Support on QFX Series Switches, EX4600 Line of Switches, and QFabric Systems

IN THIS SECTION

Juniper Networks data center switches differ in some aspects of class-of-service (CoS) support because of differences in the way the switches are used in networks, and because of hardware differences such as different chipsets or different interface capabilities.

This topic summarizes CoS support on QFX Series switches, the EX4600 line of switches, and QFabric systems.

CoS Feature Support

The first two tables list CoS feature support for newer ELS-CLI-based platforms (Table 18 on page 38) such as the QFX5000 line, the EX4600 line, and QFX10000 switches, and for legacy-CLI-based platforms (Table 19 on page 40) such as QFX3500 switches and QFabric systems. Some legacy-CLI-based platforms can also run the ELS CLI.

Table 18: QFX10000, QFX5000 Line, and EX4600 Line CoS Features

Feature	QFX10000	QFX 5000 Line, EX4600 Line	QFX5220/QFX5130
Class of service (CoS)—Class-based queuing with prioritization	Yes	Yes	Yes
CoS—Separate unicast and multi-destination classifiers, forwarding classes, and output queues	No	Yes	Yes (except multi-destination classifiers. Use firewall filters to classify multicast traffic.)
CoS—Shared unicast and multidestination classifiers, forwarding classes, and output queues	Yes	No	No
CoS support on link aggregation groups (LAGs)	Yes	Yes	Yes
Enhanced transmission selection (ETS) hierarchical port scheduling	Yes (starting in Junos OS Release 17.3)	QFX5100, QFX 5110, EX4600—Yes QFX5120, QFX5200, QFX5210, EX4650—No	No
Direct port scheduling	Yes	Yes	Yes
Queue shaping	Yes **NOTE:** Uses the **transmit-rate** statement with the **exact** option.	Yes **NOTE:** Uses the **shaping-rate** statement.	Yes

Table 18: QFX10000, QFX5000 Line, and EX4600 Line CoS Features *(continued)*

Feature	QFX10000	QFX 5000 Line, EX4600 Line	QFX5220/QFX5130
Explicit congestion notification (ECN)	Yes	Yes	Yes
Priority-based flow control (PFC)	Yes	Yes	Yes
Re-marking of bridged packets	Yes	Yes	Yes
Weighted random early detection (WRED) packet drop profiles and tail drop	Yes	Yes	Yes
802.3X Ethernet PAUSE	Yes	Yes	No
Layer 2 ingress packet classification and egress rewrite rules	Yes	Yes	Yes
MPLS EXP ingress packet classification and egress rewrite rules	Yes	Yes	No
Layer 3 ingress packet classification and egress rewrite rules	Yes	Yes	Yes (Both IPv4 and IPv6 traffic must share the same classifier.)
Virtual output queue (VOQ) architecture	Yes	No	No

Table 18: QFX10000, QFX5000 Line, and EX4600 Line CoS Features *(continued)*

Feature	QFX10000	QFX 5000 Line, EX4600 Line	QFX5220/QFX5130
Software shared buffer configurability	No (uses VOQ)	Yes	Yes, with the following restrictions: • **multicast** partition is not supported in the egress shared buffer pool. See buffer-partition (Egress). • **lossy** and **lossless** partitions must have the same percentage values for ingress and egress shared buffer pools.
Shared buffer Alpha configurability	No	Yes	Yes
Buffer monitoring	No	Yes	Yes
CoS command to detect the source of RED-dropped packets	Yes	No	No

Table 19 on page 40 shows CoS support for legacy-CLI-based switches.

Table 19: QFX3500 and QFX3600 Switch, and QFabric System CoS Features (As of Software Release 15.1X53-D30)

Feature	QFX3500	QFX3600	QFabric System
Class of service (CoS)—Class-based queuing with prioritization	Yes	Yes	Yes
CoS—Separate unicast and multidestination classifiers, forwarding classes, and output queues	Yes	Yes	Yes
CoS support on link aggregation groups (LAGs)	Yes	Yes	Yes

Table 19: QFX3500 and QFX3600 Switch, and QFabric System CoS Features (As of Software Release 15.1X53-D30) ***(continued)***

Feature	QFX3500	QFX3600	QFabric System
Enhanced transmission selection (ETS) hierarchical port scheduling	Yes	Yes	Yes
Direct port scheduling	No	No	No
Queue shaping	Yes	Yes	Yes
Explicit congestion notification (ECN)	Yes	Yes	Yes
Priority-based flow control (PFC)	Yes	Yes	Yes
Re-marking of bridged packets	Yes	Yes	Yes
Priority remapping on native Fibre Channel interfaces	Yes	No	No
Weighted random early detection (WRED) tail-drop profiles	Yes	Yes	Yes
802.3X Ethernet PAUSE	Yes	Yes	Yes
Layer 2 ingress packet classification and egress rewrite rules	Yes	Yes	Yes
MPLS EXP ingress packet classification and egress rewrite rules	Yes	Yes	Yes
Layer 3 ingress packet classification and egress rewrite rules	Yes	Yes	Yes
Software buffer configurability	Yes	Yes	No

Classifier and Rewrite Rule Ethernet Interface Type Support

The next two tables in this topic list CoS Ethernet support for classifiers and rewrite rules on different interface types for QFX10000 switches (Table 20 on page 42), and for QFX5100, QFX5110, QFX5120,

QFX5200, QFX5210, QFX5220, QFX3500, QFX3600, EX4600, and EX4650 switches, and QFabric systems (Table 21 on page 42).

On QFX10000 switches, you cannot apply classifiers or rewrite rules to Layer 2 or Layer 3 physical interfaces. 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. Table 20 on page 42 shows on which interfaces you can configure and apply classifiers and rewrite rules.

Table 20: 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

On QFX5100, QFX5110, QFX5120, QFX5200, QFX5210, QFX3500, QFX3600, EX4600, and EX4650 switches, and QFabric systems, you cannot apply classifiers or rewrite rules to Layer 2 physical interfaces or to Layer 3 logical interfaces. Table 21 on page 42 shows on which interfaces you can configure and apply classifiers and rewrite rules.

Table 21: Ethernet Interface Support for Classifier and Rewrite Rule Configuration (QFX5100, QFX5110, QFX5120, QFX5200, QFX5210, EX4600, EX4650, QFX3500, and QFX3600 Switches, and QFabric Systems)

CoS Classifiers and Rewrite Rules	Layer 2 Physical Interfaces	Layer 2 Logical Interface (Unit 0 Only)	Layer 3 Physical Interfaces (If at Least One Logical Layer 3 Interface Is Defined)	Layer 3 Logical Interfaces
Fixed classifier	No	Yes	Yes	No

Table 21: Ethernet Interface Support for Classifier and Rewrite Rule Configuration (QFX5100, QFX5110, QFX5120, QFX5200, QFX5210, EX4600, EX4650, QFX3500, and QFX3600 Switches, and QFabric Systems) ***(continued)***

CoS Classifiers and Rewrite Rules	Layer 2 Physical Interfaces	Layer 2 Logical Interface (Unit 0 Only)	Layer 3 Physical Interfaces (If at Least One Logical Layer 3 Interface Is Defined)	Layer 3 Logical Interfaces
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 mutidestination and DSCP multidestination classifiers are applied to all interfaces and cannot be applied to individual interfaces. No DSCP IPv6 multidestination classifier is supported. IPv6 multidestination traffic uses the DSCP multidestination classifier.

On QFX5220 and QFX5130 switches, you cannot apply classifiers or rewrite rules to Layer 2 or Layer 3 physical interfaces. Table 22 on page 43 shows on which interfaces you can configure and apply classifiers and rewrite rules.

Table 22: Ethernet Interface Support for Classifier and Rewrite Rule Configuration (QFX5220 and QFX5130 Switches)

CoS Classifiers and Rewrite Rules	Layer 2 Physical Interfaces	Layer 2 Logical Interfaces	Layer 3 Physical Interfaces	Layer 3 Logical Interfaces
Fixed classifier	No	Yes	No	Yes
DSCP classifier	No	Yes	No	Yes

Table 22: Ethernet Interface Support for Classifier and Rewrite Rule Configuration (QFX5220 and QFX5130 Switches) *(continued)*

CoS Classifiers and Rewrite Rules	Layer 2 Physical Interfaces	Layer 2 Logical Interfaces	Layer 3 Physical Interfaces	Layer 3 Logical Interfaces
DSCP IPv6 classifier	No	No	No	No
IEEE 802.1p classifier	No	Yes	No	Yes
EXP classifier	No	No	No	No
DSCP rewrite rule	No	Yes	No	Yes
DSCP IPv6 rewrite rule	No	No	No	No
IEEE 802.1p rewrite rule	No	Yes	No	Yes
EXP rewrite rule	No	No	No	No

NOTE: QFX5220 and QFX5130 switches do not support DSCP IPV6 classifiers and rewrite rules. Instead, attach DSCP classifiers and rewrite rules on family **inet6**.

CoS Operational Comparison Between QFX5100, QFX5120, QFX5130, QFX5200, QFX5210, and QFX5220 Switches

CoS feature support is mostly the same for QFX5100, QFX5120, QFX5130, QFX5200, QFX5210, and QFX 5220 switches, but there are some CoS operational differences due to different chipsets among these platforms. Table 23 on page 45 details both the similarities and differences for CoS on QFX5100, QFX5120, QFX5200, QFX5210, and QFX5220 switches.

Table 23: CoS Operational Comparison Between QFX5100, QFX5120, QFX5130, QFX5200, QFX5210, and QFX5220 Switches

CoS Feature	QFX5100	QFX5120	QFX5130	QFX5200	QFX5210	QFX5220	Change in Operation
Memory Management	Central memory management unit (MMU) shared by all ports	Central MMU shared by all ports	Ingress traffic manager (ITM) architecture – Buffers divided equally among 2 ITMS	Crosspoint architecture with quad pipe	Crosspoint architecture with quad pipe	ITM architecture – Buffers divided equally among 2 ITMS	ITM architecture requires special buffer management.
Pipes	2	2	8	4	4	8	No customer visible change.
Cell Accounting	Global access pipes	Global access pipes	Local to ITM (66MB/ITM)	Local to Cross point (4MB / cross point)	Local to Cross point (10.5MB / cross point)	Local to ITM (32MB/ITM)	No customer visible change.
Shared Buffer	60k Cells (Each cell 208Bytes), 12MB	About 131K Cells (Each cell 256 Bytes), 32MB	About 543K cells (Each cell 254 bytes), 132MB	(QFX5200-32C) 80K Cells (Each cell 208 Bytes), 16MB (QFX5200-48Y) 108K Cells (Each cell 208 Bytes), 22MB	About 210K Cells (Each cell 208 Bytes), 42MB	About 264K Cells (Each cell 254 Bytes), 64MB	No customer visible change, except QFX5200 and QFX5210 support larger packet buffer space than QFX5100.
Shared buffer pool per pipe	4 pools per pipe	4 pools per pipe	4 pools per pipe	4 pools per pipe	4 pools per pipe	4 pools per pipe	N/A

Table 23: CoS Operational Comparison Between QFX5100, QFX5120, QFX5130, QFX5200, QFX5210, and QFX5220 Switches *(continued)*

CoS Feature	QFX5100	QFX5120	QFX5130	QFX5200	QFX5210	QFX5220	Change in Operation
Queuing and Scheduling	LLS and three- level hierarchy	Fixed hierarchical scheduling (FHS) and two-level hierarchy	Fixed hierarchical scheduling (FHS) and two-level hierarchy	Fixed hierarchical scheduling (FHS) and two-level hierarchy	Fixed hierarchical scheduling (FHS) and two-level hierarchy	Fixed hierarchical scheduling (FHS) and two-level hierarchy	ETS and FC-Set are not supported on QFX5120, QFX5130, QFX5200, QFX5210, and QFX5220 due to FHS.
# Unicast Queues	8	8	8	8	8	8	N/A
# Multicast Queues	4	2	4	2	2	2	N/A
CPU Queues	44	44	44	44	44	44	N/A
Host Path Scheduling	48 queues directly attached to port	48 queues attached to L0	48 queues attached to L0	48 queues attached to L0	48 queues attached to L0	48 queues attached to L0	No customer visible change.
FC2Q	4 profiles	4 profiles	4 profiles	4 profiles	4 profiles	4 profiles	N/A
DSCP classifier table	128 profiles	128 profiles	64 profiles	128 profiles	128 profiles	64 profiles	N/A
802.1p classifier table	64 profiles	64 profiles	64 profiles	64 profiles	64 profiles	64 profiles	No customer visible change. SDK API change just affects software development effort.

Table 23: CoS Operational Comparison Between QFX5100, QFX5120, QFX5130, QFX5200, QFX5210, and QFX5220 Switches *(continued)*

CoS Feature	QFX5100	QFX5120	QFX5130	QFX5200	QFX5210	QFX5220	Change in Operation
PFC	Common headroom buffer	Common headroom buffer	Per ITM headroom buffer	Per pipe headroom buffer	Per pipe headroom buffer	Per ITM headroom buffer	Available and used head room buffer is maintained separately for each pipe on QFX5200 and QFX5210.
Rewrite	128 profiles	128 profiles	128 profiles	128 profiles	128 profiles	128 profiles	No customer visible change. SDK API change just affects software development effort.
WRED	128 profiles per pipe	128 profiles per pipe	128 profiles per pipe	128 profiles per pipe	128 profiles per pipe	128 profiles per pipe	N/A
Queueing Levels	Four levels physical queue level, logical queue level, CoS level, and port level	Three levels, logical queue level, CoS level, and port level.	Three levels, logical queue level, CoS level, and port level.	Three levels, logical queue level, CoS level, and port level.	Three levels, logical queue level, CoS level, and port level.	Three levels, logical queue level, CoS level, and port level.	N/A

Table 23: CoS Operational Comparison Between QFX5100, QFX5120, QFX5130, QFX5200, QFX5210, and QFX5220 Switches *(continued)*

CoS Feature	QFX5100	QFX5120	QFX5130	QFX5200	QFX5210	QFX5220	Change in Operation
Multidestination Traffic	Default scheduler map reserves 20% bandwidth for multicast and 80% of unicast traffic reserved between BE, FCoE, NoLoss and NC traffic types.	Same as QFX5100 switches	By default all multicast traffic mapped to Q8. Q8 is given 20% bandwidth in default scheduler. To classify multicast traffic to different queues (Q9,10,11) use firewall filters.	Each level 0 node is receiving both multicast and unicast traffic, so it is not possible to differentiate at the port level to apply shaping on multicast traffic.	Same as QFX5200 switches	By default all multicast traffic mapped to Q8. Q8 is given 20% bandwidth in default scheduler. To classify multicast traffic to different queue (Q9) use firewall filters.	N/A

The following limitations on QFX5200 and QFX5210 switches do not exist on QFX5100 switches.

- CoS flexible hierarchical scheduling (ETS) is not supported on QFX5200 or QFX5210 switches.
- QFX5200 and QFX5210 switches support only one queue with **strict-high** priority because these switches do not support flexible hierarchical scheduling.

> **NOTE:** QFX5100 switches support multiple queues with **strict-high** priority when you configure a forwarding class set.

- QFX5200 CoS policers do not support global management counters accessed by all ports. Only management counters local to a pipeline are supported—this means that QFX5200 management counters work only on traffic received on ports that belong to the pipeline in which the counter is created.
- Due to the cross-point architecture on QFX5200 and QFX5210 switches, all buffer usage counters are maintained separately. When usage counters are displayed with the command **show class-of-service shared-buffer**, various pipe counters are displayed separately.
- On QFX5200 and QFX5210 switches, port schedulers are supported instead of FC-SET.

- On QFX5200 and QFX5210 switches, it is not possible to group multiple forwarding classes into a forwarding class set (fc-set) and apply output traffic control profile on the fc-set. ETS for an fc-set is not supported. Because each L0 node schedules both the unicast and multicast queue of L1 node, it is not possible to differentiate multicast and unicast traffic at the port level and apply minimum bandwidth between unicast and multicast. It can only be supported at CoS level L0.
- Because QFX5200 and QFX5210 switches do not support flexible hierarchical scheduling, it is not possible to apply a traffic control profile for a group of forwarding classes.

QFX10000 Switch Classifier and Rewrite Rule Support (Scaling)

You can configure enough classifiers on QFX10000 switches to handle most, if not all, network scenarios. Table 24 on page 49 shows how many of each type of classifiers you can configure, and how many entries you can configure per classifier.

Table 24: Classifier Support by Classifier Type on QFX10000 Switches

Classifier Type	Default Classifier Name	Maximum Number of Classifiers	Maximum Number of Entries per Classifier
IEEE 802.1p (Layer 2)	ieee8021p-default (for ports in trunk mode) ieee8021p-untrust (for ports in access mode)	64	16
DSCP (Layer 3)	dscp-default	64	64
DSCP IPv6 (Layer 3)	dscp-ipv6-default	64	64
EXP (MPLS)	exp-default	64	8
Fixed	There is no default fixed classifier	8	16

The number of fixed classifiers supported (8) equals the number of supported forwarding classes (fixed classifiers assign all incoming traffic on an interface to one forwarding class).

There are no default rewrite rules. You can configure enough rewrite rules on QFX10000 switches to handle most, if not all, network scenarios. Table 25 on page 50 shows how many of each type of rewrite rule you can configure, and how many entries you can configure per rewrite rule.

Table 25: Rewrite Rule Support by Rewrite Rule Type on QFX10000 Switches

Rewrite Rule Type	Maximum Number of Rewrite Rule Sets	Maximum Number of Entries per Rewrite Rule Set
IEEE 802.1p (Layer 2)	64	128
DSCP (Layer 3)	32	128
DSCP IPv6 (Layer 3)	32	128
EXP (MPLS)	64	128

RELATED DOCUMENTATION

Understanding Applying CoS Classifiers and Rewrite Rules to Interfaces | 116
Understanding Default CoS Scheduling and Classification | 298
Understanding CoS Classifiers | 84
Understanding CoS Output Queue Schedulers | 313
Understanding CoS Virtual Output Queues (VOQs) on QFX10000 Switches | 377
Understanding CoS Port Schedulers on QFX Switches | 343
Understanding CoS Hierarchical Port Scheduling (ETS) | 407
Understanding CoS WRED Drop Profiles | 253
Understanding CoS Rewrite Rules | 111

CHAPTER 2

CoS on Interfaces

IN THIS CHAPTER

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 26 on page 51.

Table 26: 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.
rewrite-value (Fibre Channel Interfaces)	*forwarding-class*	*code-point*	(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 | 23

CoS on Virtual Chassis Switch Ports

IN THIS SECTION

QFX Series and EX4600 Virtual Chassis devices have access ports to connect to external peer devices. Virtual Chassis devices also have Virtual Chassis ports (VCPs) to interconnect members of the Virtual Chassis, in a similar way that QFabric system Node devices have fabric (fte) ports to connect to the QFabric system Interconnect device. VCPs are not used for external access.

Class of service (CoS) on Virtual Chassis access ports is the same as CoS on these devices when they are in standalone mode or used as QFabric system Node devices. However, CoS on VCPs differs in several ways from CoS on QFabric system Node device fabric ports.

This topic describes CoS support on Virtual Chassis access interfaces and on VCPs:

Access Interface CoS Support

IN THIS SECTION

CoS on Virtual Chassis access interfaces is the same as CoS on standalone device and Node device access interfaces, except for shared buffer settings. The documentation for QFX Series and EX4600 switch CoS on access interfaces applies to Virtual Chassis access interfaces, except some of the shared buffer documentation.

Similarities in CoS Support on Virtual Chassis Access Interfaces Compared to Standalone Device (or QFabric system Node device) Access Interfaces

Virtual Chassis access interfaces support the following CoS features in the same way as access interfaces on standalone devices and QFabric system Node devices:

- Forwarding classes—The default forwarding classes, queue mapping, and packet drop attributes (Table 27 on page 53) are the same:

  Table 27: Default Forwarding Class Configuration

Default Forwarding Class	Default Queue Mapping	Default Packet Drop Attribute
best-effort (be)	0	drop
fcoe	3	no-loss
no-loss	4	no-loss
network-control (nc)	7	drop
mcast	8	drop

- Packet classification—Classifier default settings and configuration are the same. Support for behavior aggregate, multifield, multidestination, and fixed classifiers is the same.
- Enhanced transmission selection (ETS)—This data center bridging (DCB) feature that supports hierarchical scheduling has the same defaults and user configuration, including forwarding class set (priority group) and traffic control profile configuration.
- Priority-based flow control (PFC)—This DCB feature that supports lossless transport has the same defaults and user configuration, including support for six lossless priorities (forwarding classes).
- Ethernet PAUSE—This feature has the same defaults and configuration.
- Queue scheduling—This feature has the same defaults, configuration, and scheduler-to-forwarding-class mapping. Queue scheduling is a subset of hierarchical scheduling.
- Priority group (forwarding class set) scheduling—This feature has the same defaults and configuration. Priority group scheduling is a subset of hierarchical scheduling.
- WRED profiles—This feature has the same defaults and configuration.
- Code-point aliases—This feature has the same defaults and configuration.

- Rewrite rules—This feature has the same defaults and configuration (no default rewrite rules applied to egress traffic).
- Host outbound traffic—This feature has the same defaults and configuration.

Differences in CoS Support on Virtual Chassis Access Interfaces Compared to Standalone Device (or QFabric system Node device) Access Interfaces

The default shared buffer settings and the way in which you configure shared buffers are the same on Virtual Chassis access interfaces as on standalone and QFabric system Node devices. The difference is that on Virtual Chassis access interfaces, the shared buffer configuration is global and applies to all access ports on all members of the Virtual Chassis, while on standalone or QFabric system Node devices, you can configure different buffer settings on different access interfaces.

You cannot configure different shared buffer settings for different Virtual Chassis members. All members of a Virtual Chassis use the same shared buffer configuration.

VCP Interface CoS Support

IN THIS SECTION

CoS on the VCP interfaces that connect the Virtual Chassis members is similar to CoS on the fabric interfaces of QFabric system Node devices, but there are several important differences:

Similarities in CoS Support on VCP Interfaces and QFabric System Node Device Fabric Interfaces

VCP interfaces support full hierarchical scheduling (ETS). ETS includes the following CoS features. VCP interfaces support no other CoS features.

- Creating forwarding class sets (priority groups) and mapping forwarding classes to forwarding class sets.
- Scheduling individual output queues. The scheduler defaults and configuration are the same as the scheduler on access interfaces.
- Scheduling priority groups (forwarding class sets) using a traffic control profile. The defaults and configuration are the same as on access interfaces.

NOTE: You cannot attach classifiers, congestion notification profiles, scheduler maps, or rewrite rules to VCP interfaces. Also, you cannot configure buffer settings on VCP interfaces. Similar to fabric interfaces on QFabric system Node devices, you can only attach forwarding class sets and traffic control profiles to VCP interfaces.

The behavior of lossless traffic across 40-Gigabit VCP interfaces is the same as the behavior of lossless traffic across QFabric system Node device fabric ports. The system automatically enables flow control for lossless forwarding classes (priorities). The system dynamically calculates buffer headroom that is allocated from the global lossless-headroom buffer for the lossless forwarding classes on each 40-Gigabit VCP interface. If there is not enough global lossless-headroom buffer space to support the number of lossless flows on a 40-Gigabit VCP interface, the system generates a syslog message.

NOTE: After you configure lossless transport on a Virtual Chassis, check the syslog messages to ensure that there is sufficient buffer space to support the configuration.

NOTE: If you break out a 40-Gigabit VCP interface into 10-Gigabit VCP interfaces, lossless transport is not supported on the 10-Gigabit VCP interfaces. Lossless transport is supported only on 40-Gigabit VCP interfaces. (10-Gigabit access interfaces support lossless transport.)

Differences in CoS Support on VCP Interfaces and QFabric System Node Device Fabric Interfaces

Although most of the CoS behavior on VCP interfaces is similar to CoS behavior on the fabric ports of QFabric system Node devices, there are some important differences:

- Hierarchical scheduling (queue and priority group scheduling)—On QFabric system Node device fabric interfaces, you can apply a different hierarchical scheduler (traffic control profile) to different priority groups (forwarding class sets) on different interfaces. However, on VCP interfaces, the schedulers that you apply to priority groups are global to all VCP interfaces. One hierarchical scheduler controls scheduling for a priority group on all VCP interfaces.

  You attach a scheduler to VCP interfaces using the global identifier (vcp-*) for VCP interfaces. For example, if you want to apply a traffic control profile (traffic control profiles contain both queue and priority group scheduling configuration) named *vcp-hpc-tcp* to a forwarding class set named *vcp-hpc-fcset*, you include the following statement in the configuration:

  ```
  [edit]
  user@switch# set class-of-service interfaces vcp-* forwarding-class-set vcp-hpc-fcset
  output-traffic-control-profile vcp-hpc-tcp
  ```

The system applies the hierarchical scheduler *vcp-hpc-tcp* to the traffic mapped to the priority group *vcp-hpc-fcset* on all VCP interfaces.

- You cannot attach classifiers, congestion notification profiles, or rewrite rules to VCP interfaces. Also, you cannot configure buffer settings on VCP interfaces. Similar to QFabric system Node device fabric interfaces, you can only attach forwarding class sets and traffic control profiles to VCP interfaces.
- Lossless transport is supported only on 40-Gigabit VCP interfaces. If you break out a 40-Gigabit VCP interface into 10-Gigabit VCP interfaces, lossless transport is not supported on the 10-Gigabit VCP interfaces.

CPU-Generated Host Outbound Traffic

CPU-generated host outbound traffic is forwarded on the network-control forwarding class, which is mapped to queue 7. If you use the default scheduler, the network-control queue receives a guaranteed minimum bandwidth (transmit rate) of 5 percent of port bandwidth. The guaranteed minimum bandwidth is more than sufficient to ensure lossless transport of host outbound traffic.

However, if you configure and apply a scheduler instead of using the default scheduler, you must ensure that the network-control forwarding class (or whatever forwarding class you configure for host outbound traffic) receives sufficient guaranteed bandwidth to prevent packet loss.

TIP: If you configure a scheduler instead of using the default scheduler, we recommend that you configure the network-control queue (or the queue you configure for host outbound traffic if it is not the network-control queue) as a strict-high priority queue. Strict-high priority queues receive the bandwidth required to transmit their entire queues before other queues are served. To limit the amount of bandwidth a strict-high priority queue can consume (and to prevent the strict-high priority queue from starving other queues), apply a shaping rate to the strict-high priority traffic in the scheduler configuration.

As with all strict-high priority traffic, if you configure the network-control queue (or any other queue) as a strict-high priority queue, you must also create a separate forwarding class set (priority group) that contains only strict-high priority traffic, and apply the strict-high priority forwarding class set and its traffic control profile (hierarchical scheduler) to the VCP interfaces.

RELATED DOCUMENTATION

Understanding Default CoS Settings | 25
Understanding CoS Classifiers | 84
Understanding CoS Forwarding Classes | 139

CoS on Virtual Chassis Fabric (VCF) EX4300 Leaf Devices (Mixed Mode)

IN THIS SECTION

A Virtual Chassis Fabric (VCF) uses QFX5100 switches as spine devices and can use QFX5100, QFX3500, QFX3600, and EX4300 switches as leaf devices. When a VCF includes more than one type of leaf device (mixed mode), the CoS feature support on the VCF depends on the capability of the lowest-featured device. In mixed mode, the supported CoS features are the “lowest common denominator” of the features supported by the leaf devices. If one leaf device does not support a particular feature, that feature is not supported on the VCF even if every other leaf device supports the feature.

NOTE: EX4300 leaf devices do not support several CoS features that are supported on QFX5100, QFX3600, and QFX3500 devices. However, even when a VCF includes an EX4300 leaf device, other leaf devices might support those CoS features.

VCF CoS in Mixed Mode with an EX4300 Leaf Device

In mixed mode, if all of the leaf devices are QFX5100, QFX3500, and QFX3600 switches, the full QFX Series CoS feature set is available, including data center bridging (DCB) features such as enhanced transmission selection (ETS, IEEE 802.1Qaz), priority-based flow control (PFC, IEEE 802.1Qbb), and Data Center Bridging Exchange Protocol (DCBX, an extension of LLDP, IEEE 802.1AB).

However, the EX4300 leaf device does not support DCB standards (ETS, PFC, DCBX). The lack of support for DCB standards means that the EX4300 leaf device does not support lossless transport. So a VCF that includes an EX4300 as a leaf device does not support lossless storage traffic such as Fibre Channel over Ethernet (FCoE).

In addition, a VCF with an EX4300 leaf device either does not support or has limited support for some other CoS features that the QFX Series switches support, including some buffer configuration features, some packet rewrite features, and Ethernet PAUSE (IEEE 802.3X).

Table 28 on page 58 summarizes the CoS support on a VCF in mixed mode with one or more EX4300 leaf devices.

Table 28: Support of QFX CoS Features on a VCF in Mixed Mode with an EX4300 Leaf Device

QFX Series CoS Feature	Support in Mixed Mode with an EX4300 Leaf Device
Forwarding Classes	The EX4300 leaf device uses the QFX Series default forwarding classes, the default QFX Series forwarding class to queue mapping, and the QFX Series maximum number of supported forwarding classes (12).
Lossless Forwarding Classes	Not supported. For example, the QFX Series default lossless forwarding classes **fcoe** and **no-loss** are not treated as lossless forwarding classes. Traffic mapped to lossless forwarding classes (default lossless forwarding classes or user-defined lossless forwarding classes) is treated as best-effort traffic.
Shared buffer configuration	Ingress shared buffer configuration is not supported. Egress shared buffer configuration does not support partitioning into three buffer pools. If there is a shared buffer configuration, only the total egress shared buffer configuration is used. Ingress shared buffer configuration and egress buffer partitioning configuration is ignored.
Classifier on a Layer 2 interface	One classifier per protocol is supported on a port. On a physical port, for a particular protocol, the same Layer 2 classifier is used on all of the logical interfaces.
Classifier on a Layer 3 interface	Supported.
Multi-destination classifier	Supported. The EX4300 leaf device uses the same default classifier as the QFX5100 spine device. As on QFX Series switches, a multi-destination classifier is global and is applied to all VCF interfaces. Multi-destination classifiers are valid only for multicast forwarding classes. You can configure two multi-destination classifiers, one for IEEE 802.1p traffic and one for DSCP traffic (the DSCP multi-destination classifier applies to both IPv4 and IPv6 traffic).

Table 28: Support of QFX CoS Features on a VCF in Mixed Mode with an EX4300 Leaf Device *(continued)*

QFX Series CoS Feature	Support in Mixed Mode with an EX4300 Leaf Device
Congestion notification profile	Not supported. If a congestion notification profile is configured on the QFX5100 spine device, it is ignored because the EX4300 leaf device does not support lossless transport, so end-to-end lossless behavior is not possible
Ethernet PAUSE (IEEE 802.3X)	Not supported. If Ethernet PAUSE is configured, it is ignored.
Hierarchical scheduling (ETS)	Translated into port-based scheduling. The EX4300 device does not support ETS scheduling. A VCF translates ETS scheduling configured on a QFX5100 spine device into port scheduling on an EX4300 leaf device. The hierarchical structure of mapping forwarding classes into forwarding class sets (fc-sets) is ignored. "Scheduling on an EX4300 VCF Leaf Device" on page 60 provides details on how a VCF translates QFX Series ETS scheduling into port scheduling on an EX4300 leaf device.
Hierarchical scheduling (ETS) on a spine device VCP port	On QFX5100 VCP ports, the hierarchical mapping of forwarding classes to forwarding class sets is supported. However, scheduling on an EX4300 leaf device is translated into port scheduling.
Drop profile (WRED)	QFX Series drop profiles are supported. The EX4300 device as a standalone switch supports four packet loss priorities. However, as part of a mixed mode VCF, the EX4300 leaf device supports only the three packet loss priorities that the QFX Series switches support: • low • medium-high • high Supporting only three packet loss priorities means that the behavior of the EX4300 switch as a leaf device is different from the behavior as a standalone switch.
Rewrite rules on a Layer 2 interface	Supported, but with a limit of one rewrite rule per physical interface. All traffic uses the same rewrite rule.
Rewrite rules on a Layer 3 interface	Supported, but with a limit of one rewrite rule per physical interface. The same rewrite rule is used on all traffic on the interface.

Table 28: Support of QFX CoS Features on a VCF in Mixed Mode with an EX4300 Leaf Device *(continued)*

QFX Series CoS Feature	Support in Mixed Mode with an EX4300 Leaf Device
Rewrite value for FCoE traffic	Not supported. If a rewrite value for FCoE traffic, is configured, it is ignored. (A mixed mode VCF does not support lossless traffic.)

In addition to the CoS limitations shown in Table 28 on page 58, using wild cards in a LAG configuration is not supported in mixed mode with one or more EX4300 leaf devices.

Scheduling on an EX4300 VCF Leaf Device

Because the EX4300 leaf device does not support ETS, the VCF translates the ETS scheduling configuration into the port scheduling configuration that the EX4300 device supports. The QFX5100 spine device uses two-tier ETS scheduling, as described in detail in "Understanding CoS Hierarchical Port Scheduling (ETS)" on page 407.

Briefly, ETS allocates port bandwidth into forwarding class sets (priority groups) and forwarding classes (priorities) in a hierarchical manner. Each forwarding class set consists of individual forwarding classes, with each forwarding class mapped to an output queue.

Port bandwidth (minimum guaranteed bandwidth and maximum bandwidth) is allocated to each forwarding class set. Forwarding class set bandwidth is in turn allocated to the forwarding classes in the forwarding class set. If a forwarding class does not use its bandwidth allocation, other forwarding classes within the same forwarding class set can share the unused bandwidth. If the forwarding classes in a forwarding class set do not use the bandwidth allocated to that forwarding class set, other forwarding class sets on the port can share the unused bandwidth. (This is how ETS increases port bandwidth utilization, by sharing unused bandwidth among forwarding classes and forwarding class sets.)

However, the EX4300 leaf device supports port scheduling, not ETS. Port scheduling is a "flat" scheduling method that allocates bandwidth directly to forwarding classes in a non-hierarchical manner.

The VCF translates the two tiers of the ETS scheduling configuration (forwarding class sets and forwarding classes) into a single port scheduling configuration as follows:

- The bandwidth allocated to a forwarding class set is divided equally among the forwarding classes in the forwarding class set. (Traffic control profiles schedule bandwidth allocation to forwarding class sets.) The minimum guaranteed bandwidth (**guaranteed-rate**) and maximum bandwidth limit (**shaping-rate**) of the forwarding class set determine the guaranteed minimum bandwidth and the maximum bandwidth the forwarding classes receive, *unless* those values are different in the forwarding class scheduler configuration.
- If there is an explicit forwarding class bandwidth scheduler configuration, it overrides the forwarding class set configuration. Bandwidth scheduling values that are not explicitly configured in a forwarding

class scheduler use the values from the forwarding class set (the traffic control profile configuration). Forwarding class schedulers control the minimum guaranteed bandwidth (**transmit-rate**), the maximum bandwidth (**shaping-rate**), and the priority (**priority**) for each forwarding class (output queue). Because the priority value is not configured at the forwarding class set level, the priority configured in the forwarding class scheduler is always used.

The following two scenarios illustrate how a VCF translates an ETS configuration into a port scheduling configuration:

Scenario 1

A forwarding class set named **fc-set-1** has a configured guaranteed minimum bandwidth (**guaranteed-rate**) of 4G, and a configured maximum bandwidth (**shaping-rate**) of 5G.

Forwarding class set **fc-set-1** consists of two forwarding classes, named **fc-1** and **fc-2**:

- Forwarding class **fc-1** has a guaranteed minimum bandwidth (**transmit-rate**) of 2.5G. There is no configured maximum bandwidth (**shaping-rate**).
- Forwarding class **fc-2** has a guaranteed minimum bandwidth (**transmit-rate**) of 1.5G. There is no configured maximum bandwidth (**shaping-rate**).

On the EX4300 leaf device, the ETS configuration above is translated approximately to the following port scheduling configuration:

- Guaranteed minimum bandwidth—Because guaranteed minimum bandwidth has been explicitly configured in the forwarding class scheduler, forwarding class **fc-1** receives a transmit rate of 2.5G and forwarding class **fc-2** receives a transmit rate of 1.5G.

  NOTE: If there had been no forwarding class scheduler **transmit-rate** configuration, then the forwarding class set minimum guaranteed bandwidth of 4G would have been split evenly between the forwarding classes, with each forwarding class receiving a minimum guaranteed bandwidth rate of 2G.

- Maximum bandwidth—Because there is no explicit maximum bandwidth (**shaping-rate** configuration for the forwarding classes, the forwarding classes that belong to the forwarding class set receive an equal share of the maximum bandwidth configured at the forwarding class set level in the traffic control profile. Because the forwarding class set maximum bandwidth is 5G, forwarding classes **fc-1** and **fc-2** each receive a maximum bandwidth of 2.5G.

In this scenario, the minimum guaranteed bandwidth and the maximum bandwidth configured at the forwarding class set hierarchy level are achieved on the forwarding classes that belong to the forwarding class set. (This does not always happen, as Scenario 2 shows.) However, unused bandwidth is not shared the same way. For example, if forwarding class **fc-1** experienced a burst of traffic at 3.5G, it would be limited to a maximum of 2.5G and traffic would be dropped. Using ETS, if forwarding class **fc-2** was not using its allocated maximum bandwidth, then **fc-1** could use (share) that unused bandwidth. But flat port scheduling does not share the unused bandwidth.

Scenario 2

A forwarding class set named **fc-set-2** has a configured guaranteed minimum bandwidth (**guaranteed-rate**) of 6G, and a configured maximum bandwidth (**shaping-rate**) of 9G.

Forwarding class set **fc-set-2** consists of three forwarding classes, named **fc-3**, **fc-4**, and **fc-5**:

- Forwarding class **fc-3** has a guaranteed minimum bandwidth (**transmit-rate**) of 1G. There is no configured maximum bandwidth (**shaping-rate**).
- Forwarding class **fc-4** has a maximum bandwidth (**shaping-rate**) of 2G. There is no configured guaranteed minimum bandwidth (**transmit-rate**).
- Forwarding class **fc-5** has a guaranteed minimum bandwidth (**transmit-rate**) of 3G. There is no configured maximum bandwidth (**shaping-rate**).

On the EX4300 leaf device, the ETS configuration above is translated approximately to the following port scheduling configuration:

- Guaranteed minimum bandwidth—Two forwarding classes (**fc-3** and **fc-5**) have an explicitly configured transmit rate, and one forwarding class (**fc-4**) does not. Forwarding classes **fc-3** and **fc-5** receive the minimum guaranteed bandwidth configured in their schedulers, so forwarding class **fc-3** receives 1G guaranteed minimum bandwidth and forwarding class **fc-5** receives 3G guaranteed minimum bandwidth.

  Forwarding class **fc-4** does not have an explicitly configured transmit rate, so the port derives the minimum guaranteed bandwidth from the forwarding class set guaranteed rate. Forwarding class set **fc-set-2** has a minimum guaranteed bandwidth (**guaranteed-rate**) of 6G, and there are three forwarding classes in the forwarding class set. Forwarding class **fc-4** receives an equal share (one third) of the forwarding class set minimum guaranteed bandwidth. So forwarding class **fc-4** is allocated a guaranteed minimum bandwidth (**transmit-rate**) of 2G (6G divided by 3 forwarding classes = 2G).
- Maximum bandwidth—Forwarding class **fc-4** has an explicitly configured shaping rate, and forwarding classes **fc-3** and **fc-5** do not. Forwarding class **fc-4** receives the maximum bandwidth configured in its scheduler, so forwarding class **fc-4** receives a maximum bandwidth of 2G.

Forwarding classes **fc-3** and **fc-5** do not have explicitly configured shaping rates, so the port derives the maximum bandwidth from the forwarding class set shaping rate. Forwarding class set **fc-set-2** has a maximum bandwidth (**shaping-rate**) of 9G, and there are three forwarding classes in the forwarding class set. Forwarding classes **fc-3** and **fc-5** each receive an equal share (one third) of the forwarding class set shaping rate. So forwarding classes **fc-3** and **fc-5** are allocated a maximum bandwidth of 3G each (9G divided by 3 forwarding classes = 3G).

Forwarding class **fc-4** receives less maximum bandwidth than forwarding classes **fc-3** and **fc-5** because the explicitly configured shaping rate for forwarding class **fc-4** is only 2G, and the explicit forwarding class configuration overrides the forwarding class set configuration.

NOTE: Scenario 2 shows that in some cases, the guaranteed minimum bandwidth (**guaranteed-rate**) and the maximum bandwidth (**shaping-rate**) configured for a forwarding class set might not be achieved at the forwarding class (queue) level. In Scenario 2, forwarding class set **fc-set-2** has a shaping rate of 9G, but the sum of the implemented forwarding class shaping rates is only 8G [(3G for **fc-3**) + (2G for **fc-4**) + (3G for **fc-5**)].

RELATED DOCUMENTATION

Understanding Default CoS Settings | 25
Understanding CoS Hierarchical Port Scheduling (ETS) | 407

Understanding CoS on OVSDB-Managed VXLAN Interfaces

IN THIS SECTION

You can configure class of service (CoS) features on OVSDB-managed VXLAN interfaces on QFX5100 and QFX10000 Series switches. An OVSDB-managed VXLAN interface uses an OVSDB controller to create and manage the VXLAN interfaces and tunnels. OVSDB-managed VXLAN interfaces support:

- Packet classifiers on ingress interfaces. On network-facing interfaces (interfaces that connect to the network, for example, switch interfaces that connect to a VXLAN gateway), you can configure DSCP classifiers. Fixed classifiers, 802.1p classifiers, and MPLS EXP classifiers are not supported on VXLAN interfaces.

  > **NOTE:** MF Filters on access-facing interfaces are applied as a group config and not as a normal filter.

- Packet rewrite rules (to change the code point bits of outgoing packets). On network-facing interfaces, you can configure DSCP rewrite rules. Rewrite rules are not supported on access-facing interfaces, and are not supported for IEEE 802.1p code points.

  > **NOTE:** Rewrite rules rewrite the DSCP code point on the VXLAN header only. Rewrite rules do not rewrite the DSCP code point on the inner packet header.

- Packet schedulers on egress interfaces. You can configure schedulers on network-facing and access-facing interfaces.

  > **NOTE:** You cannot configure CoS on manually configured VXLAN interfaces.

CoS configuration on OVSDB-managed VXLAN interfaces uses the same CLI statements and configuration constructs as CoS configuration on regular Ethernet interfaces. However, feature support differs on OVSDB-managed VXLAN interfaces and regular Ethernet interfaces. The following sections describe the differences between CoS support on OVSDB-managed VXLAN interfaces and regular Ethernet interfaces:

Classifier and Rewrite Rule Interface Support

The switch 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)

You can apply CoS classifiers and rewrite rules only to the following interfaces:

- Layer 2 physical interfaces. All underlying logical Layer 2 interfaces on the physical interface use the classifier and rewrite rule configuration on the physical interface. All OVSDB-managed VXLAN traffic on the interface uses the same Layer 2 CoS classifiers and rewrite rules.
- Layer 3 physical interfaces if at least one logical Layer 3 interface is configured on the physical interface. All underlying logical Layer 3 interfaces on the physical interface use the classifier and rewrite rule configuration on the physical interface. All OVSDB-managed VXLAN traffic on the interface uses the same Layer 3 CoS classifiers and rewrite rules.

Table 29 on page 65 shows on which interfaces you can configure and apply classifiers and rewrite rules on *network-facing* interfaces.

Table 29: OSVDB-Managed VXLAN Interface Support for Classifier and Rewrite Rule Configuration on Network-Facing Interfaces

CoS Classifiers and Rewrite Rules	Layer 2 Physical Interfaces	Layer 2 Logical Interfaces	Layer 3 Physical Interfaces (If at Least One Logical Layer 3 Interface Is Defined)	Layer 3 Logical Interfaces
Fixed classifier	Not Supported			
DSCP classifier	Yes	No	Yes	No
DSCP IPv6 classifier	Yes	No	Yes	No
IEEE 802.1p classifier	Not Supported			
EXP classifier	Not Supported			
DSCP rewrite rule	Yes	No	Yes	No
DSCP IPv6 rewrite rule	Yes	No	Yes	No
IEEE 802.1p rewrite rule	Not Supported			
EXP rewrite rule	Not Supported			

NOTE: The switch encapsulates packets in VXLAN after packet classification, and before packet rewrite and scheduling.

Classifiers on OVSDB-Managed VXLAN Interfaces

Classifiers map incoming packets to a CoS service level, based on the code points in the header of the incoming packet. At the ingress interface, the switch reads the code point value in the packet header, then assigns the packet to the forwarding class and loss priority mapped to that code point value. The forwarding class is mapped to an egress queue and to scheduling properties. OVSDB-managed VXLAN interfaces support packet classification based on DSCP code points on all ingress interfaces, and packet classification based on DSCP multi-field (MF DSCP) code points or for behavior aggregate (BA) classification, the DSCP, DSCP IPv6, or IP precedence bits of the IP header convey the behavior aggregate class information on access-facing interfaces.

If you do not configure classifiers, the switch uses the default CoS settings to classify incoming traffic, as described in "Understanding Default CoS Scheduling and Classification" on page 298.

Classifier configuration on an OVSDB-managed VXLAN switch interface is similar to classifier configuration on any other type of ingress interface (see "Understanding CoS Classifiers" on page 84). However, on OVSDB-managed VXLAN interfaces, there is a difference in the way you can apply classifiers to Layer 2 interfaces compared to non-VXLAN interfaces. On OVSDB-managed VXLAN interfaces, you apply classifiers to Layer 2 physical interfaces, and the underlying logical interfaces use the classifier configuration applied on the physical interface. On non-VXLAN interfaces, you apply Layer 2 classifiers to logical interface unit 0 (all other logical interfaces on the port use the classifier configured on unit 0), and not to physical interfaces.

Classifiers on Access-Facing Interfaces

When a packet enters an ingress switch from a server (or other source), you can map it to a forwarding class and a loss priority based on its DSCP multi-field (MF DSCP) classifiers code points. The forwarding class is mapped to an egress queue and to scheduling properties. For behavior aggregate (BA) classification, the DSCP, DSCP IPv6, or IP precedence bits of the IP header convey the behavior aggregate class information.

Classifiers on Network-Facing Interfaces

When a packet enters an egress switch from the network, you can map it to a forwarding class and a loss priority based on its DSCP code points by applying a classifier to the Layer 3 physical interface. The forwarding class is mapped to an egress queue and to scheduling properties.

By default, before a packet exits the network-facing interface on the ingress switch, the switch copies the DSCP code points from the packet header into the VXLAN header, so the DSCP code points are not rewritten. However, you can configure a rewrite rule on the egress interface (network-facing interface) of the ingress switch if you want to change the value of the DSCP code points.

On the egress switch, the network-facing interface reads the DSCP code points from the VXLAN header and assigns packets to forwarding classes (which are mapped to egress queues) and loss priorities based on the DSCP code points.

NOTE: You cannot classify traffic using an IEEE 802.1p classifier.

Rewrite Rules on OVSDB-Managed VXLAN Interfaces

When packets exit a network, edge switches might need to change the CoS settings of the packets. Rewrite rules change the value of the code points in the packet header by rewriting the code points to a different value in the outgoing packet. See "Understanding CoS Rewrite Rules" on page 111 for detailed information about rewrite rules.

On OVSDB-managed VXLAN interfaces, you can apply DSCP rewrite rules to packets on network-facing physical interfaces. You cannot apply rewrite rules to access-facing OVSDB-managed VXLAN interfaces, and you cannot apply rewrite rules to IEEE 802.1p code points on network-facing interfaces.

By default, before a packet exits the network-facing interface on the ingress switch, the switch copies the DSCP code points from the packet header into the VXLAN header, so the DSCP code points are not rewritten. The VXLAN header needs to contain the correct DSCP code points because the network-facing ingress port of the egress switch uses the DSCP code points in the VXLAN header to classify the incoming packets.

If you want to change the value of the DSCP code points before the switch transmits packets across the network to the egress switch, you can configure a DSCP rewrite rule and apply it to the egress (network-facing) interface on the ingress switch.

NOTE: Rewrite rules on OVSDB-managed VXLAN interfaces rewrite only the DSCP code point value in the VXLAN header. Rewrite rules on OVSDB-managed VXLAN interfaces do not rewrite the inner (IP) packet header DSCP code point value, so the DSCP code point value in the IP packet header remains unchanged.

Schedulers on OVSDB-Managed VXLAN Interfaces

Packet scheduling (the allocation of port resources such as bandwidth, scheduling priority, and buffers) on OVSDB-managed VXLAN interfaces uses enhanced transmission selection (ETS) hierarchical port scheduling, the same as other interfaces on the switch.

ETS hierarchical port scheduling allocates port bandwidth to traffic in two tiers. ETS provides better port bandwidth utilization and greater flexibility to allocate port resources to forwarding classes (this equates to allocating port resources to output queues because queues are mapped to forwarding classes) and to groups of forwarding classes called forwarding class sets (fc-sets).

First, ETS allocates port bandwidth to fc-sets (also known as priority groups). Each fc-set consists of one or more forwarding classes that carry traffic that requires similar CoS treatment. The bandwidth each fc-set receives is then allocated to the forwarding classes in that fc-set. Each forwarding class is mapped to an output queue. The scheduling properties of a forwarding class are assigned to the queue to which the forwarding class is mapped. Traffic control profiles control the allocation of port bandwidth to fc-sets. Queue schedulers control the allocation of fc-set bandwidth to forwarding classes. See "Understanding CoS Output Queue Schedulers" on page 313, "Understanding CoS Traffic Control Profiles" on page 372, and "Understanding CoS Hierarchical Port Scheduling (ETS)" on page 407 for detailed information about scheduling.

> **NOTE:** It is important to take into account the overhead due to VXLAN header encapsulation when you calculate the amount of bandwidth to allocate to VXLAN traffic. When a virtual tunnel endpoint (VTEP) encapsulates a packet in VXLAN, the VXLAN header adds 50 bytes to the packet.
>
> When you configure the queue scheduler transmit rate, which is the minimum amount of guaranteed bandwidth allocated to traffic mapped to a particular queue, and the traffic control profile guaranteed rate, which is the minimum amount of guaranteed bandwidth allocated to traffic mapped to a particular priority group (fc-set), be sure to configure a high enough bandwidth allocation to account for the VXLAN header overhead.

RELATED DOCUMENTATION

Configuring CoS on OVSDB-Managed VXLAN Interfaces

On QFX5100 and QFX10000 Series switches, you can configure packet classification, packet scheduling, and packet code point rewrite (rewrite rules) class of service (CoS) features on OVSDB-managed VXLAN interfaces. An OVSDB-managed VXLAN interface uses an OVSDB controller to create and manage the VXLAN interfaces and tunnels.

Classifier, scheduler, and rewrite rule configuration on OVSDB-managed VXLAN interfaces uses the same CLI statements as CoS configuration on regular Ethernet interfaces. However, feature support differs on OVSDB-managed VXLAN interfaces compared to regular Ethernet interfaces in several ways, depending on whether a switch interface is access-facing (connected to servers and other devices accessing the network) or network-facing (connected to the network, for example, switch interfaces that connect to a VXLAN gateway).

- Classifiers—On access-facing ingress interfaces, you can configure either BA or MF DSCP classifiers.

  On network-facing ingress interfaces, you can configure only DSCP classifiers.

- Rewrite rules—On network-facing interfaces, you can configure DSCP rewrite rules. Access-facing interfaces do not support rewrite rules. IEEE 802.1p rewrite rules are not supported.

  > **NOTE:** Rewrite rules rewrite the DSCP code point on the VXLAN header only. Rewrite rules do not rewrite the DSCP code point on the inner packet header. If you do not configure a rewrite rule, by default, the code point value in the packet header is copied into the VXLAN header.

- Schedulers—Egress interfaces use enhanced transmission selection (ETS) hierarchical port scheduling, the same as regular Ethernet interfaces, and the same features are supported. You can configure packet scheduling on access-facing and network-facing egress interfaces.

For more information about CoS feature support on OVSDB-managed VXLAN interfaces, (see "Understanding CoS on OVSDB-Managed VXLAN Interfaces" on page 63.)

> **NOTE:** This topic covers CoS configuration on OVSDB-managed VXLAN interfaces. It does not cover OVSDB or VXLAN configuration. See *Understanding Dynamically Configured VXLANs in an OVSDB Environment* for information about OVSDB-managed VXLANs.

NOTE: If you do not configure CoS on an interface, the interface uses the default CoS properties. If you configure some CoS properties on an interface, the interface uses the configured CoS for those properties and default CoS for unconfigured properties. The only difference in the default settings on OVSDB-managed VXLAN interfaces is that if you do not configure a rewrite rule, by default, the code point value in the packet header is copied into the VXLAN header. (There is no default rewrite rule on other interfaces.) See "Understanding Default CoS Scheduling and Classification" on page 298 for information about default scheduler and classifier settings.

The following three procedures show how to configure classifiers, rewrite rules, and ETS hierarchical port scheduling on OVSDB-managed VXLAN interfaces.

You can configure classifiers based on the default classifier or a previously configured classifier, or you can create completely new classifiers that do not use any default values. This example is for a network interface.

1. To configure a classifier on an ingress interface using the default classifier or a previously configured classifier as a template (the switch uses the default values for any values that you do not explicitly configure), include the **import** statement and specify **default** or the classifier name as the classifier to import, and associate the classifier with a forwarding class, a loss priority, and one or more code points:

   `[edit class-of-service classifiers]`

   `user@switch#` **set (dscp) import (default | *classifier-name*) forwarding-class *forwarding-class-name* loss-priority *level* code-points *code***

   To create a classifier that is not based on the default classifier or a previously existing classifier, create a new classifier and associate it with a forwarding class, a loss priority, and one or more code points:

   `[edit class-of-service classifiers]`

   `user@switch#` **set (dscp | c-dscp) forwarding-class *forwarding-class-name* loss-priority *level* low-code-point *code***

   NOTE: On network-facing ingress interfaces, only BA DSCP classifiers are supported. Access-facing ingress interfaces support both BA and MF DSCP classification.

2. Apply the classifier to one or more OVSDB-managed VXLAN interfaces on the switch:

   `[edit class-of-service interfaces]`

   `user@switch#` **set interfaces *interface-name* classifiers dscp |c-dscp**

You can configure rewrite rules based on the default rewrite rule or a previously existing rewrite rule. The default rewrite rule writes the inner packet header value to the VXLAN outer header, or you can create completely new classifiers that do not use any default values. You can configure rewrite rules only on network-facing interfaces, and the only supported rewrite rules are DSCP rewrite rules.

1. To configure a rewrite rule on a network-facing egress interface using the default rewrite rule or a previously configured rewrite rule as a template (the switch uses the default values for any values that you do not explicitly configure), include the **import** statement and specify **default** or the rewrite rule name as the rewrite rule to import, and associate the rewrite rule with a forwarding class, a loss priority, and one or more code points:

   `[edit` class-of-service rewrite-rules`]`

   `user@switch#` **set dscp *rewrite-name* import (*rewrite-name* | default) forwarding-class *forwarding-class-name* loss-priority *level* code-points [*aliases*] [*bit-patterns*]**

   To create a rewrite rule that is not based on the default rewrite rule or a previously existing rewrite rule, create a new rewrite rule and associate it with a forwarding class, a loss priority, and one or more code points:

   `[edit class-of-service rewrite-rules]`

   `user@switch#` **set dscp *rewrite-name* forwarding-class *forwarding-class-name* loss-priority *level* code-points [*aliases*] [*bit-patterns*]**

   NOTE: Rewrite rules are not supported on access-facing interfaces.

2. Apply the rewrite rule to one or more OVSDB-managed VXLAN interfaces on the switch:

   `[edit class-of-service` interfaces`]`

   `user@switch#` **set *interface-name* unit *unit* rewrite-rules dscp *rewrite-name***

ETS hierarchical port scheduling allocates port bandwidth to traffic in two tiers. ETS provides better port bandwidth utilization and greater flexibility to allocate port resources to forwarding classes (this equates to allocating port resources to output queues because queues are mapped to forwarding classes) and to groups of forwarding classes called forwarding class sets (fc-sets).

First, ETS allocates port bandwidth to fc-sets (also known as priority groups). Each fc-set consists of one or more forwarding classes that carry traffic that requires similar CoS treatment. The bandwidth each fc-set receives is then allocated to the forwarding classes in that fc-set. Each forwarding class is mapped to an output queue. The scheduling properties of a forwarding class are assigned to the queue to which the forwarding class is mapped. Traffic control profiles control the allocation of port bandwidth to fc-sets. Queue schedulers control the allocation of fc-set bandwidth to forwarding classes. See "Understanding CoS Output Queue Schedulers" on page 313, "Understanding CoS Traffic Control Profiles" on page 372, and "Understanding CoS Hierarchical Port Scheduling (ETS)" on page 407 for detailed information about scheduling.

Schedulers define the CoS properties of the output queues mapped to forwarding classes. After you configure a scheduler, you use a scheduler map to map the scheduler to one or more forwarding classes. Mapping the scheduler to a forwarding class applies the scheduling properties to the traffic in the forwarding class.

Schedulers define the following characteristics for the forwarding classes (queues) mapped to the scheduler:

- **transmit-rate**—Minimum bandwidth, also known as the *committed information rate (CIR)*, set as a percentage rate or as an absolute value in bits per second. 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.

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

  NOTE: You cannot configure a transmit rate for strict-high priority queues. Queues (forwarding classes) with a configured transmit rate cannot be included in an fc-set that has strict-high priority queues.

- **shaping-rate**—Maximum bandwidth, also known as the *peak information rate (PIR)*, set as a percentage rate or as an absolute value in bits per second.

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

- **priority**—One of two bandwidth priorities that queues associated with a scheduler can receive:
  - **low**—The scheduler has low priority.
  - **strict-high**—The scheduler has strict-high priority. You can configure only one queue as a strict-high priority queue. Strict-high priority allocates the scheduled 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.

    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.
- **drop-profile-map**—Drop profile mapping to a loss priority and protocol to apply weighted random early detection (WRED) packet drop characteristics to the scheduler.

  NOTE: If ingress port congestion occurs because of egress port congestion, apply a drop profile to the traffic on the congested egress port so that traffic is dropped at the egress interface instead of at the ingress interface. (Ingress interface congestion can affect uncongested ports when an ingress port transmits traffic to both congested and uncongested egress ports.)
- **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**—Enables ECN 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.

A traffic control profile defines the CoS properties of an fc-set, and the amount of port resources allocated to the group of forwarding classes (queues) in the fc-set. After you configure a traffic control profile, you apply it (with an associated fc-set) to an interface, to configure scheduling on that interface for traffic that belongs to the forwarding classes in the fc-set .

A traffic control profile defines the following characteristics for the fc-set (priority group) mapped to the traffic control profile when you apply traffic control profile and fc-set to an interface:

- **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 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 in each fc-set.

The following procedure shows how to configure scheduler properties, map schedulers to forwarding classes, map forwarding classes to fc-sets, configure traffic control profile properties, and apply traffic control profiles and fc-sets to interfaces (to apply the ETS ports scheduling configuration to interfaces).

> **NOTE:** You do not have to explicitly configure all of the scheduler and traffic control profile characteristics. Some characteristics are disabled by default, such as ECN, and should only be enabled under certain conditions. You can have a mix of configured CoS properties and default CoS properties.

1. Name the queue scheduler and define the minimum guaranteed bandwidth for the queue:

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

2. Define the maximum bandwidth for the queue:

   ```
   [edit class-of-service schedulers scheduler-name]
   ```

user@switch# **set** shaping-rate **(*rate* | percent *percentage*)**

3. Define the queue priority:

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

4. Define the drop profile using a drop profile map:

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

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

   [edit class-of-service schedulers *scheduler-name*]
   user@switch# **set** buffer-size **percent 20**

6. Enable ECN, if desired (queue should handle best-effort traffic):

   [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***

   This completes the characteristics you can configure in a scheduler, and scheduler mapping to forwarding classes. The next steps show how to configure traffic control profiles.

8. 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*)**

9. 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)
```

10. Attach a scheduler map to the traffic control profile; the scheduler map associates the schedulers and forwarding classes (queues) in the scheduler map with the traffic control profile:

    ```
    [edit class-of-service traffic-control-profiles traffic-control-profile-name]
    user@switch# set scheduler-map scheduler-map-name
    ```

    This completes the characteristics you can configure in a traffic control profile. The next step shows how to assign forwarding classes to fc-sets.

11. Assign one or more forwarding classes to the fc-set:

    ```
    [edit class-of-service]
    user@switch# set forwarding-class-sets forwarding-class-set-name class forwarding-class-name
    ```

    This completes assigning forwarding classes to fc-sets. The next steps show how to apply ETS hierarchical port scheduling to interfaces.

12. To apply ETS hierarchical port scheduling to interfaces, associate an fc-set and a traffic control profile with interfaces. The fc-set determines the forwarding class(es) and queue(s) that use the specified interface. The traffic control profile determines the amount of port resources allocated to the fc-set, and the mapping of forwarding classes to schedulers in the traffic control profile determines the allocation of fc-set resources to the forwarding classes that are members of the fc-set.

    ```
    user@switch# set interfaces interface-name forwarding-class-set fc-set-name
    output-traffic-control-profile tcp-name
    ```

RELATED DOCUMENTATION

Example: Configuring Queue Schedulers | 325

Example: Configuring Traffic Control Profiles (Priority Group Scheduling) | 385

Example: Configuring CoS Hierarchical Port Scheduling (ETS) | 415

Example: Configuring Unicast Classifiers | 100

Configuring CoS WRED Drop Profiles | 261

Example: Configuring ECN | 284

Understanding CoS on OVSDB-Managed VXLAN Interfaces | 63

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 only to logical interfaces; on some switches, you can 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.

> **NOTE:** OCX Series switches and NFX250 Network Services platform do not support congestion notification profiles.

- 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 switches 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 only to logical interfaces; on some switches, you can 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**.

RELATED DOCUMENTATION

Monitoring Interfaces That Have CoS Components

Understanding Junos CoS Components | 17

CHAPTER 3

CoS Code-Point Aliases

IN THIS CHAPTER

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" on page 79.

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 30 on page 80 shows the default mapping of code-point aliases to IEEE code points.

Table 30: 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 31 on page 80 shows the default mapping of code-point aliases to DSCP and DSCP IPv6 code points.

Table 31: 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

Table 31: Default DSCP and DSCP IPv6 Code-Point Aliases *(continued)*

CoS Value Types	Mapping
af33	011110
af41	100010
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 | 17
Defining CoS Code-Point Aliases | 82

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 | 82
Understanding CoS Code-Point Aliases | 79

Monitoring CoS Code-Point Value Aliases

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 32 on page 83 summarizes key output fields for CoS value aliases.

Table 32: Summary of Key CoS Value Alias Output Fields

Field	Values
Code point type	Type of the CoS value: • **dscp**—Examines Layer 3 packet headers for IP packet classification. • **dscp-ipv6**—Examines Layer 3 packet headers for IPv6 packet classification. • **ieee-802.1**—Examines Layer 2 packet headers for packet classification. • **exp**—Examines MPLS packet headers for packet classification. **NOTE:** OCX Series switches do not support MPLS.
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 | 82

CHAPTER 4

CoS Classifiers

IN THIS CHAPTER

Understanding CoS Classifiers

IN THIS SECTION

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" on page 132. 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. Except on QFX10000 switches, classifiers and forwarding classes handle either unicast or multidestination traffic. You cannot mix unicast and multidestination traffic in the same classifier or forwarding class. On QFX10000 switches, a classifier can assign both unicast and multidestination traffic to the same forwarding class.

Interfaces and Output Queues

On Gigabit Ethernet interfaces, 10-Gigabit Ethernet interfaces, and link aggregation (LAG) interfaces, 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" on page 116 describes the interaction between classifiers and interfaces in greater detail.

> **NOTE:** On QFX10000 switches you can apply different classifiers to different Layer 3 logical interfaces. You cannot apply classifiers to physical interfaces.

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.

Except on QFX10000 switches, you can configure both 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 (except when you configure a global EXP classifier; in that case, MPLS traffic uses the EXP classifier providing 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).

On QFX10000 switches, you can configure either a DSCP or a DSCP IPv6 classifier and also an IEEE 802.1p classifier on the same interface. IP traffic uses the DSCP or DSCP IPv6 classifier. If you configure an interface as **family mpls**, then the interface uses the default MPLS EXP classifier. If you configure an MPLS EXP classifier, then all MPLS traffic on the switch uses the global EXP classifier. All other traffic uses the IEEE classifier. You can configure up to 64 EXP classifiers with up to 8 entries per classifier (one entry for each forwarding class) and apply them to logical interfaces.

Except on QFX10000 switches, 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 EXP classifier, on QFX10000 switches either the default or the global EXP classifier, specified in this configuration statement to classify MPLS traffic.

Output Queues for Unicast and Multidestination Traffic

NOTE: This section applies to switches except QFX10000.

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:

NOTE: QFX5200 switches support 10 output queues, with 8 queues dedicated to unicast traffic and 2 queues dedicated to 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 multidestination traffic queues. You can apply only multidestination BA classifiers to multidestination queues. A multidestination BA classifier should contain only forwarding classes that are mapped to multidestination queues.

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

Classifier Support by Type

> **NOTE:** This section applies only to QFX10000 switches.

You can configure enough classifiers to handle most, if not all, network scenarios. Table 33 on page 87 shows how many of each type of classifiers you can configure, and how many entries you can configure per classifier.

Table 33: Classifier Support by Classifier Type

Classifier Type	Default Classifier Name	Maximum Number of Classifiers	Maximum Number of Entries per Classifier
IEEE 802.1p (Layer 2)	ieee8021p-default (for ports in trunk mode) ieee8021p-untrust (for ports in access mode)	64	16
DSCP (Layer 3)	dscp-default	64	64
DSCP IPv6 (Layer 3)	dscp-ipv6-default	64	64
EXP (MPLS)	exp-default	64	8
Fixed	There is no default fixed classifier	8	16

The number of fixed classifiers supported (8) equals the number of supported forwarding classes (fixed classifiers assign all incoming traffic on an interface to one forwarding class).

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 multidestination 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 34 on page 88 lists different types of interfaces and the corresponding implicit default BA classifiers.

Table 34: Default BA Classification

Type of Interface	Default BA Classification
Layer 2 interface in trunk mode or, except on QFX10000, tagged-access mode	**ieee8021p-default**
(QFX10000 only) Layer 2 interface in access mode	**ieee8021p-untrusted**
Layer 3 interface	**dscp-default** **dscp-ipv6-default**
(Except QFX10000) Layer 2 interface in access mode	**ieee8021p-untrusted**
(QFX10000 only) MPLS interface	**exp-default**

NOTE: Default BA classifiers assign traffic only to the **best-effort**, **fcoe**, **no-loss**, **network-control**, and, except on QFX10000 switches, **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. For other than QFX10000 switches, this applies to unicast classifiers.

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 on QFX10000 switches you configure an EXP classifier, or on other switches a global EXP classifier, and apply it on interfaces configured as **family mpls**, then MPLS traffic uses that classifier on those interfaces.

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

NOTE: This section applies to switches except QFX10000.

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 classifer. 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 35 on page 90. 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 35: 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)

> **NOTE:** This section applies to switches except QFX10000.

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" on page 132. 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.

RELATED DOCUMENTATION

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

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" on page 108 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.

> **NOTE:** OCX Series switches do not support MPLS EXP classifiers.

On switches except QFX10000 and NFX Series devices, unicast traffic must use different classifiers than multidestination (mulitcast, broadcast, and destination lookup fail) traffic. You use the **multi-destination** statement at the **[edit class-of-service]** hierarhcy level to configure a multidestination BA classifier.

On QFX10000 switches and NFX Series devices, unicast and multidestination traffic use the same classifiers and forwarding classes.

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.

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) | 415
Example: Configuring Unicast Classifiers | 100
Configuring a Global MPLS EXP Classifier | 108
Configuring Rewrite Rules for MPLS EXP Classifiers | 136
Monitoring CoS Classifiers | 108
Understanding CoS Classifiers | 84
Understanding CoS Classifiers
Understanding CoS MPLS EXP Classifiers and Rewrite Rules | 132
Understanding Applying CoS Classifiers and Rewrite Rules to Interfaces | 116
Understanding Applying CoS Classifiers and Rewrite Rules to Interfaces

Example: Configuring Classifiers

IN THIS SECTION

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.

Requirements

This example uses the following hardware and software components:

- One switch.
- Junos OS Release 15.1X53-D10 or later for the QFX Series.

Overview

Junos OS 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 36 on page 97, 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 36: 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
nc	Network-control traffic	Low loss priority code point: **110**	drop

Configuring Classifiers

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
   ```

Verification

IN THIS SECTION

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) | 415

Defining CoS BA Classifiers (DSCP, DSCP IPv6, IEEE 802.1p) | 94

Configuring a Global MPLS EXP Classifier | 108

Configuring Rewrite Rules for MPLS EXP Classifiers | 136

Monitoring CoS Classifiers | 108

Understanding CoS Classifiers | 84

Understanding Applying CoS Classifiers and Rewrite Rules to Interfaces | 116

Example: Configuring Unicast Classifiers

IN THIS SECTION

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.

Requirements

This example uses the following hardware and software components:

- One switch except QFX10000 (this example was tested on a Juniper Networks QFX3500 switch)
- Junos OS Release 11.1 or later for the QFX Series

Overview

Junos OS supports two 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) value or IEEE 802.1p 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.

> **NOTE:** You must assign unicast traffic and multidestination (multicast, broadcast, and destination lookup fail) traffic to different classifiers. One classifier cannot include both unicast and multidestination forwarding classes. A unicast classifier can include only forwarding classes for unicast traffic.

This example describes how to configure a BA classifier called **ba-ucast-classifier** as the default IEEE 802.1 map and apply it to ingress interface **xe-0/0/10**. The BA classifier assigns loss priorities, as shown in Table 37 on page 101, to incoming packets in the four forwarding classes.

You can use the same procedure to set multifield classifiers (except that you use firewall filter rules).

Table 37: ba-ucast-classifier Loss Priority Assignments

Unicast Forwarding Class	For CoS Traffic Type	ba-ucast-classifier Assignment	Packet Drop Attribute
be	Best-effort traffic	Low loss priority code point: **000**	Low loss priority code point: **000**
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**	Low loss priority code point: **100**
nc	Network-control traffic	Low loss priority code point: **110**	drop

Configuring Unicast Classifiers

To configure a unicast IEEE 802.1 BA classifier named **ba-ucast-classifier** as the default IEEE 802.1 map:

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-ucast-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-ucast-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-ucast-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-ucast-classifier forwarding-class nc loss-priority low code-points 110**

5. Apply the unicast 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-ucast-classifier**

Verification

IN THIS SECTION

To verify the unicast classifier configuration, perform these tasks:

Verifying the Unicast Classifier Configuration

Purpose

Verify that you configured the unicast 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-ucast-classifier**:

`user@switch>` **show configuration class-of-service classifiers ieee-802.1 ba-ucast-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 unicast classifier **ba-ucast-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-ucast-classifier;
    }
}
```

RELATED DOCUMENTATION

Example: Configuring CoS Hierarchical Port Scheduling (ETS) | **415**

Example: Configuring Multidestination (Multicast, Broadcast, DLF) Classifiers | **104**

Defining CoS BA Classifiers (DSCP, DSCP IPv6, IEEE 802.1p) | **94**

Configuring a Global MPLS EXP Classifier | **108**

Defining CoS BA Classifiers (DSCP, DSCP IPv6, IEEE 802.1p) | **94**

Monitoring CoS Classifiers | **108**

Understanding CoS Classifiers | **84**

Understanding CoS Classifiers

Understanding Applying CoS Classifiers and Rewrite Rules to Interfaces | **116**

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

IN THIS SECTION

Packet classification associates incoming packets with a particular CoS servicing level. Behavior aggregate (BA) classifiers examine the CoS value in the packet header to determine 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 incoming CoS value.

Beginning with Junos OS Release 17.1, EX4300 switches support multidestination classifiers. On EX4300 switches, you can apply multidestination classifiers globally or to a specific interface. If you apply multidestination classifiers both globally and to a specific interface, the classifications on the interface take precedence.

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, except on an EX4300 switch.

Unicast and multidestination traffic must use different classifiers.

Requirements

This example uses the following hardware and software components:

- One switch (this example was tested on a Juniper Networks QFX3500 Switch)
- Junos OS Release 11.1 or later for the QFX Series.

Overview

Junos OS 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 CoS value.

- 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 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.

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.

> **NOTE:** You must assign unicast traffic and multicast traffic to different classifiers. One classifier cannot include both unicast and multicast forwarding classes. A multidestination classifier can include only forwarding classes for multicast traffic.

The following example describes how to configure a BA classifier called **ba-mcast-classifier**, which is applied to all of the switch interfaces. The BA classifier assigns loss priorities, as shown in Table 38 on page 105, to incoming packets in the multidestination forwarding class.

You can also use firewall filters to set multifield classifiers.

Table 38: BA-mcast-classifier Loss Priority Assignments

Multicast Forwarding Class	Traffic Type	ba-mcast-classifier Assignment
mcast	Best-effort multicast traffic	Low loss priority code point: **000**

Configuring Multidestination Classifiers

To configure a multicast IEEE 802.1 BA classifier named **ba-mcast-classifier**:

1. Associate code point **000** with forwarding class **mcast** and loss priority **low**:

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

2. Configure the classifier as a multidestination classifier:

```
[edit class-of-service]
user@switch# set multi-destination classifiers ieee-802.1 ba-mcast-classifier
```

Verification

IN THIS SECTION

To verify the multidestination classifier configuration, perform these tasks:

Verifying the IEEE 802.1 Multidestination Classifier

Purpose

Verify that the classifier **ba-mcast-classifier** is configured as the IEEE 802.1 multidestination classifier:

Action

Verify the results of the classifier configuration using the operational mode command **show configuration class-of-service multi-destination classifiers ieee-802.1**:

`user@switch>` **show configuration class-of-service multi-destination classifiers ieee-802.1**

```
ba-mcast-classifier;
```

Verifying the Multidestination Classifier Configuration

Purpose

Verify that you configured the multidestination 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-mcast-classifier**:

`user@switch>` **show configuration class-of-service classifiers ieee-802.1 ba-mcast-classifier**

```
    forwarding-class mcast {
        loss-priority low code-points 000;
    }
```

Release History Table

Release	Description
17.1	Beginning with Junos OS Release 17.1, EX4300 switches support multidestination classifiers.

RELATED DOCUMENTATION

Example: Configuring Unicast Classifiers | 100
Defining CoS BA Classifiers (DSCP, DSCP IPv6, IEEE 802.1p) | 94
Monitoring CoS Classifiers | 108
Understanding CoS Classifiers | 84
Understanding CoS Classifiers
Understanding Applying CoS Classifiers and Rewrite Rules to Interfaces | 116

Understanding Host Inbound Traffic Classification

The destination address of traffic that enters the switch can be an external device such as another switch, a router, or a server, or the destination can be the host (the switch Routing Engine or CPU). When the destination is an external device, the DSCP and IEEE 802.1p code-point bits of incoming traffic are preserved as the traffic travels through the switch to the egress port. At the egress port, the code-point bits are either preserved when the packets are sent to the next hop or they are rewritten according to the rewrite rule attached to the egress interface.

When the destination of incoming traffic is the host, DSCP bits are preserved. However, IEEE 802.1p bits are not preserved. The IEEE 802.1p bits of traffic destined for the host are set to zero (0). This does not affect system behavior because the switch prioritizes traffic destined for the host based on the protocol type. For example, the switch gives a higher priority to BPDU traffic than to ping traffic.

Configuring a Global MPLS EXP Classifier

EXP packet classification associates incoming packets with a particular MPLS CoS servicing level. EXP behavior aggregate (BA) classifiers examine the MPLS EXP value in the packet header to determine the CoS settings applied to the packet. EXP BA classifiers allow you to set the forwarding class and loss priority of an MPLS packet based on the incoming CoS value.

You can configure up to 64 EXP classifiers, however, the switch uses only one MPLS EXP classifier as a global classifier, which is applied only on interfaces configured as **family mpls**. All **family mpls** switch interfaces use the global EXP classifier to classify MPLS traffic.

There is no default EXP classifier. If you want to classify incoming MPLS packets using the EXP bits, you must configure a global EXP classifier. The global classifier applies to all MPLS traffic on all **family mpls** interfaces.

If a global EXP classifier is configured, MPLS traffic on **family mpls** interfaces uses the EXP classifier. 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. DSCP classifiers are not applied to MPLS traffic.

To configure an MPLS EXP classifier using the CLI:

1. Create an EXP classifier and associate it with a forwarding class, a loss priority, and a code point:

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

2. Apply the EXP classifier to the switch interfaces:

   `[edit class-of-service]`
   `user@switch#` **set system-defaults classifiers exp** ***classifier-name***

Monitoring CoS Classifiers

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 39 on page 109 summarizes key output fields for CoS classifiers.

Table 39: Summary of Key CoS Classifier Output Fields

Field	Values
Classifier	Name of a classifier.
Code point type	Type of classifier: • **dscp**—All classifiers of the DSCP type. • **ieee-802.1**—All classifiers of the IEEE 802.1 type. • **ieee-mcast**—All classifiers of the IEEE 802.1 multicast type. **NOTE:** QFX10000 switches do not use different classifiers for unicast and multidestination (multicast, broadcast, destination lookup fail) traffic, so multicast-specific classifiers are not supported. • **exp**—All classifiers of the MPLS exp type. **NOTE:** OCX Series switches do not support MPLS.
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.

CHAPTER 5

CoS Rewrite Rules

IN THIS CHAPTER

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.

> **NOTE:** OCX Series switches do not support MPLS and do not support EXP rewrite rules.

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 40 on page 112 shows how many of each type of rewrite rules you can configure, and how many entries you can configure per rewrite rule.

Table 40: 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

Table 40: Configuring Rewrite Rules *(continued)*

Rewrite Rule Type	Maximum Number of Rewrite Rules	Maximum Number of Entries per Rewrite Rule
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.

NOTE: OCX Series witches do not support IRBs/RVIs.

RELATED DOCUMENTATION

Understanding Junos CoS Components | **17**

Defining CoS Rewrite Rules | **114**

Configuring Rewrite Rules for MPLS EXP Classifiers | **136**

Defining CoS Rewrite Rules

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: OCX Series switches do not support MPLS, so they do not support EXP rewrite rules.

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.

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.

RELATED DOCUMENTATION

Monitoring CoS Rewrite Rules | 137
Configuring Rewrite Rules for MPLS EXP Classifiers | 136
Understanding CoS Rewrite Rules | 111
Understanding CoS MPLS EXP Classifiers and Rewrite Rules | 132

Understanding Applying CoS Classifiers and Rewrite Rules to Interfaces

IN THIS SECTION

At ingress interfaces, classifiers group incoming traffic into classes based on the IEEE 802.1p, DSCP, or MPLS EXP class of service (CoS) code point bits 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 41 on page 116 shows the supported types of classifiers and rewrite rules supports:

Table 41: 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.
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.

Table 41: Supported Classifiers and Rewrite Rules *(continued)*

Classifier or Rewrite Rule Type	Description
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, QFX3500, and QFX3600 switches, and QFabric systems, use one global EXP classifier on all **family mpls** switch interfaces. QFX10000 switches do not support global EXP classifiers. You can apply the same EXP classifier or different EXP classifiers to different **family mpls** interfaces.
DSCP multidestination classifier (also used for IPv6 multidestination traffic) **NOTE:** 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 **NOTE:** 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, EX4600, QFX3500, and QFX3600 switches, and on QFabric systems, unlike DSCP and IEEE 802.1p classifiers, EXP classifiers are global and apply only to all interfaces that are configured as **family mpls**. On QFX10000 switches, you apply EXP classifiers to individual logical interfaces, and different interfaces can use different EXP classifiers.

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 multidestination traffic, multidestination classifiers are global and apply to all interfaces; you cannot apply a multidestination 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

IN THIS SECTION

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)

You can apply CoS classifiers and rewrite rules only to the following interfaces:

- Layer 2 logical interface unit 0

  > **NOTE:** On a Layer 2 interface, the CoS you configure on logical interface unit 0 applies to all of the logical units on that interface.

- On QFX5100, QFX5200, EX4600, QFX3500, and QFX3600 switches, and on QFabric systems, Layer 3 physical interfaces if at least one logical Layer 3 interface is configured on the physical interface

  > **NOTE:** 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.

- On QFX10000 switches, Layer 3 logical interfaces. You can apply different classifiers and rewrite rules to different Layer 3 logical interfaces.

Ethernet Interface Support for Most QFX Series Switches, and QFabric Systems

You cannot apply classifiers or rewrite rules to Layer 2 physical interfaces or to Layer 3 logical interfaces. Table 42 on page 120 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 42: Ethernet Interface Support for Classifier and Rewrite Rule Configuration (QFX5100, QFX5200, EX4600, QFX3500, and QFX3600 Switches, and QFabric Systems)

CoS Classifiers and Rewrite Rules	Layer 2 Physical Interfaces	Layer 2 Logical Interface (Unit 0 Only)	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 mutidestination and DSCP multidestination classifiers are applied to all interfaces and cannot be applied to individual interfaces. No DSCP IPv6 multidestination classifier is supported. IPv6 multidestination traffic uses the DSCP multidestination classifier.

Ethernet Interface Support for QFX10000 Switches

You cannot apply classifiers or rewrite rules to Layer 2 or Layer 3 physical interfaces. 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. Table 43 on page 121 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 43: 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 RVI 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.

> **NOTE:** Tagged-access mode is available only on QFX3500 and QFX3600 devices when used as standalone switches or as QFabric system Node devices.

- 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 multidestination classifier is the IEEE 802.1p multidestination classifier.
- There is no default MPLS EXP classifier. If you want to classify MPLS traffic using EXP bits on these switches, on QFX10000 switches, configure an EXP classifier and apply it to a logical interface that is configured as **family mpls**. On QFX5100, QFX5200, EX4600, QFX3500 and QFX3600 switches, and on QFabric systems, configure an EXP classifier and configure it as the global system default 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

IN THIS SECTION

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 (QFX5200, QFX5100, EX4600, QFX3500, and QFX3600 Switches, and QFabric Systems)

QFX10000 switches do not support configuring classifiers on physical 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 (All Switches)

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. Behaviors for applying classifiers to physical interfaces do not pertain to QFX10000 switches.

- 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. On QFX10000 switches, you can apply different EXP classifiers to different interfaces.

  However, on On QFX5200, QFX5100, EX4600, QFX3500, and QFX3600 switches, and on QFabric systems, the switch uses only one MPLS EXP classifier as a global classifier on all **family mpls** interfaces. After you configure an MPLS EXP classifier, you can configure it as the global EXP classifier by including the EXP classifier in the **[edit class-of-service system-defaults classifiers exp]** hierarchy level.

  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

IN THIS SECTION

On QFX5100, QFX5200, EX4600, QFX3500, and QFX3600 switches used as standalone switches or as QFabric system Node devices, you can apply classifiers and rewrite rules only on Layer 2 logical interface unit 0 and Layer 3 physical interfaces (if the Layer 3 physical interface has at least one defined logical interface). On QFX10000 switches, you can apply classifiers and rewrite rules only to Layer 2 logical interface unit 0 and to Layer 3 logical interfaces. 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:** On QFX5100, QFX5200, EX4600, QFX3500, and QFX3600 switches used as standalone switches or as QFabric system Node devices, EXP classifiers, are global and apply to all switch interfaces. See "Defining CoS BA Classifiers (DSCP, DSCP IPv6, IEEE 802.1p)" on page 94 for how to configure multidestination classifiers and see "Configuring a Global MPLS EXP Classifier" on page 108 for how to configure EXP classifiers.
>
> On switches that use different classifiers for unicast and multidestination traffic, multidestination 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.

> **NOTE:** QFX10000 switches cannot be misconfigured in the following two ways because you can configure classifiers only on logical interfaces. Only switches that allow classifier configuration on physical and logical interfaces can experience the following misconfigurations.

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.

QFX5100, QFX5200, EX4600, QFX3500, and QFX3600 Switch Scenarios

The following scenarios also apply the QFX5100, QFX5200, EX4600, QFX3500, and QFX3600 switches when they are used as QFabric system Node devices.

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
   ```

RELATED DOCUMENTATION

Troubleshooting an Unexpected Rewrite Value

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 | 807

Understanding CoS MPLS EXP Classifiers and Rewrite Rules

IN THIS SECTION

You can use class of service (CoS) within MPLS networks to prioritize certain types of traffic during periods of congestion by applying packet classifiers and rewrite rules to the MPLS traffic. MPLS classifiers are global and apply to all interfaces configured as **family mpls** interfaces.

When a packet enters a customer-edge interface on the ingress provider edge (PE) switch, the switch associates the packet with a particular CoS servicing level before placing the packet onto the label-switched path (LSP). The switches within the LSP utilize the CoS value set at the ingress PE switch to determine the CoS service level. The CoS value embedded in the classifier is translated and encoded in the MPLS header by means of the experimental (EXP) bits.

EXP classifiers map incoming MPLS packets to a forwarding class and a loss priority, and assign MPLS packets to output queues based on the forwarding class mapping. EXP classifiers are behavior aggregate (BA) classifiers.

EXP rewrite rules change (rewrite) the CoS value of the EXP bits in outgoing packets on the egress queues of the switch so that the new (rewritten) value 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.

NOTE: On QFX5200, QFX5100, QFX3500, QF3600, and EX4600 switches, and on QFabric systems, there is no default EXP classifier. If you want to classify incoming MPLS packets using the EXP bits, you must configure a global EXP classifier. The global EXP classifier applies to all MPLS traffic on interfaces configured as **family mpls**.

On QFX10000 switches, there is a no default EXP classifier. If you want to classify incoming MPLS packets using the EXP bits, you must configure EXP classifiers and apply them to logical interfaces configured as **family mpls**. (You cannot apply classifiers to physical interfaces.). You can configure up to 64 EXP classifiers.

There is no default EXP rewrite rule. If you want to rewrite the EXP bit value at the egress interface, you must configure EXP rewrite rules and apply them to logical interfaces.

EXP classifiers and rewrite rules are applied only to interfaces that are configured as **family mpls** (for example, **set interfaces xe-0/0/35 unit 0 family mpls**.)

This topic includes:

EXP Classifiers

On QFX5200, QFX5100, EX4600, QFX3500, and QFX3600 switches, and on QFabric systems, unlike DSCP and IEEE 802.1p BA classifiers, EXP classifiers are global to the switch and apply to all switch interfaces that are configured as **family mpls**. On QFX10000 switches, you apply EXP classifiers to individual logical interfaces, and different interfaces can use different EXP classifiers.

When you configure and apply an EXP classifier, MPLS traffic on all **family mpls** interfaces uses the EXP classifier, even on interfaces that also have a fixed classifier. If an interface has both an EXP classifier and a fixed classifier, the EXP classifier is applied to MPLS traffic and the fixed classifier is applied to all other traffic.

Also unlike DSCP and IEEE 802.1p BA classifiers, there is no default EXP classifier. If you want to classify MPLS traffic based on the EXP bits, you must explicitly configure an EXP classifier and apply it to the switch interfaces. Each EXP classifier has eight entries that correspond to the eight EXP CoS values (0 through 7, which correspond to CoS bits 000 through 111).

You can configure up to 64 EXP classifiers.

However, on QFX5200, QFX5100, EX4600, and legacy CLI switches, 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 that classifier as the global EXP classifier by including the EXP classifier in the **[edit class-of-service system-defaults classifiers exp]** hierarchy level. All switch interfaces configured as **family mpls** use the global EXP classifier to classify MPLS traffic.

On these switches, only one EXP classifier can be configured as the global EXP classifier at any time. If you want to change the global EXP classifier, delete the global EXP classifier configuration (use the **user@switch# delete class-of-service system-defaults classifiers exp** configuration statement), then configure the new global EXP classifier.

QFX10000 switches do not support global EXP classifiers. You can configure one EXP classifier and apply it to multiple logical interfaces, or configure multiple EXP classifiers and apply different EXP classifiers to different logical interfaces.

If an EXP classifier is not configured, then if a fixed classifier is applied to the interface, the MPLS traffic uses the fixed classifier. (Switches that have a default EXP classifier use the default classifier.) If no EXP classifier and no fixed classifier are applied to the interface, MPLS traffic is treated as best-effort traffic using the 802.1 default untrusted classifier. DSCP classifiers are not applied to MPLS traffic.

On QFX5200, QFX5100, EX4600, and legacy CLI switches, 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.

NOTE: The switch uses only the outermost label of incoming EXP packets for classification.

NOTE: MPLS packets with 802.1Q tags are not supported.

EXP Rewrite Rules

As MPLS packets enter or exit a network, edge switches might be required to alter the class-of-service (CoS) settings of the packets. EXP rewrite rules set the value of the EXP CoS bits within the header of the outgoing MPLS packet on **family mpls** interfaces. Each rewrite rule reads the current forwarding class and loss priority associated with the packet, locates the chosen CoS value from a table, and writes that CoS value into the packet header, replacing the old CoS value. EXP rewrite rules apply only to MPLS traffic.

EXP rewrite rules apply only to logical interfaces. You cannot apply EXP rewrite rules to physical interfaces.

There are no default EXP rewrite rules. If you want to rewrite the EXP value in MPLS packets, you must configure EXP rewrite rules and apply them to logical interfaces. If no rewrite rules are applied, all MPLS labels that are pushed have a value of zero (0). The EXP value remains unchanged on MPLS labels that are swapped.

You can configure up to 64 EXP rewrite rules, but you can only apply 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.

You can apply an EXP rewrite rule to an interface that has a DSCP, DSCP IPv6, or IEEE 802.1p rewrite rule. Only MPLS traffic uses the EXP rewrite rule. MPLS traffic does not use DSCP or DSCP IPv6 rewrite rules.

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.

Schedulers

The schedulers for using CoS with MPLS are the same as for the other CoS configurations on the switch. Default schedulers are provided only for the best-effort, fcoe, no-loss, and network-control default forwarding classes. If you configure a custom forwarding class for MPLS traffic, you need to configure a scheduler to support that forwarding class and provide bandwidth to that forwarding class.

Configuring Rewrite Rules for MPLS EXP Classifiers

You configure EXP rewrite rules to alter CoS values in outgoing MPLS packets on the outbound **family mpls** interfaces of a switch to match 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 an EXP 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 a logical **family mpls** interface. EXP rewrite rules can only be enabled on logical **family mpls** interfaces, not on physical interfaces or on interfaces of other family types. You can also apply an existing EXP rewrite rule on a logical interface.

> **NOTE:** There are no default rewrite rules.

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 **family mpls** 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:** 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.

To create an EXP rewrite rule for MPLS traffic and enable it on a logical interface:

1. Create an EXP rewrite rule:

   `user@switch#` **set** class-of-service rewrite-rules exp ***rewrite-rule-name*** forwarding-class ***forwarding-class-name*** loss-priority ***level*** code-points [***aliases***] [***bit-patterns***]

   For example, to configure an EXP rewrite rule named **exp-rr-1** for a forwarding class named **mpls-1** with a loss priority of **low** that rewrites the EXP code point value to **001**:

user@switch# **set class-of-service rewrite-rules exp exp-rr-1 forwarding-class mpls-1 loss-priority low code-points 001**

2. Apply the rewrite rule to a logical interface:

   user@switch # **set class-of-service interfaces *interface-name* unit *logical-unit* rewrite-rules exp *rewrite-rule-name***

   For example, to apply a rewrite rule named **exp-rr-1** to logical interface **xe-0/0/10.0**:

   user@switch# **set class-of-service interfaces xe-0/0/10 unit 0 rewrite-rules exp exp-rr-1**

   > **NOTE:** In this example, all forwarding classes assigned to port xe-0/0/10 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 interface.

Monitoring CoS Rewrite Rules

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 44 on page 138 summarizes key output fields for CoS rewrite rules.

Table 44: Summary of Key CoS Rewrite Rule Output Fields

Field	Values
Rewrite rule	Name of the rewrite rule.
Code point type	Rewrite rule type: • **dscp**—For IPv4 DiffServ traffic. • **dscp-ipv6**—For IPv6 Diffserv traffic. • **ieee-802.1**—For Layer 2 traffic. • **exp**—For MPLS traffic. **NOTE:** OCX Series switches do not support MPLS.
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 | 114

CHAPTER 6

CoS Forwarding Classes and Forwarding Class Sets

IN THIS CHAPTER

Understanding CoS Forwarding Classes

IN THIS SECTION

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.

Except on NFX Series devices, a classifier must associate each packet with one of the following four (QFX10000 switches) or five (other switches) 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—(Except QFX10000) Delivery of multidestination (multicast, broadcast, and destination lookup fail) packets.

On NFX Series devices, a classifier must associate each packet with one of the following four default forwarding classes or with a user-configured forwarding class to assign an output queue to the packet:

- 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.

The switch supports up to eight (QFX10000 and NFX Series devices), 10 (QFX5200 switches), or 12 (other switches) 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**.

On QFX10000 and NFX Series devices, unicast and multidestination (multicast, broadcast, and destination lookup fail) traffic use the same forwarding classes and output queues.

Except on QFX10000 and NFX Series devices, a switch supports 8 queues for unicast traffic (queues 0 through 7) and 2 (QFX5200 switches) or 4 (other switches) output queues for multidestination traffic (queues 8 through 11). Forwarding classes mapped to unicast queues are associated with unicast traffic, and forwarding classes mapped to multidestination queues are associated with multidestination traffic. You cannot map unicast and multidestination 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 45 on page 141 shows the four default forwarding classes that apply to all switches but not NFX Series devices. Except on QFX10000, these forwarding classes apply to unicast traffic. 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 45: Default Forwarding Classes

Forwarding Class Name	Default Queue Mapping	Comments
best-effort	0	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. By default, this is a lossy forwarding class with a packet drop attribute of **drop**.
fcoe	3	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. **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.
no-loss	4	By default, this is a lossless forwarding class with a packet drop attribute of **no-loss**.
network-control	7	The software delivers packets in this service class with a high priority. (These packets are not delay-sensitive.) 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. By default, this is a lossy forwarding class with a packet drop attribute of **drop**.

NOTE: Table 46 on page 142 applies only to multidestination traffic except on QFX10000 switches and NFX Series devices.

Table 46: Default Forwarding Classes for Multidestination Packets

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

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

IN THIS SECTION

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.

- (Except QFX10000 and NFX Series devices) Only unicast forwarding classes can be mapped to unicast queues (0 through 7), and only multidestination forwarding classes can be mapped to multidestination queues (8 through 11).
- (Except QFX10000 and NFX Series devices) Strict-high priority queues cannot be mapped to multidestination 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: Except on QFX10000 and NFX Series devices, 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 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

On QFX10000 switches and NFX Series devices, 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 or QFabric system Node device to other devices cannot exceed 300 meters.
- The internal cable length from the QFabric system Node device to the QFabric system Interconnect device cannot exceed 150 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: QFX10002-60C does not support PFC and lossless queues; that is, default lossless queues (fcoe and no-loss) will be lossy queues.

NOTE: Junos OS Release 12.2 introduces changes to the way lossless forwarding classes (the **fcoe** and **no-loss** forwarding classes) are handled.

In Junos OS Release 12.1, both explicitly configuring the **fcoe** and **no-loss** forwarding classes, and using the default configuration for these forwarding classes, resulted in the same lossless behavior for traffic mapped to those forwarding classes.

However, in Junos OS Release 12.2, 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.

If your CoS configuration from Junos OS Release 12.1 or earlier includes the explicit configuration of the **fcoe** or the **no-loss** forwarding class, then when you upgrade to Junos OS Release 12.2, those forwarding classes are not lossless. To preserve the lossless treatment of these forwarding classes, delete the explicit **fcoe** and **no-loss** forwarding class configuration before you upgrade to Junos OS Release 12.2.

See *Overview of CoS Changes Introduced in Junos OS Release 12.2* for detailed information about this change and how to delete an existing lossless configuration.

In Junos OS Release 12.3, the default behavior of the **fcoe** and **no-loss** forwarding classes is the same as in Junos OS Release 12.2. However, in Junos OS Release 12.3, you can configure up to six lossless forwarding classes. All explicitly configured lossless forwarding classes must include the new **no-loss** packet drop attribute or the forwarding class is lossy.

RELATED DOCUMENTATION

Overview of CoS Changes Introduced in Junos OS Release 12.2
Understanding Junos CoS Components | 17
Understanding CoS Packet Flow | 23
Understanding CoS Flow Control (Ethernet PAUSE and PFC) | 197
Example: Configuring Forwarding Classes | 157
Defining CoS Forwarding Classes | 146

Defining CoS Forwarding Classes

Forwarding classes allow you to group packets for transmission. The switch supports a total of eight (QFX10000 and NFX Series devices), 10 (QFX5200 switches), or 12 (other switches) forwarding classes. To forward traffic, you map (assign) the forwarding classes to output queues.

The QFX10000 switches and NFX Series devices have eight output queues, queues 0 through 7. These queues support both unicast and multidestination traffic.

Except on QFX10000 and NFX Series devices, the switch has 10 output queues (QFX5200) or 12 output queues (other switches). 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:

> **NOTE:** Except on QFX10000, these are the default unicast forwarding classes.

- **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

> **NOTE:** QFX10002-60C does not support PFC and lossless queues; that is, default lossless queues (fcoe and no-loss) will be lossy queues.

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. Except on QFX10000 or NFX Series devices, all forwarding classes mapped to a particular queue must be of the same type, either unicast or multicast. 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, which is not supported on OCX Series switches. On OCX Series switches, do not map traffic to the default fcoe and no-loss forwarding classes.

```
[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, if you are using Junos OS Release 12.2 or later, 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 introduced in Junos OS Release 12.3 in the forwarding class configuration..

NOTE: On switches that do not run ELS software, Junos OS Release 11.3R1 and earlier supported an alternate method of mapping forwarding classes to queues that allowed you to map only one forwarding class to a queue using the statement:

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

The **queue** statement has been deprecated and is no longer valid in Junos OS Release 11.3R2 and later. If you have a configuration that uses the **queue** statement to map forwarding classes to queues, edit the configuration to replace the **queue** statement with the **class** statement.

RELATED DOCUMENTATION

Example: Configuring CoS Hierarchical Port Scheduling (ETS) | 415

Example: Configuring Forwarding Classes | 157

Monitoring CoS Forwarding Classes | 169

Understanding CoS Forwarding Classes | 139

Understanding CoS Port Schedulers on QFX Switches | 343

Forwarding Policy Options Overview

Class-of-service (CoS)-based forwarding (CBF) enables you to control next-hop selection based on a packet's class of service and, in particular, the value of the IP packet's precedence bits.

For example, you might want to specify a particular interface or next hop to carry high-priority traffic while all best-effort traffic takes some other path. When a routing protocol discovers equal-cost paths, Junos picks a path at random or load-balance across the paths through either hash selection or round robin. CBF allows path selection based on class.

To configure CBF properties, include the following statements at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
forwarding-policy {
  next-hop-map map-name {
    forwarding-class class-name {
      next-hop [ next-hop-name ];
```

```
            lsp-next-hop [ lsp-regular-expression ];
            non-lsp-next-hop;
            discard;
        }
        forwarding-class-default {
            discard;
            lsp-next-hop [ lsp-regular-expression ];
            next-hop [next-hop-name];
            non-lsp-next-hop;
        }
    }
    class class-name {
        classification-override {
            forwarding-class class-name;
        }
    }
}
```

NOTE: Beginning with Junos OS Release 17.1R1, QFX10000 Series switches support CoS-based forwarding. **[set class-of-service forwarding-policy class]** is not supported on QFX10000 Series switches.

Beginning with Junos OS Release 17.2, MX routers with MPCs or MS-DPCs, VMX, PTX3000 routers, PTX5000 routers, and VPTX support configuring CoS-based forwarding (CBF) for up to 16 forwarding classes. All other platforms support CBF for up to 8 forwarding classes. To support up to 16 forwarding classes for CBF on MX routers, enable **enhanced-ip** at the **[edit chassis network-services]** hierarchy level. Enabling **enhanced-ip** is not necessary on PTX routers to support 16 forwarding classes for CBF.

Release History Table

Release	Description
17.2R1	Beginning with Junos OS Release 17.2, MX routers with MPCs or MS-DPCs, VMX, PTX3000 routers, PTX5000 routers, and VPTX support configuring CoS-based forwarding (CBF) for up to 16 forwarding classes.
17.1R1	Beginning with Junos OS Release 17.1R1, QFX10000 Series switches support CoS-based forwarding. **[set class-of-service forwarding-policy class]** is not supported on QFX10000 Series switches.

RELATED DOCUMENTATION

Configuring CoS-Based Forwarding

You can apply CoS-based forwarding (CBF) only to a defined set of routes. Therefore, you must configure a policy statement as in the following example:

```
[edit policy-options]
policy-statement my-cos-forwarding {
    from {
        route-filter destination-prefix match-type;
    }
    then {
        cos-next-hop-map map-name;
    }
}
```

This configuration specifies that routes matching the route filter are subject to the CoS next-hop mapping specified by ***map-name***. For more information about configuring policy statements, see the *Routing Policies, Firewall Filters, and Traffic Policers User Guide*.

NOTE: On M Series routers (except the M120 and M320 routers), forwarding-class-based matching and CBF do not work as expected if the forwarding class has been set with a multifield filter on an input interface.

Beginning with Junos OS Release 17.2, MX routers with MPCs or MS-DPCs, VMX, PTX3000 routers, and PTX5000 routers support configuring CoS-based forwarding (CBF) for up to 16 forwarding classes. All other platforms support CBF for up to 8 forwarding classes. To support up to 16 forwarding classes for CBF on MX routers, enable **enhanced-ip** at the **[edit chassis network-services]** hierarchy level.

You can configure CBF on a device with the supported number or fewer forwarding classes plus a default forwarding class only. Under this condition, the forwarding class to queue mapping can be either one-to-one or one-to-many. However, you cannot configure CBF when the number of forwarding classes configured exceeds the supported number. Similarly, with CBF configured, you cannot configure more than the supported number of forwarding classes plus a default forwarding class.

To specify a CoS next-hop map, include the **forwarding-policy** statement at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
forwarding-policy {
  next-hop-map map-name {
    forwarding-class class-name {
      discard;
      lsp-next-hop [ lsp-regular-expression ];
      next-hop [ next-hop-name ];
      non-lsp-next-hop;
    }
    forwarding-class-default {
      discard;
      lsp-next-hop [ lsp-regular-expression ];
      next-hop [next-hop-name];
      non-lsp-next-hop;
    }
  }
}
```

When you configure CBF with OSPF as the interior gateway protocol (IGP), you must specify the next hop as an interface name or next-hop alias, not as an IPv4 or IPv6 address. This is true because OSPF adds routes with the interface as the next hop for point-to-point interfaces; the next hop does not contain the IP address. For an example configuration, see "Example: Configuring CoS-Based Forwarding" on page 153.

For Layer 3 VPNs, when you use class-based forwarding for the routes received from the far-end provider edge (PE) router within a VRF instance, the software can match the routes based on the attributes that come with the received route only. In other words, the matching can be based on the route within RIB-in. In this case, the **route-filter** statement you include at the **[edit policy-options policy-statement my-cos-forwarding from]** hierarchy level has no effect because the policy checks the **bgp.l3vpn.0** table, not the ***vrf*.inet.0** table.

Junos OS applies the CoS next-hop map to the set of next hops previously defined; the next hops themselves can be located across any outgoing interfaces on the routing device. For example, the following configuration associates a set of forwarding classes and next-hop identifiers:

```
[edit class-of-service forwarding-policy]
next-hop-map map1 {
  forwarding-class expedited-forwarding {
    next-hop next-hop1;
    next-hop next-hop2;
  }
  forwarding-class best-effort {
```

```
        next-hop next-hop3;
        lsp-next-hop lsp-next-hop4;
    }
    forwarding-class-default {
        lsp-next-hop lsp-next-hop5;
    }
}
```

In this example, **next-hop *N*** is either an IP address or an egress interface for some next hop, and **lsp-next-hop *N*** is a regular expression corresponding to any next hop with that label. Q1 through Q*N* are a set of forwarding classes that map to the specific next hop. That is, when a packet is switched with Q1 through Q*N*, it is forwarded out the interface associated with the associated next hop.

This configuration has the following implications:

- A single forwarding class can map to multiple standard next hops or LSP next hops. This implies that load sharing is done across standard next hops or LSP next hops servicing the same class value. To make this work properly, Junos OS creates a list of the equal-cost next hops and forwards packets according to standard load-sharing rules for that forwarding class.
- If a forwarding class configuration includes LSP next hops and standard next hops, the LSP next hops are preferred over the standard next hops. In the preceding example, if both **next-hop3** and **lsp-next-hop4** are valid next hops for a route to which **map1** is applied, the forwarding table includes entry **lsp-next-hop4** only.
- If **next-hop-map** does not specify all possible forwarding classes, the default forwarding class is selected as the default. *default-forwarding class* defines the next hop for traffic that does not meet any forwarding class in the next hop map. If the default forwarding class is not specified in the next-hop map, a default is designated randomly. The default forwarding class is the class associated with queue 0.
- For LSP next hops, Junos OS uses UNIX **regex(3)**-style regular expressions. For example, if the following labels exist: **lsp**, **lsp1**, **lsp2**, **lsp3**, the statement **lsp-next-hop lsp** matches **lsp**, **lsp1**, **lsp2**, and **lsp3**. If you do not want this behavior, you must use the anchor characters **lsp-next-hop " ^lsp$"**, which match **lsp** only.
- The route filter does not work because the policy checks against the **bgp.l3vpn.0** table instead of the ***vrf*.inet.0** table.

The final step is to apply the route filter to routes exported to the forwarding engine. This is shown in the following example:

```
routing-options {
    forwarding-table {
        export my-cos-forwarding;
    }
}
```

This configuration instructs the routing process to insert routes to the forwarding engine matching **my-cos-forwarding** with the associated next-hop CBF rules.

The following algorithm is used when you apply a configuration to a route:

- If the route is a single next-hop route, all traffic goes to that route; that is, no CBF takes effect.
- For each next hop, associate the proper forwarding class. If a next hop appears in the route but not in the **cos-next-hop** map, it does not appear in the forwarding table entry.
- The default forwarding class is used if not all forwarding classes are specified in the next-hop map. If the default is not specified, the default is assigned to the lowest class defined in the next-hop map.

Release History Table

Release	Description
17.2R1	Beginning with Junos OS Release 17.2, MX routers with MPCs or MS-DPCs, VMX, PTX3000 routers, and PTX5000 routers support configuring CoS-based forwarding (CBF) for up to 16 forwarding classes.

RELATED DOCUMENTATION

Load Balancing VPLS Non-Unicast Traffic Across Member Links of an Aggregate Interface

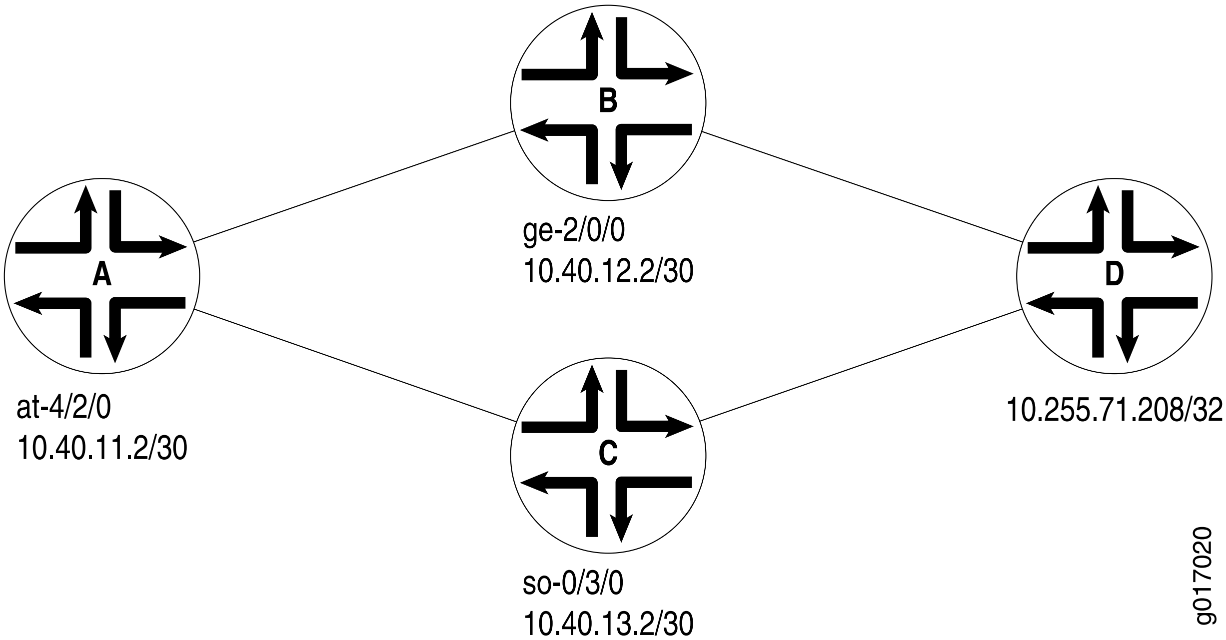
Forwarding Policy Options Overview | 148

Example: Configuring CoS-Based Forwarding

Router A has two routes to destination **10.255.71.208** on Router D. One route goes through Router B, and the other goes through Router C, as shown in Figure 5 on page 154.

Configure Router A with CoS-based forwarding (CBF) to select Router B for queue 0 and queue 2, and Router C for queue 1 and queue 3.

Figure 5: Sample CoS-Based Forwarding



When you configure CBF with OSPF as the IGP, you must specify the next hop as an interface name, not as an IPv4 or IPv6 address. The next hops in this example are specified as **ge-2/0/0.0** and **so-0/3/0.0**.

```
[edit class-of-service]
forwarding-policy {
    next-hop-map my_cbf {
        forwarding-class be {
            next-hop ge-2/0/0.0;
        }
        forwarding-class ef {
            next-hop so-0/3/0.0;
        }
        forwarding-class af {
            next-hop ge-2/0/0.0;
        }
        forwarding-class nc {
            next-hop so-0/3/0.0;
        }
    }
}
classifiers {
    inet-precedence inet {
        forwarding-class be {
            loss-priority low code-points [ 000 100 ];
        }
```

```
        forwarding-class ef {
            loss-priority low code-points [ 001 101 ];
        }
        forwarding-class af {
            loss-priority low code-points [ 010 110 ];
        }
        forwarding-class nc {
            loss-priority low code-points [ 011 111 ];
        }
    }
}
forwarding-classes {
    queue 0 be;
    queue 1 ef;
    queue 2 af;
    queue 3 nc;
}
interfaces {
    at-4/2/0 {
        unit 0 {
            classifiers {
                inet-precedence inet;
            }
        }
    }
}

[edit policy-options]
policy-statement cbf {
    from {
        route-filter 10.255.71.208/32 exact;
    }
    then cos-next-hop-map my_cbf;
}

[edit routing-options]
graceful-restart;
forwarding-table {
    export cbf;
}

[edit interfaces]
traceoptions {
    file trace-intf size 5m world-readable;
```

```
    flag all;
  }
  so-0/3/0 {
    unit 0 {
      family inet {
        address 10.40.13.1/30;
      }
      family iso;
      family mpls;
    }
  }
  ge-2/0/0 {
    unit 0 {
      family inet {
        address 10.40.12.1/30;
      }
      family iso;
      family mpls;
    }
  }
  at-4/2/0 {
    atm-options {
      vpi 1 {
        maximum-vcs 1200;
      }
    }
    unit 0 {
      vci 1.100;
      family inet {
        address 10.40.11.2/30;
      }
      family iso;
      family mpls;
    }
  }
```

RELATED DOCUMENTATION

Forwarding Policy Options Overview | 148

Example: Configuring Forwarding Classes

IN THIS SECTION

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. Except on QFX10000, unicast traffic and multidestination (multicast, broadcast, and destination lookup fail) traffic use separate forwarding classes and output queues.

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 (this example was tested on a Juniper Networks QFX3500 Switch)
- Junos OS Release 11.1 or later for the QFX Series or Junos OS Release 14.1X53-D20 or later for the OCX Series

Configuring forwarding classes for QFX10000 switches

- One QFX10000 switch
- Junos OS Release 15.1X53-D10 or later for the QFX Series

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 multidestination 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 multidestination traffic. Except for OCX Series switches, switches support up to six lossless forwarding classes. (OCX Series switches do not support lossless Layer 2 transport.)

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

- **mcast**—Multidestination 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: On switches that do not run ELS software, if you are using Junos OS Release 12.2, 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.

In Junos OS Release 12.3 and later, you can include the *no-loss* packet drop attribute in explicit forwarding class configurations to configure a lossless forwarding class.

NOTE: On switches that do not run ELS software, Junos OS Release 11.3R1 and earlier supported an alternate method of mapping forwarding classes to queues that allowed you to map only one forwarding class to a queue using the statement:

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

The **queue** statement has been deprecated and is no longer valid in Junos OS Release 11.3R2 and later. If you have a configuration that uses the **queue** statement to map forwarding classes to queues, edit the configuration to replace the **queue** statement with the **class** statement.

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

Configuration

Step-by-Step Procedure

Table 47 on page 160 shows the configuration forwarding-class-to-queue mapping for this example:

Table 47: 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

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

Configuration

Step-by-Step Procedure

Table 48 on page 161 shows the configuration forwarding-class-to-queue mapping for this example:

Table 48: Forwarding-Class-to-Queue Example Configuration on QFX10000

Forwarding Class	Queue
best-effort	**0**
be1	**1**
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

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;
```

RELATED DOCUMENTATION

Example: Configuring CoS Hierarchical Port Scheduling (ETS) | 415

Defining CoS Forwarding Classes | 146

Monitoring CoS Forwarding Classes | 169

Overview of CoS Changes Introduced in Junos OS Release 11.3

Overview of CoS Changes Introduced in Junos OS Release 12.2

Understanding CoS Forwarding Classes | 139

Understanding CoS Forwarding Classes

Understanding CoS Forwarding Class Sets (Priority Groups)

A forwarding class set is the Junos OS configuration construct that equates to a priority group in enhanced transmission selection (ETS, described in IEEE 802.1Qaz). The switch implements ETS using a two-tier hierarchical scheduler.

A priority group is a group of forwarding classes. Each forwarding class is mapped to an output queue and an IEEE 802.1p priority (code points). Classifying traffic into a forwarding class based on its code points, and mapping the forwarding class to a queue, defines the traffic assigned to that queue. The forwarding classes that belong to a priority group share the port bandwidth allocated to that priority group. The traffic mapped to forwarding classes in one priority group usually shares similar traffic-handling requirements.

You can configure up to three unicast forwarding class sets and one multicast forwarding class set. Only unicast forwarding classes can belong to unicast forwarding class sets. Only multicast forwarding classes can belong to the multicast forwarding class set.

If you configure a strict-high priority forwarding class (you can configure only one strict-high priority forwarding class), you must observe the following rules when configuring forwarding class sets:

- You must create a separate forwarding class set for the strict-high priority forwarding class.
- Only one forwarding class set can contain the strict-high priority forwarding class.
- A strict-high priority forwarding class cannot belong to the same forwarding class set as forwarding classes that are not strict-high priority.
- A strict-high priority forwarding class cannot belong to a multidestination forwarding class set.
- You cannot configure a guaranteed minimum bandwidth (guaranteed rate) for a forwarding class set that includes a strict-high priority forwarding class. (You also cannot configure a guaranteed minimum bandwidth for a strict-high forwarding class.)
- We recommend that you always apply a shaping rate to a strict-high priority forwarding class to prevent it from starving the queues mapped to other forwarding classes. If you do not apply a shaping rate to limit the amount of bandwidth a strict-high priority forwarding class can use, then the strict-high priority forwarding class can use all of the available port bandwidth and starve other forwarding classes on the port.

You must use hierarchical scheduling if you explicitly configure CoS. The two-tier hierarchical scheduler defines bandwidth resources for the forwarding class set (priority group), and then allocates those resources among the forwarding classes (priorities) that belong to the forwarding class set.

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 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 priority group is transparent. It does not appear in the configuration and is used for Data Center Bridging Capability Exchange Protocol (DCBX) advertisement (except on OCX Series switches, which do not support DCBX).

When you explicitly configure forwarding class sets and apply them to interfaces, on those interfaces, forwarding classes that you do not map to a forwarding class set receive no guaranteed bandwidth. Forwarding classes that belong to the default forwarding class set might receive bandwidth if the other forwarding class sets are not using all of the port bandwidth. However, the amount of bandwidth received by forwarding classes that are not members of a forwarding class set is not guaranteed. In this case, the bandwidth a forwarding class receives if it is not a member of a forwarding class set depends on whether unused port bandwidth is available and therefore is not deterministic.

To guarantee bandwidth for forwarding classes in a predictable manner, be sure to map all forwarding classes that you expect to carry traffic on an interface to a forwarding class set, and apply the forwarding class set to the interface.

RELATED DOCUMENTATION

Defining CoS Forwarding Class Sets

A forwarding class set is a priority group for enhanced transmission selection (ETS) traffic control. Each forwarding class set consists of one or more forwarding classes. Classifiers map traffic into forwarding classes based on code points (priority), and forwarding classes are mapped to output queues.

You can configure up to three unicast forwarding class sets and one multicast forwarding class set.

To configure a forwarding class set using the CLI:

1. Assign one or more forwarding classes to the forwarding class set:

   ```
   [edit class-of-service]
   user@switch# set forwarding-class-sets forwarding-class-set-name class forwarding-class-name
   ```

2. Map the forwarding class set to an interface:

   ```
   [edit class-of-service]
   user@switch# set interfaces interface-name forwarding-class-set forwarding-class-set-name
   ```

RELATED DOCUMENTATION

Example: Configuring Forwarding Class Sets

IN THIS SECTION

A forwarding class set (fc-set) is a priority group for enhanced transmission selection (ETS) traffic control. Each fc-set consists of one or more forwarding classes (priorities). Classifiers map traffic to forwarding classes based on code points, and forwarding classes are mapped to output queues.

ETS enables you to configure link resources (bandwidth and bandwidth sharing characteristics) for an fc-set, and then allocate the fc-set's resources among the forwarding classes that belong to the fc-set. This is called two-tier, or hierarchical, scheduling. Traffic control profiles control the scheduling for the fc-set (priority group), and schedulers control the scheduling for individual forwarding classes (priorities).

Requirements

This example uses the following hardware and software components:

- One switch (this example was tested on a Juniper Networks QFX3500 Switch)
- Junos OS Release 11.1 or later for the QFX Series or Junos OS Release 14.1X53-D20 or later for the OCX Series.

Overview

You can configure up to three unicast fc-sets and one multicast fc-set. A common way to configure unicast priority groups is to configure separate fc-sets for local area network (LAN) traffic, storage area network (SAN) traffic, and high-performance computing (HPC) traffic, and then assign the appropriate forwarding classes to each fc-set.

NOTE: If you configure a **strict-high** priority forwarding class, you must create an fc-set that is dedicated only to **strict-high** priority traffic. You can only configure one strict-high priority forwarding class, and only one fc-set can contain a strict-high priority queue. Queues that are not strict-high priority cannot belong to the same fc-set as a strict-high priority queue. The multidestination fc-set cannot contain a strict-high priority queue.

To apply ETS, you use a traffic control profile to map one or more fc-sets to a physical egress port. You can map up to three unicast forwarding class sets and one multidestination forwarding class set to each port. When you map an fc-set to a port, the port uses hierarchical scheduling to allocate port resources to the priority group (fc-set) and to allocate the priority group resources to the queues (forwarding classes) that belong to the priority group.

This example describes how to:

- Configure three fc-sets called **lan-pg**, **san-pg**, and **hpc-pg**.
- Assign forwarding classes to each of the fc-sets.
- Apply the fc-sets and their output traffic control profiles to an egress interface.

This example does not describe how to configure the forwarding classes assigned to the fc-sets or how to configure traffic control profiles (scheduling). "Example: Configuring CoS Hierarchical Port Scheduling (ETS)" on page 415 provides a complete example of how to configure ETS, including forwarding class and scheduling configuration. Table 49 on page 166 shows the configuration components for this example:

Table 49: Components of the Forwarding Class Sets Configuration Example

Component	Settings
Hardware	QFX3500 switch
LAN traffic priority group	Forwarding class set: **lan-pg** Forwarding classes: **best-effort-1**, **best-effort-2**
SAN traffic priority group	Forwarding class set: **san-pg** Forwarding classes: **fcoe**, **fcoe-2** **NOTE:** OCX Series switches do not support FCoE traffic or lossless Layer 2 transport. If you were configuring this example on an OCX Series switch, you could omit this priority group, or rename it and map different forwarding classes to it.
HPC traffic priority group	Forwarding class set: **hpc-pg** Forwarding classes: **nc**, **high-perf**

Table 49: Components of the Forwarding Class Sets Configuration Example *(continued)*

Component	Settings
Egress interface	**xe-0/0/7**

Configuring Forwarding Class Sets

1. Define the **lan-pg** priority group (fc-set) and assign to it the forwarding classes **best-effort-1** and **best-effort-2**:

```
[edit class-of-service]
user@switch# set forwarding-class-sets lan-pg class best-effort-1
user@switch# set forwarding-class-sets lan-pg class best-effort-2
```

2. Define the **san-pg** priority group and assign to it the forwarding classes **fcoe** and **fcoe-2**:

```
[edit class-of-service]
user@switch# set forwarding-class-sets san-pg class fcoe
user@switch# set forwarding-class-sets san-pg class fcoe-2
```

3. Define the **hpc-pg** priority group and assign to it the forwarding classes **nc** and **high-perf**:

```
[edit class-of-service]
user@switch# set forwarding-class-sets hpc-pg class nc
user@switch# set forwarding-class-sets hpc-pg class high-perf
```

4. Map the three forwarding class sets to an interface (the output traffic control profiles associated with the forwarding class sets determine the class of service scheduling for the priority groups):

```
[edit class-of-service]
user@switch# set interfaces xe-0/0/7 forwarding-class-set lan-pg output-traffic-control-profile
lan-tcp
user@switch# set interfaces xe-0/0/7 forwarding-class-set san-pg output-traffic-control-profile
san-tcp
user@switch# set interfaces xe-0/0/7 forwarding-class-set hpc-pg output-traffic-control-profile
hpc-tcp
```

Verification

IN THIS SECTION

To verify the priority group configuration, perform these tasks:

Verifying Forwarding Class Set Membership

Purpose

Verify that you configured the **lan-pg**, **san-pg**, and **hpc-pg** priority groups with the correct forwarding classes.

Action

List the forwarding class set member configuration using the operational mode command **show configuration class-of-service forwarding-class-sets**:

`user@switch>` **show configuration class-of-service forwarding-class-sets**

```
lan-pg {
    class best-effort-1;
    class best-effort-2;
}
san-pg {
    class fcoe;
    class fcoe-2;
}
hpc-pg {
    class high-perf;
    class nc;
}
```

Verifying the Egress Interface Configuration

Purpose

Verify that egress interface **xe-0/0/7** is associated with the **lan-pg**, **san-pg**, and **hpc-pg** priority groups and with the correct output traffic control profiles.

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 {
    lan-pg {
        output-traffic-control-profile lan-tcp;
    }
    san-pg {
        output-traffic-control-profile san-tcp;
    }
    hpc-pg {
        output-traffic-control-profile hpc-tcp;
    }
}
```

RELATED DOCUMENTATION

Example: Configuring CoS Hierarchical Port Scheduling (ETS) | **415**
Example: Configuring Queue Schedulers | **325**
Example: Configuring Traffic Control Profiles (Priority Group Scheduling) | **385**
Defining CoS Forwarding Class Sets | **164**
Understanding CoS Forwarding Class Sets (Priority Groups) | **162**

Monitoring CoS Forwarding Classes

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 multidestination (multicast, broadcast, destination lookup fail) traffic. These switches support 12 forwarding classes and output queues, eight for unicast traffic and four for multidestination traffic.

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

Table 50 on page 170 summarizes key output fields on switches that use different forwarding classes and output queues for unicast and multidestination traffic.

Table 50: Summary of Key CoS Forwarding Class Output Fields on Switches that Separate Unicast and Multidestination Traffic

Field	Values
Forwarding Class	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: • **best-effort**—Provides no special CoS handling of packets. Loss priority is typically not carried in a CoS value. • **fcoe**—Provides guaranteed delivery for Fibre Channel over Ethernet (FCoE) traffic. • **no-loss**—Provides guaranteed delivery for TCP lossless traffic • **network-control**—Packets can be delayed but not dropped. By default, the following multidestination forwarding class is assigned to queue 8: • **mcast**—Provides no special CoS handling of packets.
Queue	Queue number corresponding to (mapped to) the forwarding class name. 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: • Queue 0—**best-effort** • Queue 3—**fcoe** • Queue 4—**no-loss** • Queue 7—**network-control** • Queue 8—**mcast**

Table 50: Summary of Key CoS Forwarding Class Output Fields on Switches that Separate Unicast and Multidestination Traffic *(continued)*

Field	Values
No-Loss	Packet drop attribute associated with each forwarding class: • Disabled—The forwarding class is configured for lossy transport (packets might drop during periods of congestion) • Enabled—The forwarding class is configured for lossless transport **NOTE:** 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. OCX Series switches do not support lossless transport.

NOTE: OCX Series switches do not support the default lossless forwarding classes **fcoe** and **no-loss**, and do not support the no-loss packet drop attribute used to configure lossless forwarding classes. On OCX Series switches, do not map traffic to the default **fcoe** and **no-loss** forwarding classes (both of these default forwarding classes carry the no-loss packet drop attribute), and do not configure the no-loss packet drop attribute on forwarding classes.

Table 51 on page 171 summarizes key output fields on switches that use the same forwarding classes and output queues for unicast and multidestination traffic.

Table 51: Summary of Key CoS Forwarding Class Output Fields on Switches That Do Not Separate Unicast and Multidestination Traffic

Field	Values
Forwarding Class	Names of forwarding classes assigned to queue numbers. By default, the following forwarding classes are assigned to queues 0, 3, 4, and 7, respectively: • **best-effort**—Provides no special CoS handling of packets. Loss priority is typically not carried in a CoS value. • **fcoe**—Provides guaranteed delivery for Fibre Channel over Ethernet (FCoE) traffic. • **no-loss**—Provides guaranteed delivery for TCP lossless traffic • **network-control**—Packets can be delayed but not dropped.

Table 51: Summary of Key CoS Forwarding Class Output Fields on Switches That Do Not Separate Unicast and Multidestination Traffic *(continued)*

Field	Values
Queue	Queue number corresponding to (mapped to) the forwarding class name. By default, four queues (0, 3, 4, and 7) are assigned to forwarding classes: • Queue 0—**best-effort** • Queue 3—**fcoe** • Queue 4—**no-loss** • Queue 7—**network-control**
No-Loss	Packet drop attribute associated with each forwarding class: • Disabled—The forwarding class is configured for lossy transport (packets might drop during periods of congestion). • Enabled—The forwarding class is configured for lossless transport. **NOTE:** 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. OCX Series switches do not support lossless transport.

CHAPTER 7

Lossless Traffic Flows, Ethernet PAUSE Flow Control, and PFC

IN THIS CHAPTER

Understanding CoS IEEE 802.1p Priorities for Lossless Traffic Flows

IN THIS SECTION

The switch supports up to six lossless forwarding classes. (Junos OS Release 12.3 increased support for lossless priorities from two lossless forwarding classes—the default **fcoe** and **no-loss** forwarding classes—to a maximum of six lossless forwarding classes.) Each forwarding class is mapped to an IEEE 802.1p code point (priority).

> **NOTE:** Junos OS Release 13.1 introduced support for up to six lossless forwarding classes on QFabric systems. Throughout this document, features introduced on standalone switches in Junos OS Release 12.3 are introduced on QFabric systems in Junos OS Release 13.1 unless otherwise noted.
>
> Only switches with native Fibre Channel (FC) interfaces, such as the QFX3500, support native FC traffic and configuration as an FCoE-FC gateway. Throughout this document, features that pertain to native FC traffic and to FCoE-FC gateway configuration apply only to switches that support native FC interfaces.

Video: Why Use PFC in a Data Center Network?

The default configuration is the same as the default configuration in Junos OS Release 12.2 and is backward-compatible. If you need only two (or fewer) lossless forwarding classes, use the default configuration, in which the **fcoe** and **no-loss** forwarding classes are lossless. If you need more than two lossless forwarding classes, you can use the two default lossless forwarding classes and configure additional lossless forwarding classes. If you do not want to use the default lossless forwarding classes, you can change them, or use only the lossless forwarding classes that you explicitly configure.

Default Lossless Priority Configuration

If you do not explicitly configure forwarding classes, the system uses the default forwarding class configuration, which provides two default lossless forwarding classes (*fcoe* and *no-loss*). (If you change the forwarding class configuration, the changes apply to all traffic on that device because forwarding classes are global to a particular device.)

If you do not explicitly configure classifiers, and you do not explicitly configure flow control to pause output queues (configured in the output stanza of the CNP), the default classifier and the default output queue pause configurations are applied to all Ethernet interfaces on the switches (or Node devices). You can override the default classifier and the default output queue pause configuration on a per-interface basis by applying an explicit configuration to an Ethernet interface. The default configuration is used on all Ethernet interfaces that do not have an explicit configuration.

NOTE: If you do not configure flow control on output queues, the default configuration uses a one-to-one mapping of IEEE 802.1p code points (priorities) to output queues by number. For example, priority 0 (code point 000) is mapped to queue 0, priority 1 (code point 001) is mapped to queue 1, and so on. If you do not use the default configuration, you must explicitly configure flow control on each output queue that you want to enable for PFC pause in the output stanza of the CNP.

In the default configuration, only queue 3 and queue 4 are enabled to respond to pause messages from the connected peer. For queue 3 to respond to pause messages, priority 3 (code point 011) must be enabled for PFC in the input stanza of the CNP. For queue 4 to respond to pause messages, priority 4 (code point 100) must be enabled for PFC in the input stanza of the CNP.

The default configuration provides the following lossless behavior:

- Two default lossless forwarding classes (the **no-loss** packet drop attribute is applied to these forwarding classes automatically):
  fcoe—Mapped to output queue 3
  no-loss—Mapped to output queue 4
- A default classifier that maps the fcoe forwarding class to IEEE 802.1p priority 3 (011) and the no-loss forwarding class to IEEE 802.1p priority 4 (100)
- Priority-based flow control (PFC) enabled on Ethernet interface output queues 3 and 4 when those queues carry lossless traffic (traffic that is mapped to the fcoe and no-loss forwarding classes, respectively).

  On switches that can be configured as an FCoE-FC gateway, native FC interfaces (NP_Ports), with default flow control enabled on output queue 3 (IEEE 802.1p priority 3) for FCoE/FC traffic.
- DCBX is enabled on all interfaces in autonegotiation mode, and automatically exchanges FCoE application protocol type, length, and values (TLVs) on interfaces that carry FCoE traffic. However, if you explicitly configure DCBX protocol TLV exchange for any application, then you must explicitly configure protocol TLV exchange for every application for which you want DCBX to exchange TLVs, including FCoE.
- On Ethernet ports, PFC buffer calculations use the following default values to determine the headroom buffer size:
  Cable length—100 meters (approximately 328 feet)
  MRU for priority 3 traffic—2500 bytes
  MRU for priority 4 traffic—9216 bytes
  Maximum transmission unit (MTU)—1522 (or the configured MTU value for the interface)

  NOTE: If you configure flow control on a priority that is not one of the default flow control priorities, the default MRU value is 2500 bytes. For example, if you configure flow control on priority 5 and you do not configure an MRU value, the default MRU value is 2500 bytes.

NOTE: In addition, to support lossless transport, PFC must be enabled explicitly on the lossless IEEE 802.1p priorities (code points) on ingress Ethernet interfaces; no default PFC configuration is applied at ingress interfaces. If you do not enable PFC on lossless priorities, those priorities might experience packet loss during periods of congestion. For example, if you want lossless FCoE traffic and you are using the default fcoe forwarding class, you use a CNP to enable PFC on priority 3 (code point 011), and apply that CNP to all ingress interfaces that carry FCoE traffic.

You can override the default classifier and the default output queue pause configuration on a per-interface basis by applying an explicit configuration to an Ethernet interface.

The default CoS configuration is backward-compatible with the *default* CoS configuration of software releases before Junos OS Release 12.3. If you explicitly configure lossless transport, ensure that the input and output queues corresponding to the lossless forwarding classes are explicitly configured for PFC pause.

Table 52 on page 176 summarizes the default forwarding classes and their mapping to output queues, IEEE 802.1p priorities, and drop attributes.

Table 52: Mapping of Default Forwarding Class to Queue, IEEE 802.1p Priority, and Drop Attribute

Forwarding Class Name	Output Queue	Priority	Drop Attribute
best-effort	0	0	drop
fcoe	3	3	no-loss
no-loss	4	4	no-loss
network-control	7	7	drop

On switches that use the same forwarding classes and output queues for unicast and multidestination (multicast, broadcast, and destination lookup fail) traffic, these forwarding classes carry both unicast and multidestination traffic. Only unicast traffic is treated as lossless traffic. Multidestination traffic is not treated as lossless traffic, even on lossless output queues.

On switches that use different forwarding classes and output queues for unicast and multidestination traffic, there is one default multidestination forwarding class named *mcast*, which is mapped to output queue 8 with a drop attribute of drop. (Incoming multidestination traffic on all IEEE 802.1p priorities is mapped to the mcast forwarding class by default.)

Configuring Lossless Priorities

IN THIS SECTION

To configure more than two lossless priorities (forwarding classes), or to change the default mapping of lossless forwarding classes to priorities and paused output queues, you must explicitly configure the switch instead of using the default configuration. Configuring lossless priorities includes:

- Configuring forwarding classes with the no-loss packet drop attribute.
- Using a CNP to configure PFC on ingress interfaces and flow control (PFC) on egress interfaces.
- Configuring a classifier to map IEEE 802.1p priorities (code points) to the correct forwarding classes (the forwarding classes for which you want lossless transport).

> **NOTE:** If you expect a large amount of lossless traffic on your network and configure multiple lossless traffic classes, ensure that you reserve enough scheduling resources (bandwidth) and buffer space to support the lossless flows. (For switches that support shared buffer configuration, "Understanding CoS Buffer Configuration" on page 635 describes how to configure buffers and provides a recommended buffer configuration for networks with larger amounts of lossless traffic. Buffer optimization is automatic on switches that use virtual output queues.)

In addition, on Ethernet interfaces, DCBX must exchange the appropriate application protocol TLVs for the lossless traffic. On switches that can act as an FCoE-FC gateway, you need to remap the FCoE priority on native FC interfaces if your network uses a priority other than 3 (IEEE code point 011) for FCoE traffic. This section describes:

Configuring Lossless Forwarding Classes (Packet Drop Attribute)

Junos OS Release 12.3 introduced the *no-loss* parameter for forwarding class configuration. (Although it uses the same name, this is not the no-loss default forwarding class. It is a packet drop attribute you can specify to configure any forwarding class as a lossless forwarding class.)

NOTE: On switches that use different forwarding classes for unicast and multidestination traffic, the forwarding class must be a unicast forwarding class. On switches that use the same forwarding classes for unicast and multidestination traffic, only unicast traffic receives lossless treatment.

You can configure up to six forwarding classes (depending on system architecture and the availability of system resources) as lossless forwarding classes by including the **no-loss** drop attribute at the **[edit class-of-service forwarding-classes class *forwarding-class-name* queue-num *queue-number*]** hierarchy level.

If you use the default fcoe or no-loss forwarding classes, they include the no-loss drop attribute by default. If you explicitly configure the fcoe or no-loss forwarding classes and you want to retain their lossless behavior, you *must* include the no-loss drop attribute in the configuration.

NOTE: All forwarding classes mapped to the same output queue must have the same packet drop attribute. (All forwarding classes mapped to the same output queue must be either lossy or lossless. You cannot map both a lossy and a lossless forwarding class to the same queue.)

To avoid fate sharing (a congested flow affecting an uncongested flow), use a one-to-one mapping of lossless forwarding classes to IEEE 802.1p code points (priorities) and queues. Map each lossless forwarding class to a different queue, and classify incoming traffic into forwarding classes so that each forwarding class transports traffic of only one priority (code point).

The fcoe and no-loss forwarding classes are special cases, because in the default configuration, they are configured for lossless behavior (providing that you also enable PFC on the priorities mapped to the fcoe and no-loss forwarding classes in the CNP input stanza).

Table 53 on page 178 summarizes the possible configurations of the fcoe and no-loss forwarding classes in Junos OS Release 12.3 and later, and the result of those configurations in terms of lossless traffic behavior. It is assumed that PFC, DCBX, and classifiers are properly configured.

Table 53: FCoE and No-Loss Forwarding Class Configuration in Junos OS Release 12.3

Explicit (User-Configured) or Default Forwarding Class Configuration	Packet Drop Attribute	Result and Notes
Default	Default	The fcoe and no-loss forwarding classes are lossless. **NOTE:** Even if you explicitly configure other forwarding classes (lossy or lossless forwarding classes), the fcoe and no-loss forwarding classes remain lossless because they are not explicitly configured.

Table 53: FCoE and No-Loss Forwarding Class Configuration in Junos OS Release 12.3 ***(continued)***

Explicit (User-Configured) or Default Forwarding Class Configuration	Packet Drop Attribute	Result and Notes
Explicit	Not specified in the explicit forwarding class configuration	The fcoe and no-loss forwarding classes are lossy because they do not include the no-loss drop attribute.
Explicit	No-loss	The fcoe and no-loss forwarding classes are lossless.
Explicit, configured in Junos OS Release 12.2 or earlier	Not specified (packet drop attribute was not available before Junos OS Release 12.3)	The fcoe and no-loss forwarding classes are lossy in Junos OS Release 12.3 and later because they do not include the no-loss drop attribute. **NOTE:** To retain lossless behavior, before you upgrade to Junos OS Release 12.3, delete the explicit configuration so that the system uses the default configuration. Alternatively, you can reconfigure the forwarding classes with the no-loss packet drop attribute after upgrading to Junos OS Release 12.3 or later.

For all other forwarding classes except the **fcoe** and **no-loss** forwarding classes, you must explicitly configure lossless transport by specifying the no-loss packet drop attribute, because the default configuration for all other forwarding classes is lossy (the no-loss packet drop attribute is not applied).

Congestion Notification Profiles (PFC Configuration)

Use CNPs to configure lossless PFC characteristics on input and output interfaces.

The input stanza of a CNP enables PFC on specified IEEE 802.1p priorities (code points) and fine-tunes headroom buffer settings by configuring the maximum receive unit (MRU) value and cable length on ingress interfaces.

The output stanza of a CNP enables PFC (flow control) on output queues for specified IEEE 802.1p priorities so that the queues can respond to PFC pause messages from the connected peer on the priority of your choice. (By default, output queues 3 and 4 respond to received PFC messages when those queues carry lossless traffic in the fcoe and no-loss forwarding classes, respectively.)

To achieve lossless transport, the priority paused at the ingress interfaces must match the priority paused at the egress interfaces for a given traffic flow. For example, if you configure ingress interfaces to pause traffic tagged with IEEE 802.1p priority 5 (code point 101) and priority 5 traffic is mapped to output queue 5, then you must also configure the corresponding output interfaces to pause priority 5 on queue 5. In addition, the forwarding class mapped to queue 5 must be configured as a lossless forwarding class (using the no-loss drop attribute).

CAUTION: Any change to the PFC configuration on a port temporarily blocks the entire port (not just the priorities affected by the PFC change) so that the port can implement the change, then unblocks the port. Blocking the port stops ingress and egress traffic, and causes packet loss on all queues on the port until the port is unblocked.

A change to the PFC configuration means any change to a CNP, including changing the input portion of the CNP (enabling or disabling PFC on a priority, or changing the MRU or cable-length values) or changing the output portion the CNP that enables or disables output flow control on a queue. A PFC configuration change only affects ports that use the changed CNP.

The following actions change the PFC configuration:

- Deleting or disabling a PFC configuration (input or output) in a CNP that is in use on one or more interfaces. For example:
    1. An existing CNP with an input stanza that enables PFC on priorities 3, 5, and 6 is configured on interfaces xe-0/0/20 and xe-0/0/21.
    2. We disable the PFC configuration for priority 6 in the input CNP, and then commit the configuration.
    3. The PFC configuration change causes all traffic on interfaces xe-0/0/20 and xe-0/0/21 to stop until the PFC change has been implemented. When the PFC change has been implemented, traffic resumes.
- Configuring a CNP on an interface. (This changes the PFC state by enabling PFC on one or more priorities.)
- Deleting a CNP from an interface. (This changes the PFC state by disabling PFC on one or more priorities.)

Configuring Input Interface Flow Control (PFC and Headroom Buffer Calculation)

On Ethernet interfaces, the input stanza of the CNP enables PFC on specified priorities so that the ingress interface can send a pause message to the connected peer during periods of congestion. Input CNPs also fine-tune the headroom buffers used for PFC support by allowing you to configure the MRU value and cable length (if you do not want to use the default configuration).

Headroom buffers support lossless transport by storing the traffic that arrives at an interface after the interface sends a PFC flow control message to pause incoming traffic. Until the connected peer receives the flow control message and pauses traffic, the interface continues to receive traffic and must buffer it (and the traffic that is still on the wire after the peer pauses) to prevent packet loss.

The system uses the MRU and the length of the attached physical cable to calculate buffer headroom allocation. The default configuration values are:

- MRU for priority 3 traffic—2500 bytes
- MRU for priority 4 traffic—9216 bytes
- Cable length—100 meters (approximately 328 feet)

NOTE: If you configure flow control on a priority that is not one of the default flow control priorities, the default MRU value is 2500 bytes. For example, if you configure flow control on priority 5 and you do not explicitly configure an MRU value, the default MRU value is 2500 bytes.

You can fine-tune the MRU and the cable length to adjust the size of the headroom buffer on an interface. The switch has a shared global buffer pool and dynamically allocates headroom buffer space to lossless queues as needed.

A lower MRU or a shorter cable length reduces the amount of headroom buffer required on an interface and leaves more headroom buffer space for other interfaces. A higher MRU or a longer cable length increases the amount of headroom buffer space required on an interface and leaves less headroom buffer space for other interfaces.

In many cases, you can better utilize the headroom buffers by reducing the MRU value (for example, an MRU of 2180 is sufficient for most FCoE networks) and by reducing the cable length value if the physical cable is less than 100 meters long.

NOTE: When you configure the headroom buffers by changing the MRU or the cable length, and commit the configuration, the system performs a commit check and rejects the configuration if sufficient headroom buffer space is not available.

However, the system does not perform a commit check but instead returns a syslog error if:

- The buffers are configured on a LAG interface.
- The default classifier is used on the interface (instead of a user-configured classifier).
- The interface has not been created yet.

Configuring Output Interface Flow Control (PFC)

On Ethernet interfaces, you can use the output stanza of the CNP to configure flow control on output queues and enable PFC pause response on specified IEEE 802.1p priorities.

NOTE: On switches that use different output queues for unicast and multidestination traffic, the queue must be a unicast output queue.

By default, output queues 3 and 4 are enabled for PFC pause on priorities 3 (IEEE 802.1p code point 011) and 4 (IEEE 802.1p code point 100). The default PFC pause response supports the default lossless forwarding class configuration, which maps the fcoe forwarding class to queue 3 and priority 3, and maps the no-loss forwarding class to queue 4 and priority 4.

Configuring PFC on output queues enables you to pause any priority on any output queue on any Ethernet interface. Output flow control enables you to use more than two output queues to support lossless traffic flows (you can configure up to six lossless forwarding classes and map them to different output queues that are enabled for PFC pause). Output queue flow control also enables you to support multiple lossless forwarding classes (each mapped to a different priority and output queue) for one class of traffic.

NOTE: Output flow control only works when PFC is enabled in the CNP input stanza on the corresponding priorities on the interface. For example, if you enable output flow control on priority 5 (IEEE 802.1p code point 101), then you must also enable PFC in the CNP on the input stanza on priority 5.

For example, if the converged Ethernet network uses two different priorities for FCoE traffic (for example, priority 3 and priority 5), then you can classify those priorities into different lossless forwarding classes that are mapped to different output queues:

1. Configure two lossless forwarding classes for FCoE traffic, with each forwarding class mapped to a different output queue. For example, you could use the default fcoe forwarding class, which is mapped to queue 3, and you could configure a second lossless forwarding class called fcoe1 and map it to queue 5. The fcoe forwarding class is for priority 3 FCoE traffic (code point 011), and the fcoe1 forwarding class is for priority 5 (code point 101) FCoE traffic.

2. Configure a classifier that maps each forwarding class to the desired IEEE 802.1p code point (priority). If FCoE traffic on both priorities uses one interface, the classifier must classify both forwarding classes to the correct priorities. If FCoE traffic of different priorities uses different interfaces, the classifier configuration on each interface must map the correct priority to the corresponding lossless forwarding class.

3. Apply the classifier to the interfaces that carry FCoE traffic. The classifier determines the mapping of forwarding classes to priorities on each interface.

To configure lossless transport for these forwarding classes, you also need to:

- Enable PFC on the two priorities (3 and 5 in this example) at the ingress interfaces in the CNP input stanza.
- Configure PFC on the output queues and priorities for the forwarding classes in the CNP output stanza so that the interface can respond to pause messages received from the connected peer.

> **NOTE:** When you configure the CNP on an interface, all ingress and egress traffic is blocked until the configuration is implemented, then the interface is unblocked and traffic resumes. During the time the interface is blocked, all queues on the interface experience packet loss.

- Configure DCBX to exchange application protocol TLVs on both FCoE priorities.

> **NOTE:** If you do not configure flow control to pause output queues, the default configuration uses a one-to-one mapping of IEEE 802.1p code points (priorities) to output queues by number. For example, priority 0 (code point 000) is mapped to queue 0, priority 1 (code point 001) is mapped to queue 1, and so on. By default, only queues 3 and 4 are enabled to respond to pause messages from the connected peer, and you must explicitly enable PFC on the corresponding priorities in the CNP input stanza to achieve lossless behavior.
>
> If you do not use the default configuration, you must explicitly configure flow control on each output queue that you want to enable for PFC pause. For example, if you explicitly configure flow control on output queue 5, the default configuration is no longer valid, and only output queue 5 is enabled for PFC pause. Output queues 3 and 4 are no longer enabled for PFC pause, so traffic using those queues no longer responds to PFC pause messages even if the corresponding forwarding class is configured with the no-loss drop attribute. To retain the pause configuration on output queues 3 and 4 and configure flow control on queue 5, you need to explicitly configure flow control on queues 3, 4, and 5.

On switches that use different output queues for unicast and multidestination traffic, you cannot configure flow control to pause a multidestination output queue. You can configure flow control to pause only unicast output queues. On switches that use the same output queues for unicast and multidestination traffic, only unicast traffic receives lossless treatment.

Output Interface Flow Control Profiles

Configuring the CNP output stanza creates an output flow control profile that tells egress ports the queues on which the Ethernet interface should respond to PFC pause messages. Although you can create an unlimited number of CNPs that contain input stanzas only, the number of CNPs that you can configure with output stanzas is limited:

- For standalone switches that are not part of a QFabric system, you can configure up to two output interface flow control profiles. (You can configure up to two CNPs with output stanzas.)

- For QFabric systems, you can configure one output interface flow control profile per Node device. (You can configure one CNP with an output stanza per Node device.)

There are a total of four output flow control profiles.

The system has a default output flow control profile that is applied to all Ethernet interfaces when the CNP attached to the interface has only an input stanza and does not include an output stanza. The default profile responds to PFC pause messages received on queue 3 (for priority 3, for the default fcoe forwarding class) and on queue 4 (for priority 4, for the default no-loss forwarding class), and is effective only if PFC is configured on those priorities in the CNP input stanza.

Additionally, the system has two internal output flow control profiles that it applies automatically to fabric (FTE) ports and to native FC interfaces (NP_Ports). When the switch is not part of a QFabric system, the profile normally used for FTE ports is available for user configuration and provides a second user-configurable profile. (That is why standalone switches have two user-configurable output flow control profiles, but Node devices on a QFabric system have only one user-configurable output flow control profile.)

Because one output CNP can configure PFC pause response on multiple output queues (priorities), one user-configurable output CNP is usually flexible enough to specify the desired PFC response on all programmed interfaces.

NOTE: Each port can use one output flow control profile. You cannot apply more than one profile to one port.

Output flow control profiles can be expressed in table format. For example, Table 54 on page 184 shows the default output flow control profile that pauses priorities 3 and 4 on queues 3 and 4 (remember that PFC must also be enabled on code points 3 and 4 in the CNP input stanza in order for PFC to work):

Table 54: Default Output Flow Control Profile

IEEE 802.1p Priority Specified in Received PFC Frame	Paused Output Queue
0 (000)	–
1 (001)	–
2 (010)	–
3 (011)	3
4 (100)	4
5 (101)	–

Table 54: Default Output Flow Control Profile *(continued)*

IEEE 802.1p Priority Specified in Received PFC Frame	Paused Output Queue
6 (110)	–
7 (111)	–

Table 55 on page 185 is an example of a user-configured output flow control profile. Using the example from the preceding section, the CNP output stanza configures flow control on output queue 5, and also explicitly configures output flow control on queues 3 and 4 for the fcoe and no-loss forwarding classes. (If you explicitly configure an output CNP, you must explicitly configure every output queue that you want to respond to PFC messages, because the user-configured profile overrides the default profile. If this example did not include queues 3 and 4, those queues would no longer respond to received PFC messages.)

Table 55: User-Configred Output Flow Control Profile

IEEE 802.1p Priority Specified in Received PFC Frame	Paused Output Queue
0 (000)	–
1 (001)	–
2 (010)	–
3 (011)	3
4 (100)	4
5 (101)	5
6 (110)	–
7 (111)	–

Remember that you must also enable PFC on code points 3, 4, and 5 in the CNP input stanza for this configuration to work. When you configure the CNP on an interface, all ingress and egress traffic is blocked until the configuration is implemented, then the interface is unblocked and traffic resumes. During the time the interface is blocked, all queues on the interface experience packet loss.

Configuring PFC Across Layer 3 Interfaces on QFX5210, QFX5200, QFX5100, EX4600, and QFX10000 Switches

Enabling PFC on traffic flows is based on the IEEE 802.1p code point (priority) in the priority code point (PCP) field of the Ethernet frame header (sometimes known as the CoS bits). To enable PFC on traffic that crosses Layer 3 interfaces, the traffic must be classified by its IEEE 802.1p code point, not by its DSCP (or DSCP IPv6) code point.

See "Understanding PFC Functionality Across Layer 3 Interfaces" on page 214 for a conceptual overview of how to enable PFC on traffic across Layer 3 interfaces. See "Example: Configuring PFC Across Layer 3 Interfaces" on page 217 for an example of how to configure PFC on traffic that traverses Layer 3 interfaces.

Configuring DCBX (Application Protocol TLV Exchange)

For applications that require lossless transport, DCBX exchanges application protocol TLVs with the connected peer interface. By default, DCBX advertises FCoE application protocol TLVs on all interfaces that are enabled for DCBX, and by default, DCBX is enabled on all interfaces. DCBX advertises no other applications by default.

For each application (for example, iSCSI) that you want to configure for lossless transport, you must enable the interfaces which carry that application traffic to exchange DCBX protocol TLVs with the connected peer. The TLV exchange allows the peer interfaces to negotiate a compatible configuration to support the application.

If you configure DCBX to advertise any application, the default DCBX advertisement is overridden, and DCBX advertises only the configured applications. If you want an interface to advertise only the FCoE application, you do not have to configure DCBX application protocol TLV exchange; instead, you can use the default configuration.

If you want DCBX to advertise other applications, you must explicitly configure an application map and apply it to the interfaces on which you want to exchange protocol TLVs for those applications. If you want to exchange FCoE application protocol TLVs in addition to other application protocol TLVs, you must also explicitly configure the FCoE application in the application map. "Understanding DCBX Application Protocol TLV Exchange" on page 468 describes how application mapping works.

> **NOTE:** Lossless transport also requires that you enable PFC on the correct priority (IEEE 802.1p code point) on the ingress interfaces using an input CNP. If the priority you pause at the ingress interfaces is not mapped to queue 3 or queue 4 (the two output queues that are enabled for PFC pause flow control by default), then you must also enable the output queues that correspond to paused input priorities to pause using the output stanza of the CNP.

Fate Sharing Among Traffic Classes

You can configure different lossless (or lossy) traffic flows to share fate—that is, to receive the same CoS treatment.

Fate sharing is not desirable for I/O convergence. Instead of independent control of the fate of each type of flow, different types of flows receive the same treatment. Fate sharing is particularly undesirable for lossless flows. If one lossless flow experiences congestion and must be paused, that affects flows that share fate with the congested flow even if the other flows are not experiencing congestion, and also can cause ingress port congestion. If your network requires that all 802.1p priorities be lossless, you can achieve that by allowing some fate sharing among the eight priorities by spreading them across up to six lossless forwarding classes.

If the number of lossless priorities is less than or equal to the number of configured lossless forwarding classes, then you can avoid fate sharing by configuring a one-to-one mapping of forwarding classes to IEEE 802.1p code points (priorities) and output queues. (Each forwarding class should be mapped to a different output queue and classified to a different priority.)

If you want to configure different traffic flows to share fate, two fate-sharing configurations are supported: mapping one forwarding class to more than one IEEE 802.1p code point (priority), and mapping two forwarding classes to the same output queue:

1. If you map one lossless forwarding class to more than one priority, the traffic tagged with each of the priorities uses the same CoS properties associated (the CoS properties associated with the forwarding class). For example, configuring a forwarding class called fc1, mapping it to queue 1, and mapping it to code points 101 and 110 using a classifier named classify1 results in the traffic tagged with priorities 101 and 110 sharing fate:

   `user@switch#` **set class-of-service forwarding-classes class fc1 queue-num 1 no-loss**

   `user@switch#` **set class-of-service classifiers ieee-802.1 classify1 forwarding class fc1 loss-priority low code-points 101**

   `user@switch#` **set class-of-service classifiers ieee-802.1 classify1 forwarding class fc1 loss-priority low code-points 110**

   In this case, if the traffic mapped to either priority experiences congestion, both priorities are paused because they are mapped to the same forwarding class and are therefore treated similarly.

2. If you map multiple lossless forwarding classes to the same output queue, the traffic mapped to the forwarding classes uses the same output queue. This increases the amount of traffic on the queue, and can create congestion that affects all of the traffic flows that are mapped to the queue. For example, configuring two forwarding classes called fc1 and fc2, mapping both forwarding classes to queue 1, and mapping the forwarding classes to code points 101 and 110 (respectively) using a classifier named classify1, results in the traffic tagged with priorities 101 and 110 sharing fate on the same output queue:

   `user@switch#` **set class-of-service forwarding-classes class fc1 queue-num 1 no-loss**

   `user@switch#` **set class-of-service forwarding-classes class fc2 queue-num 1 no-loss**

   `user@switch#` **set class-of-service classifiers ieee-802.1 classify1 forwarding class fc1 loss-priority low code-points 101**

   `user@switch#` **set class-of-service classifiers ieee-802.1 classify1 forwarding class fc2 loss-priority low code-points 110**

   In this case, even though the two forwarding classes use different IEEE 802.1p priorities, if one forwarding class experiences congestion, it affects the other forwarding class. The reason is that if the output queue is paused because of congestion on either forwarding class, all traffic that uses that queue is paused. Since both forwarding classes are mapped to the queue, the traffic mapped to both forwarding classes is paused.

> **NOTE:** 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 (all of the forwarding classes must be lossy, or all of the forwarding classes mapped to a queue must be lossless).

Transit Switch Configuration Versus FCoE-FC Gateway Configuration

On a transit switch (all Ethernet ports, no native FC ports) that forwards FCoE traffic (or other traffic that requires lossless transport across the Ethernet network), the configuration of classifiers, lossless forwarding classes, DCBX, and PFC on ingress and egress interfaces to support lossless transport is as described in this document.

When a switch acts as an FCoE-FC gateway (if native FC interfaces are supported on your switch), the system uses native FC interfaces (NP_Ports) to connect to the FC switch (or FCoE forwarder) at the FC network edge. You cannot apply CNPs or DCBX to native FC interfaces, only to Ethernet interfaces.

On an FCoE-FC gateway, the Ethernet interface configuration of classifiers, DCBX, and PFC is the same as the Ethernet interface configuration on a transit switch. The configuration of lossless forwarding classes is also the same.

However, supporting lossless transport on native FC interfaces requires that you rewrite the IEEE 802.1p priority value *if* your network uses any priority other than 3 (IEEE code point 011) for FCoE traffic. If your network uses priority 3 for FCoE traffic, you can and should use the default configuration on native FC interfaces.

By default, native FC interfaces tag packets with priority 3 when they encapsulate the incoming FC packets in Ethernet. If your FCoE network uses a different priority than 3 for FCoE traffic, you need to rewrite the priority value to the value that your network uses on the FC interface, classify the FCoE traffic to the correct priority on the Ethernet interfaces, and enable PFC on the correct priority on the Ethernet interfaces, as described in *Understanding CoS IEEE 802.1p Priority Remapping on an FCoE-FC Gateway*.

Configuration Results and Commit Checks

Different configurations of forwarding classes and their drop attributes, classifiers, CNPs (PFC flow control), and Ethernet PAUSE (IEEE 802.3X flow control) result in different system behaviors.

Table 56 on page 189 describes the results of the possible lossless transport configurations in each case. The assumption in the *Result* column is that the system's buffer headroom calculation resulted in a successful configuration.

However, if the system calculates that there is insufficient buffer space to support the configuration, a commit check prevents you from committing the configuration on an individual Ethernet interface. For LAG interfaces, the system does not issue a commit check error but instead issues a syslog message.

NOTE: After you configure lossless transport for a LAG interface, be sure to check the syslog messages to confirm that the commit was successful.

Table 56: Results of Lossless Priority Configuration

Classifier Configuration	Congestion Notification Profile Configuration	Ethernet PAUSE (IEEE 802.3X) Configuration	Result
None (default classifier)	None	None	System default configuration. No flows are lossless. To achieve lossless behavior for the default fcoe and no-loss forwarding classes, you must configure an input CNP to enable PFC on their IEEE 802.1p code points (011 and 100 respectively).
Classifier with no lossless forwarding classes	None	None	No lossless traffic flows are configured; all traffic is best effort.
Classifier with at least one lossless forwarding class	None	None	Because no CNP is attached to interfaces, PFC is not enabled on the code point of the lossless traffic and no headroom buffer is allocated to the lossless queue, so packets can drop during periods of congestion. This configuration does not achieve lossless behavior.
None (default classifier)	PFC enabled on the fcoe and no-loss forwarding class code points (priorities)	None	The default classifier classifies traffic into two lossless forwarding classes, fcoe and no-loss. The CNP enables PFC on the priorities mapped to both lossless forwarding classes, resulting in lossless behavior for traffic mapped to the fcoe and no-loss forwarding classes.

Table 56: Results of Lossless Priority Configuration *(continued)*

Classifier Configuration	Congestion Notification Profile Configuration	Ethernet PAUSE (IEEE 802.3X) Configuration	Result
None (default classifier)	None	Flow control enabled	The system calculates buffer headroom for the physical link based on the interface MTU and the default cable length. The system does not calculate buffer headroom for individual output queues. Because Ethernet PAUSE is enabled on the link instead of PFC being enabled on the lossless priorities, the entire link is paused during periods of congestion. This configuration results in lossless behavior for all of the forwarding classes on the link, but because all traffic is paused, this can cause greater overall network congestion.
Classifier with at least one lossless forwarding class	PFC enabled on the lossless forwarding class code points (priorities)	None	Headroom buffer allocated only to priorities that are mapped to the lossless forwarding classes and on which PFC is enabled. This configuration achieves lossless behavior for the lossless forwarding classes.
Classifier with no lossless forwarding classes	None	Flow control enabled	The system calculates buffer headroom for the physical link based on the interface MTU and the default cable length, and it pauses all traffic on the link during periods of congestion.
Classifier with at least one lossless forwarding class	None	Flow control enabled	The system calculates buffer headroom for the physical link based on the interface MTU and the default cable length, and it pauses all traffic on the link during periods of congestion.
Classifier with at least one lossless forwarding class	PFC enabled on the lossless forwarding class code points (priorities)	Flow control enabled on a *different* interface than the interface with the CNP	The system checks the available buffer space for both the PFC-enabled priorities and for the other link. If sufficient buffer space is available, the lossless forwarding classes configured with PFC on one interface and also all of the traffic on the link with Ethernet PAUSE enabled achieve lossless behavior.

NOTE: If you attempt to configure both PFC and Ethernet PAUSE on a link, the system returns a commit error. PFC and Ethernet PAUSE are mutually exclusive configurations on an interface.

Configuration Rules and Recommendations

Keep in mind the following configuration rules and recommendations when you configure lossless traffic flows:

- You can configure a maximum of six lossless forwarding classes (forwarding classes with the no-loss packet drop attribute).
- All forwarding classes that you map to the same queue must have the same packet drop attribute (all of the forwarding classes must be lossy, or all of the forwarding classes must be lossless).
- Do not configure weighted random early detection (WRED) on lossless forwarding classes. (Do not associate a drop profile with a forwarding class that has the no-loss packet drop attribute.)
- On switches that use different forwarding classes and output queues for unicast and multidestination traffic, you cannot configure flow control to pause a multidestination output queue. You can configure PFC flow control only to pause unicast output queues.
- On switches that use different forwarding classes and output queues for unicast and multidestination traffic, forwarding classes mapped to multidestination queues (queues 8 through 11) cannot have the no-loss packet drop attribute. (Multidestination forwarding classes cannot be configured as lossless forwarding classes.)

Lossless Transport Features Introduced in Junos OS Release 12.3 (Legacy Non-ELS CLI)

Support for lossless transport introduced in Junos OS Release 12.3 includes:

- Configuring up to six lossless forwarding classes.
- Configuring PFC pause on output queues to program the output queues that can respond to PFC pause messages received from the connected peer. The priorities you pause on output queues must match the priorities on which you enable PFC on the corresponding ingress interfaces. For example, if you program output queues to pause priorities 3 (011) and 5 (101), then you must also enable pause on priorities 3 and 5 on the corresponding ingress interfaces. Configuring flow control on the output queues and enabling PFC on the corresponding input queues allows you to pause up to six priorities (forwarding classes).
- Controlling the headroom buffer on Ethernet interfaces by configuring the maximum receive unit (MRU) size for the traffic mapped to an IEEE 802.1p priority (configured per priority) and the length of the

attached cable (configured per interface). The MRU size can range up to full jumbo packet size (9216 bytes).

- Remapping (rewriting) IEEE 802.1p priorities on native Fibre Channel (FC) interfaces when the system is acting as an FCoE-FC gateway. If the Ethernet (FCoE) network uses a different IEEE 802.1p priority than priority 3 (011) for FCoE traffic, then you can use priority remapping to classify FCoE traffic into a lossless forwarding class mapped to that different priority (see *Understanding CoS IEEE 802.1p Priority Remapping on an FCoE-FC Gateway*).

Lossless transport still requires configuring previously existing features, including enabling PFC on the lossless priorities on ingress interfaces, and configuring classifiers to classify incoming traffic into lossless forwarding classes based on the IEEE 802.1p priority tag of the packet.

> **NOTE:** If you expect a large amount of lossless traffic on your network and configure multiple lossless traffic classes, ensure that you reserve enough scheduling resources (bandwidth) and lossless headroom buffer space to support the lossless flows. ("Understanding CoS Buffer Configuration" on page 635 describes how to configure buffers and provides a recommended buffer configuration for networks with larger amounts of lossless traffic.)

Backward Compatibility with Junos OS Releases Earlier Than Release 12.3 (Legacy Non-ELS CLI)

The addition of the no-loss packet drop attribute to forwarding class configuration means that when you upgrade from an earlier release to Junos OS Release 12.3, the new software might not preserve the lossless forwarding class configuration of the fcoe and no-loss forwarding classes.

If you used the default forwarding class configuration for the fcoe and no-loss forwarding classes, the CoS configuration is backward-compatible. You do not have to do anything to preserve the lossless behavior of traffic that uses those forwarding classes when you upgrade to Junos OS Release 12.3. (This is because the default configuration of these two forwarding classes includes the no-loss packet drop attribute.)

However, if you explicitly configured the fcoe or the no-loss forwarding class by including the **set forwarding-classes class *forwarding-class-name* queue-num *queue-number*** statement at the **[edit class-of-service]** hierarchy level, then those forwarding classes are no longer lossless, they are lossy. (They are lossy because explicit configuration in releases earlier than Junos OS Release 12.3 did not use the no-loss packet drop attribute.) In Junos OS Release 12.3 and later, you must include the no-loss packet drop attribute in explicit forwarding class configurations to configure a lossless forwarding class.

For example, before Junos OS Release 12.3, the following explicit configuration resulted in a lossless forwarding class:

```
user@switch# set class-of-service forwarding-classes class fcoe queue-num 3
```

However, in Junos OS Release 12.3, this configuration is lossy because it does not include the no-loss packet drop attribute. To preserve lossless behavior, after upgrading to Junos OS Release 12.3, you need to add the no-loss drop attribute:

`user@switch#` **set class-of-service forwarding-classes class fcoe queue-num 3 no-loss**

Alternatively, you can delete the explicit configuration before you upgrade to Junos OS Release 12.3 so that the system uses the default forwarding class, which is lossless:

`user@switch#` **delete class-of-service forwarding-classes class fcoe queue-num 3**

> **NOTE:** The explicit configuration of other forwarding classes does not affect the lossless (or lossy) state of the fcoe and no-loss forwarding classes, because only the fcoe and no-loss forwarding classes were lossless forwarding classes before Junos OS Release 12.3. For example, if you explicitly configured the best-effort forwarding class but you used the default fcoe and no-loss forwarding classes in Junos OS Release 12.2, then when you upgrade to Junos OS Release 12.3, the fcoe and no-loss forwarding classes are still lossless (and the best-effort forwarding classes retains its explicit configuration).

> **NOTE:** To achieve lossless behavior for the traffic belonging to any forwarding class, you must also use a CNP to enable PFC on the IEEE 802.1p priority mapped to the forwarding class and apply the CNP to the relevant interfaces, and ensure that DCBX exchanges the protocol TLVs for the application with the connected peer.

RELATED DOCUMENTATION

Understanding DCBX Application Protocol TLV Exchange | 468

Understanding CoS Flow Control (Ethernet PAUSE and PFC) | 197

Understanding PFC Functionality Across Layer 3 Interfaces | 214

Example: Configuring Lossless FCoE Traffic When the Converged Ethernet Network Does Not Use IEEE 802.1p Priority 3 for FCoE Traffic (FCoE Transit Switch) | 565

Example: Configuring Two or More Lossless FCoE Priorities on the Same FCoE Transit Switch Interface | 576

Example: Configuring Two or More Lossless FCoE IEEE 802.1p Priorities on Different FCoE Transit Switch Interfaces | 587

Configuring CoS PFC (Congestion Notification Profiles)

A congestion notification profile (CNP) enables priority-based flow control (PFC) on specified IEEE 802.1p priorities (code points). A CNP has two components:

- Input CNP:
  - Enable PFC on a specified priority.
  - Configure the maximum receive unit (MRU) on an interface for traffic that matches the PFC priority (optional).
  - Specify the length of the attached cable on the ingress interface (optional)
- Output CNP (optional): Configure flow control to enable PFC pause on specific output queues for specified priorities.

> **NOTE:** By default, output queues 3 and 4 (which are mapped to default lossless forwarding classes **fcoe** and **no-loss**, respectively) are configured to respond to PFC pause messages received from the connected peer on priorities 3 and 4 (code points 011 and 100, respectively). If you explicitly configure flow control on any output queue, you must configure flow control on every output queue that you want to respond to pause messages. (The explicit configuration overrides the default configuration.)
>
> To achieve lossless behavior, the output queue priorities on which you enable PFC flow control must match the PFC priorities on which you enable PFC on the input interfaces. For example, if you program output queues to pause priorities 3 (011) and 5 (101) in the output component of the CNP, then you must also enable pause on priorities 3 and 5 on the input component of the CNP. (In addition, the forwarding classes mapped to the paused output queues must be lossless forwarding classes.)

Associating a CNP with an interface enables PFC on the ingress traffic that matches the priority specified in the input CNP, and programs the queues listed in the output CNP to pause when the interface receives a PFC pause message from the connected peer. Configure PFC on a priority end to end along the entire data path to create a lossless lane of traffic on the network.

NOTE: You must enable PFC on the priority used by FCoE traffic on ingress interfaces (input CNP). Enable PFC on the FCoE priority on every interface that carries FCoE traffic. By convention, FCoE traffic uses priority 3 (code point **011**), which maps to queue 3. If your network uses priority 3 for FCoE traffic, the default forwarding class and classifier configuration support lossless transport, but you must still configure a CNP and apply it to the correct ingress interfaces to enable PFC and achieve lossless transport.

If your network does not use priority 3 for FCoE traffic, you need to configure a classifier that classifies FCoE traffic into a lossless forwarding class, based on the priority your network uses for FCoE traffic. If you are not using the default lossless forwarding class configuration, then you also need to ensure that the output queue mapped to the lossless FCoE forwarding class is programmed to pause.

You can attach only one CNP to an interface. There is no limit to the total number of CNPs you can create.

Configuring a CNP consists of:

- Naming the CNP.
- Specifying the IEEE 802.1 code point (priority) on which you want to enable PFC on ingress interfaces (input CNP).
- Optionally, specifying the MRU and the length of the attached cable on ingress interfaces (input CNP).
- Optionally, configuring flow control (PFC pause) on specified output queues if you want queues other than queues 3 and 4 to respond to pause messages received from the connected peer (output CNP).
- Mapping the CNP to an interface.

NOTE: Configuring or changing PFC on an interface blocks the entire port until the PFC change is completed. After a PFC change is completed, the port is unblocked and traffic resumes. Blocking the port stops ingress and egress traffic, and causes packet loss on all queues on the port until the port is unblocked.

NOTE: On QFX5100, QFX5200, and QFX5210, once the headroom buffer is exhausted, any new CNP configuration is not allocated headroom buffer, even if headroom buffer is freed by deletion of an existing CNP. CNP configuration has to be applied again to re-allocate the headroom buffer.

1. Enable PFC on the desired priority in the input CNP and optionally configure the interface MRU for traffic on that priority:

   `[edit class-of-service]`

   `user@switch#` **set congestion-notification-profile *cnp-name* input ieee-802.1 code-point *code-point bits* pfc mru *mru-value***

   For example, to configure a CNP named **fcoe-cnp** that enables PFC on IEEE 802.1 code point **011** and configures an MRU value of **2240**:

   `[edit class-of-service]`

   `user@switch#` **set congestion-notification-profile fcoe-cnp input ieee-802.1 code-point 011 pfc mru 2240**

2. (Optional) Configure the length of the cable attached to the ingress interface:

   `[edit class-of-service]`

   `user@switch#` **set congestion-notification-profile *cnp-name* input cable-length *cable-length-value***

   For example, to configure a CNP named **fcoe-cnp** that sets the length of the ingress interface cable to **100** meters:

   `[edit class-of-service]`

   `user@switch#` **set congestion-notification-profile fcoe-cnp input cable-length 100**

3. (Optional) Configure flow control on output queues:

   `[edit class-of-service]`

   `user@switch#` **set congestion-notification-profile *cnp-name* output ieee-802.1 code-point *code-point-bits* flow-control-queue [*queue* | *list-of-queues*]**

   For example, to configure a CNP named **fcoe-cnp** that enables PFC pause flow control on output queues 3 and 5 for FCoE traffic that uses priority 3 (code point **011**) and on output queue 4 for traffic that uses priority 4 (code point **100**):

   `[edit class-of-service]`

   `user@switch#` **set congestion-notification-profile *cnp-name* output ieee-802.1 code-point 011 flow-control-queue [3 5]**

   `user@switch#` **set congestion-notification-profile *cnp-name* output ieee-802.1 code-point 100 flow-control-queue 4**

4. Map the CNP to an interface:

   `[edit class-of-service]`

```
user@switch# set interfaces interface congestion-notification-profile cnp-name
```

For example, to map the CNP **fcoe-cnp** to the interface **xe-0/0/7**:

```
[edit class-of-service]
user@switch# set interfaces xe-0/0/7 congestion-notification-profile fcoe-cnp
```

RELATED DOCUMENTATION

Example: Configuring CoS PFC for FCoE Traffic | 490

Assigning CoS Components to Interfaces | 77

Monitoring Interfaces That Have CoS Components

Understanding CoS Flow Control (Ethernet PAUSE and PFC) | 197

Understanding CoS IEEE 802.1p Priorities for Lossless Traffic Flows | 173

Understanding CoS Flow Control (Ethernet PAUSE and PFC)

IN THIS SECTION

Flow control supports lossless transmission by regulating traffic flows to avoid dropping frames during periods of congestion. Flow control stops and resumes the transmission of network traffic between two connected peer nodes on a full-duplex Ethernet physical link. Controlling the flow by pausing and restarting it prevents buffers on the nodes from overflowing and dropping frames. You configure flow control on a per-interface basis.

Two methods of peer-to-peer flow control are supported:

- IEEE 802.3X Ethernet PAUSE

  NOTE: QFX10000 switches do not support Ethernet PAUSE. Information about Ethernet PAUSE does not apply to QFX10000 switches.

  OCX Series switches support symmetric Ethernet PAUSE flow control on Layer 3 tagged interfaces. OCX Series switches do not support asymmetric Ethernet PAUSE flow control. Information about asymmetric flow control does not apply to OCX Series switches.

- IEEE 802.1Qbb priority-based flow control (PFC)

  NOTE: OCX Series switches do not support PFC or lossless Layer 2 transport. Information about PFC, lossless transport, and congestion notification profiles does not apply to OCX Series switches.

  NOTE: QFX10002-60C devices do not support PFC and lossless queues; that is, the default lossless queues (fcoe and no-loss) will be lossy queues.

Video: Why Use PFC in a Data Center Network?

General Information about Ethernet PAUSE and PFC and When to Use Them

Ethernet PAUSE and PFC are link-level flow control mechanisms.

> **NOTE:** For end-to-end congestion control for best-effort traffic, see "Understanding CoS Explicit Congestion Notification" on page 274.

Ethernet PAUSE pauses transmission of all traffic on a physical Ethernet link.

PFC decouples the pause function from the physical Ethernet link and enables you to divide traffic on one link into eight priorities. You can think of the eight priorities as eight “lanes” of traffic that are mapped to forwarding classes and output queues. Each priority maps to a 3-bit IEEE 802.1p CoS code point value in the VLAN header. You can enable PFC on one or more priorities (IEEE 802.1p code points) on a link. When PFC-enabled traffic is paused on a link, traffic that is not PFC-enabled continues to flow (or is dropped if congestion is severe enough).

Use Ethernet PAUSE when you want to prevent packet loss on all of the traffic on a link. Use PFC to prevent traffic loss only on a specified type of traffic that require lossless treatment, for example, Fibre Channel over Ethernet (FCoE) traffic.

> **NOTE:** Depending on the amount of traffic on a link or assigned to a priority, pausing traffic can cause ingress port congestion and spread congestion through the network.

Ethernet PAUSE and PFC are mutually exclusive configurations on an interface. Attempting to configure both Ethernet PAUSE and PFC on a link causes a commit error.

By default, all forms of flow control are disabled. You must explicitly enable flow control on interfaces to pause traffic.

Ethernet PAUSE

IN THIS SECTION

Ethernet PAUSE is a congestion relief feature that works by providing link-level flow control for all traffic on a full-duplex Ethernet link. Ethernet PAUSE works in both directions on the link. In one direction, an interface generates and sends Ethernet PAUSE messages to stop the connected peer from sending more traffic. In the other direction, the interface responds to Ethernet PAUSE messages it receives from the connected peer to stop sending traffic.

NOTE: QFX10000 switches do not support Ethernet PAUSE. Information about Ethernet PAUSE does not apply to QFX10000 switches.

OCX Series switches support symmetric Ethernet PAUSE flow control on Layer 3 tagged interfaces. OCX Series switches do not support asymmetric Ethernet PAUSE flow control. Information about asymmetric flow control does not apply to OCX Series switches.

Ethernet PAUSE also works on aggregated Ethernet interfaces. For example, if the connected peer interfaces are called Node A and Node B:

- When the receive buffers on interface Node A reach a certain level of fullness, the interface generates and sends an Ethernet PAUSE message to the connected peer (interface Node B) to tell the peer to stop sending frames. The Node B buffers store frames until the time period specified in the Ethernet PAUSE frame elapses; then Node B resumes sending frames to Node A.
- When interface Node A receives an Ethernet PAUSE message from interface Node B, interface Node A stops transmitting frames until the time period specified in the Ethernet PAUSE frame elapses; then Node A resumes transmission. (The Node A transmit buffers store frames until Node A resumes sending frames to Node B.)

  In this scenario, if Node B sends an Ethernet PAUSE frame with a time value of 0 to Node A, the 0 time value indicates to Node A that it can resume transmission. This happens when the Node B buffer empties to below a certain threshold and the buffer can once again accept traffic.

Symmetric flow control means an interface has the same Ethernet PAUSE configuration in both directions. The Ethernet PAUSE generation and Ethernet PAUSE response functions are both configured as enabled, or they are both disabled. You configure symmetric flow control by including the **flow-control** statement at the **[edit interfaces *interface-name* ether-options]** hierarchy level.

Asymmetric flow control allows you to configure the Ethernet PAUSE functionality in each direction independently on an interface. The configuration for generating Ethernet PAUSE messages and for responding to Ethernet PAUSE messages does not have to be the same. It can be enabled in both directions, disabled in both directions, or enabled in one direction and disabled in the other direction. You configure asymmetric flow control by including the **configured-flow-control** statement at the **[edit interfaces *interface-name* ether-options]** hierarchy level.

On any particular interface, symmetric and asymmetric flow control are mutually exclusive. Asymmetric flow control overrides and disables symmetric flow control. (If PFC is configured on an interface, you cannot commit an Ethernet PAUSE configuration on the interface. Attempting to commit an Ethernet PAUSE configuration on an interface with PFC enabled on one or more queues results in a commit error. To commit the PAUSE configuration, you must first delete the PFC configuration.) Both symmetric and asymmetric flow control are supported.

Symmetric Flow Control

Symmetric flow control configures both the receive and transmit buffers in the same state. The interface can both send Ethernet PAUSE messages and respond to them (flow control is enabled), or the interface cannot send Ethernet PAUSE messages or respond to them (flow control is disabled).

When you enable symmetric flow control on an interface, the Ethernet PAUSE behavior depends on the configuration of the connected peer. With symmetric flow control enabled, the interface can perform any Ethernet PAUSE functions that the connected peer can perform. (When symmetric flow control is disabled, the interface does not send or respond to Ethernet PAUSE messages.)

Asymmetric Flow Control

Asymmetric flow control enables you to specify independently whether or not the interface receive buffer generates and sends Ethernet PAUSE messages to stop the connected peer from transmitting traffic, and whether or not the interface transmit buffer responds to Ethernet PAUSE messages it receives from the connected peer and stops transmitting traffic. The receive buffer configuration determines if the interface transmits Ethernet PAUSE messages, and the transmit buffer configuration determines if the interface receives and responds to Ethernet PAUSE messages:

- Receive buffers on—Enable Ethernet PAUSE transmission (generate and send Ethernet PAUSE frames)
- Transmit buffers on—Enable Ethernet PAUSE reception (respond to received Ethernet PAUSE frames)

You must explicitly set the flow control for both the receive buffer and the transmit buffer (**on** or **off**) to configure asymmetric Ethernet PAUSE. Table 57 on page 201 describes the configured flow control state when you set the receive (Rx) and transmit (Tx) buffers on an interface:

Table 57: Asymmetric Ethernet PAUSE Flow Control Configuration

Receive (Rx) Buffer	Transmit (Tx) Buffer	Configured Flow Control State
On	**Off**	Interface generates and sends Ethernet PAUSE messages. Interface does not respond to Ethernet PAUSE messages (interface continues to transmit even if peer requests that the interface stop sending traffic).
Off	**On**	Interface responds to Ethernet PAUSE messages received from the connected peer, but does not generate or send Ethernet PAUSE messages. (The interface does not request that the connected peer stop sending traffic.)
On	**On**	Same functionality as symmetric Ethernet PAUSE. Interface generates and sends Ethernet PAUSE messages and responds to received Ethernet PAUSE messages.
Off	**Off**	Ethernet PAUSE flow control is disabled.

The configured flow control is the Ethernet PAUSE state configured on the interface.

On 1-Gigabit Ethernet interfaces, autonegotiation of Ethernet PAUSE with the connected peer is supported. (Autonegotiation on 10-Gigabit Ethernet interfaces is not supported.) Autonegotiation enables the interface to exchange state advertisements with the connected peer so that the two devices can agree on the Ethernet PAUSE configuration. Each interface advertises its flow control state to the connected peer using a combination of the Ethernet PAUSE and ASM_DIR bits, as described in Table 58 on page 202:

Table 58: Flow Control State Advertised to the Connected Peer (Autonegotiation)

Rx Buffer State	Tx Buffer State	PAUSE Bit	ASM_DIR Bit	Description
Off	**Off**	**0**	**0**	The interface advertises no Ethernet PAUSE capability. This is equivalent to disabling flow control on an interface.
On	**On**	**1**	**0**	The interface advertises symmetric flow control (both the transmission of Ethernet PAUSE messages and the ability to receive and respond to Ethernet PAUSE messages).
On	**Off**	**0**	**1**	The interface advertises asymmetric flow control (the transmission of Ethernet PAUSE messages, but not the ability to receive and respond to Ethernet PAUSE messages).

Table 58: Flow Control State Advertised to the Connected Peer (Autonegotiation) ***(continued)***

Rx Buffer State	Tx Buffer State	PAUSE Bit	ASM_DIR Bit	Description
Off	**On**	**1**	**1**	The interface advertises both symmetric and asymmetric flow control. Although the interface does not generate and send Ethernet PAUSE requests to the peer, the interface supports both symmetric and asymmetric Ethernet PAUSE configuration on the peer because the peer is not affected if the peer does not receive Ethernet PAUSE requests. (If the interface responds to the peer's Ethernet PAUSE requests, that is sufficient to support either symmetric or asymmetric flow control on the peer.)

The flow control configuration on each switch interface interacts with the flow control configuration of the connected peer. Each peer advertises its state to the other peer. The interaction of the flow control configuration of the peers determines the flow control behavior (resolution) between them, as shown in Table 59 on page 204. The first four columns show the Ethernet PAUSE configuration on the local QFX Series or EX4600 switch and on the connected peer (also known as the *link partner*). The last two columns show the Ethernet PAUSE resolution that results from the local and peer configurations on each interface. This illustrates how the Ethernet PAUSE configuration of each interface affects the Ethernet PAUSE behavior on the other interface.

NOTE: In the Resolution columns of the table, disabling Ethernet PAUSE transmit means that the interface receive buffers do not generate and send Ethernet PAUSE messages to the peer. Disabling Ethernet PAUSE receive means that the interface transmit buffers do not respond to Ethernet PAUSE messages received from the peer.

Table 59: Asymmetric Ethernet PAUSE Behavior on Local and Peer Interfaces

Local Interface (QFX Series or EX4600 Switch)		Peer Interface			
PAUSE Bit	ASM_DIR Bit	PAUSE Bit	ASM_DIR Bit	Local Resolution	Peer Resolution
0	0	Don't care	Don't care	Disable Ethernet PAUSE transmit and receive	Disable Ethernet PAUSE transmit and receive
0	1	0	Don't care	Disable Ethernet PAUSE transmit and receive	Disable Ethernet PAUSE transmit and receive
0	1	1	0	Disable Ethernet PAUSE transmit and receive	Disable Ethernet PAUSE transmit and receive
0	1	1	1	Enable Ethernet PAUSE transmit and disable Ethernet PAUSE receive	Disable Ethernet PAUSE transmit and enable Ethernet PAUSE receive
1	0	0	Don't care	Disable Ethernet PAUSE transmit and receive	Disable Ethernet PAUSE transmit and receive
1	0	1	Don't care	Enable Ethernet PAUSE transmit and receive	Enable Ethernet PAUSE transmit and receive
1	1	0	0	Disable Ethernet PAUSE transmit and receive	Disable Ethernet PAUSE transmit and receive
1	1	0	1	Enable Ethernet PAUSE receive and disable Ethernet PAUSE transmit	Enable Ethernet PAUSE transmit and disable Ethernet PAUSE receive
1	1	Don't care	Don't care	Enable Ethernet PAUSE transmit and receive	Enable Ethernet PAUSE transmit and receive

NOTE: For your convenience, Table 59 on page 204 replicates Table 28B-3 of Section 2 of the IEEE 802.X specification.

PFC

PFC is a lossless transport and congestion relief feature that works by providing granular link-level flow control for each IEEE 802.1p code point (priority) on a full-duplex Ethernet link. When the receive buffer on a switch interface fills to a threshold, the switch transmits a pause frame to the sender (the connected peer) to temporarily stop the sender from transmitting more frames. The buffer threshold must be low enough so that the sender has time to stop transmitting frames and the receiver can accept the frames already on the wire before the buffer overflows. The switch automatically sets queue buffer thresholds to prevent frame loss.

When congestion forces one priority on a link to pause, all of the other priorities on the link continue to send frames. Only frames of the paused priority are not transmitted. When the receive buffer empties below another threshold, the switch sends a message that starts the flow again.

You configure PFC using a congestion notification profile (CNP). A CNP has two parts:

- Input—Specify the code point (or code points) on which to enable PFC, and optionally specify the maximum receive unit (MRU) and the cable length between the interface and the connected peer interface.
- Output—Specify the output queue or output queues that respond to pause messages from the connected peer.

You apply a PFC configuration by configuring a CNP on one or more interfaces. Each interface that uses a particular CNP is enabled to pause traffic identified by the priorities (code points) specified in that CNP. You can configure one CNP on an interface, and you can configure different CNPs on different interfaces. When you configure a CNP on an interface, ingress traffic that is mapped to a priority that the CNP enables for PFC is paused whenever the queue buffer fills to the pause threshold. (The pause threshold is not user-configurable.)

Configure PFC for a priority end to end along the entire data path to create a lossless lane of traffic on the network. You can selectively pause the traffic in any queue without pausing the traffic for other queues on the same link. You can create lossless lanes for traffic such as FCoE, LAN backup, or management, while using standard frame-drop congestion management for IP traffic on the same link.

Potential consequences of flow control are:

- Ingress port congestion (configuring too many lossless flows can cause ingress port congestion)
- A paused priority that causes upstream devices to pause the same priority, thus spreading congestion back through the network

By definition, PFC supports symmetric pause only (as opposed to Ethernet PAUSE, which supports symmetric and asymmetric pause). With symmetric pause, a device can:

- Transmit pause frames to pause incoming traffic. (You configure this using the input stanza of a congestion notification profile.)

- Receive pause frames and stop sending traffic to a device whose buffer is too full to accept more frames. (You configure this using the output stanza of a congestion notification profile.)

Receiving a PFC frame from a connected peer pauses traffic on egress queues based on the IEEE 802.1p priorities that the PFC pause frame identifies. The priorities are 0 through 7. By default, the priorities map to queue numbers 0 through 7, respectively, and to specific forwarding classes, as shown in Table 60 on page 206:

Table 60: Default PFC Priority to Queue and Forwarding Class Mapping

IEEE 802.1p Priority (Code Point)	Queue	Forwarding Class
0 (000)	0	best-effort
1 (001)	1	best-effort
2 (010)	2	best-effort
3 (011)	3	fcoe
4 (100)	4	no-loss
5 (101)	5	best-effort
6 (110)	6	network-control
7 (111)	7	network-control

For example, a received PFC pause frame that pauses priority 3 pauses output queue 3. If you do not want to use the default configuration, you can configure customized mapping of priorities to queues and forwarding classes.

NOTE: By convention, deployments with converged server access typically use IEEE 802.1p priority 3 for FCoE traffic. The default configuration sets the **fcoe** forwarding class as a lossless forwarding class that is mapped to queue 3. The default classifier maps incoming priority 3 traffic to the **fcoe** forwarding class. *However, you must apply PFC to the entire FCoE data path to configure the end-to-end lossless behavior that FCoE traffic requires.*

If your network uses priority 3 for FCoE traffic, we recommend that you use the default configuration. If your network uses a priority other than 3 for FCoE traffic, you can configure lossless FCoE transport on any IEEE 80.21p priority as described in "Understanding CoS IEEE 802.1p Priorities for Lossless Traffic Flows" on page 173 and *Understanding CoS IEEE 802.1p Priority Remapping on an FCoE-FC Gateway*.

To enable PFC on a priority:

1. Specify the IEEE 802.1p code point to pause in the input stanza of a CNP.
2. If you are not using the default lossless forwarding classes, specify the IEEE 802.1p code point to pause and the corresponding output queue in the output stanza of the CNP.
3. Apply the CNP to the ingress interfaces on which you want to pause the traffic.
4. If you are not using the default lossless forwarding classes, apply the CNP to the ingress interfaces on which you want to pause the traffic.

CAUTION: Any change to the PFC configuration on a port temporarily blocks the entire port (not just the priorities affected by the PFC change) so that the port can implement the change, then unblocks the port. Blocking the port stops ingress and egress traffic, and causes packet loss on all queues on the port until the port is unblocked.

A change to the PFC configuration means any change to a CNP, including changing the input portion of the CNP (enabling or disabling PFC on a priority, or changing the MRU or cable-length values) or changing the output portion of the CNP that enables or disables output flow control on a queue. A PFC configuration change only affects ports that use the changed CNP.

The following actions change the PFC configuration:

- Deleting or disabling a PFC configuration (input or output) in a CNP that is in use on one or more interfaces. For example:
  1. An existing CNP with an input stanza that enables PFC on priorities 3, 5, and 6 is configured on interfaces xe-0/0/20 and xe-0/0/21.
  2. We disable the PFC configuration for priority 6 in the input CNP, and then commit the configuration.
  3. The PFC configuration change causes all traffic on interfaces xe-0/0/20 and xe-0/0/21 to stop until the PFC change has been implemented. When the PFC change has been implemented, traffic resumes.
- Configuring a CNP on an interface. (This changes the PFC state by enabling PFC on one or more priorities.)
- Deleting a CNP from an interface. (This changes the PFC state by disabling PFC on one or more priorities.)

When you associate the CNP with an interface, the interface uses PFC to send pause requests when the output queue buffer for the lossless traffic fills to the pause threshold.

On switches that use different classifiers for unicast and multidestination traffic, you can map a unicast queue (queue 0 through 7) and a multidestination queue (queue 8, 9, 10, or 11) to the same IEEE 802.1p code point (priority) so that both unicast and multicast traffic use that priority. However, do not map multidestination traffic to lossless output queues. Starting with Junos OS Release 12.3, you can map one priority to multiple output queues.

NOTE: You can attach a maximum of one CNP to an interface, but you can create an unlimited number of CNPs that explicitly configure only the input stanza and use the default output stanza.

The output stanza of the CNP maps to a profile that interfaces use to respond to pause messages received from the connected peer. On standalone switches, you can create two CNPs with an explicitly configured output stanza.

When a switch is a Node device in a QFabric system, you can create one CNP with an explicitly configured output stanza. (One fewer profile is available on QFabric systems because the system needs a default profile for fabric interfaces, which are not used as fabric interfaces when the switches are not part of a QFabric system. "Understanding CoS IEEE 802.1p Priorities for Lossless Traffic Flows" on page 173 describes configuring output flow control.

Lossless Transport Support Summary

The switch supports up to six lossless forwarding classes. For lossless transport, you must enable PFC on the IEEE 802.1p priorities (code points) mapped to lossless forwarding classes.

CAUTION: Any change to the PFC configuration on a port temporarily blocks the entire port (not just the priorities affected by the PFC change) so that the port can implement the change, then unblocks the port. Blocking the port stops ingress and egress traffic, and causes packet loss on all queues on the port until the port is unblocked.

The following limitation applies to support lossless transport on QFabric systems only:

- The internal fiber cable length from the QFabric system Node device to the QFabric system Interconnect device cannot exceed 150 meters.

The default CoS configuration provides two lossless forwarding classes, *fcoe* and *no-loss*. If you explicitly configure lossless forwarding classes, you must include the **no-loss** packet drop attribute to enable lossless behavior, or the traffic is not lossless. For both default and explicit lossless forwarding class configuration, you must configure CNP input stanzas to enable PFC on the priority of the lossless traffic and apply the CNPs to ingress interfaces.

NOTE: The information in this note applies only to systems that do not run the ELS CLI.

Junos OS Release 12.2 introduced changes to the way the switch handles lossless forwarding classes (including the default **fcoe** and **no-loss** forwarding classes).

In Junos OS Release 12.1, either explicitly configuring the **fcoe** and **no-loss** forwarding classes or using the default configuration for these forwarding classes resulted in the same lossless behavior for traffic mapped to those forwarding classes.

However, in Junos OS Release 12.2, 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.

If your CoS configuration from Junos OS Release 12.1 or earlier includes the explicit configuration of the **fcoe** or the **no-loss** forwarding class, then when you upgrade to Junos OS Release 12.2, those forwarding classes are not lossless. To preserve the lossless treatment of these forwarding classes, delete the the explicit **fcoe** and **no-loss** forwarding class configuration before you upgrade to Junos OS Release 12.2.

See *Overview of CoS Changes Introduced in Junos OS Release 12.2* for detailed information about this change and how to delete an existing lossless configuration.

In Junos OS Release 12.3, the default behavior of the **fcoe** and **no-loss** forwarding classes is the same as in Junos OS Release 12.2. However, in Junos OS Release 12.3, you can configure up to six lossless forwarding classes. All explicitly configured lossless forwarding classes must include the new **no-loss** packet drop attribute or the forwarding class is lossy.

"Understanding CoS IEEE 802.1p Priorities for Lossless Traffic Flows" on page 173 provides detailed information about the explicit configuration of lossless priorities and about the default configuration of lossless priorities, including the input and output stanzas of the CNP.

NOTE: PFC and Ethernet PAUSE are used only on Ethernet interfaces. Fabric (fte) ports on QFabric systems (Node device fabric ports and Interconnect device fabric ports) use link-layer flow control (LLFC) to ensure the appropriate treatment of lossless traffic.

Release History Table

Release	Description
12.3	Starting with Junos OS Release 12.3, you can map one priority to multiple output queues.

RELATED DOCUMENTATION

Understanding DCB Features and Requirements | 450
Understanding CoS Explicit Congestion Notification | 274
Configuring CoS PFC (Congestion Notification Profiles) | 194
Example: Configuring CoS PFC for FCoE Traffic | 490

Enabling and Disabling CoS Symmetric Ethernet PAUSE Flow Control

Ethernet PAUSE flow control is a congestion relief feature that works by providing link-level flow control for all traffic on a full-duplex Ethernet link, including Ethernet links that belong to Ethernet link aggregated (LAG) interfaces. Ethernet PAUSE works in both directions on the link. In one direction, an interface generates and sends PAUSE messages to stop the connected peer from sending more traffic. In the other direction, the interface responds to PAUSE messages it receives from the connected peer to stop sending traffic.

Symmetric flow control means that an interface has the same PAUSE configuration in both directions. The PAUSE generation and PAUSE response functions are both configured as enabled, or they are both disabled.

Asymmetric flow control allows you to configure the PAUSE functionality in each direction independently on an interface. The configuration for generating PAUSE messages and for responding to PAUSE messages does not have to be the same. It can be enabled in both directions, disabled in both directions, or enabled in one direction and disabled in the other direction. If you do not want to PAUSE all of the traffic on a link, you can use priority-based flow control (PFC) to selectively pause traffic based on its IEEE 802.1p code point.

NOTE: OCX Series switches do not support PFC.

On any particular interface, symmetric and asymmetric flow control are mutually exclusive. If you attempt to configure both features, the switch returns a commit error. Ethernet PAUSE and PFC are also mutually exclusive features, so you cannot configure both of them on the same interface. If you attempt to configure both Ethernet PAUSE and PFC on an interface, the switch returns a commit error.

By default, all flow control features are disabled. You enable symmetric flow control on the interfaces on which you want to PAUSE all of the traffic on a link.

- To enable symmetric flow control on an interface:

```
[edit interfaces interface-name ether-options]
user@switch# set flow-control
```

- To disable symmetric flow control on an interface:

```
[edit interfaces interface-name ether-options]
user@switch# set no-flow-control
```

RELATED DOCUMENTATION

Configuring CoS Asymmetric Ethernet PAUSE Flow Control

Ethernet PAUSE flow control is a congestion relief feature that works by providing link-level flow control for all traffic on a full-duplex Ethernet link, including Ethernet links that belong to link aggregated (LAG) interfaces. Ethernet PAUSE works in both directions on the link. In one direction, an interface generates and sends PAUSE messages to stop the connected peer from sending more traffic. In the other direction, the interface responds to PAUSE messages it receives from the connected peer to stop sending traffic.

Asymmetric flow control allows you to configure the PAUSE functionality in each direction independently on an interface. The configuration for generating PAUSE messages and for responding to PAUSE messages does not have to be the same. It can be enabled in both directions, disabled in both directions, or enabled in one direction and disabled in the other direction.

Symmetric flow control means that the interface has the same configuration in both directions. The PAUSE generation and PAUSE response functions are both configured as enabled or they are both disabled. If you do not want to PAUSE all of the traffic on a link, you can use priority-based flow control (PFC) to selectively pause traffic based on its IEEE 802.1p code point.

Asymmetric flow control provides the ability to configure the receive buffer and transmit buffer Ethernet PAUSE actions independently on an interface. The buffers perform the following actions:

- The receive buffers generate and send PAUSE messages to the connected peer to ask the peer to stop sending traffic for a time period specified in the PAUSE frame. The peer interface's buffers may store outgoing frames until the PAUSE period elapses and the interface can resume sending traffic.
- The transmit buffers respond to PAUSE messages received from the connected peer to stop sending traffic to the peer. The transmit buffer may store outgoing frames until the PAUSE period elapses and the interface can resume sending traffic.

Asymmetric flow control enables you to specify independently whether or not the interface receive buffer generates and sends PAUSE messages to stop the connected peer from transmitting traffic, and whether or not the interface transmit buffer responds to PAUSE messages it receives from the connected peer and stops transmitting traffic. The receive buffer configuration determines if the interface transmits PAUSE messages, and the transmit buffer configuration determines if the interface receives and responds to PAUSE messages:

- Receive buffers on—Enable PAUSE transmission (generate and send PAUSE frames)
- Transmit buffers on—Enable PAUSE reception (respond to received PAUSE frames)

You must explicitly set both the receive buffer and the transmit buffer to configure asymmetric flow control.

- To configure asymmetric flow control on an interface:

  ```
  [edit interfaces interface-name ether-options]
  user@switch# set configured-flow-control rx-buffers (on | off) tx-buffers (on | off)
  ```

  For example, to configure interface **xe-0/0/24** to generate and send PAUSE messages but not to respond to received PAUSE messages:

  set interfaces xe-0/0/24 ether-options configured-flow-control rx-buffers on tx-buffers off

  For example, to configure interface **xe-0/0/30** to respond to received PAUSE messages but not to generate and send PAUSE messages:

  set interfaces xe-0/0/30 ether-options configured-flow-control rx-buffers off tx-buffers on

  > **NOTE:** If you configure both buffers to be **on**, that is equivalent to symmetric flow control. If you configure both buffers to be **off**, there is no flow control (flow control is disabled).

RELATED DOCUMENTATION

Enabling and Disabling CoS Symmetric Ethernet PAUSE Flow Control | 211
Configuring CoS PFC (Congestion Notification Profiles) | 194
Understanding CoS Flow Control (Ethernet PAUSE and PFC) | 197

Understanding PFC Functionality Across Layer 3 Interfaces

Priority-based flow control (PFC) allows you to select traffic flows within a link and pause them, so that the output queues associated with the flows do not overflow and drop packets. (PFC is more granular than Ethernet PAUSE, which pauses all traffic on a physical link.) PFC helps you configure lossless transport for traffic flows across a data center bridging network.

However, you might want to create a traffic flow that losslessly traverses the Layer 2 data center bridging network *and* also losslessly traverses a Layer 3 network that connects Ethernet hosts in different Layer 2 networks. On a QFX5210, QFX5200, QFX5110, QFX5100, EX4600, or QFX10000 switch running the Enhanced Layer 2 Software (ELS) CLI, in addition to configuring PFC on Layer 2 (bridging) interfaces, you can configure PFC on VLAN-tagged traffic that traverses Layer 3 interfaces. This enables you to preserve the lossless characteristics that PFC provides on VLAN-tagged traffic, even when the traffic crosses Layer 3 interfaces that connect two Layer 2 networks.

NOTE: This topic is applicable for VLAN-tagged traffic only. Starting in Junos OS Release 17.4R1, QFX5110, QFX5200, and QFX5210 switches also support DSCP-based PFC for *untagged* traffic on Layer 3 interfaces and Layer 2 access interfaces. DSCP-based PFC uses a DSCP classifier to classify the traffic based on a 6-bit DSCP value that is mapped to a 3-bit PFC priority value. For details on using DSCP-based PFC on supporting switches, see "Understanding PFC Using DSCP at Layer 3 for Untagged Traffic" on page 242.

Video: Preserving Lossless Behavior on an SDN or Overlay Network

PFC works the same way across Layer 3 interfaces as it works across Layer 2 interfaces. When an output queue buffer reaches a certain fill level threshold, the switch sends a PFC pause message to the connected peer to pause transmission of the traffic on which PFC is enabled. Pausing the incoming traffic prevents the queue buffer from overflowing and dropping packets, just as on Layer 2 interfaces. When the queue buffer fill level decreases below a certain threshold, the interface sends a message to the connected peer to restart traffic transmission.

Although PFC is a data center bridging technology, PFC also works on Layer 3 interfaces because PFC operates at the queue level. When you use an IEEE 802.1p classifier to classify incoming traffic (map incoming traffic to a forwarding class and a loss priority based on the IEEE 802.1p code point in the Ethernet frame header) and you enable PFC on the appropriate priority (IEEE 802.1p code point), PFC works on Layer 2 and Layer 3 interfaces.

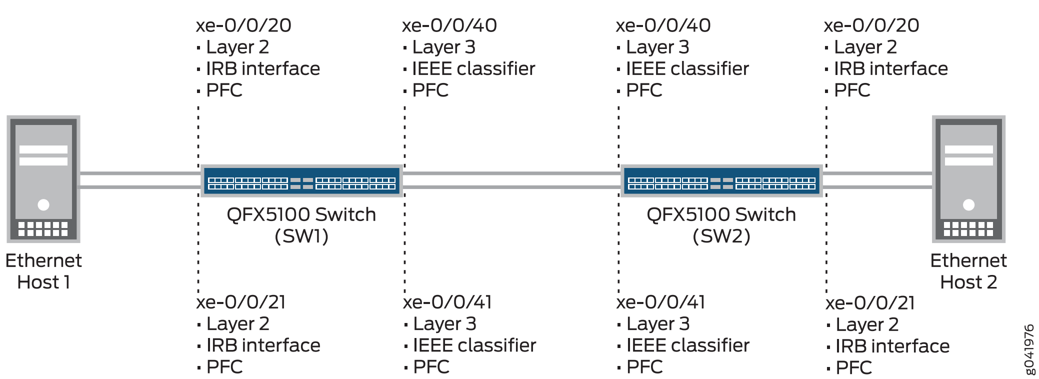
NOTE: Lossless VLAN-tagged traffic on Layer 3 interfaces *must* use an IEEE 802.1p classifier to classify incoming traffic, because PFC does not use DSCP or DSCP IPv6 code points to identify VLAN-tagged traffic for flow control. PFC cannot pause traffic flows unless the incoming traffic is classified by an IEEE 802.1p classifier. Do not apply a DSCP (or a DSCP IPv6) classifier to Layer 3 VLAN-tagged traffic on which you want to enable PFC.

Because PFC functionality relies on the mapping (classifying) of incoming traffic to IEEE 802.1p code points and on enabling PFC on the correct code point(s) at each interface, you must ensure that incoming traffic has the correct 3-bit IEEE 802.1p code point (priority) in the priority code point (PCP) field of the Ethernet frame header (sometimes known as the CoS bits).

NOTE: Layer 3 interfaces do not support FCoE traffic. FCoE traffic must use Layer 2 interfaces and cannot use Layer 3 interfaces. Therefore, you cannot enable PFC on FCoE traffic across Layer 3 interfaces.

Figure 6 on page 215 shows a topology in which two Ethernet hosts in Layer 2 networks communicate across a Layer 3 network, with PFC enabled on all of the Layer 2 and Layer 3 switch interfaces.

Figure 6: Enabling PFC Across Layer 3 Interface Hops



The Ethernet host-facing interfaces (xe-0/0/20 and xe-0/0/21 on both switches) and the Layer 3 network-facing interfaces (interfaces xe-0/0/40 and xe-0/0/41 on both switches) require different interface configurations to enable PFC on the Layer 3 interfaces. In addition, the class of service (CoS) for each interface must be configured correctly, including enabling PFC on the traffic that you want to treat as lossless traffic:

Ethernet-host facing interfaces (xe-0/0/20 and xe-0/0/21) require the following configuration:

- Set interfaces as family ethernet-switching
- Set the interface mode as trunk mode
- Create VLANs to carry the traffic
- Create IRB interfaces to place the Layer 2 VLAN traffic on Layer 3 for transport between IP networks
- Create an IEEE 802.1p classifier to classify incoming traffic into the correct forwarding class, based on the IEEE 802.1p code point
- Create a congestion notification profile (CNP) to configure PFC on the IEEE 802.1p code point of the traffic that you want treat as lossless traffic
- Apply the classifier and the CNP to the Layer 2 interfaces
- Configure CoS: lossless forwarding classes, hierarchical port scheduling (also known as enhanced transmission selection), or direct port scheduling, depending on your switch, and apply it to the Layer 2 interfaces

Layer 3 IP network-facing interfaces (xe-0/0/40 and xe-0/0/41) require the following configuration:

- Set interfaces as family inet
- Set VLAN tagging on the interfaces
- Create VLANs to carry the traffic
- Create an IEEE 802.1p classifier to classify incoming traffic into the correct forwarding class, based on the IEEE 802.1p code point (do not use a DSCP or DSCP IPv6 classifier)
- Create a congestion notification profile (CNP) to configure PFC on the IEEE 802.1p code point of the traffic that you want treat as lossless traffic on the Layer 3 interfaces
- Apply the IEEE 802.1p classifier and the CNP to the Layer 3 interfaces
- Configure CoS: lossless forwarding classes, hierarchical port scheduling (enhanced transmission selection), or direct port scheduling, depending on your switch, and apply it to the Layer 3 interfaces

> **NOTE:** Configuring or changing PFC on an interface blocks the entire port until the PFC change is completed. After a PFC change is completed, the port is unblocked and traffic resumes. Blocking the port stops ingress and egress traffic, and causes packet loss on all queues on the port until the port is unblocked.

When you configure the Layer 2 and Layer 3 interfaces correctly, the switch enables PFC on the traffic between Ethernet Host 1 and Ethernet Host 2 across the entire path between the two hosts. If any output queue in the path on which PFC is enabled experiences congestion, PFC pauses the traffic and prevents packet loss for the flow.

RELATED DOCUMENTATION

Understanding Integrated Routing and Bridging (Topic also applies to IRB interfaces.)

Example: Configuring PFC Across Layer 3 Interfaces

IN THIS SECTION

Priority-based flow control (PFC) helps ensure lossless transport across data center bridging interfaces by pausing incoming traffic when output queue buffers fill to a certain threshold. On a QFX5210, QFX5200, QFX5110, QFX5100, EX4600, or QFX10000 switch running the Enhanced Layer 2 Software (ELS) CLI, in addition to configuring PFC on Layer 2 (bridging) interfaces, you can configure PFC on VLAN-tagged traffic that traverses Layer 3 interfaces. This enables you to preserve the lossless characteristics that PFC provides on VLAN-tagged traffic, even when the traffic crosses Layer 3 interfaces that connect two Layer 2 networks.

NOTE: This topic is applicable for VLAN-tagged traffic only. Starting in Junos OS Release 17.4R1, QFX5110, QFX5200, and QFX5210 switches also support DSCP-based PFC for *untagged* traffic on Layer 3 interfaces and Layer 2 access interfaces. DSCP-based PFC uses a DSCP classifier to classify the traffic based on a 6-bit DSCP value that is mapped to a 3-bit PFC priority value. For details on configuring DSCP-based PFC on supporting switches, see "Configuring DSCP-based PFC for Layer 3 Untagged Traffic" on page 245.

Requirements

This example uses the following hardware and software components:

- Two switches

- Junos OS Release 13.2 or later for the QFX Series
- Two Ethernet hosts

Overview

On a network that uses two QFX5210, QFX5200, QFX5110, QFX5100, EX4600, or QFX10000 switches to connect hosts on two different Ethernet networks across a Layer 3 network, to configure PFC across the Layer 2 and Layer 3 interfaces, you must:

- Configure the Layer 2 and Layer 3 interfaces on the switches
- Configure VLANs to carry the traffic across the Layer 2 and Layer 3 networks
- Configure integrated routing and bridging (IRB) interfaces on the Layer 2 interfaces to move the Layer 2 VLAN traffic to Layer 3
- Configure and apply the appropriate classifiers to the interfaces
- Configure and apply congestion notification profiles (CNPs) on the interfaces to enable PFC on the traffic that you want to be lossless

  NOTE: Configuring or changing PFC on an interface blocks the entire port until the PFC change is completed. After a PFC change is completed, the port is unblocked and traffic resumes. Blocking the port stops ingress and egress traffic, and causes packet loss on all queues on the port until the port is unblocked.

- Configure lossless forwarding classes and either hierarchical port scheduling (also known as enhanced transmission selection) or direct port scheduling, depending on your switch, on the interfaces

  NOTE: PFC operates at the queue level, based on the IEEE 802.1p code point in the priority code point (PCP) field of the Ethernet frame header (sometimes known as the CoS bits). For this reason, VLAN-tagged traffic on Layer 3 interfaces on which you want to enable PFC must use an IEEE 802.1p classifier to map incoming traffic to forwarding classes (which are in turn mapped to output queues) and loss priorities. You cannot use a DSCP or DSCP IPv6 classifier to classify Layer 3 traffic if you want to enable PFC on VLAN-tagged traffic flows.

Topology

Figure 7 on page 219 shows the topology for this example.

Figure 7: Enabling PFC Across Layer 3 Interface Hops

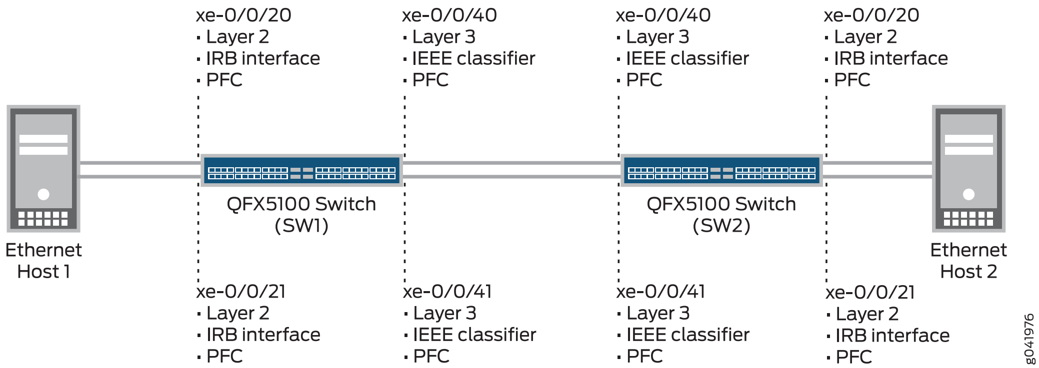


Table 61 on page 219 shows the configuration components for this example. On the two switches, the Ethernet host-facing interfaces use the same interface names and configuration, and the Layer 3 network-facing interfaces use the same interface names and configuration.

Table 61: Components of the PFC Across Layer 3 Interfaces Topology

Component	Settings
Hardware	Two switches, Switch SW1 and Switch SW2. Two Ethernet hosts
Layer 3 interfaces (xe-0/0/40 and xe-0/0/41) and VLANs	Interface xe-0/0/40: • Interface family—inet • Interface IP address—100.103.1.2/24 • VLAN tagging—enabled • Interface VLAN ID—103 Interface xe-0/0/41: • Interface family—inet • Interface IP address—100.104.1.2/24 • VLAN tagging—enabled • Interface VLAN ID—104
Layer 2 interfaces (xe-0/0/20 and xe-0/0/21) and VLAN membership	Family: Ethernet switching Interface mode—trunk Interface xe-0/0/20 VLAN membership—vlan105 Interface xe-0/0/21 VLAN membership—vlan106
VLANs for the IRB interfaces	VLAN unit 105—family inet, IP address 100.105.1.1/24 VLAN unit 106—family inet, IP address 100.106.1.1/24

Table 61: Components of the PFC Across Layer 3 Interfaces Topology *(continued)*

Component	Settings
Layer 2 IRB interfaces	Interface xe-0/0/20: • IRB interface unit—105 • IRB interface family—inet • IRB interface IP address—100.105.1.1/24 • IRB interface VLAN ID—105 • Layer 3 interface name—irb.105 Interface xe-0/0/21: • IRB interface unit—106 • IRB interface family—inet • IRB interface IP address—100.106.1.1/24 • IRB interface VLAN ID—106 • Layer 3 interface name—irb.106
Forwarding classes (both switches)	Name—lossless-3 Queue mapping—queue 3 Packet drop attribute—no-loss Name—lossless-4 Queue mapping—queue 4 Packet drop attribute—no-loss **NOTE:** Matching the forwarding class names (lossless-*3* and lossless-*4*) to the queue number and to the classified IEEE 802.1p code point (priority) creates a configuration that is logical and easy to map because the forwarding class, queue, and priority all use the same number. Name—all-others Queue mapping—queue 0 Packet drop attribute—none **NOTE:** The forwarding class *all-others* is for best-effort traffic that traverses the interfaces.
Layer 2 interface behavior aggregate (BA) classifier	Name—lossless-3-4-ieee Forwarding class lossless-3—mapped to code point 011 (IEEE 802.1p priority 3) and a packet loss priority of low Forwarding class lossless-4—mapped to code point 100 (IEEE 802.1p priority 4) and a packet loss priority of low Apply the Layer 2 IEEE 802.1p classifier to both the Layer 2 and the Layer 3 interfaces (xe-0/0/20, xe-0/0/21, xe-0/040, and xe-0/0/41).

Table 61: Components of the PFC Across Layer 3 Interfaces Topology *(continued)*

Component	Settings
Congestion notification profile (PFC, both switches)	Name—lossless-cnp PFC enabled on IEEE 802.1p code points—011 (lossless-3 forwarding class and priority), 100 (lossless-4 forwarding class and priority) Apply the CNP to both the Layer 2 and the Layer 3 interfaces (xe-0/0/20, xe-0/0/21, xe-0/040, and xe-0/0/41) to enable PFC on IEEE 802.1p code points 011 and 100.
Enhanced transmission selection (ETS) hierarchical port scheduling (only if using ETS)	Hierarchical port scheduling (ETS) includes configuring: • Schedulers to assign bandwidth to traffic • Scheduler mapping to forwarding classes • Grouping of the forwarding classes (priorities) in forwarding class sets (priority groups) • A traffic control profile to assign bandwidth to the forwarding class set and to associate the forwarding class set with the scheduler mapping Hierarchical port scheduling also includes applying the hierarchical scheduler (defined in the traffic control profile) to the interfaces. This example focuses on configuring PFC across the Layer 2 and Layer 3 interfaces. To maintain this focus, this example includes the CLI statements needed to configure hierarchical port scheduling, but does not include descriptive explanations of the configuration. The *Related Documentation* section provides links to example documents that show how to configure hierarchical port scheduling. Apply the scheduling configuration to both the Layer 2 and the Layer 3 interfaces (xe-0/0/20, xe-0/0/21, xe-0/040, and xe-0/0/41).
Direct port scheduling (only if using port scheduling instead of ETS)	Direct port scheduling includes configuring: • Schedulers to assign bandwidth to traffic • Scheduler mapping to forwarding classes Port scheduling also includes applying the scheduler map to the interfaces. This example focuses on configuring PFC across the Layer 2 and Layer 3 interfaces. To maintain this focus, this example includes the CLI statements needed to configure direct port scheduling, but does not include descriptive explanations of the configuration. The *Related Documentation* section provides links to example documents that show how to configure port scheduling. Apply the scheduling configuration to both the Layer 2 and the Layer 3 interfaces (xe-0/0/20, xe-0/0/21, xe-0/040, and xe-0/0/41).

Configuration

IN THIS SECTION

CLI Quick Configuration

To configure PFC across Layer 3 interfaces, copy the following commands, paste them in a text file, remove the 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. The same configuration applies to both Switch SW1 and Switch SW2. The configuration is separated into the configuration common to ETS and direct port scheduling, and the portions of the configuration that apply only to ETS and only to port scheduling.

Common Configuration (Applies to ETS Hierarchical Scheduling and to Port Scheduling)

```
set interfaces xe-0/0/40 vlan-tagging
set interfaces xe-0/0/40 unit 0 vlan-id 103
set interfaces xe-0/0/40 unit 0 family inet address 100.103.1.2/24
set interfaces xe-0/0/41 vlan-tagging
set interfaces xe-0/0/41 unit 0 vlan-id 104
set interfaces xe-0/0/41 unit 0 family inet address 100.104.1.2/24
set interfaces xe-0/0/20 unit 0 family ethernet-switching interface-mode trunk
set interfaces xe-0/0/20 unit 0 family ethernet-switching vlan members vlan105
set interfaces xe-0/0/21 unit 0 family ethernet-switching interface-mode trunk
set interfaces xe-0/0/21 unit 0 family ethernet-switching vlan members vlan106
set interfaces irb unit 105 family inet address 100.105.1.1/24
set interfaces irb unit 106 family inet address 100.106.1.1/24
set vlans vlan105 vlan-id 105
set vlans vlan106 vlan-id 106
set vlans vlan105 l3-interface irb.105
set vlans vlan106 l3-interface irb.106
```

set class-of-service forwarding-classes class lossless-3 queue-num 3 no-loss

set class-of-service forwarding-classes class lossless-4 queue-num 4 no-loss

set class-of-service forwarding-classes class all-others queue-num 0

set class-of-service classifiers ieee-802.1 lossless-3-4-ieee forwarding-class lossless-3 loss-priority low code-points 011

set class-of-service classifiers ieee-802.1 lossless-3-4-ieee forwarding-class lossless-4 loss-priority low code-points 100

set class-of-service congestion-notification-profile lossless-cnp input ieee-802.1 code-point 011 pfc

set class-of-service congestion-notification-profile lossless-cnp input ieee-802.1 code-point 100 pfc

set class-of-service schedulers lossless_sch transmit-rate 6g

set class-of-service schedulers lossless_sch shaping-rate percent 100

set class-of-service schedulers all-others_sch transmit-rate 4g

set class-of-service scheduler-maps lossless_map forwarding-class lossless-3 scheduler lossless_sch

set class-of-service scheduler-maps lossless_map forwarding-class lossless-4 scheduler lossless_sch

set class-of-service scheduler-maps all-others_map forwarding-class all-others scheduler all-others_sch

set class-of-service interfaces xe-0/0/20 congestion-notification-profile lossless-cnp

set class-of-service interfaces xe-0/0/20 unit 0 classifiers ieee-802.1 lossless-3-4-ieee

set class-of-service interfaces xe-0/0/21 congestion-notification-profile lossless-cnp

set class-of-service interfaces xe-0/0/21 unit 0 classifiers ieee-802.1 lossless-3-4-ieee

set class-of-service interfaces xe-0/0/40 congestion-notification-profile lossless-cnp

set class-of-service interfaces xe-0/0/40 classifiers ieee-802.1 lossless-3-4-ieee

set class-of-service interfaces xe-0/0/41 congestion-notification-profile lossless-cnp

set class-of-service interfaces xe-0/0/41 classifiers ieee-802.1 lossless-3-4-ieee

Configuration for ETS Hierarchical Scheduling

The ETS-specific portion of this example configures forwarding class set (priority group) membership and priority group CoS settings (traffic control profile), and assigns the priority group and its CoS configuration to the interfaces.

set class-of-service forwarding-class-sets lossless_fc_set class lossless-3

set class-of-service forwarding-class-sets lossless_fc_set class lossless-4

set class-of-service forwarding-class-sets all-others_fc_set class all-others

set class-of-service traffic-control-profiles lossless_tcp scheduler-map lossless_map

set class-of-service traffic-control-profiles lossless_tcp guaranteed-rate percent 60

set class-of-service traffic-control-profiles lossless_tcp shaping-rate percent 100
set class-of-service traffic-control-profiles all-others_tcp scheduler-map all-others_map
set class-of-service traffic-control-profiles all-others_tcp guaranteed-rate percent 40
set class-of-service interfaces xe-0/0/20 forwarding-class-set lossless_fc_set output-traffic-control-profile lossless_tcp
set class-of-service interfaces xe-0/0/20 forwarding-class-set all-others_fc_set output-traffic-control-profile all-others_tcp
set class-of-service interfaces xe-0/0/21 forwarding-class-set lossless_fc_set output-traffic-control-profile lossless_tcp
set class-of-service interfaces xe-0/0/21 forwarding-class-set all-others_fc_set output-traffic-control-profile all-others_tcp
set class-of-service interfaces xe-0/0/40 forwarding-class-set lossless_fc_set output-traffic-control-profile lossless_tcp
set class-of-service interfaces xe-0/0/40 forwarding-class-set all-others_fc_set output-traffic-control-profile all-others_tcp
set class-of-service interfaces xe-0/0/41 forwarding-class-set lossless_fc_set output-traffic-control-profile lossless_tcp
set class-of-service interfaces xe-0/0/41 forwarding-class-set all-others_fc_set output-traffic-control-profile all-others_tcp

Configuration for Port Scheduling

The port-scheduling-specific portion of this example assigns the scheduler maps (which set the CoS treatment of the forwarding classes in the scheduler map) to the interfaces.

```
[edit class-of-service]
```

set interfaces xe-0/0/20 scheduler-map lossless_map
set interfaces xe-0/0/20 scheduler-map all-others_map
set interfaces xe-0/0/21 scheduler-map lossless_map
set interfaces xe-0/0/21 scheduler-map all-others_map
set interfaces xe-0/0/40 scheduler-map lossless_map
set interfaces xe-0/0/40 scheduler-map all-others_map
set interfaces xe-0/0/41 scheduler-map lossless_map
set interfaces xe-0/0/41 scheduler-map all-others_map

Common Configuration (Applies to ETS Hierarchical Scheduling and to Port Scheduling)

Step-by-Step Procedure

The following step-by-step procedure shows you how to configure the VLANs, IRB interfaces, lossless forwarding classes, classifiers, PFC settings to enable PFC across Layer 3 interfaces, and the queue scheduling configuration common to ETS and direct port scheduling. For completeness, the ETS hierarchical port scheduling and direct port scheduling configurations are included separately, in the following procedures, but without explanatory text. See the *Related Documentation* links for detailed examples of the scheduling elements of the configuration.

1. Configure the Layer 3 interface VLANs and IP addresses:

   ```
   [edit interfaces]
   user@switch# set xe-0/0/40 vlan-tagging
   user@switch# set xe-0/0/40 unit 0 vlan-id 103
   user@switch# set xe-0/0/40 unit 0 family inet address 100.103.1.2/24
   user@switch# set xe-0/0/41 vlan-tagging
   user@switch# set xe-0/0/41 unit 0 vlan-id 104
   user@switch# set xe-0/0/41 unit 0 family inet address 100.104.1.2/24
   ```

2. Configure the Layer 2 interface VLAN membership and interface mode:

   ```
   [edit interfaces]
   user@switch# set xe-0/0/20 unit 0 family ethernet-switching interface-mode trunk
   user@switch# set xe-0/0/20 unit 0 family ethernet-switching vlan members vlan105
   user@switch# set xe-0/0/21 unit 0 family ethernet-switching interface-mode trunk
   user@switch# set xe-0/0/21 unit 0 family ethernet-switching vlan members vlan106
   ```

3. Configure the IRB interfaces and VLANs to transport incoming Layer 2 traffic assigned to VLANs vlan105 (of which interface xe-0/0/20 is a member) and vlan106 (of which interface xe-0/0/21 is a member) across Layer 3:

   ```
   [edit]
   user@switch# set interfaces irb unit 105 family inet address 100.105.1.1/24
   user@switch# set interfaces irb unit 106 family inet address 100.106.1.1/24
   user@switch# set vlans vlan105 vlan-id 105
   user@switch# set vlans vlan106 vlan-id 106
   user@switch# set vlans vlan105 l3-interface irb.105
   user@switch# set vlans vlan106 l3-interface irb.106
   ```

4. Configure the lossless forwarding classes and a best-effort forwarding class for any other traffic that might use the interfaces:

   [edit class-of-service]
   user@switch# **set forwarding-classes class lossless-3 queue-num 3 no-loss**
   user@switch# **set forwarding-classes class lossless-4 queue-num 4 no-loss**
   user@switch# **set forwarding-classes class all-others queue-num 0**

5. Configure the IEEE classifier for the Layer 2 and Layer 3 interfaces to classify incoming traffic into the lossless forwarding classes based on the IEEE 802.1p code point of the traffic:

   [edit class-of-service classifiers]
   user@switch# **set ieee-802.1 lossless-3-4-ieee forwarding-class lossless-3 loss-priority low code-points 011**
   user@switch# **set ieee-802.1 lossless-3-4-ieee forwarding-class lossless-4 loss-priority low code-points 100**

6. Configure the CNP to enable PFC on the lossless priorities (the lossless forwarding classes mapped to IEEE 802.1p code points 3 and 4):

   [edit class-of-service congestion-notification-profile]
   user@switch# **set lossless-cnp input ieee-802.1 code-point 011 pfc**
   user@switch# **set lossless-cnp input ieee-802.1 code-point 100 pfc**

7. Apply the Layer 2 IEEE 802.1p classifier and the CNP to the Layer 3 interfaces:

   [edit class-of-service interfaces]
   user@switch# **set xe-0/0/40 classifiers ieee-802.1 lossless-3-4-ieee**
   user@switch# **set xe-0/0/40 congestion-notification-profile lossless-cnp**
   user@switch# **set xe-0/0/41 classifiers ieee-802.1 lossless-3-4-ieee**
   user@switch# **set xe-0/0/41 congestion-notification-profile lossless-cnp**

8. Apply the Layer 2 IEEE 802.1p classifier and the CNP to the Layer 2 interfaces:

   [edit class-of-service interfaces]
   user@switch# **xe-0/0/20 unit 0 classifiers ieee-802.1 lossless-3-4-ieee**
   user@switch# **xe-0/0/20 congestion-notification-profile lossless-cnp**
   user@switch# **xe-0/0/21 unit 0 classifiers ieee-802.1 lossless-3-4-ieee**
   user@switch# **xe-0/0/21 congestion-notification-profile lossless-cnp**

9. Configure queue scheduling to support the lossless configuration and map the schedulers to the forwarding classes (statements included here for completeness; see the *Related Documentation* links for detailed examples of scheduling configuration):

   `[edit class-of-service]`
   `user@switch#` **set schedulers lossless_sch transmit-rate 6g**
   `user@switch#` **set schedulers lossless_sch shaping-rate percent 100**
   `user@switch#` **set schedulers all-others_sch transmit-rate 4g**
   `user@switch#` **set scheduler-maps lossless_map forwarding-class lossless-3 scheduler lossless_sch**
   `user@switch#` **set scheduler-maps lossless_map forwarding-class lossless-4 scheduler lossless_sch**
   `user@switch#` **set scheduler-maps all-others_map forwarding-class all-others scheduler all-others_sch**

ETS Hierarchical Scheduling Configuration

Step-by-Step Procedure

1. Configure hierarchical scheduling to support the lossless configuration (included here for completeness; see the *Related Documentation* links for detailed examples of scheduling configuration) and apply it to the Layer 2 and Layer 3 interfaces:

   ```
   [edit class-of-service interfaces]
   user@switch# set forwarding-class-sets lossless_fc_set class lossless-3
   user@switch# set forwarding-class-sets lossless_fc_set class lossless-4
   user@switch# set forwarding-class-sets all-others_fc_set class all-others
   user@switch# set traffic-control-profiles lossless_tcp scheduler-map lossless_map
   user@switch# set traffic-control-profiles lossless_tcp guaranteed-rate percent 60
   user@switch# set traffic-control-profiles lossless_tcp shaping-rate percent 100
   user@switch# set traffic-control-profiles all-others_tcp scheduler-map all-others_map
   user@switch# set traffic-control-profiles all-others_tcp guaranteed-rate percent 40
   user@switch# set interfaces xe-0/0/20 forwarding-class-set lossless_fc_set
   output-traffic-control-profile lossless_tcp
   user@switch# set interfaces xe-0/0/20 forwarding-class-set all-others_fc_set
   output-traffic-control-profile all-others_tcp
   user@switch# set interfaces xe-0/0/21 forwarding-class-set lossless_fc_set
   output-traffic-control-profile lossless_tcp
   user@switch# set interfaces xe-0/0/21 forwarding-class-set all-others_fc_set
   output-traffic-control-profile all-others_tcp
   user@switch# set interfaces xe-0/0/40 forwarding-class-set lossless_fc_set
   output-traffic-control-profile lossless_tcp
   user@switch# set interfaces xe-0/0/40 forwarding-class-set all-others_fc_set
   output-traffic-control-profile all-others_tcp
   user@switch# set interfaces xe-0/0/41 forwarding-class-set lossless_fc_set
   output-traffic-control-profile lossless_tcp
   user@switch# set interfaces xe-0/0/41 forwarding-class-set all-others_fc_set
   output-traffic-control-profile all-others_tcp
   ```

Port Scheduling Configuration

Step-by-Step Procedure

1. Apply port scheduling to support the lossless configuration on interfaces:

```
[edit class-of-service]
user@switch# set interfaces xe-0/0/20 scheduler-map lossless_map
user@switch# set interfaces xe-0/0/20 scheduler-map all-others_map
user@switch# set interfaces xe-0/0/21 scheduler-map lossless_map
user@switch# set interfaces xe-0/0/21 scheduler-map all-others_map
user@switch# set interfaces xe-0/0/40 scheduler-map lossless_map
user@switch# set interfaces xe-0/0/40 scheduler-map all-others_map
user@switch# set interfaces xe-0/0/41 scheduler-map lossless_map
user@switch# set interfaces xe-0/0/41 scheduler-map all-others_map
```

Results

Display the results of the interface, VLAN, and class-of-service configurations (the system shows only the explicitly configured parameters; it does not show default parameters). The results are valid for both Switch SW1 and Switch SW2 because the same configuration is used on both switches. The results are from the ETS hierarchical scheduling configuration, which show the more complex configuration. Direct port scheduling results would not show the traffic control profile or forwarding class set portions of the configuration, but would display the name of the scheduler map under each interface (instead of the names of the forwarding class set and output traffic control profile). Other than that, the results are the same.

Display the results of the interface configuration:

```
user@switch# show configuration interfaces
```

```
xe-0/0/20 {
    unit 0 {
        family ethernet-switching {
            interface-mode trunk;
            vlan {
                members vlan105;
            }
        }
    }
}
xe-0/0/21 {
    unit 0 {
        family ethernet-switching {
            interface-mode trunk;
            vlan {
                members vlan106;
```

```
            }
        }
    }
}
xe-0/0/40 {
    vlan-tagging;
    unit 0 {
        vlan-id 103;
        family inet {
            address 100.103.1.2/24;
        }
    }
}
xe-0/0/41 {
    vlan-tagging;
    unit 0 {
        vlan-id 104;
        family inet {
            address 100.104.1.2/24;
        }
    }
}
irb {
    unit 105 {
        family inet {
            address 100.105.1.1/24;
        }
    }
    unit 106 {
        family inet {
            address 100.106.1.1/24;
        }
    }
}
vlan {
    unit 105 {
        family inet {
            address 100.105.1.1/24;
        }
    }
    unit 106 {
        family inet {
            address 100.106.1.1/24;
        }
```

```
    }
}
```

Display the results of the vlan configuration:

`user@switch#` **show configuration vlans**

```
vlan105 {
    vlan-id 105;
    l3-interface irb.105;
}
vlan106 {
    vlan-id 106;
    l3-interface irb.106;
}
```

Display the results of the class-of-service configuration:

`user@switch#` **show configuration class-of-service**

```
classifiers {
    ieee-802.1 lossless-3-4-ieee {
        forwarding-class lossless-3 {
            loss-priority low code-points 011;
        }
        forwarding-class lossless-4 {
            loss-priority low code-points 100;
        }
    }
}
forwarding-classes {
    class lossless-3 queue-num 3 no-loss;
    class lossless-4 queue-num 4 no-loss;
    class all-others queue-num 0;
}
traffic-control-profiles {
    lossless_tcp {
        scheduler-map lossless_map;
        shaping-rate percent 100;
        guaranteed-rate percent 60;
    }
    all-others_tcp {
```

```
        scheduler-map all-others_map;
        guaranteed-rate percent 40;
    }
}
forwarding-class-sets {
    lossless_fc_set {
        class lossless-3;
        class lossless-4;
    }
    all-others_fc_set {
        class all-others;
    }
}
congestion-notification-profile {
    lossless-cnp {
        input {
            ieee-802.1 {
                code-point 011 {
                    pfc;
                }
                code-point 100 {
                    pfc;
                }
            }
        }
    }
}
interfaces {
    xe-0/0/20 {
        forwarding-class-set {
            lossless_fc_set {
                output-traffic-control-profile lossless_tcp;
            }
            all-others_fc_set {
                output-traffic-control-profile all-others_tcp;
            }
        }
        congestion-notification-profile lossless-cnp;
        unit 0 {
            classifiers {
                ieee-802.1 lossless-3-4-ieee;
            }
        }
    }
```

```
    xe-0/0/21 {
        forwarding-class-set {
            all-others_fc_set {
                output-traffic-control-profile all-others_tcp;
            }
            lossless_fc_set {
                output-traffic-control-profile lossless_tcp;
            }
        }
        congestion-notification-profile lossless-cnp;
        unit 0 {
            classifiers {
                ieee-802.1 lossless-3-4-ieee;
            }
        }
    }
    xe-0/0/40 {
        forwarding-class-set {
            lossless_fc_set {
                output-traffic-control-profile lossless_tcp;
            }
            all-others_fc_set {
                output-traffic-control-profile all-others_tcp;
            }
        }
        congestion-notification-profile lossless-cnp;
        classifiers {
            ieee-802.1 lossless-3-4-ieee;
        }
    }
    xe-0/0/41 {
        forwarding-class-set {
            lossless_fc_set {
                output-traffic-control-profile lossless_tcp;
            }
            all-others_fc_set {
                output-traffic-control-profile all-others_tcp;
            }
        }
        congestion-notification-profile lossless-cnp;
        classifiers {
            ieee-802.1 lossless-3-4-ieee;
        }
    }
```

```
}
scheduler-maps {
    lossless_map {
        forwarding-class lossless-3 scheduler lossless_sch;
        forwarding-class lossless-4 scheduler lossless_sch;
    }
    all-others_map {
        forwarding-class all-others scheduler all-others_sch;
    }
}
schedulers {
    lossless_sch {
        transmit-rate 6g;
        shaping-rate percent 100;
    }
    all-others_sch {
        transmit-rate 4g;
    }
}
```

TIP: To quickly configure the switch, issue the **load merge terminal** command, and then copy the hierarchies and paste them into the switch terminal window.

Verification

IN THIS SECTION

To verify that the PFC across Layer 3 interfaces configuration has been created and is operating properly, perform these tasks:

Verifying the Interface Configuration

Purpose

Verify that the Layer 2 Ethernet interfaces, Layer 3 IP interfaces, IRB interfaces, and VLAN interfaces have been created on the switch and are correctly configured.

Action

Display the switch interface configuration using the **show configuration interfaces** command:

`user@switch>` **show configuration interfaces**

```
xe-0/0/20 {
    unit 0 {
        family ethernet-switching {
            interface-mode trunk;
            vlan {
                members vlan105;
            }
        }
    }
}
xe-0/0/21 {
    unit 0 {
        family ethernet-switching {
            interface-mode trunk;
            vlan {
                members vlan106;
            }
        }
    }
}
xe-0/0/40 {
    vlan-tagging;
    unit 0 {
        vlan-id 103;
        family inet {
            address 100.103.1.2/24;
        }
    }
}
xe-0/0/41 {
    vlan-tagging;
```

```
    unit 0 {
        vlan-id 104;
        family inet {
            address 100.104.1.2/24;
        }
    }
}
irb {
    unit 105 {
        family inet {
            address 100.105.1.1/24;
        }
    }
    unit 106 {
        family inet {
            address 100.106.1.1/24;
        }
    }
}
vlan {
    unit 105 {
        family inet {
            address 100.105.1.1/24;
        }
    }
    unit 106 {
        family inet {
            address 100.106.1.1/24;
        }
    }
}
```

Meaning

The **show configuration interfaces** command displays all of the interfaces configured on the switch. The command output shows that:

- Interfaces xe-0/0/20 and xe-0/0/21 are Ethernet interfaces (family ethernet-switching) in trunk interface mode. Interface xe-0/0/20 is a member of VLAN vlan105, and interface xe-0/0/21 is a member of VLAN vlan106.
- Interfaces xe-0/0/40 and xe-0/0/41 are IP interfaces (family inet) with VLAN tagging enabled. Interface xe-0/0/40 has an IP address of 100.103.1.2/24 and a VLAN ID of 103. Interface xe-0/0/41 has an IP address of 100.104.1.2/24 and a VLAN ID of 104.

- Two IRB interfaces are configured, IRB unit 105 with an IP address of 100.105.1.1/24 and IRB unit 106 with an IP address of 100.106.1.1/24.
- Two VLAN interfaces are configured, VLAN unit 105 with an IP address of 100.105.1.1/24 (for IRB interface unit 105) and VLAN unit 106 with an IP address of 100.106.1.1/24 (for IRB interface unit 106).

Verifying the VLAN Configuration

Purpose

Verify that VLANs have been created on the switch and are correctly configured.

Action

Display the VLAN configuration using the **show configuration vlans** command:

`user@switch>` **show configuration vlans**

```
vlan105 {
    vlan-id 105;
    l3-interface irb.105;
}
vlan106 {
    vlan-id 106;
    l3-interface irb.106;
}
```

Meaning

The **show configuration vlans** command displays all of the VLANs configured on the switch. The command output shows that:

- VLAN vlan105 has been configured with VLAN ID 105 on IRB interface irb.105.
- VLAN vlan106 has been configured with VLAN ID 106 on IRB interface irb.106.

Verifying the PFC Configuration (Congestion Notification Profile)

Purpose

Verify that PFC has been enabled on the correct IEEE 802.1p code points (priorities) in the CNP.

Action

Display the PFC configuration using the **show configuration class-of-service congestion-notification-profile** command:

`user@switch>` **show configuration class-of-service congestion-notification-profile**

```
lossless-cnp {
    input {
        ieee-802.1 {
            code-point 011 {
                pfc;
            }
            code-point 100 {
                pfc;
            }
        }
    }
}
```

Meaning

The **show configuration class-of-service congestion-notification-profile** command displays all of the CNPs configured on the switch. The command output shows that:

- The CNP named **lossless-cnp** is configured on the switch.
- The CNP **lossless-cnp** enables PFC on IEEE 802.1p code points 100 and 100.

Verify the Forwarding Class Configuration

Purpose

Verify that the two lossless forwarding classes and the best-effort forwarding class have been configured on the switch.

Action

Display the forwarding class configuration using the **show configuration class-of-service forwarding-classes** command:

`user@switch>` **show configuration class-of-service forwarding-classes**

```
class lossless-3 queue-num 3 no-loss;
class lossless-4 queue-num 4 no-loss;
class all-others queue-num 0;
```

Meaning

The **show configuration class-of-service forwarding-classes** command displays all of the forwarding classes configured on the switch (default forwarding classes are not displayed). The command output shows that:

- Forwarding class **lossless-3** is mapped to queue 3 and is configured as a lossless forwarding class (the **no-loss** attribute is applied)
- Forwarding class **lossless-4** is mapped to queue 4 and is configured as a lossless forwarding class (the **no-loss** attribute is applied)
- Forwarding class **all-others** is mapped to queue 0. It is not a lossless forwarding class (the **no-loss** attribute is not applied).

Verifying the Classifier Configuration

Purpose

Verify that the IEEE 802.1p classifier has been configured on the switch.

Action

Display the classifier configuration using the **show configuration class-of-service classifiers** command:

`user@switch>` **show configuration class-of-service classifiers**

```
ieee-802.1 lossless-3-4-ieee {
    forwarding-class lossless-3 {
        loss-priority low code-points 011;
    }
    forwarding-class lossless-4 {
        loss-priority low code-points 100;
    }
}
```

Meaning

The **show configuration class-of-service classifiers** command displays all of the classifiers configured on the switch. The command output shows that the Layer 2 IEEE 802.1p classifier **lossless-3-4-ieee** classifiers traffic with the code point 011 into the **lossless-3** forwarding class with a loss priority of **low**, and classifies traffic with the code point 100 into the **lossless-4** forwarding class with a loss priority of **low**.

Verifying the Interface CoS Configuration (Hierarchical Scheduling, PFC, and Classifier Mapping to Interfaces)

Purpose

Verify that the interfaces have the correct hierarchical scheduling, PFC, and classifier configurations.

NOTE: The results are from the ETS hierarchical scheduling configuration, which shows the more complex configuration. Direct port scheduling results would not show the traffic control profile or forwarding class set portions of the interface configuration, but would display the name of the scheduler map under each interface instead of the names of the forwarding class set and output traffic control profile. Other than that, they are the same.

Action

Display the interface CoS configuration using the **show configuration class-of-service interfaces** command:

`user@switch>` **show configuration class-of-service interfaces**

```
xe-0/0/20 {
    forwarding-class-set {
        lossless_fc_set {
            output-traffic-control-profile lossless_tcp;
        }
        all-others_fc_set {
            output-traffic-control-profile all-others_tcp;
        }
    }
    congestion-notification-profile lossless-cnp;
    unit 0 {
        classifiers {
            ieee-802.1 lossless-3-4-ieee;
        }
    }
}
xe-0/0/21 {
    forwarding-class-set {
        all-others_fc_set {
            output-traffic-control-profile all-others_tcp;
        }
        lossless_fc_set {
            output-traffic-control-profile lossless_tcp;
        }
    }
    congestion-notification-profile lossless-cnp;
    unit 0 {
        classifiers {
            ieee-802.1 lossless-3-4-ieee;
        }
```

```
    }
}
xe-0/0/40 {
    forwarding-class-set {
        lossless_fc_set {
            output-traffic-control-profile lossless_tcp;
        }
        all-others_fc_set {
            output-traffic-control-profile all-others_tcp;
        }
    }
    congestion-notification-profile lossless-cnp;
    classifiers {
        ieee-802.1 lossless-3-4-ieee;
    }
}
xe-0/0/41 {
    forwarding-class-set {
        lossless_fc_set {
            output-traffic-control-profile lossless_tcp;
        }
        all-others_fc_set {
            output-traffic-control-profile all-others_tcp;
        }
    }
    congestion-notification-profile lossless-cnp;
    classifiers {
        ieee-802.1 lossless-3-4-ieee;
    }
}
```

Meaning

The **show configuration class-of-service interfaces** command displays all of the CoS components configured on the switch interfaces. The command output shows that:

- The configuration on Layer 2 Ethernet interfaces xe-0/0/20 and xe-0/0/21 includes:
  - Hierarchical scheduling—The forwarding class set **lossless_fc_set** with the traffic control profile **lossless_tcp** for the lossless traffic, and the forwarding class set **all-others_fc_set** with the traffic control profile **all-others_tcp** for the best-effort traffic are applied to both interfaces.
  - PFC—The **lossless-cnp** congestion notification profile is applied to both interfaces.
  - Classifiers—The Layer 2 IEEE 802.1p classifier **lossless-3-4-ieee** is applied to both interfaces.
- The configuration on Layer 3 IP interfaces xe-0/0/40 and xe-0/0/41 includes:

- Hierarchical scheduling—The forwarding class set **lossless_fc_set** with the traffic control profile **lossless_tcp** for the lossless traffic, and the forwarding class set **all-others_fc_set** with the traffic control profile **all-others_tcp** for the best-effort traffic are applied to both interfaces.
- PFC—The **lossless-cnp** congestion notification profile is applied to both interfaces.
- Classifiers—The Layer 2 IEEE 802.1p classifier **lossless-3-4-ieee** is applied to both interfaces. Traffic that would use a DSCP or a DSCP IPv6 classifier if it were configured uses the IEEE 802.1p classifier instead. Using the IEEE 802.1p classifier allows the interface to use PFC to pause traffic during periods of congestion to prevent packet loss.

RELATED DOCUMENTATION

Understanding PFC Functionality Across Layer 3 Interfaces | 214

Understanding PFC Using DSCP at Layer 3 for Untagged Traffic

IN THIS SECTION

Protocols such as Remote Direct Memory Access (RDMA) over converged Ethernet version 2 (RoCEv2) require lossless behavior for traffic across Layer 3 connections to Layer 2 Ethernet subnetworks. Traditionally, priority-based flow control (PFC) can be used to prevent traffic loss when congestion occurs on Layer 2 or Layer 3 interfaces for VLAN-tagged traffic by selectively pausing traffic on any of eight priorities corresponding to IEEE 802.1p code points in the VLAN headers of incoming traffic on an interface. However, *untagged* traffic—traffic without VLAN tagging—cannot be examined for IEEE 802.1p code points on which to pause traffic.

Starting in Junos OS Release 17.4R1, to support lossless traffic flow at Layer 3 for untagged traffic, QFX Series switches support enabling PFC for Layer 3 interfaces and Layer 2 access interfaces using Distributed Services code point (DSCP) values in the Layer 3 IP header of incoming traffic, rather than IEEE 802.1p code point values in a Layer 2 VLAN header.

Overview of DSCP-based PFC

PFC is a data center bridging technology operating at Layer 2, and DSCP information is exchanged in IP headers at Layer 3. However, QFX Series switches can be configured for DSCP-based PFC, which preserves lossless behavior across Layer 3 network connections for untagged traffic.

PFC operates by generating pause frames for traffic identified on configured code points in incoming traffic to notify the peer to pause transmission when the link is congested. With DSCP-based PFC enabled, pause frames are triggered based on a configured 6-bit DSCP value (corresponding to decimal values 0-63) in the Layer 3 IP header of incoming traffic.

However, PFC on the QFX Series can only send pause frames with a 3-bit PFC priority—one of 8 code points corresponding to decimal values 0-7—which, for VLAN-tagged traffic, usually corresponds to the IEEE 802.1p code points in the incoming traffic VLAN headers. Untagged traffic provides no reference for IEEE 802.1p code point values, so to trigger PFC on a DSCP value, the DSCP value must be mapped explicitly in the configuration to a PFC priority to use in the PFC pause frames sent to the peer when congestion occurs for that code point. You can map traffic on a DSCP value to a PFC priority when you define the no-loss forwarding class with which you want to classify DSCP-based PFC traffic. The forwarding class must also be mapped to an output queue with no-loss behavior.

NOTE: You cannot assign the same PFC priority to more than one forwarding class because the mapped PFC priority value is used as the forwarding class ID when DSCP-based PFC is configured.

A DSCP classifier (instead of an IEEE 802.1p classifier) is also required to specify that incoming traffic with the above-configured DSCP value belongs to the no-loss forwarding class. Any DSCP values for which DSCP-based PFC is enabled on a interface must be specified in either the default DSCP classifier or in a user-defined DSCP classifier associated with the interface.

To enable DSCP-based PFC on an interface, define an input congestion notification profile with the same DSCP value (and desired buffering parameters), and associate it with the interface.

The peer device should have a matching PFC configuration for the mapped PFC priority code points.

Limitations of DSCP-based PFC

The following are limitations of DSCP-based PFC:

- You cannot configure both DSCP-based PFC and IEEE 802.1p PFC under the same congestion notification profile, or associate both a DSCP-based congestion notification profile and an IEEE 802.1p congestion notification profile with the same interface.
- DSCP-based PFC is supported on Layer 3 interfaces and Layer 2 access interfaces for untagged traffic only. PFC behavior is unpredictable if VLAN-tagged packets are received on an interface with DSCP-based PFC enabled.
- Each no-loss forwarding class can only be associated with a unique 3-bit PFC priority value from 0 through 7.

Release History Table

Release	Description
17.4R1	Starting in Junos OS Release 17.4R1, to support lossless traffic flow at Layer 3 for untagged traffic, QFX Series switches support enabling PFC for Layer 3 interfaces and Layer 2 access interfaces using Distributed Services code point (DSCP) values in the Layer 3 IP header of incoming traffic, rather than IEEE 802.1p code point values in a Layer 2 VLAN header.

RELATED DOCUMENTATION

Configuring DSCP-based PFC for Layer 3 Untagged Traffic

You can configure DSCP-based PFC to support lossless behavior for untagged traffic across Layer 3 connections to Layer 2 subnetworks for protocols such as Remote Direct Memory Access (RDMA) over converged Ethernet version 2 (RoCEv2).

With DSCP-based PFC, pause frames are generated to notify the peer that the link is congested based on a configured 6-bit Distributed Services code point (DSCP) value in the Layer 3 IP header of incoming traffic, rather than a 3-bit IEEE 802.1p code point in the Layer 2 VLAN header.

Because PFC on the QFX Series can only send pause frames corresponding to PFC priority code points, the 6-bit configured DSCP value must be mapped to a 3-bit PFC priority to use in pause frames when DSCP-based PFC is triggered. Configuring the mapping involves mapping the PFC priority value to a no-loss forwarding class when you map the forwarding class to a queue, defining a congestion notification profile to enable PFC on traffic with the desired DSCP value, and configuring a DSCP classifier to associate the PFC priority-mapped forwarding class (along with the loss priority) with the configured DSCP value on which to trigger PFC pause frames.

The peer device should have output PFC and a corresponding flow control queue configured to match the PFC priority configuration on the QFX Series device.

To configure DSCP-based PFC:

1. Map a lossless forwarding class to a PFC priority—a 3-bit value represented in decimal form (0-7)—to use in the PFC pause frames.

   You must also assign an output queue to the forwarding class with the **queue-num** option. The **no-loss** option is required in this case to support lossless behavior for DSCP-based PFC, and the **pfc-priority** statement specifies the priority value mapping, as follows:

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

2. Define an input congestion notification profile to enable PFC on traffic specified by the desired 6-bit DSCP value, and optionally configure the maximum receive unit (MRU) at this time (used to determine PFC buffer headroom space reserved for the link):

   ```
   [edit class-of-service]
   user@switch# set congestion-notification-profile name input dscp code-point code-point-bits pfc mru mru-value
   ```

NOTE: You cannot configure both DSCP-based PFC and IEEE 802.1p PFC under the same congestion notification profile.

3. Set up a DSCP classifier for the confgured DSCP value and no-loss forwarding class mapped in the previous steps:

   `[edit class-of-service]`

   `user@switch#` **set classifiers dscp** ***classifier-name*** **forwarding-class** ***class-name*** **loss-priority** ***level*** **code-points** ***code-point-bits***

4. Assign the classifier and congestion notification profile set up in the previous steps to an interface on which you are enabling DSCP-based PFC:

   `[edit class-of-service]`

   `user@switch#` **set interfaces** ***interface-name*** **classifiers dscp** ***classifier-name***

   `user @swtich#` **set interfaces** ***interface-name*** **congestion-notification-profile** ***profile-name***

For example, with the following sample commands configuring DSCP-based PFC for interface xe-0/0/1, PFC pause frames will be generated with PFC priority 3 when incoming traffic with DSCP value 110000 becomes congested:

set interfaces xe-0/0/1 unit 0 family inet address 10.1.1.2/24

set class-of-service forwarding-classes class fc1 queue-num 1 no-loss

set class-of-service forwarding-classes class fc1 pfc-priority 3

set class-of-service congestion-notification-profile dpfc-cnp input dscp code-point 110000 pfc

set class-of-service classifiers dscp dpfc forwarding-class fc1 loss-priority low code-points 110000

set class-of-service interfaces xe-0/0/1 congestion-notification-profile dpfc-cnp

set class-of-service interfaces xe-0/0/1 classifiers dscp dpfc

RELATED DOCUMENTATION

Understanding PFC Using DSCP at Layer 3 for Untagged Traffic | 242

Configuring CoS PFC (Congestion Notification Profiles) | 194

Defining CoS BA Classifiers (DSCP, DSCP IPv6, IEEE 802.1p) | 94

Defining CoS Forwarding Classes | 146

CHAPTER 8

CoS and Host Outbound Traffic

IN THIS CHAPTER

Understanding Host Routing Engine Outbound Traffic Queues and Defaults

The host Routing Engine and CPU generate outbound traffic that is transmitted using different protocols. You cannot configure a classifier to map different types of outbound traffic that the host generates to forwarding classes (queues). The traffic that the host generates is assigned to forwarding classes by default as shown in Table 62 on page 248.

If you want to separate host outbound traffic from other traffic or if you want to assign that traffic to a particular queue, you can configure a single forwarding class for all traffic that the host generates. If you configure a forwarding class for outbound host traffic, that forwarding class is used globally for all traffic generated by the host. (That is, the host outbound traffic is mapped to the selected queue on all egress interfaces.) Configuring a forwarding class for host outbound traffic does not affect transit or incoming traffic.

Whether you use the default host outbound traffic forwarding class configuration or configure a forwarding class for all host outbound traffic, the configuration applies to all Layer 2 and Layer 3 protocols and to all application-level traffic such as FTP and ping operations.

If you configure a queue for host outbound traffic, the queue must be properly configured on all interfaces.

NOTE: Fibre Channel over Ethernet (FCoE) Initialization Protocol (FIP) packets generated by the CPU are always transmitted on the **fcoe** queue (queue 3), even if you configure a queue for host outbound traffic. This helps to ensure lossless behavior for FCoE traffic. QFabric systems classify FIP control packets into the same traffic class (**fcoe**) across the Interconnect device (fabric) and the egress Node device.

This does not apply to OCX Series switches, which do not support FCoE.

By default, traffic generated by the host is sent to the best effort queue (queue 0) or to the network control queue (queue 7). Table 62 on page 248 lists the default host traffic to output queue mapping.

Table 62: Routing Engine Protocol Default Queue Mapping

Routing Engine Protocol	Default Queue Mapping
Address Resolution Protocol (ARP) reply	Queue 0
ARP request	Queue 0
Border Gateway Protocol (BGP)	Queue 0
BGP TCP Retransmission	Queue 7

Table 62: Routing Engine Protocol Default Queue Mapping *(continued)*

Routing Engine Protocol	Default Queue Mapping
Fibre Channel over Ethernet (FCoE) Initialization Protocol (FIP)	Queue 3
File Transfer Protocol (FTP)	Queue 0
Internet Control Message Protocol (ICMP) reply	Queue 0
ICMP request	Queue 0
Internet Group Management Protocol (IGMP) query	Queue 7
IGMP report	Queue 0
Link Aggregation Control Protocol (LACP)	Queue 7
Open Shortest Path First (OSPF) hello	Queue 7
OSPF protocol data unit (PDU)	Queue 7
OSPF link state advertisements (LSAs)	Queue 7
Protocol Independent Multicast (PIM)	Queue 7
PIM hello	Queue 7
Simple Network Management Protocol (SNMP)	Queue 0
Secure Shell (SSH)	Queue 0
Telnet	Queue 0
Virtual Router Redundancy Protocol (VRRP)	Queue 7
VLAN Spanning Tree Protocol (VSTP)	Queue 7
xnm-clear-text	Queue 0
xnm-ssl	Queue 0

RELATED DOCUMENTATION

Changing the Host Outbound Traffic Default Queue Mapping

If you do not want to use the default mapping of host Routing Engine and CPU outbound traffic to queues, you can change the default output queue. You can also change the default DSCP bits used in the type of service (ToS) field of packets generated by the Routing Engine.

Configuring a queue for host outbound traffic maps all traffic that the host generates to one forwarding class (queue). The configuration is global and applies to all host-generated traffic on the switch. Configuring a forwarding class for host outbound traffic does not affect transit or incoming traffic.

> **NOTE:** Fibre Channel over Ethernet (FCoE) Initialization Protocol (FIP) packets generated by the CPU are always transmitted on the **fcoe** queue (queue 3), even if you configure a queue for host outbound traffic. This helps to ensure lossless behavior for FCoE traffic. QFabric systems classify FIP control packets into the same traffic class (**fcoe**) across the Interconnect device (fabric) and the egress Node device.
>
> This does not apply to OCX Series switches, which do not support FCoE.

To change the host outbound traffic egress queue by including the **host-outbound-traffic** statement at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
host-outbound-traffic {
  forwarding-class class-name;
  dscp-code-point code-point;
}
```

For example, to map host outbound traffic to queue 7 (the network control forwarding class) and set the DSCP code point value to 101010:

```
[edit class-of-service]
host-outbound-traffic {
  forwarding-class network-control;
```

```
    dscp-code-point 101010
}
```

RELATED DOCUMENTATION

Understanding Host Routing Engine Outbound Traffic Queues and Defaults | 248

2

PART

Weighted Random Early Detection (WRED) and Explicit Congestion Notification (ECN)

CHAPTER 9

WRED and Drop Profiles

IN THIS CHAPTER

Understanding CoS WRED Drop Profiles

IN THIS SECTION

When the number of packets queued is greater than the ability of the switch to empty an output queue, the queue requires a method for determining which packets to drop to relieve the congestion. Weighted random early detection (WRED) drop profiles define the drop probability of packets of different packet loss probabilities (PLPs) as the output queue fills. During periods of congestion, as the output queue fills, the switch drops incoming packets as determined by a drop profile, until the output queue becomes less congested.

Depending on the drop probabilities, a drop profile can drop many packets long before the buffer becomes full, or it can drop only a few packets even if the buffer is almost full.

You configure drop profiles in the drop profile section of the class-of-service (CoS) configuration hierarchy. You apply drop profiles using a drop profile map in queue scheduler configuration. For each queue scheduler, you can configure separate drop profiles for each PLP using the **loss-priority** attribute (low, medium-high, and high). This enables you to treat traffic of different PLPs in different ways during periods of congestion.

NOTE: Do not apply drop profiles to lossless traffic (traffic that belongs to a forwarding class that has the **no-loss** drop attribute.). Lossless traffic uses priority-based flow control (PFC) to control congestion.

OCX Series switches do not support lossless transport and do not support PFC.

NOTE: You cannot apply drop profiles to multidestination queues on switches that support them.

Drop Profile Parameters

Drop profiles specify two values, which work as pairs:

- Fill level—The queue fullness value, which represents a percentage of the memory used to store packets in relation to the total amount of memory allocated to the queue.
- Drop probability—The percentage value that corresponds to the likelihood that an individual packet is dropped.

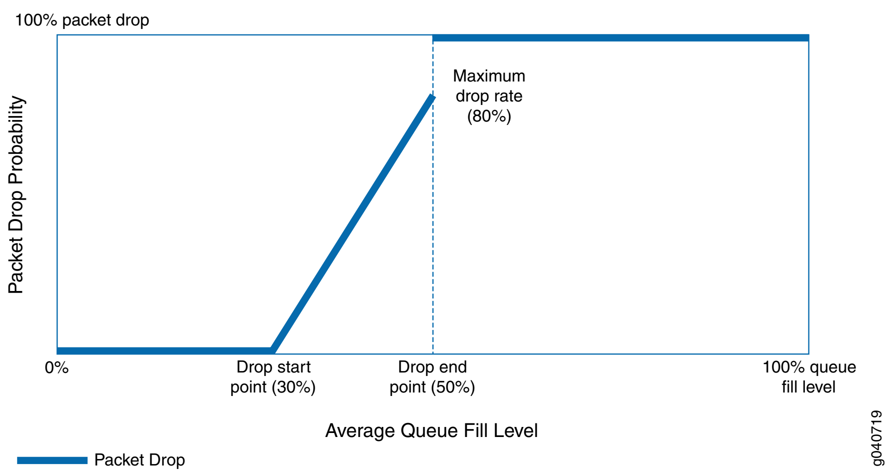
Defining Drop Profiles on Switches Except QFX10000

You set two queue fill levels and two drop probabilities in each drop profile. The first fill level and the first drop probability create one value pair and the second fill level and the second drop probability create a second value pair.

The first fill level value specifies the percentage of queue fullness at which packets begin to drop, known as the drop start point. Until the queue reaches this level of fullness, no packets are dropped. The second fill level value specifies the percentage of queue fullness at which all packets are dropped, known as the drop end point.

The first drop probability value is always **0** (zero). This pairs with the drop start point and specifies that until the queue fullness level reaches the first fill level, no packets drop. When the queue fullness exceeds the drop start point, packets begin to drop until the queue exceeds the second fill level, when all packets drop. The second drop probability value, known as the maximum drop rate, specifies the likelihood of dropping packets when the queue fullness reaches the drop end point. As the queue fills from the drop start point to the drop end point, packets drop in a smooth, linear pattern (called an interpolated graph) as shown in Figure 8 on page 255. After the drop end point, all packets drop.

Figure 8: WRED-Drop Profile Packet Drop Pattern



The thick line in Figure 8 on page 255 shows the packet drop characteristics for a sample WRED profile. At the drop start point, the queue reaches a fill level of 30 percent. At the drop end point, the queue fill level reaches 50 percent, and the maximum drop rate is 80 percent.

No packets drop until the queue fill level reaches the drop start point of 30 percent. When the queue reaches the 30 percent fill level, packets begin to drop. As the queue fills, the percentage of packets dropped increases in a linear fashion. When the queue fills to the drop end point of 50 percent, the rate of packet drop has increased to the maximum drop rate of 80 percent. When the queue fill level exceeds the drop end point of 50 percent, all of the packets drop until the queue fill level drops below 50 percent.

Defining Drop Profiles on QFX10000 Switches

Each queue fill level pairs with a drop probability. As the queue fills to different levels, every time it reaches a fill level configured in a drop profile, the queue applies the drop probability paired with that fill level to the traffic in the queue that exceeds the fill level. You can configure up to 32 pairs of fill levels and drop probabilities to create a customized packet drop probability curve with up to 32 points of differentiation.

Packets are not dropped until they reach the first configured queue fill level. When the queue reaches the first fill level, packets begin to drop at the configured drop probability rate paired with the first fill level. When the queue reaches the second fill level, packets begin to drop at the configured drop probability rate paired with the second fill level. This process continues for the number of fill level/drop probability pairs that you configure in the drop profile.

Drop profiles are interpolated, not segmented. An interpolated drop profile gradually increases the drop probability along a curve between each configured fill level. When the queue reaches the next fill level, the drop probability reaches the drop probability paired with that fill level. A segmented drop profile "jumps" from one fill level and drop probability setting to another in a stepped fashion. The drop probability of traffic does not change as the queue fills until the next fill level is reached.

An example of interpolation is a drop profile with three fill level/drop probability pairs:

- 25 percent queue fill level paired with a 30 percent drop probability
- 50 percent queue fill level paired with a 60 percent drop probability
- 75 percent queue fill level paired with a 100 percent drop probability (all packets that exceed the 75 percent queue fill level are dropped)

The queue drops no packets until its fill level reaches 25 percent. During periods of congestion, when the queue fills above 25 percent full, the queue begins to drop packets at a rate of 30 percent of the packets above the fill level.

However, as the queue continues to fill, it does not continue to drop packets at the 30 percent drop probability. Instead, the drop probability gradually increases as the queue fills to the 50 percent fullness level. When the queue reaches the 50 percent fill level, the drop probability has increased to the configured drop probability pair for the fill level, which is 60 percent.

As the queue continues to fill, the drop probability does not remain at 60 percent, but continues to rise as the queue fills. When the queue reaches the final fill level at 75 percent full, the drop probability has risen to 100 percent and all packets that exceed the 75 percent fill level are dropped.

Default Drop Profile

If you do not configure drop profiles and apply them to queue schedulers, the switch uses the default drop profile for lossy traffic classes. In the default drop profile, when the fill level is 0 percent, the drop probability is 0 percent. When the fill level is 100 percent, the drop probability is 100 percent. During periods of congestion, as soon as packets arrive on a queue, the default profile might begin to drop packets.

Packet Drop Method

When a packet reaches the head of a queue, the switch calculates a random number between 0 and 100. The switch plots the random number against the drop profile using the current fill level of the queue. When the random number falls above the graph line, the queue transmits the packet out the egress interface. When the number falls below graph the line, the switch drops the packet.

Packet Drop Example for Switches Except QFX10000

To create the linear drop pattern from the drop start point to the drop end point, the drop probabilities are derived using a linear approximation with eight sections, or steps, from the minimum queue fill level to the maximum queue fill level. The fill levels are divided into the eight sections equally, starting at the minimum fill level and ending at the maximum fill level. As the queue fills, the percentage of dropped packets increases. The percentage of packets dropped is based on the maximum drop rate.

For example, the default drop profile (which specifies a maximum drop rate of 100 percent) has the following drop probabilities at each section, or step, in the eight-section linear drop pattern:

- First section—The minimum drop probability is 6.25 percent of the maximum drop rate. The maximum drop probability is 12.5 percent of the maximum drop rate.
- Second section—The minimum drop probability is 18.75 percent of the maximum drop rate. The maximum drop probability is 25 percent of the maximum drop rate.
- Third section—The minimum drop probability is 30.25 percent of the maximum drop rate. The maximum drop probability is 37.5 percent of the maximum drop rate.
- Fourth section—The minimum drop probability is 43.75 percent of the maximum drop rate. The maximum drop probability is 50 percent of the maximum drop rate.
- Fifth section—The minimum drop probability is 56.25 percent of the maximum drop rate. The maximum drop probability is 62 percent of the maximum drop rate.
- Sixth section—The minimum drop probability is 68.75 percent of the maximum drop rate. The maximum drop probability is 75.5 percent of the maximum drop rate.
- Seventh section—The minimum drop probability is 81.25 percent of the maximum drop rate. The maximum drop probability is 87.5 percent of the maximum drop rate.
- Eighth section—The minimum drop probability is 92.75 percent of the maximum drop rate. The maximum drop probability is 100 percent of the maximum drop rate.

Packets drop even when there is no congestion, because packet drops begin at the drop start point regardless of whether congestion exists on the port. The default drop profile example represents the worst-case scenario, because the drop start point fill level is 0 percent, so packet drop begins when the queue starts to receive packets.

You can specify when packets begin to drop by configuring a drop start point at a fill level greater than 0 percent. For example, if you configure a drop profile that has a drop start point of 30 percent, packets do not drop until the queue is 30 percent full. We recommend that you configure drop profiles that are appropriate to your network traffic conditions.

The smaller the gap between the minimum drop rate (which is always 0) and the maximum drop rate, the smaller the gap between the minimum drop probability and the maximum drop probability at each section (step) of the linear drop pattern. The default drop profile, which has the maximum gap between the minimum drop rate (0 percent) and the maximum drop rate (100 percent), has the highest gap between the minimum drop probability and the maximum drop probability at each step. Configuring a lower maximum drop rate for a drop profile reduces the gap between the minimum drop probability and the maximum drop probability.

Drop Profile Maps

Drop profile maps are part of scheduler configuration. A drop profile map maps drop profiles to packet loss priorities. Specifying the drop profile map in a scheduler associates the drop profile with the forwarding classes (queues) that you map to the scheduler in a scheduler map.

You configure loss priority for a queue in the classifier section of the CoS configuration hierarchy, and the loss priority is applied to the traffic assigned to the forwarding class at the ingress interface.

Each scheduler can have multiple drop profile maps.

Congestion Prevention

Configuring drop profiles on output queues enables you to control how congestion affects other queues on a port. If you do not configure drop profiles and map them to output queues, the switch uses the default drop profile on queues that forward lossy traffic.

For example, if an ingress port forwards traffic to more than one egress port, and at least one of the egress ports experiences congestion, that can cause ingress port congestion. Ingress port congestion (ingress buffer exceeds its resource allocation) can cause frames to drop at the ingress port instead of at the egress port. Ingress port frame drop affects all of the egress ports to which the congested ingress port forwards traffic, not just the congested egress port.

> **NOTE:** Do not configure drop profiles for the **fcoe** and **no-loss** forwarding classes. FCoE and other lossless traffic queues require lossless behavior (traffic queues that are configured with the **no-loss** packet drop attribute). Use priority-based flow control (PFC) to prevent frame drop on lossless priorities.
>
> OCX Series switches do not support lossless transport and do not support PFC.

Configuring a WRED Drop Profile and Applying it to an Output Queue

To configure a WRED packet drop profile and apply it to an output queue:

1. Configure a drop profile:
   - On switches except QFX10000 use 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***.
   - On QFX10000 switches use 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) 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. On switches except QFX10000, 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. On switches except QFX10000, associate the traffic control profile with an interface using the statement **set class-of-service interfaces *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) 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.
6. On QFX10000 switches, 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.

Drop Profiles on Explicit Congestion Notification Enabled Queues

You must configure a WRED drop profile on queues that you enable for explicit congestion notification (ECN). On ECN-enabled queues, the drop profile sets the threshold for when the queue should mark a packet as experiencing congestion (see "Understanding CoS Explicit Congestion Notification" on page 274). When a queue fills to the level at which the WRED drop profile has a packet drop probability greater than zero (0), the switch might mark a packet as experiencing congestion. Whether or not a switch marks a packet as experiencing congestion is the same probability as the drop probability of the queue at that fill level.

On ECN-enabled queues, the switch does not use the drop profile to control dropping packets that are not ECN-capable packets (packets marked non-ECT, ECN code bits 00) during periods of congestion. Instead, the switch uses the tail-drop algorithm to drop non-ECN-capable packets during periods of congestion. When a queue fills to its maximum level of fullness, tail-drop simply drops all subsequently arriving packets until there is space in the queue to buffer more packets. All non-ECN-capable packets are treated the same way.

To apply a WRED drop profile to non-ECT traffic, configure a multifield (MF) classifier to assign non-ECT traffic to a different output queue that is not ECN-enabled, and then apply the WRED drop profile to that queue.

RELATED DOCUMENTATION

Understanding Junos CoS Components | 17
Example: Configuring CoS Hierarchical Port Scheduling (ETS) | 415
Understanding CoS Explicit Congestion Notification | 274
Example: Configuring WRED Drop Profiles | 264
Example: Configuring Drop Profile Maps | 271
Example: Configuring Unicast Classifiers | 100
Configuring CoS WRED Drop Profiles | 261
Configuring CoS Drop Profile Maps | 270
Defining CoS BA Classifiers (DSCP, DSCP IPv6, IEEE 802.1p) | 94

Configuring CoS WRED Drop Profiles

IN THIS SECTION

You can configure an interpolated weighted random early detection (WRED) profile to control traffic congestion by controlling packet drop characteristics for different packet loss priorities.

Drop profiles specify two values, which work as pairs:

- Fill level—The queue fullness value, which represents a percentage of the memory used to store packets in relation to the total amount of memory allocated to the queue.
- Drop probability—The percentage value that corresponds to the likelihood that an individual packet is dropped.

> **NOTE:** Do not enable WRED on lossless traffic flows (forwarding classes configured with the **no-loss** packet drop attribute). Use priority-based flow control (PFC) to prevent packet loss on lossless forwarding classes.
>
> Except on QFX10000, you cannot enable WRED on multidestination (multicast) queues on. You can enable WRED only on unicast queues.
>
> OCX Series switches do not support lossless flows or PFC.

NOTE: On ECN-enabled queues, the drop profile sets the threshold for when the queue should mark a packet as experiencing congestion (see "Understanding CoS Explicit Congestion Notification" on page 274). On ECN-enabled queues, the switch does not use the drop profile to control dropping packets that are not ECN-capable packets during periods of congestion. Instead, the switch uses the tail-drop algorithm to drop non-ECN-capable packets during periods of congestion. When a queue fills to its maximum level of fullness, tail-drop simply drops all subsequently arriving packets until there is space in the queue to buffer more packets. All non-ECN-capable packets are treated the same way.

Drop Profiles on Switches Except QFX10000

Interpolated means that the switch creates a smooth drop curve from a drop start point to a drop end point, with a maximum drop rate that is reached at the drop end point.

The dropstart point is the average queue fill level when the WRED algorithm starts to drop packets. Before the drop start point, no packets are scheduled to drop. Specify the drop start point using the first of two **fill-level** statements.

The drop end point is the average queue fill level at which all subsequently arriving packets are dropped. When the queue fill levels falls below the drop end point, packets begin to be forwarded again. (At the drop end point, the packet drop probability becomes 100 percent.) Specify the drop end point using the second of two **fill-level** statements.

The minimum drop rate is always **0**. Specify the minimum drop rate using the first of two **drop-probability** statements. The maximum drop rate is the drop probability when the average queue fill level reaches the drop end point. Specify the maximum drop rate using the second of two **drop-probability** statements.

The drop rate is zero until the queue fill level reaches the drop start point. As the queue continues to fill, packets drop in smooth linear curveuntil the queue reaches the drop end point, when packets drop at the maximum drop rate. If the queue fills beyond the drop end point, all packets that match the drop profile are dropped.

To configure a WRED profile using the CLI on switches except QFX10000:

Name the drop profile and set the drop start point, drop end point, minimum drop rate, and maximum drop rate for the drop profile:

`[edit class-of-service]`

`user@switch#` **set drop-profile** ***drop-profile-name*** **interpolate fill-level** ***percentage*** **fill-level** ***percentage*** **drop-probability 0 drop-probability** ***percentage***

Drop Profiles on QFX 10000 Switches

Each queue fill level pairs with a drop probability. As the queue fills to different levels, every time it reaches afill level configured in a drop profile, the queue applies thedrop probability paired with that fill level to the traffic in the queue that exceeds the fill level. You can configure up to 32 pairs of fill levels and drop probabilities to create a customized packet drop probability curve with up to 32 points of differentiation.

Packets are not dropped until they reach the first configured queue fill level. When the queue reaches the firstfill level, packets begin to drop at the configured drop probability rate paired with the first fill level. When the queue reaches the second fill level, packets begin to drop at the configured drop probability rate paired with the second fill level. This process continues for the number of fill level/drop probability pairs that you configure in the drop profile.

Drop profiles are *interpolated*. An interpolated drop profile gradually increases the drop probability along a curve between each configured fill level. When the queue reaches the next fill level, the drop probability reaches the drop probability paired with that fill level.

To configure a WRED profile using the CLI on QFX10000 switches:

Name the drop profile and set the fill levels and their associated drop probabilities as percentages. For every fill level, there must be a paired drop probability (you must configure the same number of fill levels and drop probabilities).

```
[edit class-of-service]
user@switch# set drop-profile drop-profile-name interpolate fill-level level1 level2 ... level32
drop-probability probability1 probability2 ... probability32
```

RELATED DOCUMENTATION

Example: Configuring CoS Hierarchical Port Scheduling (ETS) | **415**

Example: Configuring WRED Drop Profiles | **264**

Defining CoS Queue Schedulers | **321**

Defining CoS Queue Schedulers for Port Scheduling | **357**

Configuring CoS Drop Profile Maps | **270**

Understanding CoS WRED Drop Profiles | **253**

Example: Configuring WRED Drop Profiles

IN THIS SECTION

You can configure interpolated weighted random early detection (WRED) profiles to control traffic congestion by controlling packet drop characteristics for different packet loss priorities.

> **NOTE:** Do not enable WRED on lossless traffic flows. Use priority-based flow control (PFC) to prevent packet loss on lossless forwarding classes. (OCX Series switches do not support lossless flows or PFC.)
>
> Except on QFX10000 switches, you cannot enable WRED on multidestination (multicast) queues. You can enable WRED only on unicast queues.

Requirements

This example uses the following hardware and software components:

- One switch
- Junos OS Release 11.1 or later for the QFX Series or Junos OS Release 14.1X53-D20 or later for the OCX Series or Junos OS Release 15.1X53-D10 or later for the QFX10000.

Overview

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. Drop profiles specify two values, which work as pairs:

- Fill level—The queue fullness value, which represents a percentage of the memory used to store packets in relation to the total amount of memory allocated to the queue.

- Drop probability—The percentage value that corresponds to the likelihood that an individual packet is dropped.

> **NOTE:** On ECN-enabled queues, the drop profile sets the threshold for when the queue should mark a packet as experiencing congestion (see "Understanding CoS Explicit Congestion Notification" on page 274). On ECN-enabled queues, the switch does not use the drop profile to control dropping packets that are not ECN-capable packets during periods of congestion. Instead, the switch uses the tail-drop algorithm to drop non-ECN-capable packets during periods of congestion. When a queue fills to its maximum level of fullness, tail-drop simply drops all subsequently arriving packets until there is space in the queue to buffer more packets. All non-ECN-capable packets are treated the same way.

Configuring WRED Drop Profiles on Switches Except QFX10000

Configuration

Step-by-Step Procedure

Interpolated means that the switch creates a smooth drop curve from a drop start point to a drop end point, with a maximum drop rate that is reached at the drop end point:

- Drop start point—Percentage of average queue fill level when the WRED algorithm starts to drop packets. Before the drop start point, no packets are scheduled to drop.
- Drop end point—Average queue fill level at which all subsequently arriving packets are dropped. When the queue fill levels falls below the drop end point, packets begin to be forwarded again. (At the drop end point, the packet drop probability becomes 100 percent.)
- Maximum drop rate—Drop probability when the average queue fill level reaches the drop end point.

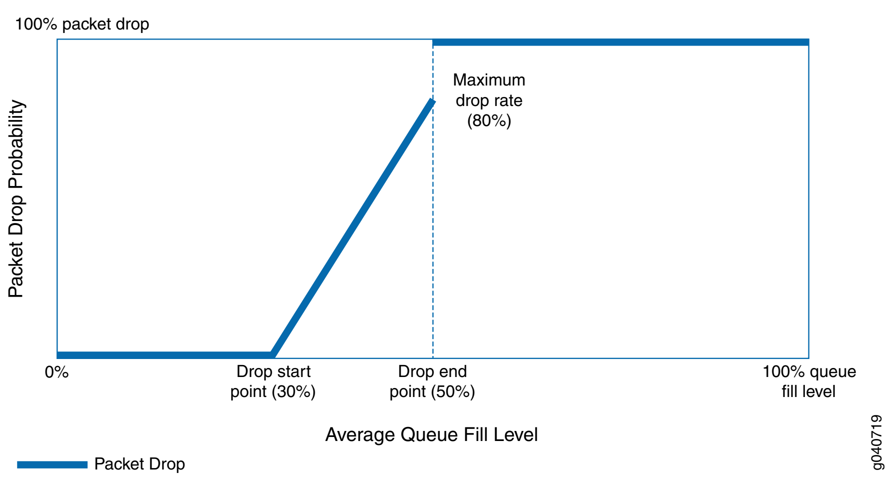
You set the drop start point and the drop end point by specifying two queue fill level percentage values. The first value is the drop start point and the second value is the drop end point.

You set the maximum drop rate by specifying two drop probability percentage values. The first value is always zero (0), which is the minimum drop rate, the probability of dropping a packet at the drop start point. The second value is the maximum drop rate at the drop end point.

The drop rate is zero until the queue fill level reaches the drop start point. As the queue continues to fill, packets drop in smooth linear curve until the queue reaches the drop end point, when packets drop at the maximum drop rate. If the queue fills beyond the drop end point, all packets that match the drop profile are dropped.

Figure 9 on page 266 shows the graph for a drop profile with a drop start point of 30 percent, a drop end point of 50 percent, and a maximum drop rate of 80 percent.

Figure 9: WRED Drop Profile Packet Drop Example



The graph shows that when the queue fill level is less than 30 percent, the packet drop rate is zero. When the queue fill level reaches 30 percent, packets begin to drop. As the queue fills, a higher percentage of packets drop. When the queue fill level reaches 50 percent, the packet drop rate has climbed to 80 percent. When the queue fill level exceeds 50 percent, all packets drop.

This example describes how to configure the drop profile shown in Figure 9 on page 266. The drop profile will have:

- The name **be-dp1**
- 30 percent for the drop start point (first **fill-level** setting)
- 50 percent for the drop end point (second **fill-level** setting)
- 0 percent for the minimum drop rate (first **drop-probability** setting)
- 80 percent for the maximum drop rate (second **drop-probability** setting)

You apply a drop profile by configuring a drop profile map that maps the drop profile to a packet loss priority, and associate the drop profile and packet loss priority with a scheduler. When you map the scheduler to a forwarding class (queue), the switch applies the drop profile to the packets in the forwarding class that have a matching packet loss priority.

1. Set the drop start point at **30** percent, the drop end point at **50** percent, the minimum drop rate at **0** percent, and the maximum drop rate at **80** percent for the drop profile **be-dp1**:

   ```
   [edit class-of-service]
   user@switch# set drop-profile be-dp1 interpolate fill-level 30 fill-level 50 drop-probability 0 drop-probability 80
   ```

Verification

Verifying the Drop Profile Configuration

Purpose

Verify that you configured the drop profile **be-dp1** with the correct drop start and end points and with the correct drop rates.

Action

Verify the results of the drop profile configuration using the operational mode command **show configuration class-of-service drop-profiles be-dp1**:

```
user@switch> show configuration class-of-service drop-profiles be-dp1
```

```
interpolate {
    fill-level [ 30 50 ];
    drop-probability [ 0 80 ];
}
```

Configuring WRED Drop Profiles on QFX10000 Switches

Configuration

Step-by-Step Procedure

Each queue fill level pairs with a drop probability. As the queue fills to different levels, every time it reaches a fill level configured in a drop profile, the queue applies the drop probability paired with that fill level to the traffic in the queue that exceeds the fill level. You can configure up to 32 pairs of fill levels and drop probabilities to create a customized packet drop probability curve with up to 32 points of differentiation.

Packets are not dropped until they reach the first configured queue fill level. When the queue reaches the first fill level, packets begin to drop at the configured drop probability rate paired with the first fill level. When the queue reaches the second fill level, packets begin to drop at the configured drop probability rate paired with the second fill level. This process continues for the number of fill level/drop probability pairs that you configure in the drop profile.

Drop profiles are *interpolated*. An interpolated drop profile gradually increases the drop probability along a curve between each configured fill level. When the queue reaches the next fill level, the drop probability reaches the drop probability paired with that fill level.

This example describes how to configure a drop profile with three fill level/drop probability pairs:

- Drop profile name—**be-dp1**
- Queue fill levels—25 percent, 50 percent, 75 percent
- Drop probabilities—30 percent, 60 percent, 100 percent

Each of the three fill levels pairs with a drop probability to program the interpolated drop profile curve.

You apply a drop profile by configuring a drop profile map that maps the drop profile to a packet loss priority, and associate the drop profile and packet loss priority with a scheduler. When you map the scheduler to a forwarding class (queue), the switch applies the drop profile to the packets in the forwarding class that have a matching packet loss priority.

To configure a drop profile:

1. Set the drop start point at a **25** percent fill level, an intermediate fill level of **50** percent, and a drop end point of **75** percent. Set the paired drop probabilities to **30** percent, **60** percent, and **100** percent, respectively, for drop profile **be-dp1**:

   ```
   [edit class-of-service]
   user@switch# set drop-profile be-dp1 interpolate fill-level [ 25 50 75 ] drop-probability [ 30 60 100 ]
   ```

Verification

Verifying the Drop Profile Configuration

Purpose

Verify that you configured the drop profile **be-dp1** with the correct fill levels and drop probabilities.

Action

Verify the results of the drop profile configuration using the operational mode command **show configuration class-of-service drop-profiles be-dp1**:

`user@switch>` **show configuration class-of-service drop-profiles be-dp1**

```
interpolate {
    fill-level [ 25 50 75 ];
    drop-probability [ 30 60 100 ];
}
```

Configuring CoS Drop Profile Maps

A drop-profile map associates weighted random early detection (WRED) profiles for traffic of specified packet loss priorities with a scheduler. When you use a scheduler map to map a scheduler to a forwarding class, the drop profile map associated with the scheduler applies the specified WRED drop profile to traffic in the forwarding class that matches the specified packet loss priority.

Drop profile maps enable you to configure different drop profiles for traffic of different packet loss priorities within the same scheduler. You can associate different drop profiles with low-priority, medium-high priority, and high-priority traffic within a single scheduler, and then map that scheduler to a forwarding class. This applies the appropriate drop profile to traffic of each loss priority in a forwarding class. Drop profile maps apply to all traffic protocols.

To configure a drop-profile map:

- For the desired scheduler, configure the traffic loss priority and specify the drop profile you want to use to control the drop characteristics for traffic of that loss priority:

  `[edit` class-of-service`]`
  `user@switch#` **set** schedulers ***scheduler-name*** drop-profile-map loss-priority ***level*** **protocol any** drop-profile ***drop-profile-name***

  > **NOTE:** QFX10000 switches do not support the **protocol any** portion of the configuration. Drop profiles apply to all protocols.

Example: Configuring Drop Profile Maps

IN THIS SECTION

A drop-profile map associates weighted random early detection (WRED) profiles for traffic of specified packet loss priorities with a scheduler. When you use a scheduler map to map a scheduler to a forwarding class, the drop profile map associated with the scheduler applies the specified WRED drop profile to traffic in the forwarding class that matches the specified packet loss priority.

Requirements

This example uses the following hardware and software components:

- A Juniper Networks QFX3500 Switch
- Junos OS Release 11.1 or later for the QFX Series or Junos OS Release 14.1X53-D20 or later for the OCX Series.

Overview

Drop profile maps enable you to configure different drop profiles for traffic of different packet loss priorities within the same scheduler. You can associate different drop profiles with low-priority, medium-high priority, and high-priority traffic within a single scheduler, and then map that scheduler to a forwarding class. This applies the appropriate drop profile to traffic of each loss priority in a forwarding class. Drop profile maps apply to all traffic protocols.

The following example describes how to configure a drop profile map for a scheduler named **mylan** that includes:

- A drop profile called **lp-profile** for low-priority traffic
- A drop profile called **mh-profile** for medium-high priority traffic
- A drop profile called **h-profile** for high-priority traffic

You apply the drop profiles in the drop profile map to a forwarding class by associating the scheduler **mylan** with a forwarding class in a scheduler map.

Configuring a Drop Profile Map

CLI Quick Configuration

To quickly configure a drop profile map, 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 mylan drop-profile-map loss-priority low protocol any drop-profile lp-profile
set schedulers mylan drop-profile-map loss-priority medium-high protocol any drop-profile mh-profile
set schedulers mylan drop-profile-map loss-priority high protocol any drop-profile h-profile
```

To configure a drop profile map:

1. Configure the drop profile for low-priority traffic:

   ```
   [edit class-of-service]
   user@switch# set schedulers mylan drop-profile-map loss-priority low protocol any drop-profile lp-profile
   ```

2. Configure the drop profile for medium-high priority traffic:

   ```
   [edit class-of-service]
   user@switch# set schedulers mylan drop-profile-map loss-priority medium-high protocol any drop-profile mh-profile
   ```

3. Configure the drop profile for high-priority traffic:

   ```
   [edit class-of-service]
   user@switch# set schedulers mylan drop-profile-map loss-priority high protocol any drop-profile h-profile
   ```

Verification

Verifying the Drop Profile Map Configuration

Purpose

Verify that you configured the drop profile map for the scheduler **mylan** with the correct loss priorities and drop profiles.

Action

Verify the results of the drop profile map configuration using the operational mode command **show configuration class-of-service schedulers mylan**:

user@switch> **show configuration class-of-service schedulers mylan**

```
transmit-rate 3g;
shaping-rate percent 100;
priority low;
drop-profile-map loss-priority low protocol any drop-profile lp-profile;
drop-profile-map loss-priority medium-high protocol any drop-profile mh-profile;
drop-profile-map loss-priority high protocol any drop-profile h-profile;
```

NOTE: This example does not include configuring scheduler bandwidth and priority. This information (transmit rate, shaping rate, and priority) is shown for completeness.

RELATED DOCUMENTATION

Example: Configuring CoS Hierarchical Port Scheduling (ETS) | 415
Example: Configuring Queue Schedulers | 325
Example: Configuring Queue Schedulers for Port Scheduling | 361
Example: Configuring WRED Drop Profiles | 264
Configuring CoS Drop Profile Maps | 270
Understanding CoS WRED Drop Profiles | 253

CHAPTER 10

Explicit Congestion Notification (ECN)

IN THIS CHAPTER

Understanding CoS Explicit Congestion Notification

IN THIS SECTION

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 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.

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.

> **NOTE:** OCX Series switches do not support lossless transport and do not support PFC.

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

IN THIS SECTION

Without ECN, switches 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 switches 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 switches 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 switch might mark a packet as experiencing congestion. Whether or not a switch 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 switch, the switch 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 63 on page 276:

Table 63: 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

Table 63: ECN Bit Codes *(continued)*

ECN Bits (Code)	Meaning
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

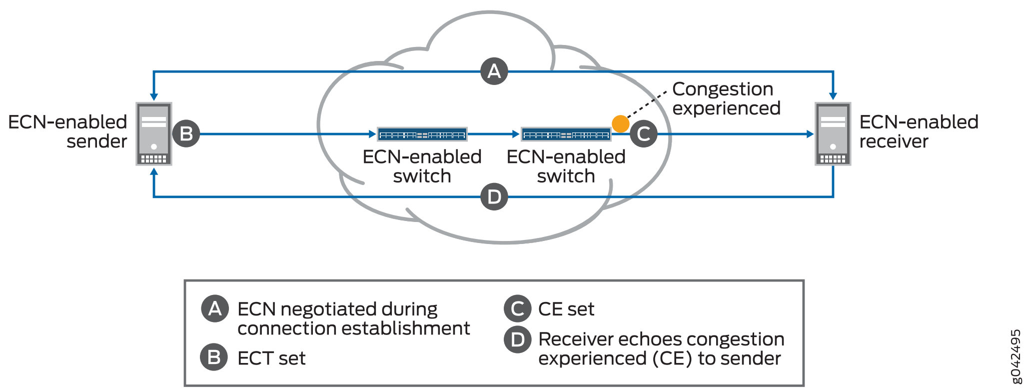
After the sending and receiving endpoints negotiate ECN, the sending endpoint marks packets as ECN-capable by setting the DiffServ ECN field to ECT(1) (01) or ECT(0) (10). Every intermediate switch between the endpoints must have ECN enabled or it does not work.

When a packet traverses a switch and experiences congestion at an output queue that uses the WRED packet drop mechanism, the switch 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 switch 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 might be dropped. See "WRED Drop Profile Control of ECN Thresholds" on page 281 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 10 on page 278 summarizes how ECN works to mitigate network congestion:

Figure 10: 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.

2. After successful negotiation of ECN capability, the ECN-capable sender sends IP packets with the ECT field set to the receiver.

   > **NOTE:** All of the intermediate devices in the path between the sender and the receiver must be ECN-enabled.

3. If the WRED algorithm on a switch egress queue determines that the queue is experiencing congestion and the packet is drop eligible, the switch 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 switch), the switch forwards the packet with CE marked.

   If there is no congestion at the switch egress queue, the switch 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.

   On QFX5210, QFX5200, QFX5100, EX4600, QFX3500, and QFX3600 switches, and on QFabric systems, packets that are not marked as ECN-capable (ECT, 00) are treated according to the WRED drop profile configuration and might be dropped during periods of congestion.

   On QFX10000 switches, the switch uses the tail-drop algorithm to drop packets that are marked ECT (00) during periods of congestion. (When a queue fills to its maximum level of fullness, tail-drop simply drops all subsequently arriving packets until there is space in the queue to buffer more packets. All non-ECN-capable packets are treated the same.)

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 64 on page 279 summarizes the behavior of traffic on ECN-enabled queues.

Table 64: 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
Non-ECT (00)	Does not matter	Drop (QFX5210, QFX5200, QFX5100, EX4600, QFX3500, QFX3600, QFabric systems). Tail drop occurs when queue reaches maximum fullness because no WRED drop probability is applied (QFX10000 switches).	No ECN bits marked
ECT (10 or 01)	ECN disabled	Drop	Packet dropped—no ECN bits marked
ECT (10 or 01)	ECN enabled	Do not drop. Mark packet as experiencing congestion (CE, bits 11).	Packet marked ECT (11) to indicate congestion
CE (11)	ECN disabled	Drop	Packet dropped—no ECN bits marked
CE (11)	ECN enabled	Do not drop. Packet is already marked as experiencing congestion, forward packet without changing the ECN marking.	Packet marked ECT (11) to indicate congestion

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.

> **NOTE:** QFX10000 switches do not support Ethernet PAUSE.
>
> OCX Series switches do not support PFC. OCX Series switches support Ethernet PAUSE on tagged Layer 3 interfaces.

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 switches. Instead, PFC controls flows between two PFC-enabled peer devices (for example, switches) 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.

PFC works this way: if a switch output queue fills to a certain threshold, the switch sends a PFC pause message to the connected peer device that is transmitting data. The pause message tells the transmitting switch to pause transmission of the flow. When the congestion clears, the switch sends another PFC message to tell the connected peer to resume transmission. (If the output queue of the transmitting switch also reaches a certain threshold, that switch can in turn send a PFC pause message to the connected peer that is transmitting to it. In this way, PFC can propagate a transmission pause back through the network.)

See "Understanding CoS Flow Control (Ethernet PAUSE and PFC)" on page 197 for more information. For QFX5100 and EX4600 switches only, you can also refer to "Understanding PFC Functionality Across Layer 3 Interfaces" on page 214.

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 switch 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. Strict-high priority queues receive all of the port bandwidth they require up to the configured maximum bandwidth limit (scheduler **transmit-rate** on QFX10000 switches, and **shaping-rate** on QFX5210, QFX5200, QFX5100, QFX3500, QFX3600, and EX4600 switches, and QFabric systems).

Different switches support different amounts of fill level/drop probability pairs in drop profiles. For example, QFX10000 switches support 32 fill level/drop probability pairs, so there can be as many as 30 intermediate fill level/drop probability pairs between the drop start and drop endpoints. QFX5210, QFX5200, QFX5100, QFX3500, QFX3600, and EX4600 switches, and QFabric systems support two fill level/drop probability pairs—by definition, the two pairs you configure on these switches are the drop start and drop end points.

> **NOTE:** Do not configure the last fill level as 100 percent.

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. (At this point on QFX5210, QFX5200, QFX5100, EX4600, QFX3500, and QFX3600 switches, and on QFabric systems, all non-ECN packets are dropped.) 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 switches 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.

Starting in Release 15.1, you can configure a WRED packet drop profile and apply it to an output queue on switches 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 switches 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

NOTE: On QFX10000 switches, when you enable a queue for ECN and apply a WRED drop profile to the queue, the WRED drop profile only sets the thresholds for marking ECN traffic as experiencing congestion (CE, 11). On ECN-enabled queues, the WRED drop profile does not set drop thresholds for non-ECT (00) traffic (traffic that is not ECN-capable). Instead, the switch uses the tail-drop algorithm on traffic is that is marked non-ECT on ECN-enabled queues during periods of congestion.

To apply a WRED drop profile to non-ECT traffic, configure a multifield (MF) classifier to assign non-ECT traffic to a different output queue that is not ECN-enabled, and then apply the WRED drop profile to that queue.

Release History Table

Release	Description
15.1	Starting in Release 15.1, you can configure a WRED packet drop profile and apply it to an output queue on switches that support port scheduling (ETS hierarchical scheduling is either not supported or not used).

RELATED DOCUMENTATION

Example: Configuring ECN | 284

Example: Configuring ECN

IN THIS SECTION

This example shows how to enable explicit congestion notification (ECN) on an output queue.

Requirements

This example uses the following hardware and software components:

- One switch.
- Junos OS Release 13.2X51-D25 or later for the QFX Series or Junos OS Release 14.1X53-D20 for the OCX Series

Overview

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 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

A weighted random early detection (WRED) packet drop profile must be applied to the output queues on which ECN is enabled. ECN uses the WRED drop profile thresholds to mark packets when the output queue experiences congestion.

ECN reduces packet loss by forwarding ECN-capable packets during periods of network congestion instead of dropping those packets. (TCP notifies the network about congestion by dropping packets.) During periods of congestion, ECN marks ECN-capable packets that egress from congested queues. When the receiver receives an ECN packet that is marked as experiencing congestion, the receiver echoes the congestion state back to the sender. The sender then reduces its transmission rate to clear the congestion.

ECN is disabled by default. You can enable ECN on best-effort traffic. ECN should not be enabled on lossless traffic queues, which uses priority-based flow control (PFC) for congestion notification, and ECN should not be enabled on strict-high priority traffic queues.

To enable ECN on an output queue, you not only need to enable ECN in the queue scheduler, you also need to:

- Configure a WRED packet drop profile.
- Configure a queue scheduler that includes the WRED drop profile and enables ECN. (This example shows only ECN and drop profile configuration; you can also configure bandwidth, priority, and buffer settings in a scheduler.)
- Map the queue scheduler to a forwarding class (output queue) in a scheduler map.
- Starting in Junos OS 15.1, enhanced transmission selection (ETS) hierarchical scheduling is supported. If you are using enhanced transmission selection (ETS) hierarchical scheduling, add the forwarding class to a forwarding class set (priority group).
- If you are using ETS, associate the queue scheduler map with a traffic control profile (priority group scheduler for hierarchical scheduling).
- If you are using ETS, apply the traffic control profile and the forwarding class set to an interface. On that interface, the output queue uses the scheduler mapped to the forwarding class, as specified by the scheduler map attached to the traffic control profile. This enables ECN on the queue and applies the WRED drop profile to the queue.

  If you are using port scheduling, apply the scheduler map to an interface. On that interface, the output queue uses the scheduler mapped to the forwarding class in the scheduler map, which enables ECN on the queue and applies the WRED drop profile to the queue.

Table 65 on page 286 shows the configuration components for this example.

Table 65: Components of the ECN Configuration Example

Component	Settings
Hardware	QFX Series switch
Drop profile (with two fill level/drop probability pairs)	Name: **be-dp** Drop start fill level: **30** percent Drop end fill level: **75** percent Drop probability at drop start (minimum drop rate): **0** percent Drop probability at drop end (maximum drop rate): **80** percent
Scheduler	Name: **be-sched** ECN: enabled Drop profile: **be-dp** Transmit rate: **25%** Buffer size: **25%** Priority: **low**

Table 65: Components of the ECN Configuration Example *(continued)*

Component	Settings
Scheduler map	Name: **be-map** Forwarding class: **best-effort** Scheduler: **be-sched** **NOTE:** By default, the **best-effort** forwarding class is mapped to output queue **0**.
Forwarding class set (ETS only)	Name: **be-pg** Forwarding class: **best-effort** (queue 0)
Traffic control profile (ETS only)	Name: **be-tcp** Scheduler map: **be-map**
Interface (ETS only)	Name: **xe-0/0/20** Forwarding class set: **be-pg** (Output) traffic control profile: **be-tcp**
Interface (port scheduling only)	Name: **xe-0/0/20**

NOTE: Only switches that support ETS hierarchical scheduling support forwarding class set and traffic control profile configuration. Direct port scheduling does not use the hierarchical scheduling structure.

NOTE: On QFX5100, EX4600, QFX3500, and QFX3600 switches, and on QFabric systems, the WRED drop profile also controls packet drop behavior for traffic that is not ECN-capable (packets marked non-ECT, ECN bit code 00).

On QFX10000 switches, when ECN is enabled on a queue, the WRED drop profile only sets the ECN thresholds, it does not control packet drop on non-ECN packets. On ECN-enabled queues, QFX10000 switches use the tail-drop algorithm on non-ECN packets during periods of congestion. If you do not enable ECN, then the queue uses the WRED packet drop mechanism.

Configuration

CLI Quick Configuration

To quickly configure the drop profile, scheduler with ECN enabled, and to map the scheduler to an output queue on an interface, 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.

ETS Quick Configuration

```
[edit class-of-service]
set drop-profile be-dp interpolate fill-level 30 fill-level 75 drop-probability 0 drop-probability 80
set schedulers be-sched explicit-congestion-notification
set schedulers be-sched drop-profile-map loss-priority low protocol any drop-profile be-dp
set schedulers be-sched transmit-rate percent 25
set schedulers be-sched buffer-size percent 25
set schedulers be-sched priority low
set scheduler-maps be-map forwarding-class best-effort scheduler be-sched
set forwarding-class-sets be-pg class best-effort
set traffic-control-profiles be-tcp scheduler-map be-map
set interfaces xe-0/0/20 forwarding-class-set be-pg output-traffic-control-profile be-tcp
```

Port Scheduling Quick Configuration (QFX10000 Switches)

```
[edit class-of-service]
set drop-profile be-dp interpolate fill-level 30 fill-level 75 drop-probability 0 drop-probability 80
set schedulers be-sched explicit-congestion-notification
set schedulers be-sched drop-profile-map loss-priority low protocol any drop-profile be-dp
set schedulers be-sched transmit-rate percent 25
set schedulers be-sched buffer-size percent 25
set schedulers be-sched priority low
set scheduler-maps be-map forwarding-class best-effort scheduler be-sched
set interfaces xe-0/0/20 scheduler-map be-map
```

Configuring ECN

Step-by-Step Procedure

To configure ECN:

1. Configure the WRED packet drop profile **be-dp**. This example uses a drop start point of **30** percent, a drop end point of **75** percent, a minimum drop rate of **0** percent, and a maximum drop rate of **80** percent:

   `[edit class-of-service]`

   `user@switch#` **set drop-profile be-dp interpolate fill-level 30 fill-level 75 drop-probability 0 drop-probability 80**

2. Create the scheduler **be-sched** with ECN enabled and associate the drop profile **be-dp** with the scheduler:

   `[edit class-of-service]`

   `user@switch#` **set schedulers be-sched explicit-congestion-notification**

   `user@switch#` **set schedulers be-sched drop-profile-map loss-priority low protocol any drop-profile be-dp**

   `user@switch#` **set be-sched transmit-rate percent 25**

   `user@switch#` **set be-sched buffer-size percent 25**

   `user@switch#` **set be-sched priority low**

3. Map the scheduler **be-sched** to the **best-effort** forwarding class (output queue 0) using scheduler map **be-map**:

   `[edit class-of-service]`

   `user@switch#` **set scheduler-maps be-map forwarding-class best-effort scheduler be-sched**

4. If you are using ETS, add the forwarding class **best-effort** to the forwarding class set **be-pg**; if you are using direct port scheduling, skip this step:

   `[edit class-of-service]`

   `user@switch#` **set forwarding-class-sets be-pg class best-effort**

5. If you are using ETS, associate the scheduler map **be-map**with the traffic control profile **be-tcp**; if you are using direct port scheduling, skip this step:

   `[edit class-of-service]`

   `user@switch#` **set traffic-control-profiles be-tcp scheduler-map be-map**

6. If you are using ETS, associate the traffic control profile **be-tcp** and the forwarding class set **be-pg** with the interface on which you want to enable ECN on the best-effort queue:

   ```
   [edit class-of-service]
   user@switch# set interfaces xe-0/0/20 forwarding-class-set be-pg output-traffic-control-profile be-tcp
   ```

   If you are using direct port scheduling, associate the scheduler map **be-map** with the interface on which you want to enable ECN on the best-effort queue:

   ```
   [edit class-of-service]
   user@switch# set interfaces xe-0/0/20 scheduler-map be-map
   ```

Verification

Verifying That ECN Is Enabled

Purpose

Verify that ECN is enabled in the scheduler **be-sched** by showing the configuration for the scheduler map **be-map**.

Action

Display the scheduler map configuration using the operational mode command **show class-of-service scheduler-map be-map**:

```
user@switch> show class-of-service scheduler-map be-map
```

```
Scheduler map: be-map, Index: 12240

  Scheduler:be-sched, Forwarding class: best-effort, Index: 115
    Transmit rate: 25 percent, Rate Limit: none, Buffer size: 25 percent,
    Buffer Limit: none, Priority: low
    Excess Priority: unspecified, Explicit Congestion Notification: enable
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any          3312    be-dp
      Medium-high     any          1       <default-drop-profile>
      High            any          1       <default-drop-profile>
```

Meaning

The **show class-of-service scheduler-map** operational command shows the configuration of the scheduler associated with the scheduler map and the forwarding class mapped to that scheduler. The output shows that:

- The scheduler associated with the scheduler map is **be-sched**.
- The scheduler map applies to the forwarding class **best-effort** (output queue 0).
- The scheduler **be-sched** has a transmit rate of **25** percent, a queue buffer size of **25** percent, and a drop priority of **low**.
- Explicit congestion notification state is **enable**.
- The WRED drop profile used for low drop priority traffic is **be-dp**.

Release History Table

Release	Description
15.1	Starting in Junos OS 15.1, enhanced transmission selection (ETS) hierarchical scheduling is supported.

RELATED DOCUMENTATION

Understanding CoS Explicit Congestion Notification | 274

Data Center Quantized Congestion Notification (DCQCN)

IN THIS SECTION

Remote Direct Memory Access (RDMA) provides the high throughput and ultra-low latency, with low CPU overhead, necessary for modern datacenter applications. RDMA is deployed using the RoCEv2 protocol, which relies on Priority-based Flow Control (PFC) to enable a drop-free network. Data Center Quantized Congestion Notification (DCQCN) is an end-to-end congestion control scheme for RoCEv2. Starting in Junos OS Release 18.1R1, Junos OS supports DCQCN by combining Explicit Congestion Notification (ECN) and PFC to overcome the limitations of PFC to support end-to-end lossless Ethernet.

Understanding Data Center Quantized Congestion Notification (DCQCN)

Priority-based Flow Control (PFC) is a lossless transport and congestion relief feature that works by providing granular link-level flow control for each IEEE 802.1p code point (priority) on a full-duplex Ethernet link. When the receive buffer on a switch interface fills to a threshold, the switch transmits a pause frame to the sender (the connected peer) to temporarily stop the sender from transmitting more frames. The buffer threshold must be low enough so that the sender has time to stop transmitting frames and the receiver can accept the frames already on the wire before the buffer overflows. The switch automatically sets queue buffer thresholds to prevent frame loss.

When congestion forces one priority on a link to pause, all of the other priorities on the link continue to send frames. Only frames of the paused priority are not transmitted. When the receive buffer empties below another threshold, the switch sends a message that starts the flow again. However, depending on the amount of traffic on a link or assigned to a priority, pausing traffic can cause ingress port congestion and spread congestion through the network.

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 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.

Data Center Quantized Congestion Notification (DCQCN) is a combination of ECN and PFC to support end-to-end lossless Ethernet. ECN helps overcome the limitations of PFC to achieve lossless Ethernet. The idea behind DCQCN is to allow ECN to do flow control by decreasing the transmission rate when congestion starts, thereby minimizing the time PFC is triggered, which stops the flow altogether.

The correct operation of DCQCN requires balancing two conflicting requirements:

1. Ensuring PFC is not triggered too early, that is, before giving ECN a chance to send congestion feedback to slow the flow.
2. Ensuring PFC is not triggered too late, thereby causing packet loss due to buffer overflow.

There are three important parameters that need to be calculated and configured properly to achieve the above key requirements:

1. **Headroom Buffers**—A PAUSE message sent to an upstream device takes some time to arrive and take effect. To avoid packet drops, the PAUSE sender must reserve enough buffer to process any packets it may receive during this time. This includes packets that were in flight when the PAUSE was sent, and the packets sent by the upstream device while it is processing the PAUSE message. In QFX5000 Series switches, headroom buffers are allocated on a per port per priority basis. Headroom buffers are carved out of the global shared buffer. You can control the amount of headroom buffers allocated for each

port and priority using the MRU and cable length parameters in the congestion notification profile. If you see minor ingress drops even after PFC is triggered, you can eliminate those drops by increasing the headroom buffers for that port and priority combination.

2. **PFC Threshold**—This is an ingress threshold. This is the maximum size an ingress priority group can grow to before a PAUSE message is sent to the upstream device. Each PFC priority gets its own priority group at each ingress port. PFC thresholds are set per priority group at each ingress port. On QFX Series devices, there are two components in the PFC threshold—the **PG MIN** threshold and the **PG shared** threshold. Once **PG MIN** and **PG shared** thresholds are reached for a priority group, PFC is generated for that corresponding priority. The switch sends a RESUME message when the queue falls below the PFC thresholds.

3. **ECN Threshold**—This is an egress threshold. The ECN threshold is equal to the WRED start-fill-level value. Once an egress queue exceeds this threshold, the switch starts ECN marking for packets on that queue. For DCQCN to be effective, this threshold must be lower than the ingress PFC threshold to ensure PFC is not triggered before the switch has a chance to mark packets with ECN. Setting a very low WRED fill level increases ECN marking probability. For example with default shared buffer setting, a WRED start-fill-level of 10 percent ensures lossless packets are ECN marked. But with a higher fill level, the probability of ECN marking is reduced. For example, with two ingress port with lossless traffic to the same egress port and a WRED start-fill-level of 50 percent, no ECN marking will occur, because ingress PFC thresholds will be met first.

Configuring Data Center Quantized Congestion Notification (DCQCN)

To enable DCQCN, configure both ECN and PFC for a traffic flow. As an example, consider a QFX5000 Series switch between a reaction point (RP) and a notification point (NP), with et-0/0/3 as the ingress port and et-0/0/4 as the egress port.

1. Configure ECN on the egress port for a lossless flow. For example:

```
[edit class-of-service]
user@host# set drop-profiles dp1 interpolate fill-level 10 drop-probability 0 fill-level 80 drop-probability
  100
user@host# set schedulers s1 drop-profile-map loss-priority any protocol any drop-profile dp1
user@host# set schedulers s1 explicit-congestion-notification
user@host# set scheduler-maps sm1 forwarding-class fcoe scheduler s1
user@host# set interfaces et-0/0/4 scheduler-map sm1
```

2. Configure PFC on the ingress port for the same lossless flow. For example:

```
[edit class-of-service]
user@host# set congestion-notification-profile cnp1 input ieee-802.1 code-point 011 pfc
user@host# set interfaces et-0/0/3 congestion-notification-profile cnp1
```

3. Configure the shared buffers. For example:

```
[edit class-of-service]
user@host# set shared-buffer ingress buffer-partition lossless percent 15
user@host# set shared-buffer ingress buffer-partition lossy percent 5
user@host# set shared-buffer ingress buffer-partition lossless-headroom percent 80
user@host# set shared-buffer egress buffer-partition lossless percent 60
user@host# set shared-buffer egress buffer-partition lossy percent 20
user@host# set shared-buffer egress buffer-partition multicast percent 20
```

4. Verify your configuration.

```
[edit class-of-service]
user@host# show
```

For example:

```
[edit class-of-service]
user@host# show
drop-profiles {
    dp1 {
        interpolate {
            fill-level [ 10 80 ];
            drop-probability [ 0 100 ];
        }
    }
}
shared-buffer {
    ingress {
        buffer-partition lossless {
            percent 15;
        }
        buffer-partition lossy {
            percent 5;
        }
        buffer-partition lossless-headroom {
            percent 80;
        }
    }
    egress {
        buffer-partition lossless {
            percent 60;
        }
```

```
        buffer-partition lossy {
            percent 20;
        }
        buffer-partition multicast {
            percent 20;
        }
    }
}
congestion-notification-profile {
    cnp1 {
        input {
            ieee-802.1 {
                code-point 011 {
                    pfc;
                }
            }
        }
    }
}
interfaces {
    et-0/0/3 {
        congestion-notification-profile cnp1;
    }
    et-0/0/4 {
        scheduler-map sm1;
    }
}
scheduler-maps {
    sm1 {
        forwarding-class fcoe scheduler s1;
    }
}
schedulers {
    s1 {
        drop-profile-map loss-priority any protocol any drop-profile dp1;
        explicit-congestion-notification;
    }
}
```

5. Save your configuration.

```
[edit class-of-service]
user@host# commit
```

RELATED DOCUMENTATION

Understanding CoS Flow Control (Ethernet PAUSE and PFC) | 197
Configuring CoS PFC (Congestion Notification Profiles) | 194
Understanding CoS Explicit Congestion Notification | 274
Example: Configuring ECN | 284

3

PART

CoS Queue Schedulers, Traffic Control Profiles, and Hierarchical Port Scheduling (ETS)

CHAPTER 11

Queue Schedulers and Scheduling Priority

IN THIS CHAPTER

Understanding Default CoS Scheduling and Classification

IN THIS SECTION

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 66 on page 300 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 66: 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 67 on page 300 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 67 on page 300 applies only to unicast traffic except on QFX10000 switches and NFX Series devices.

Table 67: Default IEEE 802.1 Classifiers for Ports in Access Mode (Untrusted Classifier)

Code Point	Forwarding Class	Loss Priority
000	best-effort	low

Table 67: Default IEEE 802.1 Classifiers for Ports in Access Mode (Untrusted Classifier) ***(continued)***

Code Point	Forwarding Class	Loss Priority
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 68 on page 301 shows the default mapping of IEEE 802.1 code-point values to multidestination (multicast, broadcast, and destination lookup fail traffic) forwarding classes and loss priorities.

NOTE: Table 68 on page 301 does not apply to QFX10000 switches or NFX Series devices.

Table 68: 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 69 on page 302 shows the default mapping of DSCP code-point values to forwarding classes and loss priorities for DSCP IP and DCSP IPv6.

NOTE: Table 69 on page 302 applies only to unicast traffic except on QFX10000 switches and NFX Series devices.

Table 69: 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
cs2 (010000)	best-effort	low
cs3 (011000)	best-effort	low

Table 69: Default DSCP IP and IPv6 Classifiers *(continued)*

Code Point	Forwarding Class	Loss Priority
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 multidestination classifiers for multidestination traffic. DSCP IPv6 multidestination classifiers are not supported for multidestination traffic.

Table 70 on page 303 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 70: 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 71 on page 304:

Table 71: 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 71 on page 304 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 multidestination (multicast, broadcast, and destination lookup fail) traffic use the same forwarding classes and output queues.

On switches other than QFX10000 and NFX Series devices, multidestination queue 11 receives enough bandwidth from the default multidestination scheduler to handle CPU-generated multidestination 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 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, multidestination 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 multidestination traffic receives 2-Gbps of bandwidth.

> **NOTE:** Except on QFX5200, QFX5210, and QFX10000 switches and NFX Series devices, which do not support queue 11, multidestination queue 11 also receives a small amount of default bandwidth from the multidestination scheduler. CPU-generated multidestination 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 multidestination 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 | **23**
Understanding CoS Hierarchical Port Scheduling (ETS) | **407**
Understanding Default CoS Settings | **25**
Understanding CoS Virtual Output Queues (VOQs) on QFX10000 Switches | **377**
Understanding Applying CoS Classifiers and Rewrite Rules to Interfaces | **116**
Understanding DCB Features and Requirements | **450**
Understanding Default CoS Scheduling on QFabric System Interconnect Devices (Junos OS Release 13.1 and Later Releases)
Example: Configuring Unicast Classifiers | **100**
Example: Configuring Queue Schedulers | **325**

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 direct port scheduling, and some switches support both methods of 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, QFX3500, and QFX3600 switches and on QFabric systems, 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, QFX3500, and QFX3600 switches, and on QFabric systems.

    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 multidestination (multicast, broadcast, and destination lookup fail) traffic, a strict-high priority queue cannot belong to a multidestination 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, QFX3500, and QFX3600 switches, and on QFabric systems, 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.

- On QFabric systems, if any queue that contains outgoing packets does not transmit packets for 12 consecutive seconds, the port automatically resets. Failure of a queue to transmit packets for 12 consecutive seconds might be due to:
  - A strict-high priority queue consuming all of the port bandwidth
  - Several queues consuming all of the port bandwidth
  - Any queue or port receiving continuous priority-based flow control (PFC) or 802.3x Ethernet PAUSE messages (received PFC and PAUSE messages prevent a queue or a port, respectively, from transmitting packets because of network congestion)
  - Other conditions that prevent a queue from obtaining port bandwidth for 12 consecutive seconds

  If the cause is a strict-high priority queue consuming all of the port bandwidth, use rate shaping to configure a maximum rate for the strict-high priority queue and prevent it from using all of the port bandwidth. To configure rate shaping, include the **shaping-rate (*rate* | percent *percentage*)** statement at the **[edit class-of-service schedulers *scheduler-name*]** hierarchy level and apply the shaping rate to the strict-high priority scheduler. We recommend that you always apply a shaping rate to strict-high priority traffic to prevent the strict-high priority queue from starving other queues.

  If several queues consume all of the port bandwidth, you can use a scheduler to rate shape those queues and prevent them from using all of the port bandwidth.

- 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 multidestination 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 multidestination 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 multidestination traffic into best-effort forwarding classes.

Understanding CoS Output Queue Schedulers

IN THIS SECTION

Output queue scheduling defines the class-of-service (CoS) properties of output queues. Output queues are mapped to forwarding classes, and classifiers map incoming traffic into forwarding classes based on IEEE 802.1p or DSCP code points. Output queue properties include the amount of interface bandwidth assigned to the queue, the size of the memory buffer allocated for storing packets, the priority of the queue, and the weighted random early detection (WRED) drop profiles associated with the queue. Queue scheduling works with priority group scheduling to create a two-tier hierarchical scheduler.

The hierarchical scheduler allocates port bandwidth to a group of queues (forwarding classes) called a priority group (forwarding class set), and queue scheduling determines the portion of the priority group's bandwidth that a particular queue can use. So the first scheduling tier is allocating port bandwidth to a forwarding class set, and the second scheduling tier is allocating forwarding class set bandwidth to forwarding classes (queues).

Scheduler maps associate queue schedulers with forwarding classes. The queue mapped to a forwarding class receives the scheduling resources assigned to that forwarding class. You associate a scheduler map with a traffic control profile, and then associate the traffic control profile with a forwarding class set (priority group) and a port interface to apply scheduling to a port. In conjunction with the priority group scheduling configured in the traffic control profile, queue scheduling configures the packet schedulers and weighted random early detection (WRED) packet drop processes for 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.

Output Queue Scheduling Components

Table 72 on page 315 provides a quick reference to the scheduler components you can configure to determine the bandwidth properties of output queues (forwarding classes), and Table 73 on page 316 provides a quick reference to some related scheduling configuration components.

Table 72: Output Queue Scheduler Components

Output Queue Scheduler Component	Description
Buffer size	Sets the size of the queue buffer. See "Understanding CoS Buffer Configuration" on page 635.
Drop profile map	Maps a drop profile to a loss priority. Drop profile map components include: • Drop profile—Sets the probability of dropping packets as the queue fills up. • Loss priority—Sets the traffic loss priority to which a drop profile applies. See "Configuring CoS Drop Profile Maps" on page 270.
Explicit congestion notification	Enables explicit congestion notification (ECN) on the queue. See "Understanding CoS Explicit Congestion Notification" on page 274.
Priority	Sets the scheduling priority applied to the queue. See "Defining CoS Queue Scheduling Priority" on page 334.
Shaping rate	Sets the maximum bandwidth the queue can consume. **TIP:** On QFX5200 Series switches, a granularity of 64kbps is supported for the shaping rate. See "Understanding CoS Priority Group Shaping and Queue Shaping (Maximum Bandwidth)" on page 398.
Transmit rate	Sets the minimum guaranteed bandwidth for the queue. Extra bandwidth is shared among queues in proportion to the minimum guaranteed bandwidth of each queue. See "Understanding CoS Priority Group and Queue Guaranteed Minimum Bandwidth" on page 389.

Table 73: Other Scheduling Components

Other Scheduling Components	Description
Forwarding class	Maps traffic to an output queue. Classifiers map forwarding classes to IEEE 802.1p, DSCP, or EXP code points. A forwarding class, an output queue, and code point bits are mapped to each other and identify the same traffic. (The code point bits identify incoming traffic. Classifiers assign traffic to forwarding classes based on the code point bits. Forwarding classes are mapped to output queues. This mapping determines the output queue each class of traffic uses on the switch egress interfaces.)
Output queue	Buffers traffic before the switch forwards the traffic out the egress interface. Output queues are mapped to forwarding classes. The switch applies CoS properties defined in schedulers to output queues, by mapping forwarding classes to schedulers in scheduler maps. The queue mapped to the forwarding class has the CoS properties defined in the scheduler mapped to that forwarding class.
Scheduler map	Maps schedulers to forwarding classes (forwarding classes are mapped to queues, so a forwarding class represents a queue, and the scheduler mapped to a forwarding class determines the CoS properties of the output queue mapped to that forwarding class).
Traffic control profile	Configures scheduling for the forwarding class set (priority group), and associates a scheduler map with the forwarding class set to apply queue scheduling to the forwarding classes in the forwarding class set. Extra port bandwidth is shared among forwarding class sets in proportion to the minimum guaranteed bandwidth of each forwarding class set.
Forwarding class set	Name of a priority group. You map forwarding classes to forwarding class sets. A forwarding class set consists of one or more forwarding classes.

Default Schedulers

Each forwarding class requires a scheduler to set the CoS properties of the forwarding class and its output queue. You can use the default schedulers or you can define new schedulers for the associated forwarding classes. For any other forwarding class, you must explicitly configure a scheduler. For more information, see “Default Scheduling” on page 303.

Transmit Rate (Minimum Guaranteed Bandwidth)

The transmit rate determines the minimum guaranteed bandwidth for each forwarding class. The switch applies the minimum bandwidth guarantee to the output queue mapped to the forwarding class. The transmit rate also determines how much excess (extra) bandwidth each low-priority queue can share; each queue shares extra bandwidth in proportion to its transmit rate. You specify the rate in bits per second as a fixed value such as 1 Mbps or as a percentage of the total forwarding class set minimum guaranteed bandwidth (the guaranteed rate set in the traffic control profile). Either the default scheduler or a scheduler you configure allocates a portion of the outgoing interface bandwidth to each forwarding class in proportion to the transmit rate.

> **NOTE:** 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 may round small fixed rates to a lower percentage. For example, a fixed rate of 350 Mbps is rounded down to 3 percent.

You cannot configure a transmit rate for a strict-high priority queue. Queues with a configured transmit rate cannot be included in a forwarding class set that has a strict-high priority queue (you cannot mix strict-high priority queues and queues that are not strict-high priority in the same forwarding class set).

The allocated bandwidth can exceed the configured minimum rate if additional bandwidth is available from other queues in the forwarding class set that are not using all of their allocated bandwidth. During periods of congestion, the configured transmit rate is the guaranteed bandwidth minimum for the queue. This behavior enables you to ensure that each queue receives the amount of bandwidth appropriate to its level of service and is also able to share unused bandwidth.

> **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 combined minimum bandwidth for the queues that is greater than the minimum bandwidth guaranteed for the entire set of queues.)

For more information, see “Understanding CoS Priority Group and Queue Guaranteed Minimum Bandwidth” on page 389.

Sharing Extra Bandwidth

Extra bandwidth is available to low-priority queues when a forwarding class set does not use its full amount of minimum guaranteed bandwidth (guaranteed-rate). Extra bandwidth is shared among the forwarding classes in a forwarding class set in proportion to the minimum guaranteed bandwidth (transmit-rate) of each queue.

For example, in a forwarding class set, Queue A has a transmit rate of 1 Gbps, Queue B has a transmit rate of 1 Gbps, and Queue C has a transmit rate of 2 Gbps. After servicing the minimum guaranteed bandwidth of these queues, the forwarding class set has an extra 2 Gbps of bandwidth available, and all three queues still have packets to forward. The queues receive the extra bandwidth in proportion to their transmit rates, so Queue A receives an extra 500 Mbps, Queue B receives an extra 500 Mbps, and Queue C receives an extra 1 Gbps.

Shaping Rate (Maximum Bandwidth)

The shaping rate sets the maximum bandwidth that a forwarding class can consume. You specify the rate in bits per second as a fixed value, such as 3 Mbps or as a percentage of the total forwarding class set maximum bandwidth (the shaping rate set in the traffic control profile).

The maximum bandwidth for a queue depends on the total bandwidth available to the forwarding class set to which the queue belongs, and on how much bandwidth the other queues in the forwarding class set consume.

> **NOTE:** On QFabric systems, if any queue that contains outgoing packets does not transmit packets for 12 consecutive seconds, the port automatically resets. A strict-high priority queue (or several queues with higher priorities than the starved queue) can consume all of the port bandwidth and prevent another queue from transmitting packets. To prevent a queue from being starved for bandwidth, you can configure a shaping rate on the queue or queues to prevent them from consuming all of the port bandwidth.

> **NOTE:** We recommend that you always configure a shaping rate in the scheduler for strict-high priority queues to prevent them from starving other queues.

For more information, see "Understanding CoS Priority Group Shaping and Queue Shaping (Maximum Bandwidth)" on page 398.

Scheduling Priority

Scheduling priority determines the order in which an interface transmits traffic from its output queues. This ensures that queues containing important traffic receive prioritized access to the outgoing interface bandwidth. The priority setting in the scheduler determines the priority for the queue.

For more information, see “Defining CoS Queue Scheduling Priority” on page 334.

Scheduler Drop-Profile Maps

Drop-profile maps associate drop profiles with queue schedulers and packet loss priorities (PLPs). Drop profiles set thresholds for dropping packets during periods of congestion, based on the queue fill level and a percentage probability of dropping packets at the specified queue fill level. At different fill levels, a drop profile sets different probabilities of dropping a packet during periods of congestion.

Classifiers assign incoming traffic to forwarding classes (which are mapped to output queues), and also assign a PLP to the incoming traffic. The PLP can be low, medium-high, or high. You can classify traffic with different PLPs into the same forwarding class to differentiate treatment of traffic within the forwarding class.

In a drop profile map, you can configure a different drop profile for each PLP and associate (map) the drop profiles to a queue scheduler. A scheduler map maps the queue scheduler to a forwarding class (output queue). Traffic classified into the forwarding class uses the drop characteristics defined in the drop profiles that the drop profile map associates with the queue scheduler. The drop profile the traffic uses depends on the PLP that the classifier assigns to the traffic. (You can map different drop profiles to the forwarding class for different PLPs.)

In summary:

- Classifiers assign one of three PLPs (low, medium-high, high) to incoming traffic when classifiers assign traffic to a forwarding class.
- Drop profiles set thresholds for packet drop at different queue fill levels.
- Drop profile maps associate a drop profile with each PLP, and map the drop profiles to schedulers.
- Scheduler maps map schedulers to forwarding classes, and forwarding classes are mapped to output queues. The scheduler mapped to a forwarding class determines the CoS characteristics of the output queue mapped to the forwarding class, including the drop profile mapping.

Buffer Size

Most of the total system buffer space is divided into two buffer pools, shared buffers and dedicated buffers. Shared buffers are a global pool that the ports share dynamically as needed. Dedicated buffers are a reserved portion of the buffer pool that is distributed evenly to all of the ports. Each port receives an equal allocation of dedicated buffer space. The dedicated buffer allocation to ports is not configurable because it is reserved for the ports.

The queue buffers are allocated from the dedicated buffer pool assigned to the port. By default, ports divide their allocation of dedicated buffers among the egress queues in the same proportion as the default scheduler sets the minimum guaranteed transmission rates (**transmit-rate**) for traffic. Only the queues included in the default scheduler receive dedicated buffers.

If you do not use the default configuration, you can explicitly configure the queue buffer size in either of two ways:

- As a percentage—The queue receives the specified percentage of dedicated port buffers when the queue is mapped to the scheduler and the scheduler is mapped to a port.
- As a remainder—After the port services the queues that have an explicit percentage buffer size configuration, the remaining port dedicated buffer space is divided equally among the other queues to which a scheduler is attached. (No default or explicit scheduler means no dedicated buffer allocation for the queue.) If you configure a scheduler and you do not specify a buffer size as a percentage, *remainder* is the default setting.

> **NOTE:** The total of all of the explicitly configured buffer size percentages for all of the queues on a port cannot exceed 100 percent.

For a complete discussion about queue buffer configuration in the context of ingress and egress port buffer configuration, see "Understanding CoS Buffer Configuration" on page 635.

Explicit Congestion Notification

Explicit congestion notification (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. ECN enables end-to-end congestion notification between two endpoints on TCP/IP based networks. ECN is disabled by default.

For more information, see "Understanding CoS Explicit Congestion Notification" on page 274.

Scheduler Maps

A scheduler map associates a forwarding class with a scheduler configuration. After configuring a scheduler, you must include it in a scheduler map, associate the scheduler map with a traffic control profile, and then associate the traffic control profile with an interface and a forwarding class set to implement the configured queue scheduling.

You can associate up to four user-defined scheduler maps with traffic control profiles. For more information, see *Default Schedulers Overview*.

RELATED DOCUMENTATION

Defining CoS Queue Schedulers

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, DSCP, or MPLS EXP 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.

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

- **transmit-rate**—Minimum bandwidth, also known as the *committed information rate (CIR)*, set as a percentage rate or as an absolute value in bits per second. 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.

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

  > **NOTE:** You cannot configure a transmit rate for strict-high priority queues. Queues (forwarding classes) with a configured transmit rate cannot be included in a forwarding class set that has strict-high priority queues.

- **shaping-rate**—Maximum bandwidth, also known as the *peak information rate (PIR)*, set as a percentage rate or as an absolute value in bits per second.

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

- **priority**—One of two bandwidth priorities that queues associated with a scheduler can receive:
  - **low**—The scheduler has low priority.
  - **strict-high**—The scheduler has strict-high priority. You can configure only one queue as a strict-high priority queue. Strict-high priority allocates the scheduled 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.

    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.

- **drop-profile-map**—Drop profile mapping to a loss priority and protocol, 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**—Enables ECN 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: 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: 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.

OCX Series switches do not support lossless transport or PFC. On OCX Series switches, do not map traffic to the default lossless fcoe and no-loss forwarding classes.

To apply scheduling properties to traffic, map schedulers to forwarding classes using a scheduler map, and then associate the scheduler map with interfaces. (You associate a scheduler map with an interface using a traffic control profile; see "Example: Configuring CoS Hierarchical Port Scheduling (ETS)" on page 415 for an example of the complete hierarchical scheduling process.) Using different scheduler maps, you can map different schedulers to the same traffic (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:

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

2. Set the maximum bandwidth for the queue:

   `[edit class-of-service schedulers scheduler-name]` user@switch# **set** **shaping-rate** **(*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)** **protocol** ***protocol*** **drop-profile** ***drop-profile-name***

5. Configure the size of the port dedicated 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 using hierarchical scheduling. See "Example: Configuring CoS Hierarchical Port Scheduling (ETS)" on page 415 for a detailed example of hierarchical scheduling.

   [edit class-of-service]
   user@switch# **set traffic-control-profiles *tcp-name* scheduler-map *scheduler-map-name***
   user@switch# **set interfaces *interface-name* forwarding-class-set *fc-set-name* output-traffic-control-profile *tcp-name***

RELATED DOCUMENTATION

Example: Configuring Queue Schedulers

IN THIS SECTION

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.

Requirements

This example uses the following hardware and software components:

- One switch (this example was tested on a Juniper Networks QFX3500 Switch)
- Junos OS Release 11.1 or later for the QFX Series or Junos OS Release 14.1X53-D20 or later for the OCX Series

Overview

Scheduler parameters define the following characteristics for the queues mapped to the scheduler:

- **transmit-rate**—Minimum bandwidth, also known as the *committed information rate (CIR)*. Each queue mapped to the scheduler receives a minimum of either the configured amount of absolute bandwidth or the configured percentage of bandwidth. 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. You cannot configure a transmit rate for strict-high priority queues. Queues (forwarding classes) with a configured transmit rate cannot be included in a forwarding class set that has strict-high priority queues.

  NOTE: The **transmit-rate** setting works only if you also configure the **guaranteed-rate** in the traffic control profile that is attached to the forwarding class set to which the queue belongs. If you do not configure the **guaranteed-rate**, the **transmit-rate** does not work. The sum of all queue transmit rates in a forwarding class set should not exceed the traffic control profile guaranteed rate. If you configure transmit rates whose sum exceeds the forwarding class set guaranteed rate, the commit check fails, and the system rejects the configuration.

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

  NOTE: You cannot configure a transmit rate for strict-high priority queues. Queues (forwarding classes) with a configured transmit rate cannot be included in a forwarding class set that has strict-high priority queues.

- **shaping-rate**—Maximum bandwidth, also known as the *peak information rate (PIR)*. Each queue 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.

- **priority**—One of two bandwidth priorities that queues associated with a scheduler can receive:
  - **low**—The scheduler has low priority.
  - **strict-high**—The scheduler has strict-high priority. You can configure only one queue as a strict-high priority queue. Strict-high priority allocates the scheduled 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.

    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.
- **drop-profile-map**—Mapping of a drop profile to a loss priority and protocol to apply WRED to the scheduler.
- **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**—Enables ECN 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: 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: 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.

OCX Series switches do not support lossless transport or PFC. On OCX Series switches, do not map traffic to the default lossless fcoe and no-loss forwarding classes.

Scheduler maps associate schedulers with forwarding classes (queues). After defining schedulers and mapping them to queues in a scheduler map, to configure hardware queue scheduling (hierarchical port scheduling) you:

1. Associate a scheduler map with a traffic control profile (a traffic control profile schedules resources for a group of forwarding classes, called a *forwarding class set* or *priority group*).

2. Attach a forwarding class and a traffic control profile to an interface.

   “Example: Configuring CoS Hierarchical Port Scheduling (ETS)” on page 415 provides a complete example of hierarchical scheduling.

You can associate up to four user-defined scheduler maps with forwarding class sets.

This process configures the bandwidth properties and WRED characteristics that you map to forwarding classes (and thus to output queues) in a scheduler map. The traffic control profile uses the scheduler CoS properties to determine the resources that should be allocated to the individual output queues from the total resources available to the priority group.

Table 74 on page 328 shows the configuration components for this example.

Table 74: Components of the Queue Scheduler Configuration Example

Component	Settings
Hardware	QFX3500 switch
Scheduler	Name: **be-sched** Transmit rate: **20%** Shaping rate: **40%** Buffer size: **20%** Priority: **low** Drop profile: **be-dp** ECN: **disable** (default)
Scheduler map	Name: **be-map** Forwarding class to associate with the **be-sched** scheduler: **best-effort**

Table 74: Components of the Queue Scheduler Configuration Example *(continued)*

Component	Settings
Traffic control profile	Name: **be-tcp** **NOTE:** This topic does not describe how to define a traffic control profile.
Forwarding class set	Name: **lan-pg**

Configuring a CoS Scheduler

CLI Quick Configuration

To quickly configure a queue scheduler, 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 percent 20

set schedulers be-sched shaping-rate percent 40

set schedulers be-sched buffer-size percent 20

set schedulers be-sched priority low

set schedulers be-sched drop-profile-map loss-priority low protocol any drop-profile be-dp

set scheduler-maps be-map forwarding-class best-effort scheduler be-sched

set traffic-control-profiles be-tcp scheduler-map be-map

set interfaces xe-0/0/7 forwarding-class-set lan-pg output-traffic-control-profile be-tcp

To configure a CoS scheduler:

1. Create scheduler (**be-sched**) with a minimum guaranteed bandwidth of 2 Gbps, a maximum bandwidth of 4 Gbps, and low priority, and map it to the drop profile **be-dp**:

   ```
   [edit class-of-service schedulers]
   user@switch# set be-sched transmit-rate percent 20
   user@switch# set be-sched shaping-rate percent 40
   user@switch# set be-sched buffer-size percent 20
   user@switch# set be-sched priority low
   user@switch# set be-sched drop-profile-map loss-priority low protocol any drop-profile be-dp
   ```

   > **NOTE:** Because ECN is disabled by default, no ECN configuration is shown.

2. Configure scheduler map (**be-map**) to associate the scheduler (**be-sched**) with the forwarding class (**best-effort**):

   ```
   [edit class-of-service scheduler-maps]
   user@switch# set be-map forwarding-class best-effort scheduler be-sched
   ```

3. Associate the scheduler map **be-map** with a traffic control profile (**be-tcp**):

   ```
   [edit class-of-service traffic-control-profiles]
   user@switch# set be-tcp scheduler-map be-map
   ```

4. Associate the traffic control profile **be-tcp** with a forwarding class set (**lan-pg**) and a 10-Gigabit Ethernet interface (**xe-0/0/7**):

   ```
   [edit class-of-service]
   user@switch# set interfaces xe-0/0/7 forwarding-class-set lan-pg output-traffic-control-profile
   be-tcp
   ```

Verification

IN THIS SECTION

To verify that the queue scheduler has been created and is mapped to the correct interfaces, perform these tasks:

Verifying the Scheduler Configuration

Purpose

Verify that the queue scheduler **be-sched** has been created with a minimum guaranteed bandwidth of 2 Gbps, a maximum bandwidth of 4 Gbps, the priority set to **low**, and the drop profile **be-dp**.

Action

Display the scheduler using the operational mode command **show configuration class-of-service schedulers be-sched**:

```
user@switch> show configuration class-of-service schedulers be-sched
```

```
transmit-rate percent 20;
shaping-rate percent 40;
buffer-size percent 20;
priority low;
drop-profile-map loss-priority low protocol any drop-profile be-dp;
```

Verifying the Scheduler Map Configuration

Purpose

Verify that the scheduler map **be-map** has been created and associates the forwarding class **best-effort** with the scheduler **be-sched**, and also that the scheduler map is attached to the traffic control profile **be-tcp**.

Action

Display the scheduler map using the operational mode command **show configuration class-of-service scheduler-maps be-map**:

`user@switch>` **show configuration class-of-service scheduler-maps be-map**

```
forwarding-class best-effort scheduler be-sched;
```

Display the traffic control profile to verify that the scheduler map **be-map** is attached 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;
```

NOTE: This topic does not describe how to configure a traffic control profile or its allocation of port bandwidth. Using a traffic control profile to configure the port resource allocation to the priority group is necessary to implement hierarchical scheduling.

Verifying That the Scheduler Is Associated with the Interface

Purpose

Verify that the forwarding class set (**lan-pg**) and the traffic control profile (**be-tcp**) that are associated with the queue scheduler are attached to the interface **xe-0/0/7**.

Action

List the 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 {
    lan-pg {
        output-traffic-control-profile be-tcp;
    }
}
```

RELATED DOCUMENTATION

Example: Configuring CoS Hierarchical Port Scheduling (ETS) | 415
Example: Configuring Minimum Guaranteed Output Bandwidth | 392
Example: Configuring Maximum Output Bandwidth | 400
Example: Configuring Traffic Control Profiles (Priority Group Scheduling) | 385
Example: Configuring WRED Drop Profiles | 264
Example: Configuring ECN | 284
Defining CoS Queue Schedulers | 321
Monitoring CoS Scheduler Maps | 341
Understanding CoS Output Queue Schedulers | 313
Understanding CoS Hierarchical Port Scheduling (ETS) | 407
Understanding CoS Scheduling on QFabric System Node Device Fabric (fte) Ports
Understanding Default CoS Scheduling on QFabric System Interconnect Devices (Junos OS Release 13.1 and Later Releases)
Understanding CoS Buffer Configuration | 635

Defining CoS Queue Scheduling Priority

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. On QFX5100, QFX5200, EX4600, QFX3500, and QFX3600 switches, and on QFabric systems, queues can have one of two bandwidth scheduling priorities, **strict-high** priority or **low** priority. On QFX10000 Series switches, queues can also be configured as **high** priority.

> **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.

Different switches handle traffic configured as **strict-high** priority traffic in different ways:

- QFX5100, QFX5200, QFX3500, QFX3600, and EX4600 switches, and QFabric systems—You can configure only one queue as a strict-high priority queue.

  On these switches, 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.

- QFX10000 switches—You can configure as many queues as you want as strict-high priority. However, keep in mind that too much strict-high priority traffic can starve low priority queues on the port.

  > **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.

- To configure queue priority using the CLI:

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

RELATED DOCUMENTATION

Example: Configuring Queue Scheduling Priority | 336
Monitoring CoS Scheduler Maps | 341

Example: Configuring Queue Scheduling Priority

IN THIS SECTION

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.

Requirements

This example uses the following hardware and software components:

- One switch.
- Junos OS Release 11.1 or later for the QFX Series or Junos OS Release 14.1X53-D20 or later for the OCX Series.

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. On QFX10000 switches, you can configure as many queues as you want as strict-high priority queues. On QFX5200, QFX3500, and QFX3600 switches and on QFabric systems, you can configure only one queue as a strict-high queue. 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.

  NOTE: 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.

  On QFX10000 switches, if you configure strict-high priority queues on a port, we strongly recommend that you configure a transmit rate on those queues. The transmit rate sets the amount of traffic that the switch forwards as strict-high priority; traffic in excess of the transmit 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 the queue to prevent it from starving other queues. If you do not configure a transmit 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.

  On QFX5200, QFX5100, QFX3500, QFX3600, and EX4600 switches and on QFabric systems, 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.

  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 multidestination traffic, the multidestination forwarding class set cannot contain strict-high priority queues.

- **high** (QFX10000 Series switches only)—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 75 on page 338 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 **nl-sched**. One scheduler map, **schedmap1**, associates the schedulers to the queues.

Table 75: Components of the Queue Scheduler Priority Configuration Example

Component	Settings
Hardware	One switch
Schedulers	**fcoe-sched** for FCoE traffic **nl-sched** for no-loss traffic
Priority	**low** for FCoE traffic **low** for no-loss traffic
Scheduler map	**schedmap1**: FCoE mapping: scheduler **fcoe-sched** to forwarding class **fcoe** No-loss mapping: scheduler **nl-sched** to forwarding class **no-loss**

NOTE: OCX Series switches do not support lossless transport. On OCX Series switches, the default DSCP classifier does not map traffic to the default fcoe and no-loss forwarding classes. On an OCX Series switch, you could use this example by substituting other forwarding classes (for example, best-effort or network-control) for the fcoe and no-loss forwarding classes, and naming the schedulers appropriately. The active forwarding classes (**best-effort**, **network-control**, and **mcast**) share the unused bandwidth assigned to the **fcoe** and **no-loss** forwarding classes.

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
```

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
   ```

Verification

IN THIS SECTION

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;
```

RELATED DOCUMENTATION

Defining CoS Queue Scheduling Priority | 334

Monitoring CoS Scheduler Maps

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 76 on page 341 summarizes key output fields for CoS scheduler maps.

Table 76: 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.
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. On strict-high priority queues on QFX10000 switches, defines the maximum amount of traffic on the queue that is treated as strict-high priority traffic.

Table 76: Summary of Key CoS Scheduler Maps Output Fields *(continued)*

Field	Values
Priority	Scheduling priority of traffic on a queue: • **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). On QFX3500, QFX3600, EX4600, and OCX series switches, and on QFabric system, only one queue can be configured as **strict-high** or **high** priority. On QFX10000 switches, you can configure more than one strict-high priority queue. • **low**—Packets in this queue are transmitted after packets in the **strict-high** queue.
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.

CHAPTER 12

Port Scheduling

IN THIS CHAPTER

Understanding CoS Port Schedulers on QFX Switches

IN THIS SECTION

Port scheduling defines the class-of-service (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.

Output queue properties include the amount of interface bandwidth assigned to the queue, the size of the memory buffer allocated for storing packets, the scheduling priority of the queue, and the weighted random early detection (WRED) drop profiles associated with the queue to control packet drop during periods of congestion.

Scheduler maps map schedulers to forwarding classes. The output queue mapped to a forwarding class receives the port resources and properties defined in the scheduler mapped to that forwarding class. You apply a scheduler map to an interface to apply queue scheduling to a port. You can associate different scheduler maps with different interfaces to configure port-specific scheduling for forwarding classes (output queues).

NOTE: Port scheduling is simpler to configure than enhanced transmission selection (ETS) two-tier hierarchical port scheduling. Port scheduling allocates port bandwidth to output queues directly, instead of allocating port bandwidth to output queues through a scheduling hierarchy. While port scheduling is simpler, ETS is more flexible.

ETS allocates port bandwidth in a two-tier hierarchy:

- Port bandwidth is first allocated to a priority group using the CoS properties defined in a traffic control profile. A priority group is a group of forwarding classes (which are mapped to output queues) that require similar CoS treatment.
- Priority group bandwidth is allocated to the output queues (which are mapped to forwarding classes) using the properties defined in the output queue scheduler.

NOTE: When you configure bandwidth for a queue, 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, consider the preamble and the IFG as well as the data in the calculations.

Queue Scheduling Components

Table 77 on page 345 provides a quick reference to the scheduler components you can configure to determine the bandwidth properties of output queues (forwarding classes).

Table 77: Output Queue Scheduler Components

Output Queue Scheduler Component	Description
Buffer size	Sets the size of the queue buffer.
Drop profile map	Maps a drop profile to a packet loss priority. Drop profile map components include: • Drop profile—Sets the probability of dropping packets as the queue fills up. • Loss priority—Sets the traffic packet loss priority to which a drop profile applies.
Excess rate	Sets the percentage of extra bandwidth (bandwidth that is not used by other queues) a queue can receive. If not set, the switch uses the transmit rate to determine how much extra bandwidth the queue can use. Extra bandwidth is the bandwidth remaining after all guaranteed bandwidth requirements are met.
Explicit congestion notification	Enables explicit congestion notification (ECN) on the queue.
Priority	Sets the scheduling priority applied to the queue.
Transmit rate	Sets the minimum guaranteed bandwidth on low and high priority queues. By default, if you do not configure an excess rate, extra bandwidth is shared among queues in proportion to the transmit rate of each queue. On strict-high priority queues, sets the amount of bandwidth that receives strict-high priority forwarding treatment. 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. If you configure two or more strict-high priority queues on a port, you must configure a transmit rate on those queues. However, we strongly recommend that you always configure a transmit rate on strict-high priority queues to prevent them from starving other queues.

Table 78 on page 346 provides a quick reference to some related scheduling configuration components.

Table 78: Related Scheduling Components

Related Scheduling Components	Description
Forwarding class	Maps traffic classified into the forwarding class at the switch ingress to an output queue. Classifiers map forwarding classes to IEEE 802.1p, DSCP, or EXP code points. A forwarding class, an output queue, and code point bits are mapped to each other and identify the same traffic. (The code point bits identify incoming traffic. Classifiers assign traffic to forwarding classes based on the code point bits. Forwarding classes map to output queues. This mapping determines the output queue each class of traffic uses on the switch egress interfaces.)
Output queue (virtual output queue)	Output queues are virtual, and are comprised of the physical buffers on the ingress pipeline of each Packet Forwarding Engine (PFE) chip to store traffic for every egress port. Every output queue on an egress port has buffer storage space on every ingress pipeline on all of the PFE chips on the switch. The mapping of ingress pipeline storage space to output queues is 1-to-1, so each output queue receives buffer space on each ingress pipeline. See "Understanding CoS Virtual Output Queues (VOQs) on QFX10000 Switches" on page 377 for more information.
Scheduler map	Maps schedulers to forwarding classes (forwarding classes are mapped to queues, so a forwarding class represents a queue, and the scheduler mapped to a forwarding class determines the CoS properties of the output queue mapped to that forwarding class).

Default Schedulers

If you do not configure CoS, the switch uses its default settings. Each forwarding class requires a scheduler to set the CoS properties of the forwarding class and its output queue. The default configuration has four forwarding classes: best-effort (queue 0), fcoe (queue 3), no-loss (queue 4), and network-control (queue 7). Each default forwarding class is mapped to a default scheduler. You can use the default schedulers or you can define new schedulers for these four forwarding classes. For explicitly configured forwarding classes, you must explicitly configure a queue scheduler to allocate CoS resources to the traffic mapped to each forwarding class.

Table 79 on page 347 shows the default queue schedulers.

Table 79: Default Scheduler Configuration

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

NOTE: By default, the minimum guaranteed bandwidth (transmit rate) determines the amount of excess (extra) bandwidth a queue can share. Extra bandwidth is allocated to queues in proportion to the transmit rate of each queue. You can configure bandwidth sharing (excess rate) to override the default setting and configure the excess bandwidth percentage independently of the transmit rate.

By default, only the four default schedulers shown in Table 79 on page 347 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 default bandwidth of a default queue.) If a forwarding class does not transport traffic, the bandwidth allocated to that forwarding class is available to other forwarding classes. Unicast and multidestination (multicast, broadcast, and destination lookup fail) traffic use the same forwarding classes and output queues.

Default scheduling is port scheduling. If you configure scheduling instead of using default scheduling, you can configure port scheduling or enhanced transmission selection (ETS) hierarchical port scheduling.

Default scheduling uses weighted round-robin (WRR) scheduling. Each queue receives a portion (weight) of the total available port bandwidth. The scheduling weight is based on the transmit rate (minimum guaranteed bandwidth) of the default scheduler for that queue. For example, queue 7 receives a default scheduling weight of 15 percent of available port bandwidth, and queue 4 receives a default scheduling weight of 35 percent of 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.

You should explicitly map traffic to non-default (unconfigured) queues and schedule bandwidth resources for those queues if you want to use them to forward traffic. By default, queues 1, 2, 5, and 6 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.

If you map traffic to an unconfigured queue and do not schedule bandwidth for the queue, the queue receives only the amount of bandwidth proportional to its default weight (1). The actual amount of bandwidth an unconfigured queue receives 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. Because of their scheduling weights, configured queues have higher priority for bandwidth than unconfigured queues. If a configured queue needs more bandwidth, then less bandwidth is available for unconfigured queues. However, 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 and map it to the forwarding class that is mapped to the queue, and then apply the scheduler map to the port.

Scheduling Priority

Scheduling priority determines the order in which an interface transmits traffic from its output queues. Priority settings ensure that queues containing important traffic receive prioritized access to the outgoing interface bandwidth. The priority setting in the scheduler determines queue priority (a scheduler map maps the scheduler to a forwarding class, the forwarding class is mapped to an output queue, and the output queue uses the CoS properties defined in the scheduler).

By default, all queues are low priority queues. The switch supports three levels of scheduling priority:

- Low—In the default CoS state, all queues are low priority queues. Low priority queues transmit traffic based on the weighted round-robin (WRR) algorithm. If you configure scheduling priorities higher than low priority on queues, then the higher priority queues are served before the low priority queues.
- High— (QFX10000 Series switches only) High priority queues transmit traffic based on the weighted round-robin (WRR) algorithm, and have higher scheduling priority than low priority queues.
- Strict-high—You can configure queues as **strict-high** priority. Strict-high priority queues receive preferential treatment over all other queues, and receive all of their configured bandwidth before other queues are serviced. Other queues do not transmit traffic until strict-high priority queues are empty, and they receive the bandwidth that remains after the strict-high priority queues are serviced. Because strict-high priority queues are always serviced first, strict-high priority queues can starve other queues on a port. Carefully consider how much bandwidth you want to allocate to strict-high priority queues to avoid starving other queues.

NOTE: For QFX10002, QFX10008, and QFX10016 devices, strict-high priority queues share excess bandwidth based on an excess bandwidth sharing weight of 1, which is not configurable. The actual amount of extra bandwidth that strict-high priority traffic exceeding the transmit rate receives depends on how many other queues consume excess bandwidth and the excess rates of those queues.

For QFX10002-60C, excess traffic on the strict-high queue will starve other high/low priority queues.

When you define scheduling priorities for queues instead of using the default priorities (by default all queues are low priority), the switch uses the priorities to determine the order of packet transmission from the queues. The switch services traffic of different scheduling priorities in a strict order, using round-robin (RR) scheduling to arbitrate queue transmission service among queues of the same priority. The switch transmits packets is the following order:

1. Strict-high priority traffic within the configured queue transmit rate (on strict-high priority queues, the transmit rate limits the amount of traffic treated as strict-high priority traffic). When traffic arrives on a strict-high priority queue, the switch forwards it before servicing other queues.

2. High priority traffic within the configured queue transmit rate (on high priority queues, the transmit rate sets the minimum guaranteed bandwidth)

3. Low priority traffic within the configured queue transmit rate (on low priority queues, the transmit rate sets the minimum guaranteed bandwidth)

4. All traffic that exceeds the queue transmit rate using weighted round-robin (WRR) scheduling. Traffic that exceeds the queue transmit rate contends for excess port bandwidth (bandwidth that is not consumed after the port meets all guaranteed bandwidth requirements). The switch allocates and weights excess bandwidth for low priority queues based on the configured queue excess rate, or on the transmit rate if no excess rate is configured. The switch allocates and weights excess bandwidth for strict-high priority queues based on the hard-coded weight “1”, which is not configurable. The actual amount of extra bandwidth that traffic exceeding the transmit rate gets depends on how many other queues consume excess bandwidth and the weighting of those queues.

NOTE: If you use the default CoS configuration, all queues are low priority queues and transmit traffic based on the weighted round-robin (WRR) algorithm.

Bandwidth Scheduling

IN THIS SECTION

A queue scheduler allocates port bandwidth to a queue (the scheduler is mapped to a forwarding class, and the forwarding class is mapped to a queue). The bandwidth profile, which consists of minimum guaranteed bandwidth, maximum bandwidth (queue shaping), and excess bandwidth sharing properties configured in the scheduler, defines the amount of port bandwidth a queue can consume during normal and congested transmission periods.

The scheduler regularly reevaluates whether each individual queue is within its defined bandwidth profile by comparing the amount of data the queue receives to the amount of bandwidth the scheduler allocates to the queue. When the received amount is less than the guaranteed minimum amount of bandwidth, the queue is considered to be in profile. A queue is out of profile when its received amount is larger than its guaranteed minimum amount. Out of profile queue data is transmitted only if extra (excess) bandwidth is available. Otherwise, it is buffered if buffer space is available. If no buffer space is available, the traffic might be dropped.

The switch provides features that enable you to control the allocation of port bandwidth to queues, so that you can meet the demands of different types of traffic on a port:

Minimum Guaranteed Bandwidth

The transmit rate determines the minimum guaranteed bandwidth for each forwarding class that is mapped to an output queue, and so determines the minimum bandwidth guarantee on that queue.

If you do not want to use the default configuration, you can set the minimum guaranteed bandwidth in several ways, and with several options, using the **[set class-of-service schedulers *scheduler-name* transmit-rate (rate | percent *percentage*) <exact>]**statement:

- Rate—Set the minimum guaranteed bandwidth as a fixed amount (rate) in bits-per-second of port bandwidth (for example, 2 Gbps or 800 Mbps).
- Percent—Set the minimum guaranteed bandwidth as a percentage of port bandwidth (for example, 25 percent).
- Exact—(QFX10000 switches only) Shape the queue to the transmit rate so that the transmit rate is the maximum amount of bandwidth a queue can use. The queue cannot share extra port bandwidth if you

configure the exact option. Configuring a transmit rate as *exact* is how you set a shaping rate to configure the maximum amount of bandwidth low and high priority queues can consume, and the maximum is the transmit rate. You cannot use the **exact** option on a strict-high priority queue.

> **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.

- Extra bandwidth sharing—On low and high priority queues, if you configure an excess rate, the excess rate determines the amount of extra port bandwidth a queue can use. If you do not configure an excess rate, the transmit rate determines how much excess (extra) bandwidth a low and high priority queue can share. If you do not configure an excess rate, then each queue shares extra bandwidth in proportion to its transmit rate.

  You cannot configure an excess rate on strict-high priority queues. Strict-high priority queues share extra bandwidth based on a scheduling weight of “1”, which is not configurable. The actual amount of extra bandwidth that traffic exceeding the transmit rate gets depends on how many other queues consume excess bandwidth and the excess rates of those queues.

  > **NOTE:** The sum of the transmit rates of the queues on a port should not exceed the total bandwidth of that port. (You cannot guarantee a combined minimum bandwidth for the queues on a port that is greater than the total port bandwidth.)

  > **NOTE:** 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.

The bandwidth a low or high priority queue consumes can exceed the configured minimum rate if additional bandwidth is available, and if you do not configure the transmit rate as **exact** on QFX10000 switches. During periods of congestion, the configured transmit rate is the guaranteed minimum bandwidth for the queue. This behavior enables you to ensure that each queue receives the amount of bandwidth appropriate to its required level of service and is also able to share unused bandwidth.

Maximum Bandwidth (Rate Shaping on Low and High Priority Queues and LAGs)

On QFX10000 switches, the optional **exact** keyword in the **[set class-of-service schedulers *scheduler-name* transmit-rate (rate | percent *percentage*) <exact>]** configuration statement shapes the transmission rate of low and high priority queues. When you specify the **exact** option, the switch drops traffic that exceeds the configured transmit rate, even if excess bandwidth is available. Rate shaping prevents a queue from using more bandwidth than is appropriate for the planned service level of the traffic on the queue. You cannot use the **exact** option on a strict-high priority queue.

Configuring rate shaping 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.

LAG interfaces consist 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 a 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 rate shaping to LAG traffic, let's look at a LAG interface named **ae0** that has two member links, **xe-0/0/20** and **xe-0/0/21**. On LAG **ae0**, we configure rate shaping of **2g** by including the **transmit-rate 2g exact** statement in the queue scheduler, and apply the scheduler to 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. The result is that best-effort traffic receives a maximum of 4g on the LAG, not the 2g set by the rate shaping statement. When hashing spreads the traffic assigned to an output queue (which is mapped to a forwarding class) across multiple LAG links, the effective shaping rate (cumulative maximum bandwidth) on the LAG is:

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

Limiting Bandwidth Consumed by Strict-High Priority Queues

You can limit the amount of traffic that receives strict-high priority treatment on a queue by configuring a transmit rate on the strict-high priority queue. The transmit rate sets the amount of traffic that receives strict-high priority treatment. 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 gets depends on how many other queues consume excess bandwidth and the excess rates of those queues. 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:** Configuring a transmit rate on a low or high priority queue sets the guaranteed minimum bandwidth of the queue, as described in "Minimum Guaranteed Bandwidth" on page 350.

CAUTION: If you configure strict-high priority queues, we strongly recommend that you configure a transmit rate on the queues to prevent them from starving low and high priority queues on that port. This is especially important if you configure more than one strict-high priority queue on a port. Although it is not mandatory to configure a transmit rate on strict-high priority queues, if you do not configure a transmit rate, the strict-high priority queues can consume all of the port bandwidth and starve the other queues.

Sharing Extra Bandwidth (Excess Rate on Low and High Priority Queues)

Extra bandwidth is essentially the bandwidth remaining after the switch meets all guaranteed bandwidth requirements. Extra bandwidth is available to low and high priority traffic when the queues on a port do not use all of the available port bandwidth.

By default, extra port bandwidth is shared among the forwarding classes on a port in proportion to the transmit rate of each queue. You can explicitly configure the amount of extra bandwidth a queue can share by setting an **excess-rate** in the scheduler of a low or high priority queue. The configured excess rate overrides the transmit rate and determines the percentage of extra bandwidth the queue can consume.

NOTE: You cannot configure an excess rate on a strict-high priority queue. Strict-high priority queues share excess bandwidth based on an excess bandwidth sharing weight of “1”, which is not configurable. The actual amount of extra bandwidth that strict-high priority traffic exceeding the transmit rate receives depends on how many other queues consume excess bandwidth and the excess rates of those queues.

NOTE: QFX 10002, QFX 10008, and QFX 10016 support multiple strict-high queues.

QFX 10002-60C supports only one strict-high queue.

An example of extra bandwidth allocation based on transmit rates is a port that has traffic running on three forwarding classes, **best-effort**, **fcoe**, and **network-control**. In this example, the **best-effort** forwarding class has a transmit rate of 2 Gbps, forwarding class **fcoe** has a transmit rate of 4 Gbps, and **network-control** has a transmit rate of 2 Gbps, for a total of 8 Gbps of the port bandwidth. After servicing the minimum guaranteed bandwidth of these three queues, the port has 2 Gbps of available extra bandwidth.

If all three queues still have packets to forward, the queues receive the extra bandwidth in proportion to their transmit rates, so the **best-effort** queue receives an extra 500 Mbps, the **fcoe** queue receives an extra 1 Gbps, and the **network-control** queue receives an extra 500 Mbps.

If you configure an excess rate for a queue, the excess rate determines the proportion of extra bandwidth that the queue receives in the same way that the default (transmit rate) determines the proportion of extra bandwidth a queue receives. In the previous example, if you configured an excess rate of 20 percent on the **fcoe** forwarding class, and the transmit rates of the **best-effort** and **network-control** forwarding classes remained 2g (with no configured excess rate, so the 2g transmit rate for each queue still determines the excess rate), then the 2 Gbps of extra bandwidth would be allocated evenly among the three queues because all three queues have the same excess rate.

In the previous example, if you configured an excess rate of 10 percent on the **fcoe** forwarding class, and the transmit rates of the **best-effort** and **network-control** forwarding classes remained 2g (again with no configured excess rate, so the 2g transmit rate for each queue still determines the excess rate), the 2 Gbps of extra bandwidth would be allocated 800 Mbps to the **best-effort** queue, 400 Mbps to the **fcoe** queue, and 800 Mbps to the **network-control** queue (again, in proportion to the queue excess rates).

Scheduler Drop-Profile Maps

Drop-profile maps associate drop profiles with queue schedulers and packet loss priorities (PLPs). Drop profiles set thresholds for dropping packets during periods of congestion, based on the queue fill level and a percentage probability of dropping packets at the specified queue fill level. At different fill levels, a drop profile sets different probabilities of dropping a packet during periods of congestion.

Classifiers assign incoming traffic to forwarding classes (which are mapped to output queues), and also assign a PLP to the incoming traffic. The PLP can be low, medium-high, or high. You can classify traffic with different PLPs into the same forwarding class to differentiate treatment of traffic within the forwarding class.

In a drop profile map, you can configure a different drop profile for each PLP and associate (map) the drop profiles to a queue scheduler. A scheduler map maps the queue scheduler to a forwarding class (output queue). Traffic classified into the forwarding class uses the drop characteristics defined in the drop profiles that the drop profile map associates with the queue scheduler. The drop profile the traffic uses depends on the PLP that the classifier assigns to the traffic. (You can map different drop profiles to the forwarding class for different PLPs.)

In summary:

- Classifiers assign one of three PLPs (low, medium-high, high) to incoming traffic when classifiers assign traffic to a forwarding class.
- Drop profiles set thresholds for packet drop at different queue fill levels.
- Drop profile maps associate a drop profile with each PLP, and then map the drop profiles to schedulers.
- Scheduler maps map schedulers to forwarding classes, and forwarding classes are mapped to output queues. The scheduler mapped to a forwarding class determines the CoS characteristics of the output queue mapped to the forwarding class, including the drop profile mapping.

You associate a scheduler map with an interface to apply the drop profiles and other scheduler elements to traffic in the forwarding class mapped to the scheduler on that interface.

Buffer Size

On QFX10000 switches, the buffer size is the amount of time in milliseconds of port bandwidth that a queue can use to continue to transmit packets during periods of congestion, before the buffer runs out and packets begin to drop.

The switch can use up to 100 ms total (combined) buffer space for all queues on a port. A buffer-size configured as one percent is equal to 1 ms of buffer usage. A buffer-size of 15 percent (the default value for the best effort and network control queues) is equal to 15 ms of buffer usage.

The total buffer size of the switch is 4 GB. A 40-Gigabit port can use up to 500 MB of buffer space, which is equivalent to 100 ms of port bandwidth on a 40-Gigabit port. A 10-Gigabit port can use up to 125 MB of buffer space, which is equivalent to 100 ms of port bandwidth on a 10-Gigabit port. The total buffer sizes of the eight output queues on a port cannot exceed 100 percent, which is equal to the full 100 ms total buffer available to a port. The maximum amount of buffer space any queue can use is also 100 ms (which equates to a 100 percent buffer-size configuration), but if one queue uses all of the buffer, then no other queue receives buffer space.

There is no minimum buffer allocation, so you can set the buffer-size to zero (0) for a queue. However, we recommend that on queues on which you enable PFC to support lossless transport, you allocate a minimum of 5 ms (a minimum buffer-size of 5 percent). The two default lossless queues, fcoe and no-loss, have default buffer-size values of 35 ms (35 percent).

NOTE: If you do not configure buffer-size and you do not explicitly configure a queue scheduler, the default buffer-size is the default transmit rate of the queue. If you explicitly configure a queue scheduler, the default buffer allocations are not used. If you explicitly configure a queue scheduler, configure the buffer-size for each queue in the scheduler, keeping in mind that the total buffer-size of the queues cannot exceed 100 percent (100 ms).

If you do not use the default configuration, you can explicitly configure the queue buffer size in either of two ways:

- As a percentage—The queue receives the specified percentage of dedicated port buffers when the queue is mapped to the scheduler and the scheduler is mapped to a port.
- As a remainder—After the port services the queues that have an explicit percentage buffer size configuration, the remaining port dedicated buffer space is divided equally among the other queues to which a scheduler is attached. (No default or explicit scheduler means no dedicated buffer allocation for the queue.) If you configure a scheduler and you do not specify a buffer size as a percentage, *remainder* is the default setting.

Queue buffer allocation is dynamic, shared among ports as needed. However, a queue cannot use more than its configured amount of buffer space. For example, if you are using the default CoS configuration, the best-effort queue receives a maximum of 15 ms of buffer space because the default transmit rate for the best-effort queue is 15 percent.

If a switch experiences congestion, queues continue to receives their full buffer allocation until 90 percent of the 4 GB buffer space is consumed. When 90 percent of the buffer space is in use, the amount of buffer space per port, per queue, is reduced in proportion to the configured buffer size for each queue. As the percentage of consumed buffer space rises above 90 percent, the amount of buffer space per port, per queue, continues to be reduced.

On 40-Gigabit ports, because the total buffer is 4 GB and the maximum buffer a port can use is 500 MB, up to seven 40-Gigabit ports can consume their full 100 ms allocation of buffer space. However, if an eighth 40-Gigabit port requires the full 500 MB of buffer space, then the buffer allocations are proportionally reduced because the buffer consumption is above 90 percent.

On 10-Gigabit ports, because the total buffer is 4 GB and the maximum buffer a port can use is 125 MB, up to 28 10-Gigabit ports can consume their full 100 ms allocation of buffer space. However, if a 29th 10-Gigabit port requires the full 125 MB of buffer space, then the buffer allocations are proportionally reduced because the buffer consumption is above 90 percent.

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 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.

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 rate (see “Scheduling Priority” on page 348).

Scheduler Maps

A scheduler map maps a forwarding class to a queue scheduler. After configuring a scheduler, you must include it in a scheduler map, and apply the scheduler map to an interface to implement the configured queue scheduling.

RELATED DOCUMENTATION

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

IN THIS SECTION

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.

Requirements

This example uses the following hardware and software components:

- One QFX10000 switch.
- Junos OS Release 15.1X53-D10 or later for the QFX Series

Overview

Scheduler parameters define the following characteristics for the queues mapped to the scheduler:

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. 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.

  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.

Scheduler maps map schedulers to forwarding classes, and forwarding classes are mapped to output queues. After you configure schedulers and map them to forwarding classes in a scheduler map, you attach the scheduler map to an interface to implement the configured scheduling on output queues on that interface.

This process configures the bandwidth properties, scheduling, priority, and WRED characteristics that you map to forwarding classes (and thus to output queues) in a scheduler map.

Table 80 on page 363 shows the configuration components for this example.

Table 80: Components of the Port Output Queue Scheduler Configuration Example

Component	Settings
Hardware	One switch
Scheduler	Name: **be-sched** Transmit rate: **20%** Buffer size: **20%** Excess rate: **20%** Priority: **low** Drop profile: **be-dp** ECN: **disable** (default)
Scheduler map	Name: **be-map** Forwarding class to associate with the **be-sched** scheduler: **best-effort**

Configuring a CoS Scheduler

CLI Quick Configuration

To quickly configure a queue scheduler, 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 percent 20
set schedulers be-sched buffer-size percent 20
set schedulers be-sched excess-rate percent 20
set schedulers be-sched priority low
set schedulers be-sched drop-profile-map loss-priority low protocol any drop-profile be-dp
set scheduler-maps be-map forwarding-class best-effort scheduler be-sched
set interfaces xe-0/0/7 scheduler-map be-map

To configure a CoS scheduler:

1. Create scheduler (**be-sched**) and map it to the drop profile **be-dp**:

   ```
   [edit class-of-service schedulers]
   user@switch# set be-sched transmit-rate percent 20
   user@switch# set be-sched buffer-size percent 20
   user@switch# set be-sched excess-rate percent 20
   user@switch# set be-sched priority low
   user@switch# set be-sched drop-profile-map loss-priority low protocol any drop-profile be-dp
   ```

   > **NOTE:** Because ECN is disabled by default, no ECN configuration is shown.

2. Configure scheduler map (**be-map**) to associate the scheduler (**be-sched**) with the forwarding class (**best-effort**):

   ```
   [edit class-of-service scheduler-maps]
   user@switch# set be-map forwarding-class best-effort scheduler be-sched
   ```

3. Associate the scheduler map with an interface to apply scheduling to the best-effort forwarding class output queue:

   ```
   [edit class-of-service]
   set interfaces xe-0/0/7 scheduler-map be-map
   ```

Verification

IN THIS SECTION

To verify that the queue scheduler has been created and is mapped to the correct interfaces, perform these tasks:

Verifying the Scheduler Configuration

Purpose

Verify that the queue scheduler **be-sched** has been created with a minimum guaranteed bandwidth (**transmit-rate**) of 2 Gbps, an extra bandwidth sharing rate (**excess-rate**) of 20 percent, the priority set to **low**, and the drop profile **be-dp**.

Action

Display the scheduler using the operational mode command **show configuration class-of-service schedulers be-sched**:

`user@switch>` **show configuration class-of-service schedulers be-sched**

```
transmit-rate percent 20;
buffer-size percent 20;
excess-rate percent 20;
priority low;
drop-profile-map loss-priority low protocol any drop-profile be-dp;
```

Verifying the Scheduler Map Configuration

Purpose

Verify that the scheduler map **be-map** has been created and associates the forwarding class **best-effort** with the scheduler **be-sched**.

Action

Display the scheduler map using the operational mode command **show configuration class-of-service scheduler-maps be-map**:

`user@switch>` **show configuration class-of-service scheduler-maps be-map**

```
forwarding-class best-effort scheduler be-sched;
```

Verifying That the Scheduler Is Associated with the Interface

Purpose

Verify that the scheduler map **be-sched** is attached to interface **xe-0/0/7**.

Action

List the 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**

```
scheduler-map be-map;
```

RELATED DOCUMENTATION

Example: Configuring WRED Drop Profiles | **264**

Example: Configuring ECN | **284**

Defining CoS Queue Schedulers for Port Scheduling | **357**

Monitoring CoS Scheduler Maps | **341**

Understanding CoS Port Schedulers on QFX Switches | **343**

Understanding CoS Virtual Output Queues (VOQs) on QFX10000 Switches | **377**

CHAPTER 13

Troubleshooting Egress Bandwidth Issues

IN THIS CHAPTER

Troubleshooting Egress Bandwidth That Exceeds the Configured Minimum Bandwidth

Problem

Description: The guaranteed minimum bandwidth of a queue (forwarding class) or a priority group (forwarding class set) when measured at the egress port exceeds the guaranteed minimum bandwidth configured for the queue (transmit-rate) or for the priority group (guaranteed-rate).

> **NOTE:** On switches that support enhanced transmission selection (ETS) hierarchical scheduling, the switch allocates guaranteed minimum bandwidth first to a priority group using the guaranteed rate setting in the traffic control profile, and then allocates priority group minimum guaranteed bandwidth to forwarding classes in the priority group using the transmit rate setting in the queue scheduler.
>
> On switches that support direct port scheduling, there is no scheduling hierarchy. The switch allocates port bandwidth to forwarding classes directly, using the transmit rate setting in the queue scheduler.
>
> In this topic, if you are using direct port scheduling on your switch, ignore the references to priority groups and forwarding class sets (priority groups and forwarding class sets are only used for ETS hierarchical port scheduling). For direct port scheduling, only the transmit rate queue scheduler setting can cause the issue described in this topic.

Cause

When you configure bandwidth for a queue or a priority group, the switch accounts for the configured bandwidth as data only. The switch does not include the preamble and the interframe gap (IFG) associated with frames, so the switch does not account for the bandwidth consumed by the preamble and the IFG in its minimum bandwidth calculations.

The measured egress bandwidth can exceed the configured minimum bandwidth when small packet sizes (64 or 128 bytes) are transmitted because the preamble and the IFG are a larger percentage of the total traffic. For larger packet sizes, the preamble and IFG overhead are a small portion of the total traffic, and the effect on egress bandwidth is minor.

> **NOTE:** For ETS, 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.)
>
> For port scheduling, the sum of the queue transmit rates should not exceed the port bandwidth.

Solution

When you calculate the bandwidth requirements for queues and priority groups on which you expect a significant amount of traffic with small packet sizes, consider the transmit rate and the guaranteed rate as the minimum bandwidth for the data only. Add sufficient bandwidth to your calculations to account for the preamble and IFG so that the port bandwidth is sufficient to handle the combined minimum data rate and the preamble and IFG.

If the minimum bandwidth measured at the egress port exceeds the amount of bandwidth that you want to allocate to a queue or to a priority group, reduce the transmit rate for that queue and reduce the guaranteed rate of the priority group that contains the queue.

RELATED DOCUMENTATION

transmit-rate | 863
Example: Configuring Minimum Guaranteed Output Bandwidth | 392

Troubleshooting Egress Bandwidth That Exceeds the Configured Maximum Bandwidth

Problem

Description: The maximum bandwidth of a queue when measured at the egress port exceeds the maximum bandwidth rate shaper (**shaping-rate** statement on QFX5200, QFX5100, EX4600, QFX3500, QFX3600, and OCX1100 switches, and on QFabric systems, and **transmit-rate (rate | percentage *percent* exact** statement on QFX10000 switches) configured for the queue.

Cause

When you configure bandwidth for a queue (forwarding class) or a priority group (forwarding class set), the switch accounts for the configured bandwidth as data only. The switch does not rate-shape the preamble and the interframe gap (IFG) associated with frames, so the switch does not account for the bandwidth consumed by the preamble and the IFG in its maximum bandwidth calculations.

The measured egress bandwidth can exceed the configured maximum bandwidth when small packet sizes (64 or 128 bytes) are transmitted because the preamble and the IFG are a larger percentage of the total traffic. For larger packet sizes, the preamble and IFG overhead are a small portion of the total traffic, and the effect on egress bandwidth is minor.

Solution

When you calculate the bandwidth requirements for queues on which you expect a significant amount of traffic with small packet sizes, consider the shaping rate as the maximum bandwidth for the data only. Add sufficient bandwidth to your calculations to account for the preamble and IFG so that the port bandwidth is sufficient to handle the combined maximum data rate (shaping rate) and the preamble and IFG.

If the maximum bandwidth measured at the egress port exceeds the amount of bandwidth that you want to allocate to the queue, reduce the shaping rate for that queue.

Troubleshooting Egress Queue Bandwidth Impacted by Congestion

Problem

Description: Congestion on an egress port causes egress queues to receive less bandwidth than expected. Egress port congestion can impact the amount of bandwidth allocated to queues on the congested port and, in some cases, on ports that are not congested.

Cause

Egress queue congestion can cause the ingress port buffer to fill above a certain threshold and affect the flow to the queues on the egress port. One queue receives its configured bandwidth, but the other queues on the egress port are affected and do not receive their configured share of bandwidth.

Solution

The solution is to configure a drop profile to apply weighted random early detection (WRED) to the queue or queues on the congested ports.

Configure a drop profile on the queue that is receiving its configured bandwidth. This queue is preventing the other queues from receiving their expected bandwidth. The drop profile prevents the queue from affecting the other queues on the port.

To configure a WRED profile using the CLI:

Name the drop profile and set the drop start point, drop end point, minimum drop rate, and maximum drop rate for the drop profile:

`[edit` **class-of-service**`]`
`user@switch#` **set drop-profile *drop-profile-name* interpolate fill-level *percentage* fill-level *percentage* drop-probability 0 drop-probability *percentage***

RELATED DOCUMENTATION

drop-profile | 747

Example: Configuring WRED Drop Profiles | 264

Example: Configuring CoS Hierarchical Port Scheduling (ETS) | 415

Understanding CoS WRED Drop Profiles | 253

CHAPTER 14

Traffic Control Profiles and Priority Group Scheduling

IN THIS CHAPTER

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) | 407
Example: Configuring CoS Hierarchical Port Scheduling (ETS) | 415
Example: Configuring Traffic Control Profiles (Priority Group Scheduling) | 385
Defining CoS Traffic Control Profiles (Priority Group Scheduling) | 384

Understanding CoS Priority Group Scheduling

IN THIS SECTION

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 81 on page 374 provides a quick reference to the traffic control profile components you can configure to determine the bandwidth properties of priority groups, and Table 82 on page 374 provides a quick reference to some related scheduling configuration components.

Table 81: 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 82: Other Scheduling Components

Other Scheduling Components	Description
Forwarding class	Maps traffic to a queue (priority).

Table 82: Other Scheduling Components *(continued)*

Other Scheduling Components	Description
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 | **17**
Understanding CoS Output Queue Schedulers | **313**
Understanding CoS Hierarchical Port Scheduling (ETS) | **407**
Understanding CoS Scheduling Behavior and Configuration Considerations | **307**
Understanding CoS Scheduling on QFabric System Node Device Fabric (fte) Ports
Understanding Default CoS Scheduling on QFabric System Interconnect Devices (Junos OS Release 13.1 and Later Releases)
Example: Configuring CoS Hierarchical Port Scheduling (ETS) | **415**
Example: Configuring Minimum Guaranteed Output Bandwidth | **392**

Understanding CoS Virtual Output Queues (VOQs) on QFX10000 Switches

IN THIS SECTION

The traditional method of forwarding traffic through a switch is based on buffering ingress traffic in input queues on ingress interfaces, forwarding the traffic across the switch fabric to output queues on egress interfaces, and then buffering traffic again on the output queues before transmitting the traffic to the next hop. The traditional method of queueing packets on an ingress port is storing traffic destined for different egress ports in the same input queue (buffer).

During periods of congestion, the switch might drop packets at the egress port, so the switch might spend resources transporting traffic across the switch fabric to an egress port, only to drop that traffic instead of forwarding it. And because input queues store traffic destined for different egress ports, congestion on one egress port could affect traffic on a different egress port, a condition called *head-of-line blocking (HOLB)*.

Virtual output queue (VOQ) architecture takes a different approach:

- Instead of separate physical buffers for input and output queues, the switch uses the physical buffers on the ingress pipeline of each Packet Forwarding Engine (PFE) chip to store traffic for every egress port. Every output queue on an egress port has buffer storage space on every ingress pipeline on all of the PFE chips on the switch. The mapping of ingress pipeline storage space to output queues is 1-to-1, so each output queue receives buffer space on each ingress pipeline.
- Instead of one input queue containing traffic destined for multiple different output queues (a one-to-many mapping), each output queue has a dedicated VOQ comprised of the input buffers on each packet forwarding chip that are dedicated to that output queue (a 1-to-1 mapping). This architecture prevents communication between any two ports from affecting another port.
- Instead of storing traffic on a physical output queue until it can be forwarded, a VOQ does not transmit traffic from the ingress port across the fabric to the egress port until the egress port has the resources to forward the traffic.

A VOQ is a collection of input queues (buffers) that receive and store traffic destined for one output queue on one egress port. Each output queue on each egress port has its own dedicated VOQ, which consists of all of the input queues that are sending traffic to that output queue.

VOQ Architecture

IN THIS SECTION

A VOQ represents the ingress buffering for a particular output queue. A unique buffer ID identifies each output queue on a PFE chip. Each of the six PFE chips uses the same unique buffer ID for a particular output queue. The traffic stored using a particular buffer ID on the six PFE chips comprises the traffic destined for one particular output queue on one port, and is the VOQ for that output queue.

A switch that has 72 egress ports with 8 output queues on each port, has 576 VOQs on each PFE chip (72 x 8 = 576). Because the switch has six PFE chips, the switch has a total of 3,456 VOQs (576 x 6 = 3,456).

A VOQ is distributed across all of the PFE chips that are actively sending traffic to that output queue. Each output queue is the sum of the total buffers assigned to that output queue (by its unique buffer ID) across all of the PFE chips. So the output queue itself is virtual, not physical, although the output queue is comprised of physical input queues.

Round-Trip Time Buffering

Although there is no output queue buffering during periods of congestion (no long-term storage), there is a small physical output queue buffer on egress line cards to accommodate the round-trip time for traffic to traverse the switch fabric from ingress to egress. The round-trip time consists of the time it takes the ingress port to request egress port resources, receive a grant from the egress port for resources, and transmit the data across the switch fabric.

That means if a packet is not dropped at the switch ingress, and the switch forwards the packet across the fabric to the egress port, the packet will not be dropped and will be forwarded to the next hop. All packet drops take place in the ingress pipeline.

The switch has 4 GB of external DRAM to use as a delay bandwidth buffer (DBB). The DBB provides storage for ingress ports until the ports can forward traffic to egress ports.

Requesting and Granting Egress Port Bandwidth

When packets arrive at an ingress port, the ingress pipeline stores the packet in the ingress queue with the unique buffer ID of the destination output queue. The switch makes the buffering decision after performing the packet lookup. If the packet belongs to a class for which the maximum traffic threshold has been exceeded, the packet might not be buffered and might be dropped. To transport packets across the switch fabric to egress ports:

1. The ingress line card PFE request scheduler sends a request to the egress line card PFE grant scheduler to notify the egress PFE that data is available for transmission.

2. When there is available egress bandwidth, the egress line card grant scheduler responds by sending a bandwidth grant to the ingress line card PFE.

3. The ingress line card PFE receives the grant from the egress line card PFE, and transmits the data to the egress line card.

Ingress packets remain in the VOQ on the ingress port input queues until the output queue is ready to accept and forward more traffic.

Under most conditions, the switch fabric is fast enough to be transparent to egress class-of-service (CoS) policies, so the process of forwarding traffic from the ingress pipeline, across the switch fabric, to egress ports, does not affect the configured CoS policies for the traffic. The fabric only affects CoS policy if there is a fabric failure or if there is an issue of port fairness.

When a packet ingresses and egresses the same PFE chip (local switching), the packet does not traverse the switch fabric. However, the switch uses the same request and grant mechanism to receive egress bandwidth as packets that cross the fabric, so locally switched packets and packets that arrive at a PFE chip after crossing the switch fabric are treated fairly when the traffic is contending for the same output queue.

VOQ Advantages

IN THIS SECTION

VOQ architecture provides two major advantages:

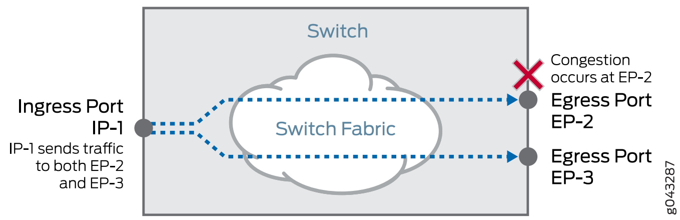
Eliminate Head-of-Line Blocking

VOQ architecture eliminates head-of-line blocking (HOLB) issues. On non-VOQ switches, HOLB occurs when congestion at an egress port affects a different egress port that is not congested. HOLB occurs when the congested port and the uncongested port share the same input queue on an ingress interface.

An example of a HOLB scenario is a switch that has streams of traffic entering one ingress port (IP-1) that are destined for two different egress ports (EP-2 and EP-3):

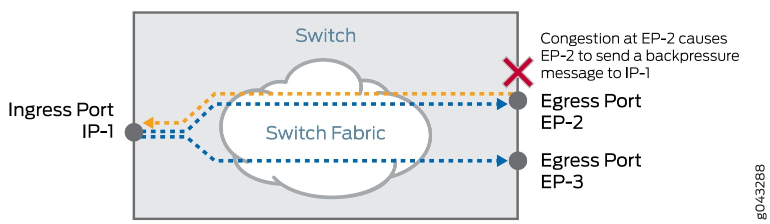
1. Congestion occurs on egress port EP-2. There is no congestion on egress port EP-3, as shown in Figure 11 on page 381.

Figure 11: Congestion Occurs on EP-2



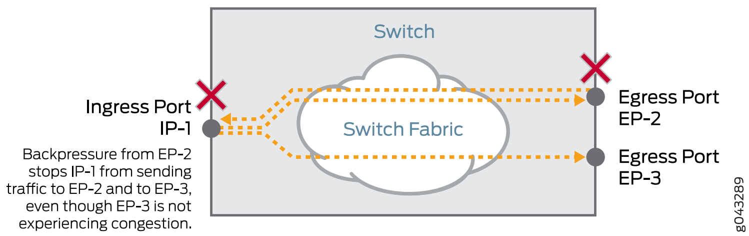
2. Egress port EP-2 sends a backpressure signal to ingress port IP-1, as shown in Figure 12 on page 381.

Figure 12: EP-2 Backpressures IP-1



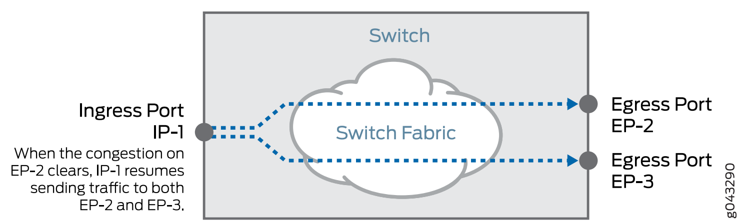
3. The backpressure signal causes the ingress port IP-1 to stop sending traffic and to buffer traffic until it receives a signal to resume sending, as shown in Figure 13 on page 382. Traffic that arrives at ingress port IP-1 destined for uncongested egress port EP-3 is buffered along with the traffic destined for congested port EP-2, instead of being forwarded to port EP-3.

Figure 13: Backpressure from EP-2 Causes IP-1 to Buffer Traffic Instead of Sending Traffic, Affecting EP-3



4. Ingress port IP-1 transmits traffic to uncongested egress port EP-3 only when egress port EP-2 clears enough to allow ingress port IP-1 to resume sending traffic, as shown in Figure 14 on page 382.

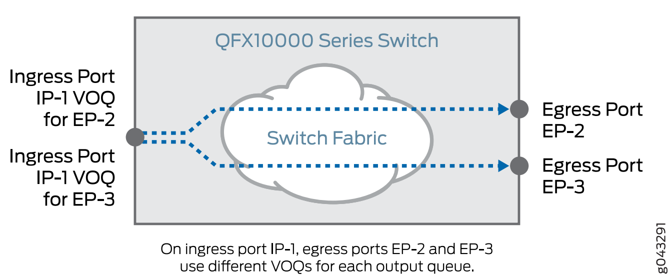
Figure 14: Congestion on EP-2 Clears, Allowing IP-1 to Resume Sending Traffic to Both Egress Ports



In this way, congested egress port EP-2 negatively affects uncongested egress port EP-3, because both egress ports share the same input queue on ingress port IP-1.

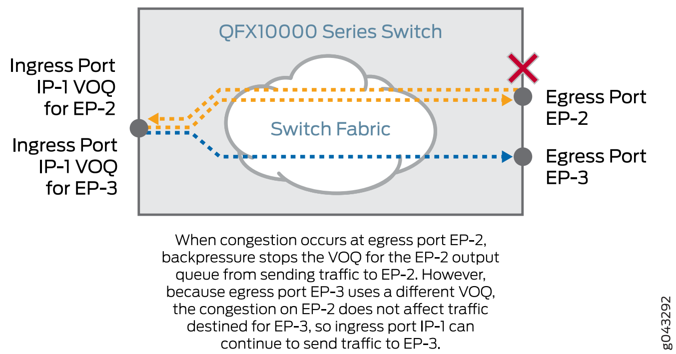
VOQ architecture avoids HOLB by creating a different dedicated virtual queue for each output queue on each interface, as shown in Figure 15 on page 382.

Figure 15: Each Egress Port Has a Separate Virtual Output Queue on IP-1



Because different egress queues do not share the same input queue, a congested egress queue on one port cannot affect an egress queue on a different port, as shown in Figure 16 on page 383. (For the same reason, a congested egress queue on one port cannot affect another egress queue on the same port—each output queue has its own dedicated virtual output queue composed of ingress interface input queues.)

Figure 16: Congestion on EP-2 Does Not Affect Uncongested Port EP-3



Performing queue buffering at the ingress interface ensures that the switch only sends traffic across the fabric to an egress queue if that egress queue is ready to receive that traffic. If the egress queue is not ready to receive traffic, the traffic remains buffered at the ingress interface.

Increase Fabric Efficiency and Utilization

Traditional output queue architecture has some inherent inefficiencies that VOQ architecture addresses.

- Packet buffering—Traditional queueing architecture buffers each packet twice in long-term DRAM storage, once at the ingress interface and once at the egress interface. VOQ architecture buffers each packet only once in long-term DRAM storage, at the ingress interface. The switch fabric is fast enough to be transparent to egress CoS policies, so instead of buffering packets a second time at the egress interface, the switch can forward traffic at a rate that does not require deep egress buffers, without affecting the configured egress CoS policies (scheduling).
- Consumption of resources—Traditional queueing architecture sends packets from the ingress interface input queue (buffer), across the switch fabric, to the egress interface output queue (buffer). At the egress interface, packets might be dropped, even though the switch has expended resources transporting the packets across the fabric and storing them in the egress queue. VOQ architecture does not send packets across the fabric to the egress interface until the egress interface is ready to transmit the traffic. This increases system utilization because no resources are wasted transporting and storing packets that are dropped later.

Independent of VOQ architecture, the Juniper Networks switching architecture also provides better fabric utilization because the switch converts packets into cells. Cells have a predictable size, which enables the switch to spray the cells evenly across the fabric links and more fully utilize the fabric links. Packets vary greatly in size, and packet size is not predictable. Packet-based fabrics can deliver no better than 65-70 percent utilization because of the variation and unpredictability of packet sizes. Juniper Networks' cell-based fabrics can deliver a fabric utilization rate of almost 95 percent because of the predictability of and control over cell size.

RELATED DOCUMENTATION

Understanding CoS Port Schedulers on QFX Switches | 343

Example: Configuring Queue Schedulers for Port Scheduling | 361

Understanding Default CoS Scheduling and Classification | 298

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 traffic-control-profile-name]`
   `user@switch#` **set scheduler-map *scheduler-map-name***

RELATED DOCUMENTATION

Example: Configuring CoS Hierarchical Port Scheduling (ETS) | 415
Example: Configuring Traffic Control Profiles (Priority Group Scheduling) | 385
Example: Configuring Minimum Guaranteed Output Bandwidth | 392
Example: Configuring Maximum Output Bandwidth | 400
Defining CoS Queue Schedulers | 321
Understanding CoS Traffic Control Profiles | 372

Example: Configuring Traffic Control Profiles (Priority Group Scheduling)

IN THIS SECTION

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:

- A Juniper Networks QFX3500 Switch
- Junos OS Release 11.1 or later for the QFX Series

Overview

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

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

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) | 415

Example: Configuring Minimum Guaranteed Output Bandwidth | 392

Example: Configuring Maximum Output Bandwidth | 400

Example: Configuring Queue Schedulers | 325

Defining CoS Traffic Control Profiles (Priority Group Scheduling) | 384

Understanding CoS Traffic Control Profiles | 372

Understanding CoS Hierarchical Port Scheduling (ETS) | 407

Understanding CoS Priority Group and Queue Guaranteed Minimum Bandwidth

IN THIS SECTION

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.

This topic covers:

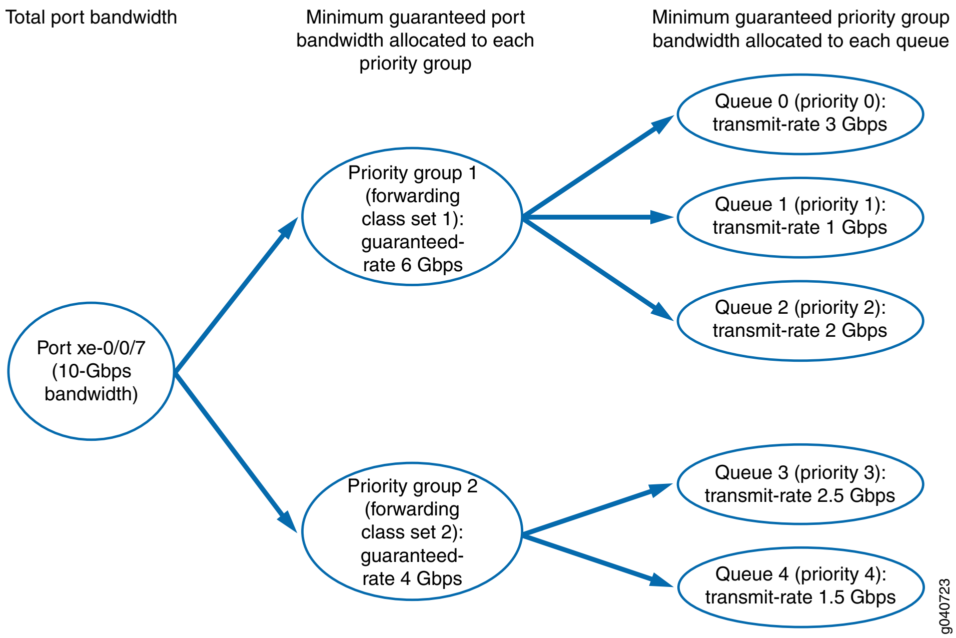
Guaranteeing Bandwidth Using Hierarchical Scheduling

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 17 on page 390 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 17: 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 | 313
Understanding CoS Traffic Control Profiles | 372
Example: Configuring CoS Hierarchical Port Scheduling (ETS) | 415
Example: Configuring Queue Schedulers | 325
Example: Configuring Traffic Control Profiles (Priority Group Scheduling) | 385
Defining CoS Queue Schedulers | 321
Defining CoS Traffic Control Profiles (Priority Group Scheduling) | 384

Example: Configuring Minimum Guaranteed Output Bandwidth

IN THIS SECTION

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 than priority groups and queues receive the bandwidth required to support the expected traffic.

Requirements

This example uses the following hardware and software components:

- A Juniper Networks QFX3500 Switch
- Junos OS Release 11.1 or later for the QFX Series or Junos OS Release 14.1X53-D20 or later for the OCX Series

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 83 on page 393 shows the configuration components for this example:

Table 83: Components of the Minimum Guaranteed Output Bandwidth Configuration Example

Component	Settings
Hardware	QFX3500 switch

Table 83: Components of the Minimum Guaranteed Output Bandwidth Configuration Example *(continued)*

Component	Settings
Minimum guaranteed queue bandwidth	Transmit rate: **2g**
Minimum guaranteed priority group bandwidth	Guaranteed rate: **4g**
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**

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

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
```

Verification

IN THIS SECTION

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

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

IN THIS SECTION

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.

This topic covers:

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.

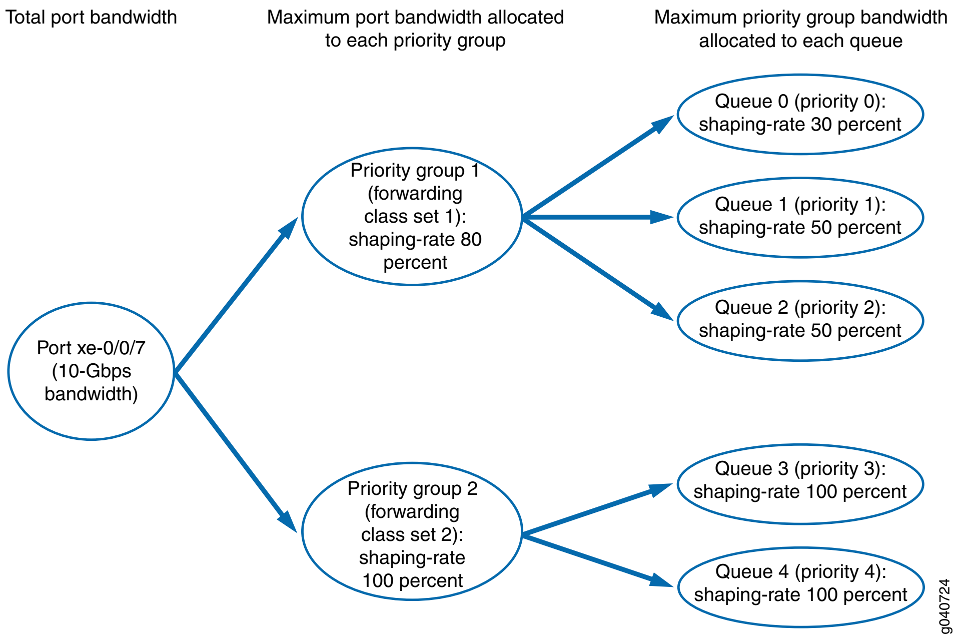
You configure the queue **shaping-rate** in the scheduler configuration, and you set the **shaping-rate** for priority groups in the traffic control profile configuration.

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 18 on page 400 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 18: Setting Maximum Bandwidth Using Hierarchical Scheduling



RELATED DOCUMENTATION

Understanding CoS Output Queue Schedulers | 313
Understanding CoS Traffic Control Profiles | 372
Example: Configuring CoS Hierarchical Port Scheduling (ETS) | 415
Example: Configuring Queue Schedulers | 325
Example: Configuring Traffic Control Profiles (Priority Group Scheduling) | 385
Defining CoS Queue Schedulers | 321
Defining CoS Traffic Control Profiles (Priority Group Scheduling) | 384

Example: Configuring Maximum Output Bandwidth

IN THIS SECTION

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.

Requirements

This example uses the following hardware and software components:

- One switch (this example was tested on a Juniper Networks QFX3500 Switch)
- Junos OS Release 11.1 or later for the QFX Series or Junos OS Release 14.1X53-D20 or later for the OCX Series

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 +/– 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 84 on page 402 shows the configuration components for this example:

Table 84: Components of the Maximum Output Bandwidth Configuration Example

Component	Settings
Hardware	QFX3500 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**

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

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**

Verification

IN THIS SECTION

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

CHAPTER 15

Hierarchical Port Scheduling (ETS)

IN THIS CHAPTER

Understanding CoS Hierarchical Port Scheduling (ETS)

IN THIS SECTION

Scheduling defines the class-of-service (CoS) properties of output queues. Output queues are mapped to forwarding classes. CoS scheduler properties include the amount of interface bandwidth assigned to the queue, the queue priority, and the drop profiles associated with the queue.

Hierarchical port scheduling is a two-tier process that provides better port bandwidth utilization and greater flexibility to allocate resources to queues (forwarding classes) and to groups of queues (forwarding class sets). Hierarchical scheduling includes the Junos OS implementation of enhanced transmission selection (ETS), as described in IEEE 802.1Qaz.

> **NOTE:** All QFX Series devices use ETS scheduling, except for QFX5120, QFX5200, QFX5210, and QFX10002-60C switches.
>
> Starting with Junos OS 17.3, QFX10000 devices, except for QFX10002-60C, support ETS scheduling. However, Juniper does not recommend configuring ETS on supported QFX10000 devices on Junos OS 18.3 or before.
>
> EX4600 switches use ETS scheduling while EX4650 switches do not.

Video: What is Enhanced Transmission Selection?

This topic describes:

Hierarchical Scheduling Tiers

The two tiers used in hierarchical scheduling are priorities and priority groups, as shown in Table 85 on page 409.

Table 85: Hierarchical Scheduling Tiers

Junos OS Configuration Construct	Equivalent ETS Construct	Description
Forwarding class	Priority	Think about priorities (forwarding classes) as output queues. You map forwarding classes to queues, so each forwarding class represents an output queue. When you use a classifier to map a forwarding class to an IEEE 802.1p code point, the code point identifies that traffic's priority for priority-based flow control (PFC). Thus the forwarding class, the queue mapped to the forwarding class, and the priority (code point) mapped to the forwarding class all identify the same traffic. **NOTE:** OCX Series switches do not support lossless transport or PFC.
Forwarding class set	Priority group	Priority groups (forwarding class sets) are groups of priorities (forwarding classes). Forwarding class membership in a forwarding class set defines the priority group to which each priority belongs. You can configure up to three unicast priority groups and one multicast priority group.

You apply scheduling properties to each hierarchical scheduling tier as descried in the next section.

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

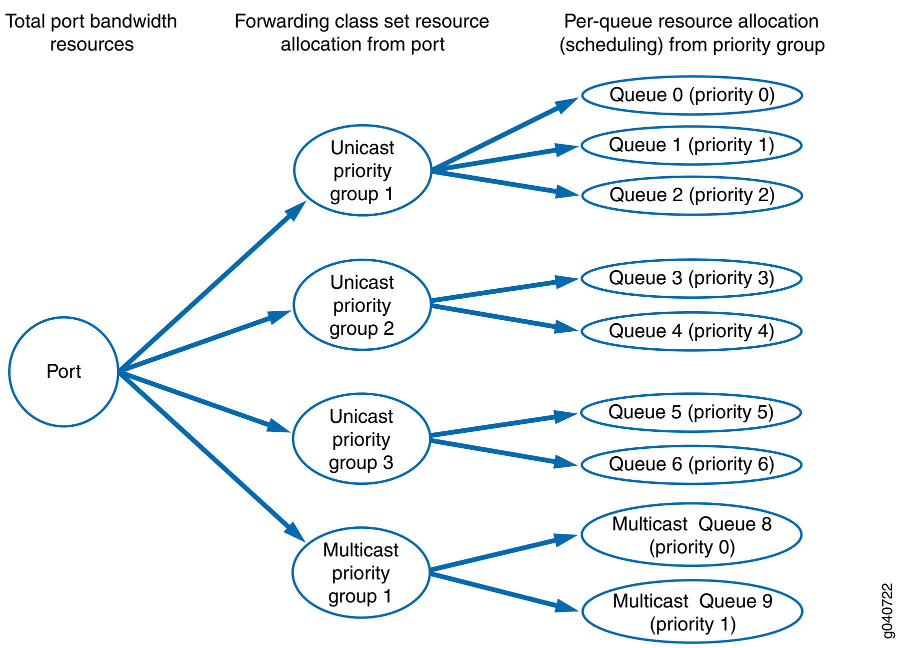
NOTE: On OCX Series switches, by default, classifiers use DSCP code points to map traffic to forwarding classes. However, hierarchical scheduling works in the same manner as when you use IEEE 802.1p code points to classify traffic. The OCX Series classifies traffic into forwarding classes based on DSCP code points, the forwarding classes are mapped to forwarding class sets, and you apply scheduling properties to each of the two tiers.

Hierarchical Scheduling and ETS

Two-tier hierarchical scheduling manages bandwidth efficiently by enabling you to define the CoS properties for each priority group and for each priority. The first tier of the hierarchical scheduler allocates port bandwidth to a priority group. The second tier of the hierarchical scheduler determines the portion of the priority group bandwidth that a priority (queue) can use.

The CoS properties of a priority group define the amount of port bandwidth resources available to the queues in that priority group. The CoS properties you configure for each queue specify the amount of the bandwidth available to the queue from the bandwidth allocated to the priority group. Figure 19 on page 410 shows the relationship of port resource allocation to priority groups, and priority group resource allocation to queues (priorities).

Figure 19: Hierarchical Scheduling Tiers



If a queue (priority) does not use its allocated bandwidth, ETS shares the unused bandwidth among the other queues in the priority group in proportion to the minimum guaranteed rate (transmit rate) scheduled for each queue. If a priority group does not use its allocated bandwidth, ETS shares the unused bandwidth among the priority groups on the port in proportion to the minimum guaranteed rate (guaranteed rate) scheduled for each priority group.

In this way, ETS improves link bandwidth utilization, and it provides each queue and each priority group with the maximum available bandwidth. For example, priorities that consist of bursty traffic can share bandwidth during periods of low traffic transmission, instead of reserving their entire bandwidth allocation when traffic loads are light.

NOTE: The available link bandwidth is the bandwidth remaining after servicing **strict-high** priority flows. Strict-high priority takes precedence over all other traffic. We recommend that you configure a **shaping-rate** (**transmit-rate** on QFX10000 switches) to limit the maximum amount of bandwidth that a strict-high priority forwarding class can use to prevent starving other queues.

ETS Advertisement in DCBX

When you configure hierarchical scheduling on a port, Data Center Bridging Capability Exchange protocol (DCBX) advertises:

- Each priority group
- The priorities in each priority group
- The bandwidth properties of each priority group and priority

When you configure hierarchical scheduling on a port, any priority that is not part of an explicitly configured priority group is assigned to the automatically generated default priority group and receives no bandwidth. The default priority group is transparent. It does not appear in the configuration.

NOTE: OCX Series switches do not support DCBX, so hierarchical scheduling information is not exchanged with connected peers on OCX Series switches.

Hierarchical Scheduling Process

Hierarchical scheduling consists of multiple configuration steps that create the priorities and the priority groups, schedule their resources, and assign them to interfaces. The steps below correspond to the six blocks in the packet flow diagram shown in Figure 20 on page 413:

1. Packet classification:

   - Configure classification of incoming traffic into forwarding classes (priorities). This consists of either using the default classifiers or configuring classifiers to map code points and loss priorities to the forwarding classes.

   - Apply the classifiers to ingress interfaces or use the default classifiers. Applying a classifier to an interface groups incoming traffic on the interface into forwarding classes and loss priorities, by applying the classifier code point mapping to the incoming traffic.

2. Configure the output queues for the forwarding classes (priorities). This consists of either using the default forwarding classes and forwarding-class-to-queue mapping, or creating your own forwarding classes and mapping them to output queues.

3. Allocate resources to the forwarding classes:

   - Define resources for the priorities. This consists of configuring schedulers to set minimum guaranteed bandwidth, maximum bandwidth, drop profiles for Weighted Random Early Detection (WRED), and bandwidth priority to apply to a forwarding class. Extra bandwidth is shared among queues in proportion to the minimum guaranteed bandwidth (transmit rate) of each queue.

   - Map resources to priorities. This consists of mapping forwarding classes to schedulers, using a scheduler map.

4. Configure priority groups. This consists of mapping forwarding classes (priorities) to forwarding class sets (priority groups) to define the priorities that belong to each priority group.

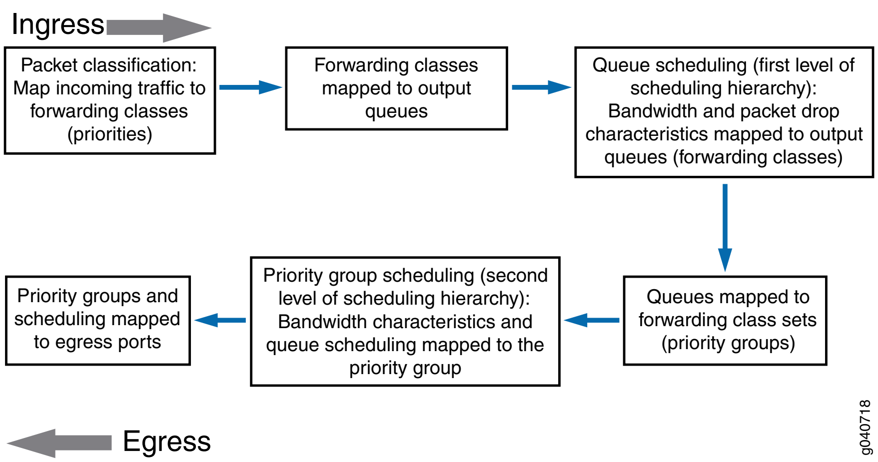
5. Define resources for the priority groups. This consists of configuring traffic control profiles to set minimum guaranteed bandwidth (**guaranteed-rate**) and maximum bandwidth (**shaping-rate** on switches other than QFX10000 switches, **transmit-rate** on QFX10000 switches) for a priority group. Traffic control profiles also specify a scheduler map, which defines the resources (schedulers) mapped to the priorities in the priority group. Extra port bandwidth is shared among priority groups in proportion to the minimum guaranteed bandwidth of each priority group.

   The traffic control profile bandwidth settings determine the port resources available to the priority group. The schedulers specified in the scheduler map determine the amount of priority group resources that each priority receives.

NOTE: QFX10000 switches do not support defining a shaping rate for priority groups. Instead, set the maximum bandwidth for a priority group by defining a transmit rate. See transmit-rate.

6. Apply hierarchical scheduling to a port. This consists of attaching one or more priority groups (forwarding class sets) to an interface. For each priority group, you also attach a traffic control profile, which contains the scheduling properties of the priority group and the priorities in the priority group. Different priority groups on the same port can use different traffic control profiles, which provides fine tuned control of scheduling for each queue on each interface.

Figure 20: Hierarchical Scheduling Packet Flow



Strict-High Priority Queues and Hierarchical Scheduling

If you configure a strict-high priority queue, you must observe the following rules:

- You must create a separate forwarding class set (priority group) for the strict-high priority queue.
- Only one forwarding class set can contain strict-high priority queues.
- Strict-high priority queues cannot belong to the same forwarding class set as queues that are not strict-high priority.
- A strict-high priority queue cannot belong to a multidestination forwarding class set.
- We recommend that you always apply a shaping-rate (transmit-rate on QFX10000 switches) to strict-high priority queues to limit the amount of bandwidth a strict-high priority queue can use. If you do not 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 a QFabric system, if a fabric (fte) interface handles strict-high priority traffic, you must define a separate forwarding class set (priority group) for strict-high priority traffic. Strict-high priority traffic cannot be mixed with traffic of other priorities in a forwarding class set. For example, you might choose to create different forwarding class sets for best effort, lossless, strict-high priority, and multidestination traffic.

Default Hierarchical Scheduling

NOTE: There is no default hierarchical scheduling on QFX10000 switches. QFX10000 switches use port scheduling by default, and you must explicitly configure hierarchical scheduling to enable ETS. Also on QFX10000 switches, changing from port scheduler to ETS or from ETS to port scheduler requires a reboot.

QFX10002-60C devices do not support ETS.

If you do not explicitly configure hierarchical scheduling, the switch uses the default settings:

- The switch automatically creates a default forwarding class set that contains all of the forwarding classes on the switch. The switch assigns 100 percent of the port output bandwidth to the default forwarding class set. The default forwarding class set is transparent. It does not appear in the configuration and is used for Data Center Bridging Capability Exchange protocol (DCBX) advertisement.

  NOTE: OCX Series switches do not support DCBX, so the ETS configuration is not advertised to connected peers.

- 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.

Release History Table

Release	Description
17.3	Starting with Junos OS 17.3, QFX10000 devices, except for QFX10002-60C, support ETS scheduling. However, Juniper does not recommend configuring ETS on supported QFX10000 devices on Junos OS 18.3 or before.

RELATED DOCUMENTATION

Example: Configuring CoS Hierarchical Port Scheduling (ETS)

IN THIS SECTION

Hierarchical port scheduling defines the class-of-service (CoS) properties of output queues, which are mapped to forwarding classes. Traffic is classified into forwarding classes based on code point (priority), so mapping queues to forwarding classes also maps queues to priorities). Hierarchical port scheduling 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.

Hierarchical port scheduling is the Junos OS implementation of enhanced transmission selection (ETS), as described in IEEE 802.1Qaz. One major benefit of hierarchical port scheduling is greater port bandwidth utilization. If a priority group on a port does not use all of its allocated bandwidth, other priority groups on that port can use that bandwidth. Also, if a priority within a priority group does not use its allocated bandwidth, other priorities within that priority group can use that bandwidth.

Configuring hierarchical scheduling is a multistep procedure that includes:

- Mapping forwarding classes to queues
- Defining forwarding class sets (priority groups)
- Defining behavior aggregate classifiers
- Configuring priority-based flow control (PFC) for lossless priorities (queues)
- Applying classifiers and PFC configuration to ingress interfaces
- Defining drop profiles
- Defining schedulers
- Mapping forwarding classes to schedulers
- Defining traffic control profiles
- Assigning priority groups and traffic control profiles to egress ports

NOTE: OCX Series switches do not support lossless transport and do not support PFC. Although this example includes configuring lossless transport with PFC, the portions of the example that do not pertain to lossless transport still apply to OCX Series switches. (You can configure hierarchical scheduling on OCX Series switches, but you cannot configure lossless transport or lossless forwarding classes.)

This example describes how to configure hierarchical scheduling:

Requirements

This example uses the following hardware and software components:

- One switch (this example was tested on a Juniper Networks QFX3500 Switch)
- Junos OS Release 11.1 or later for the QFX Series or Junos OS Release 14.1X53-D20 or later for the OCX Series

Overview

Keep the following considerations in mind when you plan the port bandwidth allocation for priority groups and for individual priorities:

- How much traffic and what types of traffic you expect to traverse the system.
- How you want to divide different types of traffic into priorities (forwarding classes) to apply different CoS treatment to different types of traffic. Dividing traffic into priorities includes:
  - Mapping the code points of ingress traffic to forwarding classes using behavior aggregate (BA) classifiers. This classifies incoming traffic into the appropriate forwarding class based on code point.
  - Mapping forwarding classes to output queues. This defines the output queue for each type of traffic.
  - Attaching the BA classifier to the desired ingress interfaces so that incoming traffic maps to the desired forwarding classes and queues.
- How you want to organize priorities into priority groups (forwarding class sets).

  Traffic that requires similar treatment usually belongs in the same priority group. To do this, place forwarding classes that require similar bandwidth, loss, and other characteristics in the same forwarding class set. For example, you can map all types of best-effort traffic forwarding classes into one forwarding class set.
- How much of the port bandwidth you want to allocate to each priority group and to each of the priorities in each priority group. The following considerations apply to bandwidth allocation:
  - Estimate how much traffic you expect in each forwarding class, and how much traffic you expect in each forwarding class set (the amount of traffic you expect in a forwarding class set is the aggregate amount of traffic in the forwarding classes that belong to the forwarding class set).
  - The combined minimum guaranteed bandwidth of the priorities (forwarding classes) in a priority group should not exceed the minimum guaranteed bandwidth of the priority group (forwarding class set). The transmit rate scheduler parameter defines the minimum guaranteed bandwidth for forwarding classes. Scheduler maps associate schedulers with forwarding classes.
  - The combined minimum guaranteed bandwidth of the priority groups (forwarding class sets) on a port should not exceed the port’s total bandwidth. The guaranteed rate parameter in the traffic control profile defines the minimum bandwidth for a forwarding class set. Associating a scheduler map with a traffic control profile sets the scheduling for the individual forwarding classes in the forwarding class set.

This example creates hierarchical port scheduling by defining priority groups for best effort, guaranteed delivery, and high-performance computing (HPC) traffic. Each priority group includes priorities that need to receive similar CoS treatment. Each priority group and each priority within each priority group receive the CoS resources needed to service their flows. Lossless priorities use PFC to prevent packet loss when the network experiences congestion.

Topology

Table 86 on page 418 shows the configuration components for this example.

> **NOTE:** OCX Series switches do not support lossless transport and do not support PFC. If you eliminate the configuration elements for the default lossless **fcoe** and **no-loss** forwarding classes (including classifier, forwarding class set, scheduler, and traffic control profile configuration for those forwarding classes) and for PFC, this example works for OCX Series switches. However, because the default **fcoe** and **no-loss** forwarding classes do not carry traffic on OCX Series switches, you can apply the bandwidth allocated to those forwarding classes to other forwarding classes. By default, the active forwarding classes (**best-effort**, **network-control**, and **mcast**) share the unused bandwidth assigned to the **fcoe** and **no-loss** forwarding classes.

Table 86: Components of the Hierarchical Port Scheduling (ETS) Configuration Topology

Property	Settings
Hardware	QFX3500 switch
Mapping of forwarding classes (priorities) to queues	**best-effort** to queue **0** **be2** to queue **1** **fcoe** (Fibre Channel over Ethernet) to queue **3** **no-loss** to queue **4** **hpc** (high-performance computing) to queue **5** **network-control** to queue **7** **NOTE:** On switches that do not support the ELS CLI, if you are using Junos OS Release 12.2 or later, use the default forwarding-class-to-queue mapping for the lossless **fcoe** and **no-loss** forwarding classes. If you explicitly configure the default lossless forwarding classes, the traffic mapped to those forwarding classes is treated as lossy (**best-effort**) traffic and does *not* receive lossless treatment. On switches that do not support the ELS CLI, in Junos OS Release 12.3 and later, you can include the *no-loss* packet drop attribute in the explicit forwarding class configuration to configure a lossless forwarding class.
Forwarding class sets (priority groups)	**best-effort-pg**: contains forwarding classes **best-effort**, **be2**, and **network control** **guar-delivery-pg**: contains forwarding classes **fcoe** and **no-loss** **hpc-pg**: contains forwarding class **hpc**

Table 86: Components of the Hierarchical Port Scheduling (ETS) Configuration Topology *(continued)*

Property	Settings
Behavior aggregate classifier (maps forwarding classes and loss priorities to incoming packets by IEEE 802.1 code point)	Name—**hsclassifier1** Code point mapping: • **000** to forwarding class **best-effort** and loss priority **low** • **001** to forwarding class **be2** and loss priority **high** • **011** to forwarding class **fcoe** and loss priority **low** • **100** to forwarding class **no-loss** and loss priority **low** • **101** to forwarding class **hpc** and loss priority **low** • **110** to forwarding class **network-control** and loss priority **low**
PFC	Congestion notification profile name—**gd-cnp** PFC enabled on code points: **011** (**fcoe** priority), **010** (**no-loss** priority)
Drop profiles **NOTE:** The **fcoe** and **no-loss** priorities (queues) do not use drop profiles because they are lossless traffic classes.	**dp-be-low**: drop start point **25**, drop end point **50**, maximum drop rate **80** **dp-be-high**: drop start point **10**, drop end point **40**, maximum drop rate **100** **dp-hpc**: drop start point **75**, drop end point **90**, maximum drop rate **75** **dp-nc**: drop start point **80**, drop end point **100**, maximum drop rate **100**
Queue schedulers	**be-sched**: minimum bandwidth **3g**, maximum bandwidth **100%**, priority **low**, drop profiles **dp-be-low** and **dp-be-high** **fcoe-sched**: minimum bandwidth **2.5g**, maximum bandwidth **100%**, priority **low** **hpc-sched**: minimum bandwidth **2g**, maximum bandwidth **100%**, priority **low**, drop profile **dp-hpc** **nc-sched**: minimum bandwidth **500m**, maximum bandwidth **100%**, priority **low**, drop profile **dp-nc** **nl-sched**: minimum bandwidth **2g**, maximum bandwidth **100%**, priority **low**

Table 86: Components of the Hierarchical Port Scheduling (ETS) Configuration Topology *(continued)*

Property	Settings
Forwarding class-to-scheduler mapping	Scheduler map **be-map**: Forwarding class **best-effort**, scheduler **be-sched** Forwarding class **be2**, scheduler **be-sched** Forwarding class **network-control**, scheduler **nc-sched** Scheduler map **gd-map**: Forwarding class **fcoe**, scheduler **fcoe-sched** Forwarding class **no-loss**, scheduler **nl-sched** Scheduler map **hpc-map**: Forwarding class **hpc**, scheduler **hpc-sched**
Traffic control profiles	**be-tcp**: scheduler map **be-map**, minimum bandwidth **3.5g**, maximum bandwidth **100%** **gd-tcp**: scheduler map **gd-map**, minimum bandwidth **4.5g**, maximum bandwidth **100%** **hpc-tcp**: scheduler map **hpc-map**, minimum bandwidth **2g**, maximum bandwidth **100%**
Interfaces	This example configures hierarchical port scheduling on interfaces **xe-0/0/20** and **xe-0/0/21**. Because traffic is bidirectional, you apply the ingress and egress configuration components to both interfaces: • Classifier Name—**hsclassifier1** • Forwarding class sets—**best-effort-pg**, **guar-deliver-pg**, **hpc-pg** • Congestion notification profile—**gd-cnp**

Figure 21 on page 421 shows a block diagram of the configuration components and the configuration flow of the CLI statements used in the example. You can perform the configuration steps in a different sequence if you want.

Figure 21: Hierarchical Port Scheduling Components Block Diagram

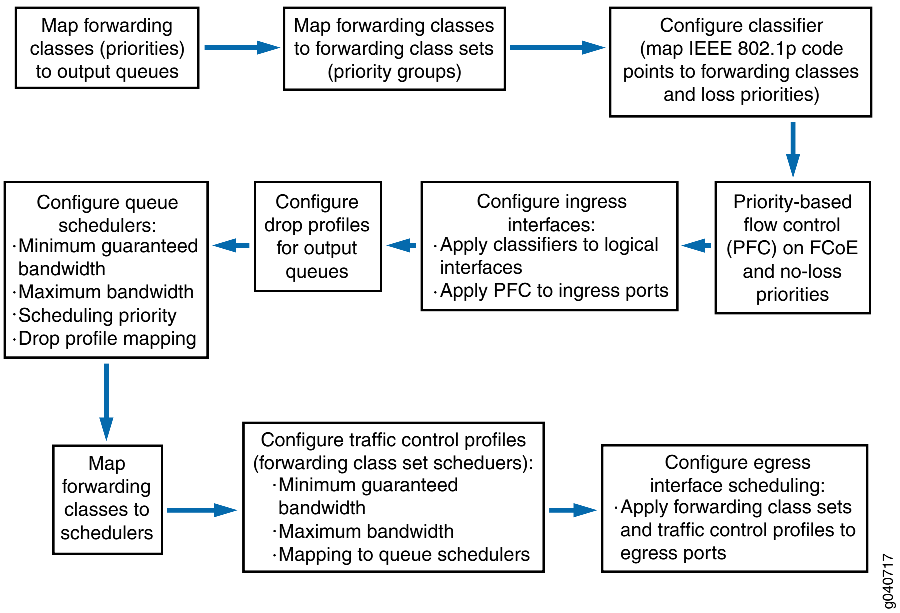
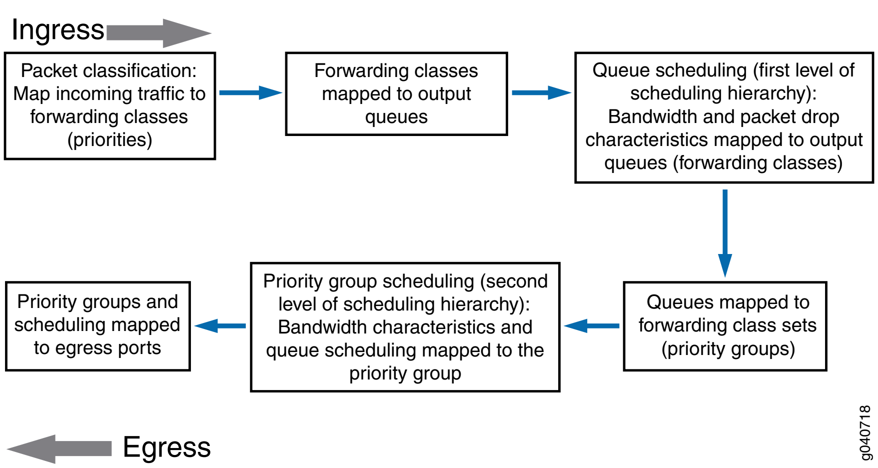


Figure 22 on page 421 shows a block diagram of the hierarchical scheduling packet flow from ingress to egress.

Figure 22: Hierarchical Port Scheduling Packet Flow Block Diagram



Configuration

CLI Quick Configuration

To quickly configure hierarchical port scheduling on systems that support lossless transport, 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 class-of-service] hierarchy level:

```
[edit class-of-service]
```

set forwarding-classes class best-effort queue-num 0

set forwarding-classes class be2 queue-num 1

set forwarding-classes class hpc queue-num 5

set forwarding-classes class network-control queue-num 7

set forwarding-class-sets best-effort-pg class best-effort

set forwarding-class-sets best-effort-pg class be2

set forwarding-class-sets best-effort-pg class network-control

set forwarding-class-sets guar-delivery-pg class fcoe

set forwarding-class-sets guar-delivery-pg class no-loss

set forwarding-class-sets hpc-pg class hpc

set classifiers ieee-802.1 hsclassifier1 forwarding-class best-effort loss-priority low code-points 000

set classifiers ieee-802.1 hsclassifier1 forwarding-class be2 loss-priority high code-points 001

set classifiers ieee-802.1 hsclassifier1 forwarding-class fcoe loss-priority low code-points 011

set classifiers ieee-802.1 hsclassifier1 forwarding-class no-loss loss-priority low code-points 100

set classifiers ieee-802.1 hsclassifier1 forwarding-class hpc loss-priority low code-points 101

set classifiers ieee-802.1 hsclassifier1 forwarding-class network-control loss-priority low code-points 110

set congestion-notification-profile gd-cnp input ieee-802.1 code-point 011 pfc

set congestion-notification-profile gd-cnp input ieee-802.1 code-point 100 pfc

set interfaces xe-0/0/20 unit 0 classifiers ieee-802.1 hsclassifier1

set interfaces xe-0/0/21 unit 0 classifiers ieee-802.1 hsclassifier1

set interfaces xe-0/0/20 congestion-notification-profile gd-cnp

set interfaces xe-0/0/21 congestion-notification-profile gd-cnp

set drop-profiles dp-be-low interpolate fill-level 25 fill-level 50 drop-probability 0 drop-probability 80

set drop-profiles dp-be-high interpolate fill-level 10 fill-level 40 drop-probability 0 drop-probability 100

set drop-profiles dp-nc interpolate fill-level 80 fill-level 100 drop-probability 0 drop-probability 100

set drop-profiles dp-hpc interpolate fill-level 75 fill-level 90 drop-probability 0 drop-probability 75

set schedulers be-sched priority low transmit-rate 3g

set schedulers be-sched shaping-rate percent 100

set schedulers be-sched drop-profile-map loss-priority low protocol any drop-profile dp-be-low

set schedulers be-sched drop-profile-map loss-priority high protocol any drop-profile dp-be-high

set schedulers fcoe-sched priority low transmit-rate 2500m

set schedulers fcoe-sched shaping-rate percent 100

set schedulers hpc-sched priority low transmit-rate 2g

set schedulers hpc-sched shaping-rate percent 100
set schedulers hpc-sched drop-profile-map loss-priority low protocol any drop-profile dp-hpc
set schedulers nc-sched priority low transmit-rate 500m
set schedulers nc-sched shaping-rate percent 100
set schedulers nc-sched drop-profile-map loss-priority low protocol any drop-profile dp-nc
set schedulers nl-sched priority low transmit-rate 2g
set schedulers nl-sched shaping-rate percent 100
set scheduler-maps be-map forwarding-class best-effort scheduler be-sched
set scheduler-maps be-map forwarding-class be2 scheduler be-sched
set scheduler-maps be-map forwarding-class network-control scheduler nc-sched
set scheduler-maps gd-map forwarding-class fcoe scheduler fcoe-sched
set scheduler-maps gd-map forwarding-class no-loss scheduler nl-sched
set scheduler-maps hpc-map forwarding-class hpc scheduler hpc-sched
set traffic-control-profiles be-tcp scheduler-map be-map guaranteed-rate 3500m
set traffic-control-profiles be-tcp shaping-rate percent 100
set traffic-control-profiles gd-tcp scheduler-map gd-map guaranteed-rate 4500m
set traffic-control-profiles gd-tcp shaping-rate percent 100
set traffic-control-profiles hpc-tcp scheduler-map hpc-map guaranteed-rate 2g
set traffic-control-profiles hpc-tcp shaping-rate percent 100
set interfaces xe-0/0/20 forwarding-class-set best-effort-pg output-traffic-control-profile be-tcp
set interfaces xe-0/0/20 forwarding-class-set guar-delivery-pg output-traffic-control-profile gd-tcp
set interfaces xe-0/0/20 forwarding-class-set hpc-pg output-traffic-control-profile hpc-tcp
set interfaces xe-0/0/21 forwarding-class-set best-effort-pg output-traffic-control-profile be-tcp
set interfaces xe-0/0/21 forwarding-class-set guar-delivery-pg output-traffic-control-profile gd-tcp
set interfaces xe-0/0/21 forwarding-class-set hpc-pg output-traffic-control-profile hpc-tcp

OCX Series Switches

Because OCX Series switches do not support lossless transport, the following subset of the configuration eliminates the lossless configuration elements and provides hierarchical port scheduling for the best-effort, be2, hpc, and network-control forwarding classes. In addition, on OCX Series switches, you would probably use DSCP classifiers and code points instead of IEEE classifiers and code points. To quickly configure hierarchical port scheduling on an OCX Series switch, 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 class-of-service] hierarchy level:

```
[edit class-of-service]
```

set forwarding-classes class best-effort queue-num 0
set forwarding-classes class be2 queue-num 1
set forwarding-classes class hpc queue-num 5
set forwarding-classes class network-control queue-num 7
set forwarding-class-sets best-effort-pg class best-effort
set forwarding-class-sets best-effort-pg class be2
set forwarding-class-sets best-effort-pg class network-control
set forwarding-class-sets hpc-pg class hpc
set classifiers ieee-802.1 hsclassifier1 forwarding-class best-effort loss-priority low code-points 000
set classifiers ieee-802.1 hsclassifier1 forwarding-class be2 loss-priority high code-points 001
set classifiers ieee-802.1 hsclassifier1 forwarding-class hpc loss-priority low code-points 101
set classifiers ieee-802.1 hsclassifier1 forwarding-class network-control loss-priority low code-points 110
set interfaces xe-0/0/20 unit 0 classifiers ieee-802.1 hsclassifier1
set interfaces xe-0/0/21 unit 0 classifiers ieee-802.1 hsclassifier1
set drop-profiles dp-be-low interpolate fill-level 25 fill-level 50 drop-probability 0 drop-probability 80
set drop-profiles dp-be-high interpolate fill-level 10 fill-level 40 drop-probability 0 drop-probability 100
set drop-profiles dp-nc interpolate fill-level 80 fill-level 100 drop-probability 0 drop-probability 100
set drop-profiles dp-hpc interpolate fill-level 75 fill-level 90 drop-probability 0 drop-probability 75
set schedulers be-sched priority low transmit-rate 3g
set schedulers be-sched shaping-rate percent 100
set schedulers be-sched drop-profile-map loss-priority low protocol any drop-profile dp-be-low
set schedulers be-sched drop-profile-map loss-priority high protocol any drop-profile dp-be-high
set schedulers hpc-sched priority low transmit-rate 2g
set schedulers hpc-sched shaping-rate percent 100
set schedulers hpc-sched drop-profile-map loss-priority low protocol any drop-profile dp-hpc
set schedulers nc-sched priority low transmit-rate 500m
set schedulers nc-sched shaping-rate percent 100
set schedulers nc-sched drop-profile-map loss-priority low protocol any drop-profile dp-nc
set scheduler-maps be-map forwarding-class best-effort scheduler be-sched
set scheduler-maps be-map forwarding-class be2 scheduler be-sched
set scheduler-maps be-map forwarding-class network-control scheduler nc-sched
set scheduler-maps hpc-map forwarding-class hpc scheduler hpc-sched
set traffic-control-profiles be-tcp scheduler-map be-map guaranteed-rate 3500m

set traffic-control-profiles be-tcp shaping-rate percent 100

set traffic-control-profiles hpc-tcp scheduler-map hpc-map guaranteed-rate 2g

set traffic-control-profiles hpc-tcp shaping-rate percent 100

set interfaces xe-0/0/20 forwarding-class-set best-effort-pg output-traffic-control-profile be-tcp

set interfaces xe-0/0/20 forwarding-class-set hpc-pg output-traffic-control-profile hpc-tcp

set interfaces xe-0/0/21 forwarding-class-set best-effort-pg output-traffic-control-profile be-tcp

set interfaces xe-0/0/21 forwarding-class-set hpc-pg output-traffic-control-profile hpc-tcp

Step-by-Step Procedure

To perform a step-by-step configuration of the forwarding classes (priorities), forwarding class sets (priority groups), classifiers, queue schedulers, PFC, traffic control profiles, and interfaces to set up hierarchical port scheduling (ETS):

1. Configure the forwarding classes (priorities) and map them to unicast output queues (do not explicitly map the **fcoe** and **no-loss** forwarding classes to output queues; use the default configuration):

```
[edit class-of-service]
user@switch# set forwarding-classes class best-effort queue-num 0
user@switch# set forwarding-classes class be2 queue-num 1
user@switch# set forwarding-classes class hpc queue-num 5
user@switch# set forwarding-classes class network-control queue-num 7
```

2. Configure forwarding class sets (priority groups) to group forwarding classes (priorities) that require similar CoS treatment:

```
[edit class-of-service]
user@switch# set forwarding-class-sets best-effort-pg class best-effort
user@switch# set forwarding-class-sets best-effort-pg class be2
user@switch# set forwarding-class-sets best-effort-pg class network-control
user@switch# set forwarding-class-sets guar-delivery-pg class fcoe
user@switch# set forwarding-class-sets guar-delivery-pg class no-loss
user@switch# set forwarding-class-sets hpc-pg class hpc
```

> **NOTE:** On OCX Series switches, you would not configure the **guar-delivery-pg** forwarding class set for lossless traffic.

3. Configure a classifier to set the loss priority and IEEE 802.1 code points assigned to each forwarding class at the ingress:

```
[edit class-of-service]
user@switch# set classifiers ieee-802.1 hsclassifier1 forwarding-class best-effort loss-priority low code-points 000
user@switch# set classifiers ieee-802.1 hsclassifier1 forwarding-class be2 loss-priority high code-points 001
user@switch# set classifiers ieee-802.1 hsclassifier1 forwarding-class fcoe loss-priority low code-points 011
user@switch# set classifiers ieee-802.1 hsclassifier1 forwarding-class no-loss loss-priority low code-points 100
user@switch# set classifiers ieee-802.1 hsclassifier1 forwarding-class hpc loss-priority low code-points 101
user@switch# set classifiers ieee-802.1 hsclassifier1 forwarding-class network-control loss-priority low code-points 110
```

> **NOTE:** On OCX Series switches, you would not configure the **fcoe** and **no-loss** portions of the classifier.

4. Configure a congestion notification profile to enable PFC on the FCoE and no-loss queue IEEE 802.1 code points:

```
[edit class-of-service]
user@switch# set congestion-notification-profile gd-cnp input ieee-802.1 code-point 011 pfc
user@switch# set congestion-notification-profile gd-cnp input ieee-802.1 code-point 100 pfc
```

> **NOTE:** This step does not apply to OCX Series switches, which do not support PFC.

5. Assign the classifier to the interfaces:

```
[edit class-of-service]
user@switch# set interfaces xe-0/0/20 unit 0 classifiers ieee-802.1 hsclassifier1
user@switch# set interfaces xe-0/0/21 unit 0 classifiers ieee-802.1 hsclassifier1
```

6. Apply the PFC configuration to the interfaces:

```
[edit class-of-service]
```

user@switch# **set interfaces xe-0/0/20 congestion-notification-profile gd-cnp**

user@switch# **set interfaces xe-0/0/21 congestion-notification-profile gd-cnp**

> **NOTE:** This step does not apply to OCX Series switches, which do not support PFC.

7. Configure the drop profile for the best-effort low loss-priority queue:

   `[edit class-of-service]`

   user@switch# **set drop-profiles dp-be-low interpolate fill-level 25 fill-level 50 drop-probability 0 drop-probability 80**

8. Configure the drop profile for the best-effort high loss-priority queue:

   `[edit class-of-service]`

   user@switch# **set drop-profiles dp-be-high interpolate fill-level 10 fill-level 40 drop-probability 0 drop-probability 100**

9. Configure the drop profile for the network-control queue:

   `[edit class-of-service]`

   user@switch# **set drop-profiles dp-nc interpolate fill-level 80 fill-level 100 drop-probability 0 drop-probability 100**

10. Configure the drop profile for the high-performance computing queue:

   `[edit class-of-service]`

   user@switch# **set drop-profiles dp-hpc interpolate fill-level 75 fill-level 90 drop-probability 0 drop-probability 75**

11. Define the minimum guaranteed bandwidth, priority, maximum bandwidth, and drop profiles for the best-effort queue:

   `[edit class-of-service]`

   user@switch# **set schedulers be-sched priority low transmit-rate 3g**

   user@switch# **set schedulers be-sched shaping-rate percent 100**

   user@switch# **set schedulers be-sched drop-profile-map loss-priority low protocol any drop-profile dp-be-low** user@switch# **set schedulers be-sched drop-profile-map loss-priority high protocol any drop-profile dp-be-high**

12. Define the minimum guaranteed bandwidth, priority, and maximum bandwidth for the FCoE queue:

    ```
    [edit class-of-service]
    ```
    user@switch# **set schedulers fcoe-sched priority low transmit-rate 2500m**
    user@switch# **set schedulers fcoe-sched shaping-rate percent 100**

    > **NOTE:** This step does not apply to OCX Series switches, which do not support lossless transport.

13. Define the minimum guaranteed bandwidth, priority, maximum bandwidth, and drop profile for the high-performance computing queue:

    ```
    [edit class-of-service]
    ```
    user@switch# **set schedulers hpc-sched priority low transmit-rate 2g**
    user@switch# **set schedulers hpc-sched shaping-rate percent 100**
    user@switch# **set schedulers hpc-sched drop-profile-map loss-priority low protocol any drop-profile dp-hpc**

14. Define the minimum guaranteed bandwidth, priority, maximum bandwidth, and drop profile for the network-control queue:

    ```
    [edit class-of-service]
    ```
    user@switch# **set schedulers nc-sched priority low transmit-rate 500m**
    user@switch# **set schedulers nc-sched shaping-rate percent 100**
    user@switch# **set schedulers nc-sched drop-profile-map loss-priority low protocol any drop-profile dp-nc**

15. Define the minimum guaranteed bandwidth, priority, and maximum bandwidth for the no-loss queue:

    ```
    [edit class-of-service]
    ```
    user@switch# **set schedulers nl-sched priority low transmit-rate 2g**
    user@switch# **set schedulers nl-sched shaping-rate percent 100**

NOTE: This step does not apply to OCX Series switches, which do not support lossless transport.

16. Map the schedulers to the appropriate forwarding classes (queues):

    ```
    [edit class-of-service]
    user@switch# set scheduler-maps be-map forwarding-class best-effort scheduler be-sched
    user@switch# set scheduler-maps be-map forwarding-class be2 scheduler be-sched
    user@switch# set scheduler-maps be-map forwarding-class network-control scheduler nc-sched
    user@switch# set scheduler-maps gd-map forwarding-class fcoe scheduler fcoe-sched
    user@switch# set scheduler-maps gd-map forwarding-class no-loss scheduler nl-sched
    user@switch# set scheduler-maps hpc-map forwarding-class hpc scheduler hpc-sched
    ```

    NOTE: On OCX Series switches, because lossless transport is not supported, you would not configure the **gd-map** scheduler map.

17. Define the traffic control profile for the best-effort priority group (queue scheduler to mapping, minimum guaranteed bandwidth, and maximum bandwidth):

    ```
    [edit class-of-service]
    user@switch# set traffic-control-profiles be-tcp scheduler-map be-map guaranteed-rate 3500m
    user@switch# set traffic-control-profiles be-tcp shaping-rate percent 100
    ```

18. Define the traffic control profile for the guaranteed delivery priority group (queue to scheduler mapping, minimum guaranteed bandwidth, and maximum bandwidth):

    ```
    [edit class-of-service]
    user@switch# set traffic-control-profiles gd-tcp scheduler-map gd-map guaranteed-rate 4500m
    user@switch# set traffic-control-profiles gd-tcp shaping-rate percent 100
    ```

    NOTE: This step does not apply to OCX Series switches, which do not support lossless transport.

19. Define the traffic control profile for the high-performance computing priority group (queue to scheduler mapping, minimum guaranteed bandwidth, and maximum bandwidth):

    ```
    [edit class-of-service]
    user@switch# set traffic-control-profiles hpc-tcp scheduler-map hpc-map guaranteed-rate 2g
    user@switch# set traffic-control-profiles hpc-tcp shaping-rate percent 100
    ```

20. Apply the three priority groups (forwarding class sets) and the appropriate traffic control profiles to the egress ports:

    ```
    [edit class-of-service]
    user@switch# set interfaces xe-0/0/20 forwarding-class-set best-effort-pg
    output-traffic-control-profile be-tcp
    user@switch# set interfaces xe-0/0/20 forwarding-class-set guar-delivery-pg
    output-traffic-control-profile gd-tcp
    user@switch# set interfaces xe-0/0/20 forwarding-class-set hpc-pg output-traffic-control-profile
    hpc-tcp
    user@switch# set interfaces xe-0/0/21 forwarding-class-set best-effort-pg
    output-traffic-control-profile be-tcp
    user@switch# set interfaces xe-0/0/21 forwarding-class-set guar-delivery-pg
    output-traffic-control-profile gd-tcp
    user@switch# set interfaces xe-0/0/21 forwarding-class-set hpc-pg output-traffic-control-profile
    hpc-tcp
    ```

    NOTE: Because OCX Series switches do not support lossless transport, on OCX Series switches, you would not apply the **guar-deliver-pg** forwarding class set and the **gd-tcp** traffic control profile to interfaces.

Results

Display the results of the configuration (the system shows only the explicitly configured parameters; it does not show default parameters such as the **fcoe** and **no-loss** lossless forwarding classes). On OCX Series switches, you would not see the lossless configuration components in the output:

```
user@switch> show configuration class-of-service
classifiers {
  ieee-802.1 hsclassifier1 {
```

```
    forwarding-class best-effort {
        loss-priority low code-points 000;
    }
    forwarding-class be2 {
        loss-priority high code-points 001;
    }
    forwarding-class fcoe {
        loss-priority low code-points 011;
    }
    forwarding-class no-loss {
        loss-priority low code-points 100;
    }
    forwarding-class hpc {
        loss-priority low code-points 101;
    }
    forwarding-class network-control {
        loss-priority low code-points 110;
    }
}
drop-profiles {
    dp-be-low {
        interpolate {
            fill-level [ 25 50 ];
            drop-probability [ 0 80 ];
        }
    }
    dp-be-high {
        interpolate {
            fill-level [ 10 40 ];
            drop-probability [ 0 100 ];
        }
    }
    dp-hpc {
        interpolate {
            fill-level [ 75 90 ];
            drop-probability [ 0 75 ];
        }
    }
    dp-nc {
        interpolate {
            fill-level [ 80 100 ];
            drop-probability [ 0 100 ];
        }
    }
```

```
    }
    forwarding-classes {
        class best-effort queue-num 0;
        class be2 queue-num 1;
        class hpc queue-num 5;
        class network-control queue-num 7;
    }
    traffic-control-profiles {
        be-tcp {
            scheduler-map be-map;
            shaping-rate percent 100;
            guaranteed-rate 3500000000;
        }
        gd-tcp {
            scheduler-map gd-map;
            shaping-rate percent 100;
            guaranteed-rate 4500000000;
        }
        hpc-tcp {
            scheduler-map hpc-map;
            shaping-rate percent 100;
            guaranteed-rate 2g;
        }
    }
    forwarding-class-sets {
        guar-delivery-pg {
            class fcoe;
            class no-loss;
        }
        best-effort-pg {
            class best-effort;
            class be2;
            class network-control;
        }
        hpc-pg {
            class hpc;
        }
    }
    congestion-notification-profile {
        gd-cnp {
            input {
                ieee-802.1 {
                    code-point 011 {
                        pfc;
```

```
                }
                code-point 100 {
                    pfc;
                }
            }
        }
    }
}
interfaces {
    xe-0/0/20 {
        forwarding-class-set {
            best-effort-pg {
                output-traffic-control-profile be-tcp;
            }
            guar-delivery-pg {
                output-traffic-control-profile gd-tcp;
            }
            hpc-pg {
                output-traffic-control-profile hpc-tcp;
            }
        }
        congestion-notification-profile gd-cnp;
        unit 0 {
            classifiers {
                ieee-802.1 hsclassifier1;
            }
        }
    }
    xe-0/0/21 {
        forwarding-class-set {
            best-effort-pg {
                output-traffic-control-profile be-tcp;
            }
            guar-delivery-pg {
                output-traffic-control-profile gd-tcp;
            }
            hpc-pg {
                output-traffic-control-profile hpc-tcp;
            }
        }
        congestion-notification-profile gd-cnp;
        unit 0 {
            classifiers {
                ieee-802.1 hsclassifier1;
```

```
            }
        }
    }
}
scheduler-maps {
    be-map {
        forwarding-class best-effort scheduler be-sched;
        forwarding-class network-control scheduler nc-sched;
        forwarding-class be2 scheduler be-sched;
    }
    gd-map {
        forwarding-class fcoe scheduler fcoe-sched;
        forwarding-class no-loss scheduler nl-sched;
    }
    hpc-map {
        forwarding-class hpc scheduler hpc-sched;
    }
}
schedulers {
    be-sched {
        transmit-rate 3g;
        shaping-rate percent 100;
        priority low;
        drop-profile-map loss-priority low protocol any drop-profile dp-be-low;
        drop-profile-map loss-priority high protocol any drop-profile dp-be-high;
    }
    fcoe-sched {
        transmit-rate 2500000000;
        shaping-rate percent 100;
        priority low;
    }
    hpc-sched {
        transmit-rate 2g;
        shaping-rate percent 100;
        priority low;
        drop-profile-map loss-priority low protocol any drop-profile dp-hpc;
    }
    nc-sched {
        transmit-rate 500m;
        shaping-rate percent 100;
        priority low;
        drop-profile-map loss-priority low protocol any drop-profile dp-nc;
    }
    nl-sched {
```

```
        transmit-rate 2g;
        shaping-rate percent 100;
        priority low;
      }
    }
```

TIP: To quickly configure the interfaces, issue the **load merge terminal** command, and then copy the hierarchy and paste it into the switch terminal window.

Verification

IN THIS SECTION

NOTE: The verification output is based on the full example configuration. On OCX Series switches, you do not see lossless configuration components in the output. Comments about lossless configuration components do not apply to OCX Series switches.

To verify that you created the hierarchical port scheduling components and they are operating properly, perform these tasks:

Verifying the Forwarding Classes (Priorities)

Purpose

Verify that you created the forwarding classes and mapped them to the correct queues. (The system shows only the explicitly configured forwarding classes. It does not show default forwarding classes such as **fcoe** and **no-loss**.)

Action

List the forwarding classes using the operational mode command **show class-of-service forwarding-class**:

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

```
Forwarding class                        ID      Queue  Policing priority  No-Loss
  best-effort                           0         0         normal        Disabled

  be2                                   1         3         normal        Disabled

  hpc                                   2         4         normal        Disabled

  network-control                       3         7         normal        Disabled

  mcast                                 8         8         normal        Disabled
```

Meaning

The **show class-of-service forwarding-class** command lists all of the configured forwarding classes, the internal identification number of each forwarding class, the queues that are mapped to the forwarding classes, the policing priority, and whether the forwarding class is lossless (no-loss packet drop attribute enabled) or lossy forwarding class (no-loss packet drop attribute disabled). The command output shows that:

- Forwarding class **best-effort** maps to queue **0** and is lossy
- Forwarding class **be2** maps to queue **1** and is lossy
- Forwarding class **hpc** maps to queue **5** and is lossy
- Forwarding class **network-control** maps to queue **7** and is lossy

In addition, the command lists the default multicast (multidestination) forwarding class and the default queue to which it is mapped.

Verifying the Forwarding Class Sets (Priority Groups)

Purpose

Verify that you created the priority groups and that the correct priorities (forwarding classes) belong to the appropriate priority group.

Action

List the forwarding class sets using the operational mode command **show class-of-service forwarding-class-set**:

user@switch> **show class-of-service forwarding-class-set**

```
Forwarding class set: best-effort-pg, Type: normal-type, Forwarding class set
index: 19907
  Forwarding class                       Index
  best-effort                              0
  be2                                      1
  network-control                          5

Forwarding class set: guar-delivery-pg, Type: normal-type, Forwarding class set
index: 43700
  Forwarding class                       Index
  fcoe                                     2
  no-loss                                  3

Forwarding class set: hpc-pg, Type: normal-type, Forwarding class set index: 60758

  Forwarding class                       Index
  hpc                                      4
```

Meaning

The **show class-of-service forwarding-class-set** command lists all of the configured forwarding class sets (priority groups), the forwarding classes (priorities) that belong to each priority group, and the internal index number of each priority group. The command output shows that:

- The forwarding class set **best-effort-pg** includes the forwarding classes **best-effort**, **be2**, and **network-control**.
- The forwarding class set **guar-delivery-pg** includes the forwarding classes **fcoe** and **no-loss**.
- The forwarding class set **hpc-pg** includes the forwarding class **hpc**.

Verifying the Classifier

Purpose

Verify that the classifier maps forwarding classes to the correct IEEE 802.1p code points and packet loss priorities.

Action

List the classifier configured for hierarchical port scheduling using the operational mode command **show class-of-service classifier name hsclassifier1**:

```
user@switch> show class-of-service classifier name hsclassifier1
```

```
Classifier: hsclassifier1, Code point type: ieee-802.1, Index: 43607
  Code point         Forwarding class                    Loss priority
  000                best-effort                         low
  001                be2                                 high
  011                fcoe                                low
  100                no-loss                             low
  101                hpc                                 low
  110                network-control                     low
```

Meaning

The **show class-of-service classifier name hsclassifier1** command lists all of the IEEE 802.1p code points and the loss priorities mapped to all of the forwarding classes in the classifier. The command output shows that the forwarding classes **best-effort**, **be2**, **no-loss**, **fcoe**, **hpc**, and **network-control** have been created and mapped to IEEE 802.1p code points and loss priorities.

Verifying Priority-Based Flow Control

Purpose

Verify that PFC is enabled on the correct priorities for lossless transport.

Action

List the congestion notification profiles using the operational mode command **show class-of-service congestion-notification**:

```
user@switch> show class-of-service congestion-notification
```

```
Type: Input, Name: gd-cnp, Index: 51687
Cable Length: 100 m
  Priority     PFC          MRU
  000          Disabled
  001          Disabled
  010          Disabled
  011          Enabled      2500
  100          Enabled      2500
  101          Disabled
  110          Disabled
  111          Disabled
Type: Output
```

```
Priority     Flow-Control-Queues
000
             0
001
             1
010
             2
011
             3
100
             4
101
             5
110
             6
111
             7
```

Meaning

The **show class-of-service congestion-notification** command lists all of the congestion notification profiles and the IEEE 802.1p code points with PFC enabled. The command output shows that PFC is enabled for code points **011** (**fcoe** priority and queue) and **100** (**no-loss** priority and queue) for the **gd-cnp** congestion notification profile.

The command also shows the default cable length (100 meters), the default maximum receive unit (2500 bytes), and the default mapping of priorities to output queues because this example does not include configuring these options.

Verifying the Output Queue Schedulers

Purpose

Verify that you created the output queue schedulers with the correct bandwidth parameters and priorities, mapped to the correct queues, and mapped to the correct drop profiles.

Action

List the scheduler maps using the operational mode command **show class-of-service scheduler-map**:

`user@switch>` **show class-of-service scheduler-map**

```
Scheduler map: be-map, Index: 64023

  Scheduler: be-sched, Forwarding class: best-effort, Index: 13005
    Transmit rate: 3000000000 bps, Rate Limit: none, Buffer size: remainder,
```

```
    Buffer Limit: none, Priority: low
    Excess Priority: unspecified
    Shaping rate: 100 percent,
    drop-profile-map-set-type: mark
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any         55387    dp-be-low
      Medium high     any             1    <default-drop-profile>
      High            any          4369    dp-be-high

  Scheduler: be-sched, Forwarding class: be2, Index: 13005
    Transmit rate: 3000000000 bps, Rate Limit: none, Buffer size: remainder,
    Buffer Limit: none, Priority: low
    Excess Priority: unspecified
    Shaping rate: 100 percent,
    drop-profile-map-set-type: mark
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any         55387    dp-be-low
      Medium high     any             1    <default-drop-profile>
      High            any          4369    dp-be-high

  Scheduler: nc-sched, Forwarding class: network-control, Index: 45740
    Transmit rate: 500000000 bps, Rate Limit: none, Buffer size: remainder,
    Buffer Limit: none, Priority: low
    Excess Priority: unspecified
    Shaping rate: 100 percent,
    drop-profile-map-set-type: mark
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any         44207    dp-nc
      Medium high     any             1    <default-drop-profile>
      High            any             1    <default-drop-profile>

Scheduler map: gd-map, Index: 61447

  Scheduler: fcoe-sched, Forwarding class: fcoe, Index: 37289
    Transmit rate: 2500000000 bps, Rate Limit: none, Buffer size: remainder,
    Buffer Limit: none, Priority: low
    Excess Priority: unspecified
    Shaping rate: 100 percent,
    drop-profile-map-set-type: mark
    Drop profiles:
      Loss priority   Protocol    Index    Name
```

```
      Low              any         44207    <default-drop-profile>
      Medium high      any             1    <default-drop-profile>
      High             any             1    <default-drop-profile>

  Scheduler: nl-sched, Forwarding class: no-loss, Index: 29359
    Transmit rate: 2000000000 bps, Rate Limit: none, Buffer size: remainder,
    Buffer Limit: none, Priority: low
    Excess Priority: unspecified
    Shaping rate: 100 percent,
    drop-profile-map-set-type: mark
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any         44207    <default-drop-profile>
      Medium high     any             1    <default-drop-profile>
      High            any             1    <default-drop-profile>

Scheduler map: hpc-map, Index: 56941

  Scheduler: hpc-sched, Forwarding class: hpc, Index: 55900
    Transmit rate: 2000000000 bps, Rate Limit: none, Buffer size: remainder,
    Buffer Limit: none, Priority: low
    Excess Priority: unspecified
    Shaping rate: 100 percent,
    drop-profile-map-set-type: mark
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any         57716    dp-hpc
      Medium high     any             1    <default-drop-profile>
      High            any             1    <default-drop-profile>
```

Meaning

The **show class-of-service scheduler-map** command lists all of the configured scheduler maps. For each scheduler map, the command output includes:

- The name of the scheduler map (**scheduler-map** field)
- The name of the scheduler (**scheduler** field)
- The forwarding classes mapped to the scheduler (**forwarding-class** field)
- The minimum guaranteed queue bandwidth (**transmit-rate** field)
- The scheduling priority (**priority** field)
- The maximum bandwidth in the priority group the queue can consume (**shaping-rate** field)
- The drop profile loss priority (**loss priority** field) for each drop profile name (**name** field)

The command output shows that:

- The scheduler map **be-map** was created and has these properties:
  - There are two schedulers, **be-sched** and **nc-sched**.
  - The scheduler **be-sched** has two forwarding classes, **best-effort** and **be2**.
  - Scheduler **be-sched** forwarding classes **best-effort** and **be2** share a minimum guaranteed bandwidth of **3,000,000,000 bps**, can consume a maximum of **100 percent** of the priority group bandwidth, and use the drop profile **dp-be-low** for low loss-priority traffic, the default drop profile for medium-high loss-priority traffic, and the drop profile **dp-be-high** for high loss-priority traffic.
  - The scheduler **nc-sched** has one forwarding class, **network-control**.
  - The **network-control** forwarding class has a minimum guaranteed bandwidth of **500,000,000 bps**, can consume a maximum of **100 percent** of the priority group bandwidth, and uses the drop profile **dp-nc** for low loss-priority traffic and the default drop profile for medium-high and high loss priority traffic.
- The scheduler map **gd-map** was created and has these properties:
  - There are two schedulers, **fcoe-sched** and **nl-sched**.
  - The scheduler **fcoe-sched** has one forwarding class, **fcoe**.
  - The **fcoe** forwarding class has a minimum guaranteed bandwidth of **2,500,000,000 bps**, and can consume a maximum of **100 percent** of the priority group bandwidth.
  - The scheduler **nl-sched** has one forwarding class, **no-loss**.
  - The **no-loss** forwarding class has a minimum guaranteed bandwidth of **2,000,000,000 bps**, and can consume a maximum of **100 percent** of the priority group bandwidth.
- The scheduler map **hpc-map** was created and has these properties:
  - There is one scheduler, **hpc-sched**.
  - The scheduler **hpc-sched** has one forwarding class, **hpc**.
  - The **hpc** forwarding class has a minimum guaranteed bandwidth of **2,000,000,000 bps**, can consume a maximum of **100 percent** of the priority group bandwidth, and uses the drop profile **dp-hpc** for low loss-priority traffic and the default drop profile for medium-high and high loss-priority traffic.

Verifying the Drop Profiles

Purpose

Verify that you created the drop profiles **dp-be-high**, **dp-be-low**, **dp-hpc**, and **dp-nc** with the correct fill levels and drop probabilities.

Action

List the drop profiles using the operational mode command **show configuration class-of-service drop-profiles**:

user@switch> **show configuration class-of-service drop-profiles**

```
dp-be-low {
    interpolate {
        fill-level [ 25 50 ];
        drop-probability [ 0 80 ];
    }
}
dp-be-high {
    interpolate {
        fill-level [ 10 40 ];
        drop-probability [ 0 100 ];
    }
}
dp-hpc {
    interpolate {
        fill-level [ 75 90 ];
        drop-probability [ 0 75 ];
    }
}
dp-nc {
    interpolate {
        fill-level [ 80 100 ];
        drop-probability [ 0 100 ];
    }
```

Meaning

The **show configuration class-of-service drop-profiles** command lists the drop profiles and their properties. The command output shows that there are four drop profiles configured, **dp-be-high**, **dp-be-low**, **dp-hpc**, and **dp-nc**. The output also shows that:

- For **dp-be-low**, the drop start point (the first fill level) is when the queue is 25 percent filled, the drop end point (the second fill level) occurs when the queue is 50 percent filled, and the drop probability at the drop end point is 80 percent.
- For **dp-be-high**, the drop start point (the first fill level) is when the queue is 10 percent filled, the drop end point (the second fill level) occurs when the queue is 40 percent filled, and the drop probability at the drop end point is 100 percent.

- For **dp-hpc**, the drop start point (the first fill level) is when the queue is 75 percent filled, the drop end point (the second fill level) occurs when the queue is 90 percent filled, and the drop probability at the drop end point is 75 percent.
- For **dp-nc**, the drop start point (the first fill level) is when the queue is 80 percent filled, the drop end point (the second fill level) occurs when the queue is 100 percent filled, and the drop probability at the drop end point is 100 percent.

Verifying the Priority Group Output Schedulers (Traffic Control Profiles)

Purpose

Verify that you created the traffic control profiles **be-tcp**, **gd-tcp**, and **hpc-tcp** with the correct bandwidth parameters and scheduler mapping.

Action

List the traffic control profiles using the operational mode command **show class-of-service traffic-control-profile**:

```
user@switch> show class-of-service traffic-control-profile
```

```
Traffic control profile: be-tcp, Index: 40535
  Shaping rate: 100 percent
  Scheduler map: be-map
  Guaranteed rate: 3500000000

Traffic control profile: gd-tcp, Index: 37959
  Shaping rate: 100 percent
  Scheduler map: gd-map
  Guaranteed rate: 4500000000

Traffic control profile: hpc-tcp, Index: 47661
  Shaping rate: 100 percent
  Scheduler map: hpc-map
  Guaranteed rate: 2000000000
```

Meaning

The **show class-of-service traffic-control-profile** command lists all of the configured traffic control profiles. For each traffic control profile, the command output includes:

- The name of the traffic control profile (**traffic-control-profile**)
- The maximum port bandwidth the priority group can consume (**shaping-rate**)
- The scheduler map associated with the traffic control profile (**scheduler-map**)
- The minimum guaranteed priority group port bandwidth (**guaranteed-rate**)

The command output shows that:

- The traffic control profile **be-tcp** can consume a maximum of **100 percent** of the port bandwidth, is associated with the scheduler map **be-map**, and has a minimum guaranteed bandwidth of **3,500,000,000 bps**.
- The traffic control profile **gd-tcp** can consume a maximum of **100 percent** of the port bandwidth, is associated with the scheduler map **gd-map**, and has a minimum guaranteed bandwidth of **4,500,000,000 bps**.
- The traffic control profile **hpc-tcp** can consume a maximum of **100 percent** of the port bandwidth, is associated with the scheduler map **hpc-map**, and has a minimum guaranteed bandwidth of **2,000,000,000 bps**.

Verifying the Interface Configuration

Purpose

Verify that the classifier, the congestion notification profile, and the forwarding class sets are configured on interfaces **xe-0/0/20** and **xe-0/0/21**.

Action

List the interfaces using the operational mode commands **show configuration class-of-service interfaces xe-0/0/20** and **show configuration class-of-service interfaces xe-0/0/21**:

`user@switch>` **show configuration class-of-service interfaces xe-0/0/20**

```
forwarding-class-set {
    best-effort-gp {
        output-traffic-control-profile be-tcp;
    }
    guar-delivery-pg {
        output-traffic-control-profile gd-tcp;
    }
    hpc-pg {
        output-traffic-control-profile hpc-tcp;
    }
}
congestion-notification-profile gd_cnp;
unit 0 {
    classifiers {
        ieee-802.1 hsclassifier1;
    }
}
```

`user@switch>` **show configuration class-of-service interfaces xe-0/0/21**

```
forwarding-class-set {
    best-effort-gp {
        output-traffic-control-profile be-tcp;
    }
    guar-delivery-pg {
        output-traffic-control-profile gd-tcp;
    }
    hpc-pg {
        output-traffic-control-profile hpc-tcp;
    }
}
congestion-notification-profile gd_cnp;
unit 0 {
    classifiers {
        ieee-802.1 hsclassifier1;
    }
}
```

Meaning

The **show configuration class-of-service interfaces *interface-name*** command shows that each interface includes the forwarding class sets **best-effort-pg**, **guar-delivery-pg**, and **hpc-pg**, congestion notification profile **gd-cnp**, and the IEEE 802.1p classifier **hsclassifier1**.

RELATED DOCUMENTATION

Defining CoS BA Classifiers (DSCP, DSCP IPv6, IEEE 802.1p) | 94

Benefits of Configuring CoS Hierarchical Port Scheduling

Assigning CoS Components to Interfaces | 77

Example: Configuring WRED Drop Profiles | 264

Example: Configuring Drop Profile Maps | 271

Example: Configuring Forwarding Classes | 157

Example: Configuring Forwarding Class Sets | 165

Example: Configuring Queue Schedulers | 325

Example: Configuring Queue Scheduling Priority | 336

Example: Configuring Traffic Control Profiles (Priority Group Scheduling) | 385

Example: Configuring Minimum Guaranteed Output Bandwidth | 392

Example: Configuring Maximum Output Bandwidth | 400

Configuring CoS PFC (Congestion Notification Profiles) | 194

Disabling the ETS Recommendation TLV

The enhanced transmission selection (ETS) Recommendation TLV communicates the ETS settings that the switch wants the connected peer interface to use. If the peer interface is "willing," the peer interface changes its configuration to match the configuration in the ETS Recommendation TLV. By default, the switch interfaces send the ETS Recommendation TLV to the peer. The settings communicated are the egress ETS settings defined by configuring hierarchical scheduling on the interface.

We recommend that you use the same ETS settings on the connected peer that you use on the switch interface and that you leave the ETS Recommendation TLV enabled. However, on interfaces that use IEEE DCBX as the DCBX mode, if you want an asymmetric configuration between the switch interface and the connected peer, you can disable the ETS Recommendation TLV.

> **NOTE:** Disabling the ETS Recommendation TLV on interfaces that use DCBX version 1.01 as the DCBX mode has no effect and does not change DCBX behavior.

If you disable the ETS Recommendation TLV, the switch still sends the ETS Configuration TLV to the connected peer. The result is that the connected peer is informed about the switch DCBX ETS configuration, but even if the peer is "willing," the peer does not change its configuration to match the switch configuration. This is asymmetric configuration—the two interfaces can have different parameter values for the ETS attribute.

To disable the ETS Recommendation TLV:

- `[edit protocols dcbx interface interface-name]`
  `user@switch#` **set enhanced-transmission-selection no-recommendation-tlv**

RELATED DOCUMENTATION

Understanding Data Center Bridging Capability Exchange Protocol for EX Series Switches

4

PART

Data Center Bridging and Lossless FCoE

CHAPTER 16

Data Center Bridging

IN THIS CHAPTER

Understanding DCB Features and Requirements

IN THIS SECTION

Data center bridging (DCB) is a set of enhancements to the IEEE 802.1 bridge specifications. DCB modifies and extends Ethernet behavior to support I/O convergence in the data center. I/O convergence includes but is not limited to the transport of Ethernet LAN traffic and Fibre Channel (FC) storage area network (SAN) traffic on the same physical Ethernet network infrastructure.

Video: What is Data Center Bridging?

A converged architecture saves cost by reducing the number of networks and switches required to support both types of traffic, reducing the number of interfaces required, reducing cable complexity, and reducing administration activities.

The Juniper Networks QFX Series and EX4600 switches support the DCB features required to transport converged Ethernet and FC traffic while providing the class-of-service (CoS) and other characteristics FC requires for transmitting storage traffic. To accommodate FC traffic, DCB specifications provide:

- A flow control mechanism called priority-based flow control (PFC, described in IEEE 802.1Qbb) to help provide lossless transport.
- A discovery and exchange protocol for conveying configuration and capabilities among neighbors to ensure consistent configuration across the network, called Data Center Bridging Capability Exchange protocol (DCBX), which is an extension of Link Layer Data Protocol (LLDP, described in IEEE 802.1AB).
- A bandwidth management mechanism called enhanced transmission selection (ETS, described in IEEE 802.1Qaz).
- A congestion management mechanism called quantized congestion notification (QCN, described in IEEE 802.1Qau).

The switch supports the PFC, DCBX, and ETS standards but does not support QCN. The switch also provides the high-bandwidth interfaces (10-Gbps minimum) required to support DCB and converged traffic.

This topic describes the DCB standards and requirements the switch supports:

Lossless Transport

IN THIS SECTION

FC traffic requires lossless transport (defined as no frames dropped because of congestion). Standard Ethernet does not support lossless transport, but the DCB extensions to Ethernet along with proper buffer management enable an Ethernet network to provide the level of class of service (CoS) necessary to transport FC frames encapsulated in Ethernet over an Ethernet network.

This section describes these factors in creating lossless transport over Ethernet:

PFC

PFC is a link-level flow control mechanism similar to Ethernet PAUSE (described in IEEE 802.3x). Ethernet PAUSE stops all traffic on a link for a period of time. PFC enables you to divide traffic on a link into eight priorities and stop the traffic of a selected priority without stopping the traffic assigned to other priorities on the link.

Pausing the traffic of a selected priority enables you to provide lossless transport for traffic assigned that priority and at the same time use standard lossy Ethernet transport for the rest of the link traffic.

Buffer Management

Buffer management is critical to the proper functioning of PFC, because if buffers are allowed to overflow, frames are dropped and transport is not lossless.

For each lossless flow priority, the switch requires sufficient buffer space to:

- Store frames sent during the time it takes to send the PFC pause frame across the cable between devices.
- Store the frames that are already on the wire when the sender receives the PFC pause frame.

The propagation delay due to cable length and speed, as well as processing speed, determines the amount of buffer space needed to prevent frame loss due to congestion.

The switch automatically sets the threshold for sending PFC pause frames to accommodate delay from cables as long as 150 meters (492 feet) and to accommodate large frames that might be on the wire when the switch sends the pause frame. This ensures that the switch sends pause frames early enough to allow the sender to stop transmitting before the receive buffers on the switch overflow.

Physical Interfaces

QFX Series switches support 10-Gbps or faster, full-duplex interfaces. The switch enables DCB capability only on 10-Gbps or faster Ethernet interfaces.

ETS

PFC divides traffic into up to eight separate streams (priorities, configured on the switch as forwarding classes) on a physical link. ETS enables you to manage the link bandwidth by:

- Grouping the priorities into priority groups (configured on the switch as forwarding class sets).
- Specifying the bandwidth available to each of the priority groups as a percentage of the total available link bandwidth.

- Allocating the bandwidth to the individual priorities in the priority group.

The available link bandwidth is the bandwidth remaining after servicing strict-high priority queues. On QFX5200, QFX5100, EX4600, QFX3500, and QFX3600 switches, and on QFabric systems, we recommend that you always configure a shaping rate to limit the amount of bandwidth a strict-high priority queue can consume by including the shaping-rate statement in the **[edit class-of-service schedulers]** hierarchy on the strict-high priority scheduler. This prevents a strict-high priority queue from starving other queues on the port. (On QFX10000 switches, configure a transmit rate on strict-high priority queues to set a maximum amount of bandwidth for strict-high priority traffic.)

Managing link bandwidth with ETS provides several advantages:

- There is uniform management of all types of traffic on the link, both congestion-managed traffic and standard Ethernet traffic.
- When a priority group does not use all of its allocated bandwidth, other priority groups on the link can use that bandwidth as needed.

  When a priority in a priority group does not use all of its allocated bandwidth, other priorities in the group can use that bandwidth.

  The result is better bandwidth utilization, because priorities that consist of bursty traffic can share bandwidth during periods of low traffic transmission instead of consuming their entire bandwidth allocation when traffic loads are light.
- You can assign traffic types with different service needs to different priorities so that each traffic type receives appropriate treatment.
- Strict priority traffic retains its allocated bandwidth.

DCBX

DCB devices use DCBX to exchange configuration information with directly connected peers (switches and endpoints such as servers). DCBX is an extension of LLDP. If you disable LLDP on an interface, that interface cannot run DCBX. If you attempt to enable DCBX on an interface on which LLDP is disabled, the configuration commit fails.

DCBX can:

- Discover the DCB capabilities of peers.
- Detect DCB feature misconfiguration or mismatches between peers.
- Configure DCB features on peers.

You can configure DCBX operation for PFC, ETS, and for Layer 2 and Layer 4 applications such as FCoE and iSCSI. DCBX is enabled or disabled on a per-interface basis.

RELATED DOCUMENTATION

Understanding DCBX

IN THIS SECTION

Data Center Bridging Capability Exchange protocol (DCBX) is an extension of Link Layer Data Protocol (LLDP). If you disable LLDP on an interface, that interface cannot run DCBX. If you attempt to enable DCBX on an interface on which LLDP is disabled, the configuration commit operation fails. Data center bridging (DCB) devices use DCBX to exchange configuration information with directly connected peers.

Video: What is DCBX Protocol?

This topic describes:

DCBX Basics

DCBX can:

- Discover the DCB capabilities of peers.
- Detect DCB feature misconfiguration or mismatches between peers.
- Configure DCB features on peers.

You can configure DCBX operation for priority-based flow control (PFC), Layer 2 and Layer 4 applications such as FCoE and iSCSI, and ETS. DCBX is enabled or disabled on a per-interface basis.

> **NOTE:** QFX5200 and QFX5210 switches do not support enhanced transmission selection (ETS) hierarchical scheduling. Use port scheduling to manage bandwidth on these switches.

By default, for PFC and ETS, DCBX automatically negotiates administrative state and configuration with each interface's connected peer. To enable DCBX negotiation for applications, you must configure the applications, map them to IEEE 802.1p code points in an application map, and apply the application map to interfaces.

The FCoE application only needs to be included in an application map when you want an interface to exchange type, length, and values (TLVs) for other applications in addition to FCoE. If FCoE is the only application you want an interface to advertise, then you do not need to use an application map. For ETS, DCBX pushes the switch configuration to peers if they are set to learn the configuration from the switch (unless you disable sending the ETS recommendation TLV on interfaces in IEEE DCBX mode).

You can override the default behavior for PFC, for ETS, or for all applications mapped to an interface by turning off autonegotiation to force an interface to enable or disable that feature. You can also disable DCBX autonegotiation for applications on an interface by excluding those applications from the application map you apply to that interface or by deleting the application map from the interface.

The default autonegotiation behavior for applications that are mapped to an interface is:

- DCBX is enabled on the interface if the connected peer device also supports DCBX.
- DCBX is disabled on the interface if the connected peer device does not support DCBX.

During negotiation of capabilities, the switch can push the PFC configuration to an attached peer if the peer is configured as "willing" to learn the PFC configuration from other peers. The Juniper Networks switch does not support self autoprovisioning and does not change its configuration during autonegotiation to match the peer configuration. (The Juniper switch is not "willing" to learn the PFC configuration from peers.)

> **NOTE:** When a port with DCBX enabled begins to exchange type, length, and value (TLV) entries, optional LLDP TLVs on that port are not advertised to neighbors, so that the switch can interoperate with a wider variety of converged network adapters (CNAs) and Layer 2 switches that support DCBX.

DCBX Modes and Support

IN THIS SECTION

This section describes DCBX support:

DCBX Modes (Versions)

The two most common DCBX modes are supported:

- IEEE DCBX—The newest DCBX version. Different TLVs have different subtypes (for example, the subtype for the ETS configuration TLV is 9); the IEEE DCBX Organizationally Unique Identifier (OUI) is 0x0080c2.
- DCBX version 1.01—The Converged Enhanced Ethernet (CEE) version of DCBX. It has a subtype of 2 and an OUI of 0x001b21.

IEEE DCBX and DCBX version 1.01 differ mainly in frame format. DCBX version 1.01 uses one TLV that includes all DCBX attribute information, which is sent as sub-TLVs. IEEE DCBX uses a unique TLV for each DCB attribute.

> **NOTE:** The switch does not support pre-CEE (pre-DCB) DCBX versions. Unsupported older versions of DCBX have a subtype of 1 and an OUI of 0x001b21. The switch drops LLDP frames that contain pre-CEE DCBX TLVs.

Table 87 on page 456 summarizes the differences between IEEE DCBX and DCBX version 1.01, including show command output:

Table 87: Summary of Differences Between IEEE DCBX and DCBX Version 1.01

Characteristic	IEEE DCBX	DCBX Version 1.01
OUI	0x0080c2	0x001b21

Table 87: Summary of Differences Between IEEE DCBX and DCBX Version 1.01 *(continued)*

Characteristic	IEEE DCBX	DCBX Version 1.01
Frame Format	Sends a separate, unique TLV for each DCBX attribute. For example, IEEE DCBX uses separate TLVs for ETS, PFC, and each application. Configuration and Recommendation information is sent in different TLVs	Sends one TLV that includes all DCBX attribute information organized in sub-TLVs. The “willing” bit determines whether or not an interface can change its configuration to match the connected peer.
Symmetric/asymmetric configuration with peer	Asymmetric or symmetric	Symmetric only
Differences in the **show dcbx interface *interface-name*** operational command	• Synchronization information is not shown because symmetric configuration is not required. • Operational state information is not shown because the operational states do not have to be symmetric. • TLV type is shown because unique TLVs are sent for each DCBX attribute. • ETS peer Configuration TLV and Recommendation TLV information is shown separately because they are different TLVs.	• Synchronization information is shown because symmetric configuration is required. • Operational state information is shown because the operational states do have to be symmetric. • TLV type is not shown because one TLV is used for all attribute information. • Recommendation TLV is not sent (DCBX Version 1.01 uses the “willing” bit to determine whether or not an interface uses the peer interface configuration).

You can configure interfaces to use the following DCBX modes:

- IEEE DCBX—The interface uses IEEE DCBX regardless of the configuration on the connected peer.
- DCBX version 1.01—The interface uses DCBX version 1.01 regardless of the configuration on the connected peer.
- Autonegotiation—The interface automatically negotiates with the connected peer to determine the DCBX version the peers use. Autonegotiation is the default DCBX mode.

If you configure a DCBX mode on an interface, the interface ignores DCBX protocol data units (PDUs) it receives from the connected peer if the PDUs do not match the DCBX version configured on the interface. For example, if you configure an interface to use IEEE DCBX and the connected peer sends DCBX version 1.01 LLDP PDUs, the interface ignores the version 1.01 PDUs. If you configure an interface to use DCBX version 1.01 and the peer sends IEEE DCBX LLDP PDUs, the interface ignores the IEEE DCBX PDUs.

> **NOTE:** On interfaces that use the IEEE DCBX mode, the **show dcbx neighbors interface *interface-name*** operational command does not include application, PFC, or ETS operational state in the output.

Autonegotiation

Autonegotiation is the default DCBX mode. Each interface automatically negotiates with its connected peer to determine the DCBX version that both interfaces use to exchange DCBX information.

When an interface connects to its peer interface, the interface advertises IEEE DCBX TLVs to the peer. If the interface receives one IEEE DCBX PDU from the peer, the interface sets the DCBX mode as IEEE DCBX. If the interface receives three DCBX version 1.01 TLVs from the peer, the interface sets DCBX version 1.01 as the DCBX mode.

Autonegotiation works slightly differently on standalone switches compared to QFabric systems:

- Standalone switches—When an interface connects to its peer interface, the interface advertises IEEE DCBX TLVs to the peer. If the interface receives an IEEE DCBX TLV from the peer, the interface sets IEEE DCBX as the DCBX mode. If the interface receives three consecutive DCBX version 1.01 TLVs from the peer, the interface sets DCBX version 1.01 as the DCBX mode.
- QFabric system—When an interface connects to its peer interface, the interface advertises DCBX version 1.01 TLVs to the peer. If the interface receives an IEEE DCBX TLVs from the peer, the interface sets IEEE DCBX as the DCBX mode. If the interface receives three consecutive DCBX version 1.01 TLVs from the peer, the interface retains DCBX version 1.01 as the DCBX mode.

> **NOTE:** If the link flaps or the LLDP process restarts, the interface starts the autonegotiation process again. The interface does not use the last received DCBX communication mode.

CNA Support for DCBX Modes

Different CNA vendors support different versions and capabilities of DCBX. The DCBX configuration you use on switch interfaces depends on the DCBX features that the CNAs in your network support.

Interface Support for DCBX

You can configure DCBX on 10-Gigabit Ethernet interfaces and on link aggregation group (LAG) interfaces whose member interfaces are all 10-Gigabit Ethernet interfaces.

DCBX Attribute Types

IN THIS SECTION

DCBX has three attribute types:

- Informational—These attributes are exchanged using LLDP, but do not affect DCBX state or operation; they only communicate information to the peer. For example, application priority TLVs are informational TLVs.
- Asymmetric—The values for these types of attributes do not have to be the same on the connected peer interfaces. Peers exchange asymmetric attributes when the attribute values can differ on each peer interface. The peer interface configurations might match or they might differ. For example, ETS Configuration and Recommendation TLVs are asymmetric TLVs.
- Symmetric—The intention is that the values for these types of attributes should be the same on both of the connected peer interfaces. Peer interfaces exchange symmetric attributes to ensure symmetric DCBX configuration for those attributes. For example, PFC Configuration TLVs are symmetric TLVs.

The following sections describe asymmetric and symmetric DCBX attributes:

Asymmetric Attributes

DCBX passes asymmetric attributes between connected peer interfaces to communicate parameter information about those attributes (features). The resulting configuration for an attribute might be different on each peer, so the parameters configured on one interface might not match the parameters on the connected peer interface.

There are two types of asymmetric attribute TLVs:

- Configuration TLV—Configuration TLVs communicate the current operational state and the state of the “willing” bit. The “willing” bit communicates whether or not the interface is willing to accept and use the configuration from the peer interface. If an interface is “willing,” the interface uses the configuration it receives from the peer interface. (The peer interface configuration can override the configuration on the “willing” interface.) If an interface is “not willing”, the configuration on the interface cannot be overridden by the peer interface configuration.
- Recommendation TLV—Recommendation TLVs communicate the parameters the interface recommends that the connected peer interface should use. When an interface sends a Recommendation TLV, if the connected peer is “willing,” the connected peer changes its configuration to match the parameters in the Recommendation TLV.

Symmetric Attributes

DCBX passes symmetric attributes between connected peer interfaces to communicate parameter information about those attributes (features), with the objective that both interfaces should use the same configuration. The intent is that the parameters configured on one interface should match the parameters on the connected peer interface.

There is one type of symmetric attribute TLV, the Configuration TLV. As with asymmetric attributes, symmetric attribute Configuration TLVs communicate the current operational state and the state of the "willing" bit. "Willing" interfaces use the peer interface parameter values for the attribute. (The attribute configuration of the peer overrides the configuration on the "willing" interface.)

DCBX Application Protocol TLV Exchange

IN THIS SECTION

DCBX advertises the switch's capabilities for Layer 2 applications such as FCoE and Layer 4 applications such as iSCSI:

Application Protocol TLV Exchange

For all applications, DCBX advertises the application's state and IEEE 802.1p code points on the interfaces to which the application is mapped. If an application is not mapped to an interface, that interface does not advertise the application's TLVs. There is an exception for FCoE application protocol TLV exchange when FCoE is the only application you want DCBX to advertise on an interface.

FCoE Application Protocol TLV Exchange

Protocol TLV exchange for the FCoE application depends on whether FCoE is the only application you want the interface to advertise or whether you want the interface to exchange other application TLVs in addition to FCoE TLVs.

If FCoE is the only application you want DCBX to advertise on an interface, DCBX exchanges FCoE application protocol TLVs by default if the interface:

- Carries FCoE traffic (traffic mapped by CoS configuration to the FCoE forwarding class)
- Has a congestion notification profile with PFC enabled on the FCoE priority (IEEE 802.1p code point)
- Does *not* have an application map

> **NOTE:** If no CoS configuration for FCoE is mapped to an interface, that interface does not exchange FCoE application protocol TLVs.

If you want DCBX to advertise FCoE and other applications on an interface, you must specify all of the applications, including FCoE, in an application map, and apply the application map to the desired interfaces.

> **NOTE:** If an application map is applied to an interface, the FCoE application must be explicitly configured in the application map, or the interface does not exchange FCoE TLVs.

When DCBX advertises the FCoE application, it advertises the FCoE state and IEEE 802.1p code points. If a peer device connected to a switch interface does not support FCoE, DCBX uses autonegotiation to mark the interface as “FCoE down,” and FCoE is disabled on that interface.

Disabling Application Protocol TLV Exchange

To disable DCBX application protocol exchange for all applications on an interface, issue the **set protocols dcbx interface *interface-name* applications no-auto-negotiation** command.

You can also disable DCBX application protocol exchange for applications on an interface by deleting the application map from the interface, or by deleting a particular application from the application map. However, when you delete an application from an application map, the application protocol is no longer exchanged on any interface which uses that application map.

DCBX and PFC

After you enable PFC on a switch interface, DCBX uses autonegotiation to control the operational state of the PFC functionality.

If the peer device connected to the interface supports PFC and is provisioned compatibly with the switch, DCBX sets the PFC operational state to enabled. If the peer device connected to the interface does not support PFC or is not provisioned compatibly with the switch, DCBX sets the operational state to disabled. (PFC must be symmetrical.)

If the peer advertises that it is “willing” to learn its PFC configuration from the switch, DCBX pushes the switch’s PFC configuration to the peer and does not check the peer’s administrative state.

You can manually override DCBX control of the PFC operational state on a per-interface basis by disabling autonegotiation. If you disable autonegotiation on an interface on which you have configured PFC, then PFC is enabled on that interface regardless of the peer configuration. To disable PFC on an interface, do not configure PFC on that interface.

DCBX and ETS

IN THIS SECTION

This section describes:

Default DCBX ETS Advertisement

If you do not configure ETS on an interface, the switch automatically creates a default priority group that contains all of the priorities (forwarding classes, which represent output queues) and assigns 100 percent of the port output bandwidth to that priority group. The default priority group is transparent. It does not appear in the configuration and is used for DCBX advertisement. DCBX advertises the default priority group, its priorities, and the assigned bandwidth.

If you configure ETS on an interface, DCBX advertises:

- Each priority group on the interface
- The priorities in each priority group
- The bandwidth properties of each priority group and priority

Any priority on that interface that is not part of an explicitly configured priority group (forwarding class set) is assigned to the automatically generated default priority group and receives no bandwidth. If you configure ETS on an interface, every forwarding class (priority) on that interface for which you want to forward traffic must belong to a forwarding class set (priority group).

ETS Advertisement and Peer Configuration

DCBX does not control the switch's ETS (hierarchical scheduling) operational state. If the connected peer is configured as "willing," DCBX pushes the switch's ETS configuration to the switch's peers if the ETS Recommendation TLV is enabled (it is enabled by default). If the peer does not support ETS or is not consistently provisioned with the switch, DCBX does not change the ETS operational state on the switch. The ETS operational state remains enabled or disabled based only on the switch hierarchical scheduling configuration and is enabled by default.

When ETS is configured, DCBX advertises the priority groups, the priorities in the priority groups, and the bandwidth configuration for the priority groups and priorities. Any priority (essentially a forwarding class or queue) that is not part of a priority group has no scheduling properties and receives no bandwidth.

You can manually override whether DCBX advertises the ETS state to the peer on a per-interface basis by disabling autonegotiation. This does not affect the ETS state on the switch or on the peer , but it does prevent the switch from sending the Recommendation TLV or the Configuration TLV to the connected peer. To disable ETS on an interface, do not configure priority groups (forwarding class sets) on the interface.

ETS Recommendation TLV

The ETS Recommendation TLV communicates the ETS settings that the switch wants the connected peer interface to use. If the peer interface is "willing," it changes its configuration to match the configuration in the ETS Recommendation TLV. By default, the switch interfaces send the ETS Recommendation TLV to the peer. The settings communicated are the egress ETS settings defined by configuring hierarchical scheduling on the interface.

We recommend that you use the same ETS settings on the connected peer that you use on the switch interface and that you leave the ETS Recommendation TLV enabled. However, on interfaces that use IEEE DCBX as the DCBX mode, if you want an asymmetric configuration between the switch interface and the connected peer, you can disable the ETS Recommendation TLV by including the **no-recommendation-tlv** statement at the **[edit protocols dcbx interface *interface-name* enhanced-transmission-selection]** hierarchy level.

NOTE: You can disable the ETS Recommendation TLV only when the DCBX mode on the interface is IEEE DCBX. Disabling the ETS Recommendation TLV has no effect if the DCBX mode on the interface is DCBX version 1.01. (IEEE DCBX uses separate application attribute TLVs, but DCBX version 1.01 sends all application attributes in the same TLV and uses sub-TLVs to separate the information.)

If you disable the ETS Recommendation TLV, the switch still sends the ETS Configuration TLV to the connected peer. The result is that the connected peer is informed about the switch DCBX ETS configuration, but even if the peer is "willing," the peer does not change its configuration to match the switch configuration. This is asymmetric configuration—the two interfaces can have different parameter values for the ETS attribute.

For example, if you want a CNA connected to a switch interface to have different bandwidth allocations than the switch ETS configuration, you can disable the ETS Recommendation TLV and configure the CNA for the desired bandwidth. The switch interface and the CNA exchange configuration parameters, but the CNA does not change its configuration to match the switch interface configuration.

RELATED DOCUMENTATION

Understanding DCBX Application Protocol TLV Exchange | 468

Understanding DCB Features and Requirements | 450

Configuring the DCBX Mode

You can configure the DCBX mode that an interface uses to communicate with the connected peer. Three DCBX modes are supported:

- Autonegotiation—The interface negotiates with the connected peer to determine the DCBX mode. This is the default DCBX mode.
- IEEE DCBX—The interface uses IEEE DCBX type, length, and value (TLV) to exchange DCBX information with the connected peer. QFX3500 Node devices come up with IEEE DCBX enabled by default and then autonegotiate with the connected peer to determine the final DCBX mode.
- DCBX Version 1.01—The interface uses Converged Enhanced Ethernet (CEE) DCBX version 1.01 TLVs to exchange DCBX information with the connected peer. QFabric system Node devices other than QFX3500 switches come up with DCBX version 1.01 enabled by default and then autonegotiate with the connected peer to determine the final DCBX mode.

> **NOTE:** Pre-CEE (pre-DCB) versions of DCBX such as DCBX version 1.00 are not supported. If an interface receives an LLDP frame with pre-CEE DCBX TLVs, the system drops the frame.

Configure the DCBX mode by specifying the mode for one interface or for all interfaces.

- To configure the DCBX mode, specify the interface and the mode:

  `[edit protocols dcbx]`
  `user@switch#` **set interface *interface-name* dcbx-version (auto-negotiate | ieee-dcbx | dcbx-version-1.01)**

  For example, to configure DCBX version 1.01 on interface **xe-0/0/21**:

  `user@switch#` **set protocols dcbx interface xe-0/0/21 dcbx-version dcbx-version-1.01**

To configure IEEE DCBX on all interfaces:

`user@switch#` **set protocols dcbx interface all dcbx-version ieee-dcbx**

RELATED DOCUMENTATION

Configuring DCBX Autonegotiation

Data Center Bridging Capability Exchange protocol (DCBX) discovers the data center bridging (DCB) capabilities of peers by exchanging feature configuration information. DCBX also detects feature misconfiguration and mismatches, and can configure DCB on peers. DCBX is an extension of the Link Layer Discovery Protocol (LLDP), and LLDP must remain enabled on every interface for which you want to use DCBX. If you attempt to enable DCBX on an interface on which LLDP is disabled, the configuration commit operation fails.

> **NOTE:** LLDP and DCBX are enabled by default on all interfaces.

The switch supports DCBX autonegotiation for:

- Priority-based flow control (PFC) configuration
- Layer 2 and Layer 4 applications such as Fibre Channel over Ethernet (FCoE) and Internet Small Computer System Interface (iSCSI)
- Enhanced transmission selection (ETS) advertisement

DCBX autonegotiation is configured on a per-interface basis for each supported feature or application. The PFC and application DCBX exchanges use autonegotiation by default. The default autonegotiation behavior is:

- DCBX is enabled on the interface if the connected peer device also supports DCBX.
- DCBX is disabled on the interface if the connected peer device does not support DCBX.

You can override the default behavior for each feature by turning off autonegotiation to force an interface to enable or disable the feature.

Autonegotiation of ETS means that when ETS is enabled on an interface (priority groups are configured), the interface advertises its ETS configuration to the peer device. In this case, priorities (forwarding classes) that are not part of a priority group (forwarding class set) receive no bandwidth and are advertised in an automatically generated default forwarding class. If ETS is not enabled on an interface (no priority groups are configured), all of the priorities are advertised in one automatically generated default priority group that receives 100 percent of the port bandwidth.

Disabling ETS autonegotiation prevents the interface from sending the Recommendation TLV or the Configuration TLV to the connected peer.

On interfaces that use IEEE DCBX mode to exchange DCBX parameters, you can disable autonegotiation of the ETS Recommendation TLV to the peer if you want an asymmetric ETS configuration between the peers. DCBX still exchanges the ETS Configuration TLV if you disable the ETS Recommendation TLV.

Autonegotiation of PFC means that when PFC is enabled on an interface, if the peer device connected to the interface supports PFC and is provisioned compatibly with the switch, DCBX sets the PFC operational state to enabled. If the peer device connected to the interface does not support PFC or is not provisioned compatibly with the switch, DCBX sets the operational state to disabled.

In addition, if the peer advertises that it is “willing” to learn its PFC configuration from the switch, DCBX pushes the switch’s PFC configuration to the peer and does not check the peer’s administrative state. The switch does not learn PFC configuration from peers (the switch does not advertise its state as “willing”).

Disabling PFC autonegotiation prevents the interface from exchanging PFC configuration information with the peer. It forces the interface to enable PFC if PFC is configured on the interface or to disable PFC if PFC is not configured on the interface. If you disable PFC autonegotiation, the assumption is that the peer is also configured manually.

Autonegotiation of applications depends on whether or not you apply an application map to an interface. If you apply an application map to an interface, the interface autonegotiates DCBX for each application in the application map. PFC must be enabled on the FCoE priority (the FCoE IEEE 802.1p code point) for the interface to advertise the FCoE application. The interface only advertises applications that are included in the application map.

For example, if you apply an application map to an interface and the application map does not include the FCoE application, then that interface does not perform DCBX advertisement of FCoE.

If you do not apply an application map to an interface, DCBX does not advertise applications on that interface, with the exception of FCoE, which is handled differently than other applications.

NOTE: If you do not apply an application map to an interface, the interface performs autonegotiation of FCoE if the interface carries traffic in the FCoE forwarding class and also has PFC enabled on the FCoE priority. On such interfaces, if DCBX detects that the peer device connected to the interface supports FCoE, the switch advertises its FCoE capability and IEEE 802.1p code point on that interface. If DCBX detects that the peer device connected to the interface does not support FCoE, DCBX marks that interface as "FCoE down" and disables FCoE on the interface.

When DCBX marks an interface as "FCoE down," the behavior of the switch depends on how you use it in the network:

- When the switch acts as an FCoE transit switch, the interface drops all of the FIP packets it receives. In addition, FIP packets received from an FCoE forwarder (FCF) are not forwarded to interfaces marked as "FCoE down."
- When the switch acts as an FCoE-FC gateway (only switches that support native Fibre Channel interfaces), it does not send or receive FCoE Initialization Protocol (FIP) packets.

Disabling autonegotiation prevents the interface from exchanging application information with the peer. In this case, the assumption is that the peer is also configured manually.

To disable DCBX autonegotiation of PFC, applications (including FCoE), and ETS using the CLI:

1. Turn off autonegotiation for PFC.

   ```
   [edit]
   user@switch# set protocols dcbx interface interface-name priority-flow-control no-auto-negotiation
   ```

2. Turn off autonegotiation for applications.

   ```
   [edit]
   user@switch# set protocols dcbx interface interface-name applications no-auto-negotiation
   ```

3. Turn off autonegotiation for ETS.

   ```
   [edit]
   user@switch# set protocols dcbx interface interface-name enhanced-transmission-selection
   no-auto-negotiation
   ```

To disable autonegotiation of the ETS Recommendation TLV so that DCBX exchanges only the ETS Configuration TLV:

- `[edit protocols dcbx interface` *`interface-name`*`]`
  `user@switch#` **set enhanced-transmission-selection no-recommendation-tlv**

RELATED DOCUMENTATION

Example: Configuring DCBX Application Protocol TLV Exchange | 476
Example: Configuring CoS PFC for FCoE Traffic | 490
Disabling the ETS Recommendation TLV | 447
Understanding DCBX Application Protocol TLV Exchange | 468

Understanding DCBX Application Protocol TLV Exchange

IN THIS SECTION

Data Center Bridging Capability Exchange protocol (DCBX) discovers the data center bridging (DCB) capabilities of connected peers. DCBX also advertises the capabilities of applications on interfaces by exchanging application protocol information through application type, length, and value (TLV) elements. DCBX is an extension of Link Layer Discovery Protocol (LLDP). LLDP must remain enabled on every interface on which you want to use DCBX.

NOTE: LLDP and DCBX are enabled by default on all interfaces.

Setting up application protocol exchange consists of:

- Defining applications
- Mapping the applications to IEEE 802.1p code points in an *application map*
- Configuring classifiers to prioritize incoming traffic and map the incoming traffic to the application by the traffic code points
- Applying the application maps and classifiers to interfaces

You need to explicitly define the applications that you want an interface to advertise. The FCoE application is a special case (see “Applications” on page 469) and only needs to be defined on an interface if you want DCBX to exchange application protocol TLVs for other applications in addition to FCoE on that interface.

You also need to explicitly map all of the defined applications that you want an interface to advertise to IEEE 802.1p code points in an application map. The FCoE application is a special case that only requires inclusion in an application map when you want an interface to use DCBX for other applications in addition to FCoE, as described later in this topic (see “Application Maps” on page 470).

This topic describes:

Applications

Before an interface can exchange application protocol information, you need to define the applications that you want to advertise. The exception is the FCoE application. If FCoE is the only application that you want the interface to advertise, then you do not need to define the FCoE application. You need to define the FCoE application only if you want interfaces to advertise other applications in addition to FCoE.

NOTE: If FCoE is the only application that you want DCBX to advertise on an interface, DCBX exchanges FCoE application protocol TLVs by default if the interface:

- Carries FCoE traffic (traffic mapped by CoS configuration to the FCoE forwarding class and applied to the interface)
- Has a congestion notification profile with PFC enabled on the FCoE priority (IEEE 802.1p code point)
- Does *not* have an application map

If you apply an application map to an interface, then all applications that you want DCBX to advertise must be defined and configured in the application map, including the FCoE application.

If no CoS configuration for FCoE is mapped to an interface, that interface does not exchange FCoE application protocol TLVs.

You can define:

- Layer 2 applications by EtherType
- Layer 4 applications by a combination of protocol (TCP or UDP) and destination port number

The EtherType is a two-octet field in the Ethernet frame that denotes the protocol encapsulated in the frame. For a list of common EtherTypes, see http://standards.ieee.org/develop/regauth/ethertype/eth.txt on the IEEE standards organization website. For a list of port numbers and protocols, see the *Service Name and Transport Protocol Port Number Registry* at http://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xml on the Internet Assigned Numbers Authority (IANA) website.

You must explicitly define each application that you want to advertise, except FCoE. The FCoE application is defined by default (EtherType 0x8906).

Application Maps

An application map maps defined applications to one or more IEEE 802.1p code points. Each application map contains one or more applications. DCBX includes the configured application code points in the protocol TLVs exchanged with the connected peer.

To exchange protocol TLVs for an application, you must include the application in an application map. The FCoE application is a special case:

- If you want DCBX to exchange application protocol TLVs for more than one application on a particular interface, you must configure the applications, define an application map to map the applications to code points, and apply the application map to the interface. In this case, you must also define the FCoE application and add it to the application map.

This is the same process and treatment required for all other applications. In addition, for DCBX to exchange FCoE application TLVs, you must enable priority-based flow control (PFC) on the FCoE priority (the FCoE IEEE 802.1p code point) on the interface.

- If FCoE is the only application that you want DCBX to advertise on an interface, then you do not need to configure an application map and apply it to the interface. By default, when an interface has no application map, and the interface carries traffic mapped to the FCoE forwarding class, and PFC is enabled on the FCoE priority, the interface advertises FCoE TLVs (autonegotiation mode). DCBX exchanges FCoE application protocol TLVs by default until you apply an application map to the interface, remove the FCoE traffic from the interface (you can do this by removing the or editing the classifier for FCoE traffic), or disable PFC on the FCoE priority.

  If you apply an application map to an interface that did not have an application map and was exchanging FCoE application TLVs, and you do not include the FCoE application in the application map, the interface stops exchanging FCoE TLVs. Every interface that has an application map must have FCoE included in the application map (and PFC enabled on the FCoE priority) in order for DCBX to exchange FCoE TLVs.

Mapping an application to code points does two things:

- Maps incoming traffic with the same code points to that application
- Allows you to configure classifiers that map incoming application traffic, by code point, to a forwarding class and a loss priority, in order to apply class of service (CoS) to application traffic and prioritize application traffic

You apply an application map to an interface to enable DCBX application protocol exchange on that interface for each application specified in the application map. All of the applications that you want an interface to advertise must be configured in the application map that you apply to the interface, with the previously noted exception for the FCoE application when FCoE is the only application for which you want DCBX to exchange protocol TLVs on an interface.

Classifying and Prioritizing Application Traffic

When traffic arrives at an interface, the interface classifies the incoming traffic based on its code points. Classifiers map code points to loss priorities and forwarding classes. The loss priority prioritizes the traffic. The forwarding class determines the traffic output queue and CoS service level.

When you map an application to an IEEE 802.1p code point in an application map and apply the application map to an interface, incoming traffic on the interface that matches the application code points is mapped to the appropriate application. The application receives the loss priority and the CoS associated with the forwarding class for those code points, and is placed in the output queue associated with the forwarding class.

You can use the default classifier or you can configure a classifier to map the application code points defined in the application map to forwarding classes and loss priorities.

Enabling Interfaces to Exchange Application Protocol Information

Each interface with the **fcoe** forwarding class and PFC enabled on the FCoE code point is enabled for FCoE application protocol exchange by default until you apply an application map to the interface. If you apply an application map to an interface and you want that interface to exchange FCoE application protocol TLVs, you must include the FCoE application in the application map. (In all cases, to achieve lossless transport, you must also enable PFC on the FCoE code point or code points.)

Except when FCoE is the only protocol you want DCBX to advertise on an interface, interfaces on which you want to exchange application protocol TLVs must include the following two items:

- The application map that contains the application(s)
- A classifier

> **NOTE:** You must also enable PFC on the code point of any traffic for which you want to achieve lossless transport.

Disabling DCBX Application Protocol Exchange

To disable DCBX application protocol exchange for all applications on an interface, issue the **set protocols dcbx interface *interface-name* applications no-auto-negotiation** command.

You can also disable DCBX application protocol exchange for applications on an interface by deleting the application map from the interface, or by deleting a particular application from the application map. However, when you delete an application from an application map, the application protocol is no longer exchanged on any interface which uses that application map.

On interfaces that use IEEE DCBX mode to exchange DCBX parameters, you can disable sending the enhanced transmission selection (ETS) Recommendation TLV to the peer if you want an asymmetric ETS configuration between the peers.

RELATED DOCUMENTATION

Understanding DCBX | 454
Configuring DCBX Autonegotiation | 465
Disabling the ETS Recommendation TLV | 447
Defining an Application for DCBX Application Protocol TLV Exchange | 473
Configuring an Application Map for DCBX Application Protocol TLV Exchange | 474
Applying an Application Map to an Interface for DCBX Application Protocol TLV Exchange | 475

Defining an Application for DCBX Application Protocol TLV Exchange

Define each application for which you want DCBX to exchange application protocol information. You can define Layer 2 and Layer 4 applications. After you define applications, you map them to IEEE 802.1p code points, and then apply the application map to the interfaces on which you want DCBX to exchange application protocol information with connected peers. (See *Related Documentation* for how to configure application maps and apply them to interfaces, and for an example of the entire procedure that also includes classifier configuration.)

> **NOTE:** In Junos OS Release 12.1, the FCoE application was configured by default, so you did not need to configure it in an application map. In Junos OS Release 12.2, if you want DCBX to advertise the FCoE application on an interface and you apply an application map to that interface, you must explicitly configure FCoE in the application map. You also must enable priority-based flow control (PFC) on the FCoE code point on all interfaces that you want to advertise FCoE. If you apply an application map to an interface, the interface sends DCBX TLVs only for the applications configured in the application map.

Define Layer 2 applications by mapping an application name to an EtherType. Define Layer 4 applications by mapping an application name to a protocol (TCP or UDP) and a destination port.

- To define a Layer 2 application, specify the name of the application and its EtherType:

  ```
  [edit applications]
  user@switch# set application application-name ether-type ether-type
  ```

  For example, to configure an application named **PTP** (for Precision Time Protocol) that uses the EtherType **0x88F7**:

  ```
  user@switch# set applications application ptp ether-type 0x88F7
  ```

- To define a Layer 4 application, specify the name of the application, its protocol (TCP or UDP), and its destination port:

  ```
  [edit]
  user@switch# set applications application application-name protocol (tcp | udp) destination-port
  port-value
  ```

  For example, to configure an application named **iscsi** (for Internet Small Computer System Interface) that uses the protocol **TCP** and the destination port **3260**:

```
user@switch# set applications application iscsi protocol tcp destination-port 3260
```

RELATED DOCUMENTATION

Configuring an Application Map for DCBX Application Protocol TLV Exchange

After you define applications for which you want to exchange DCBX application protocol information, map the applications to IEEE 802.1p code points. The IEEE 802.1p code points identify incoming traffic and allow you to map that traffic to the desired application. You then apply the application map to the interfaces on which you want DCBX to exchange application protocol information with connected peers. (See *Related Documentation* for how to define applications and apply the application map to interfaces, and for an example of the entire procedure that also includes classifier configuration.)

> **NOTE:** In Junos OS Release 12.1, the FCoE application was configured by default, so you did not need to configure it in an application map. In Junos OS Release 12.2, if you want DCBX to advertise the FCoE application on an interface and you apply an application map to that interface, you must explicitly configure FCoE in the application map. You also must enable priority-based flow control (PFC) on the FCoE code point on all interfaces that you want to advertise FCoE. If you apply an application map to an interface, the interface sends DCBX TLVs only for the applications configured in the application map.

Configure an application map by creating an application map name and mapping an application to one or more IEEE 802.1p code points.

- To define an application map, specify the name of the application map, the name of the application, and the IEEE 802.1p code points of the incoming traffic that you want to associate with the application in the application map:

  `[edit policy-options]`
  `user@switch#` **set application-maps *application-map-name* application *application-name* code-points [*aliases*] [*bit-patterns*]**

  For example, to configure an application map named **ptp-app-map** that includes an application named **PTP** (for Precision Time Protocol) and map the application to IEEE 802.1p code points **001** and **101**:

  `user@switch#` **set policy-options application-maps ptp-app-map application ptp code points [001 101]**

RELATED DOCUMENTATION

Defining an Application for DCBX Application Protocol TLV Exchange | 473
Applying an Application Map to an Interface for DCBX Application Protocol TLV Exchange | 475
Configuring DCBX Autonegotiation | 465
Example: Configuring DCBX Application Protocol TLV Exchange | 476
Example: Configuring DCBX to Support an iSCSI Application
show dcbx neighbors | 977

Applying an Application Map to an Interface for DCBX Application Protocol TLV Exchange

After you define applications and map them to IEEE 802.1p code points in an application map, apply the application map to the interfaces on which you want DCBX to exchange the application protocol information with connected peers. (See *Related Documentation* for how to define applications and configure application maps to interfaces, and for an example of the entire procedure that also includes classifier configuration.)

> **NOTE:** In Junos OS Release 12.1, the FCoE application was configured by default, so you did not need to configure it in an application map. In Junos OS Release 12.2, if you want DCBX to advertise the FCoE application on an interface and you apply an application map to that interface, you must explicitly configure FCoE in the application map. You also must enable priority-based flow control (PFC) on the FCoE code point on all interfaces that you want to advertise FCoE. If you apply an application map to an interface, the interface sends DCBX TLVs only for the applications configured in the application map.

- To apply an application map to a DCBX interface, specify the DCBX interface and the application map name:

  `[edit protocols]`
  `user@switch#` **set dcbx interface *interface-name* application-map *application-map-name***

  For example, to apply an application map named **ptp-app-map** on interface **xe-0/0/11**:

  `user@switch#` **set protocols dcbx interface xe-0/0/11 application-map ptp-app-map**

RELATED DOCUMENTATION

Defining an Application for DCBX Application Protocol TLV Exchange | 473
Configuring an Application Map for DCBX Application Protocol TLV Exchange | 474
Configuring DCBX Autonegotiation | 465
Example: Configuring DCBX Application Protocol TLV Exchange | 476
Example: Configuring DCBX to Support an iSCSI Application
show dcbx neighbors | 977

Example: Configuring DCBX Application Protocol TLV Exchange

IN THIS SECTION

Data Center Bridging Capability Exchange protocol (DCBX) discovers the data center bridging (DCB) capabilities of connected peers by exchanging application configuration information. DCBX detects feature misconfiguration and mismatches and can configure DCB on peers. DCBX is an extension of the Link Layer Discovery Protocol (LLDP). LLDP must remain enabled on every interface on which you want to use DCBX.

> **NOTE:** LLDP and DCBX are enabled by default on all interfaces.

The switch supports DCBX application protocol exchange for Layer 2 and Layer 4 applications such as the Internet Small Computer System Interface (iSCSI). You specify applications by EtherType (for Layer 2 applications) or by the destination port and protocol (for Layer 4 applications; the protocol can be either TCP or UDP).

The switch handles Fibre Channel over Ethernet (FCoE) application protocol exchange differently than other protocols in some cases:

- If FCoE is the only application for which you want to enable DCBX application protocol TLV exchange on an interface, you do not have to explicitly configure the FCoE application or an application map. By default, the switch exchanges FCoE application protocol TLVs on all interfaces that carry FCoE traffic (traffic mapped to the **fcoe** forwarding class) and have priority-based flow control (PFC) enabled on the FCoE priority (the FCoE IEEE 802.1p code point). The default priority mapping for the FCoE application is IEEE 802.1p code point 011 (the default **fcoe** forwarding class code point).
- If you want an interface to use DCBX to exchange application protocol TLVs for any other applications in addition to FCoE, you must configure the applications (including FCoE), define an application map (including FCoE), and apply the application map to the interface. If you apply an application map to an interface, you must explicitly configure the FCoE application, or the interface does not exchange FCoE application protocol TLVs.

This example shows how to configure interfaces to exchange both Layer 2 and Layer 4 applications by configuring one interface to exchange iSCSI and FCoE application protocol information and configuring another interface to exchange iSCSI and Precision Time Protocol (PTP) application protocol information.

Requirements

This example uses the following hardware and software components:

- Juniper Networks QFX Series device
- Junos OS Release 12.1 or later for the QFX Series

Overview

The switch supports DCBX application protocol exchange for:

- Layer 2 applications, defined by EtherType
- Layer 4 applications, defined by destination port and protocol

> **NOTE:** DCBX also advertises PFC and enhanced transmission selection (ETS) information. See "Configuring DCBX Autonegotiation" on page 465 for how DCBX negotiates and advertises configuration information for these features and for the applications.

DCBX is configured on a per-interface basis for each supported feature or application. For applications that you want to enable for DCBX application protocol exchange, you must:

- Define the application name and configure the EtherType or the destination port and protocol (TCP or UDP) of the application. Use the EtherType for Layer 2 applications, and use the destination port and protocol for Layer 4 protocols.
- Map the application to an IEEE 802.1p code point in an application map.
- Add the application map to DCBX interface.

In addition, for all applications (including FCoE, even when you do not use an application map), you either must create an IEEE 802.1p classifier and apply it to the appropriate ingress interfaces or use the default classifier. A classifier maps the code points of incoming traffic to a forwarding class and a loss priority so that ingress traffic is assigned to the correct class of service (CoS). The forwarding class determines the output queue on the egress interface.

If you do not create classifiers, trunk and tagged-access ports use the unicast IEEE 802.1 default trusted classifier. Table 88 on page 479 shows the default mapping of IEEE 802.1 code-point values to unicast forwarding classes and loss priorities for ports in trunk mode or tagged-access mode. Table 89 on page 479 shows the default untrusted classifier IEEE 802.1 code-point values to unicast forwarding class mapping for ports in access mode.

Table 88: Default IEEE 802.1 Classifiers for Trunk Ports and Tagged-Access Ports (Default 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 89: Default IEEE 802.1 Unicast Classifiers for Access Ports (Default 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

Topology

This example shows how to configure DCBX application protocol exchange for three protocols (iSCSI, PTP, and FCoE) on two interfaces. One interface exchanges iSCSI and FCoE application protocol information, and the other interface exchanges iSCSI and PTP application protocol information.

NOTE: You must map FCoE traffic to the interfaces on which you want to forward FCoE traffic. You must also enable PFC on the FCoE interfaces and create an ingress classifier for FCoE traffic, or else use the default classifier.

Table 90 on page 480 shows the configuration components for this example.

Table 90: Components of DCBX Application Protocol Exchange Configuration Topology

Component	Settings
Hardware	QFX Series device
LLDP	Enabled by default on Ethernet interfaces
DCBX	Enabled by default on Ethernet interfaces
iSCSI application (Layer 4)	Application name—**iscsi** protocol—**TCP** destination-port—**3260** code-points—**111**
PTP application (Layer 2)	Application name—**ptp** ether-type—**0x88F7** code-points—**001**, **101**
FCoE application (Layer 2)	Application name—**fcoe** ether-type—**0x8906** code-points—**011** **NOTE:** You explicitly configure the FCoE application because you are applying an application map to the interface. When you apply an application map to an interface, all applications must be explicitly configured and included in the application map.
Application maps	**dcbx-iscsi-fcoe-app-map**—Maps the iSCSI and FCoE applications to IEEE 802.1p code points **dcbx-iscsi-ptp-app-map**—Maps iSCSI and PTP applications to IEEE 802.1p code points

Table 90: Components of DCBX Application Protocol Exchange Configuration Topology *(continued)*

Component	Settings
Interfaces	**xe-0/0/10**—Configured to exchange FCoE and iSCSI application TLVs (uses application map **dcbx-iscsi-fcoe-app-map**, carries FCoE traffic, and has PFC enabled on the FCoE priority) **xe-0/0/11**—Configured to exchange iSCSI and PTP application TLVs (uses application map **dcbx-iscsi-ptp-app-map**)
PFC congestion notification profile for FCoE application exchange	**fcoe-cnp**: • Code point—**011** • Interface—**xe-0/0/10**
Behavior aggregate classifiers (map forwarding classes to incoming packets by the packet's IEEE 802.1 code point)	**fcoe-iscsi-cl1**: • Maps the **fcoe** forwarding class to the IEEE 802.1p code point used for the FCoE application (**011**) and a loss priority of **high** • Maps the **network-control** forwarding class to the IEEE 802.1p code point used for the iSCSI application (**111**) and a loss priority of **high** • Applied to interface **xe-0/0/10** **iscsi-ptp-cl2**: • Maps the **network-control** forwarding class to the IEEE 802.1p code point used for the iSCSI application (**111**) and a loss priority of **low** • Maps the **best-effort** forwarding class to the IEEE 802.1p code points used for the PTP application (**001** and **101**) and a loss priority of **low** • Applied to interface **xe-0/0/11**

NOTE: This example does not include scheduling (bandwidth allocation) configuration or lossless configuration for the iSCSI forwarding class.

Configuration

CLI Quick Configuration

To quickly configure DCBX application protocol exchange, 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.

set applications application iSCSI protocol tcp destination-port 3260

set applications application FCoE ether-type 0x8906

set applications application PTP ether-type 0x88F7

set policy-options application-maps dcbx-iscsi-fcoe-app-map application iSCSI code-points 111

set policy-options application-maps dcbx-iscsi-fcoe-app-map application FCoE code-points 011

set policy-options application-maps dcbx-iscsi-ptp-app-map application iSCSI code-points 111

set policy-options application-maps dcbx-iscsi-ptp-app-map application PTP code-points [001 101]

set protocols dcbx interface xe-0/0/10 application-map dcbx-iscsi-fcoe-app-map

set protocols dcbx interface xe-0/0/11 application-map dcbx-iscsi-ptp-app-map

set class-of-service congestion-notification-profile fcoe-cnp input ieee-802.1 code-point 011 pfc

set class-of-service interfaces xe-0/0/10 congestion-notification-profile fcoe-cnp

set class-of-service classifiers ieee-802.1 fcoe-iscsi-cl1 import default forwarding-class fcoe loss-priority high code-points 011

set class-of-service classifiers ieee-802.1 fcoe-iscsi-cl1 import default forwarding-class network-control loss-priority high code-points 111

set class-of-service classifiers ieee-802.1 iscsi-ptp-cl2 import default forwarding-class network-control loss-priority low code-points 111

set class-of-service classifiers ieee-802.1 iscsi-ptp-cl2 import default forwarding-class best-effort loss-priority low code-points [001 101]

set class-of-service interfaces xe-0/0/10 unit 0 classifiers ieee-802.1 fcoe-iscsi-cl1

set class-of-service interfaces xe-0/0/11 unit 0 classifiers ieee-802.1 iscsi-ptp-cl2

Configuring DCBX Application Protocol TLV Exchange

Step-by-Step Procedure

To define the applications, map the applications to IEEE 802.1p code points, apply the applications to interfaces, and create classifiers for DCBX application protocol exchange:

1. Define the iSCSI application by specifying its protocol and destination port, and define the FCoE and PTP applications by specifying their EtherTypes.

   ```
   [edit applications]
   user@switch# set application iSCSI protocol tcp destination-port 3260
   user@switch# set application FCoE ether-type 0x8906
   user@switch# set application PTP ether-type 0x88F7
   ```

2. Define an application map that maps the iSCSI and FCoE applications to IEEE 802.1p code points.

   ```
   [edit policy-options]
   user@switch# set application-maps dcbx-iscsi-fcoe-app-map application iSCSI code-points 111
   user@switch# set application-maps dcbx-iscsi-fcoe-app-map application FCoE code-points 011
   ```

3. Define the application map that maps the iSCSI and PTP applications to IEEE 802.1p code points.

   ```
   [edit policy-options]
   user@switch# set application-maps dcbx-iscsi-ptp-app-map application iSCSI code-points 111
   user@switch# set application-maps dcbx-iscsi-ptp-app-map application PTP code-points [001 101]
   ```

4. Apply the iSCSI and FCoE application map to interface **xe-0/0/10**, and apply the iSCSI and PTP application map to interface **xe-0/0/11**.

   ```
   [edit protocols dcbx]
   user@switch# set interface xe-0/0/10 application-map dcbx-iscsi-fcoe-app-map
   user@switch# set interface xe-0/0/11 application-map dcbx-iscsi-ptp-app-map
   ```

5. Create the congestion notification profile to enable PFC on the FCoE code point (**011**), and apply the congestion notification profile to interface **xe-0/0/10**.

   ```
   [edit class-of-service]
   user@switch# set congestion-notification-profile fcoe-cnp input ieee-802.1 code-point 011 pfc
   user@switch# set interfaces xe-0/0/10 congestion-notification-profile fcoe-cnp
   ```

6. Configure the classifier to apply to the interface that exchanges iSCSI and FCoE application information.

   ```
   [edit class-of-service classifiers]
   ```

user@switch# **set ieee-802.1 fcoe-iscsi-cl1 import default forwarding-class fcoe loss-priority high code-points 011**
user@switch# **set ieee-802.1 fcoe-iscsi-cl1 import default forwarding-class network-control loss-priority high code-points 111**

7. Configure the classifier to apply to the interface that exchanges iSCSI and PTP application information.

   [edit class-of-service classifiers]
   user@switch# **set ieee-802.1 iscsi-ptp-cl2 import default forwarding-class network-control loss-priority low code-points 111**
   user@switch# **set ieee-802.1 iscsi-ptp-cl2 import default forwarding-class best-effort loss-priority low code-points [001 101]**

8. Apply the classifiers to the appropriate interfaces.

   [edit class-of-service]
   user@switch# **set interfaces xe-0/0/10 unit 0 classifiers ieee-802.1 fcoe-iscsi-cl1**
   user@switch# **set interfaces xe-0/0/11 unit 0 classifiers ieee-802.1 iscsi-ptp-cl2**

Verification

IN THIS SECTION

To verify that DCBX application protocol exchange configuration has been created and is operating properly, perform these tasks:

Verifying the Application Configuration

Purpose

Verify that DCBX applications have been configured.

Action

List the applications by using the configuration mode command **show applications**:

```
user@switch# show applications
```

```
application iSCSI {
   protocol tcp;
   destination-port 3260;
}

application fcoe {
   ether-type 0x8906;
}

application ptp {
   ether-type 0x88F7;
}
```

Meaning

The **show applications** configuration mode command lists all of the configured applications and either their protocol and destination port (Layer 4 applications) or their EtherType (Layer 2 applications). The command output shows that the iSCSI application is configured with the **tcp** protocol and destination port **3260**, the FCoE application is configured with the EtherType **0x8906**, and that the PTP application is configured with the EtherType **0x88F7**.

Verifying the Application Map Configuration

Purpose

Verify that the application maps have been configured.

Action

List the application maps by using the configuration mode command **show policy-options application-maps**:

```
user@switch# show policy-options application-maps
```

```
dcbx-iscsi-fcoe-app-map {
   application iSCSI code-points 111;
   application FCoE code-points 011;
}

dcbx-iscsi-ptp-app-map {
   application iSCSI code-points 111;
```

```
    application PTP code-points [001 101];
}
```

Meaning

The **show policy-options application-maps** configuration mode command lists all of the configured application maps and the applications that belong to each application map. The command output shows that there are two application maps, **dcbx-iscsi-fcoe-app-map** and **dcbx-iscsi-ptp-app-map**.

The application map **dcbx-iscsi-fcoe-app-map** consists of the iSCSI application, which is mapped to IEEE 802.1p code point **111**, and the FCoE application, which is mapped to IEEE 802.1p code point **011**.

The application map **dcbx-iscsi-ptp-app-map** consists of the iSCSI application, which is mapped to IEEE 802.1p code point **111**, and the PTP application, which is mapped to IEEE 802.1p code points **001** and **101**.

Verifying DCBX Application Protocol Exchange Interface Configuration

Purpose

Verify that the application maps have been applied to the correct interfaces.

Action

List the application maps by using the configuration mode command **show protocols dcbx**:

`user@switch#` **show protocols dcbx**

```
interface xe-0/0/10.0 {
    application-map dcbx-iscsi-fcoe-app-map;
}

interface xe-0/0/11.0 {
    application-map dcbx-iscsi-ptp-app-map;
}
```

Meaning

The **show protocols dcbx** configuration mode command lists whether the interfaces are enabled for DCBX and lists the application map applied to each interface. The command output shows that interfaces **xe-0/0/10.0** and **xe-0/0/11.0** are enabled for DCBX, and that interface **xe-0/0/10.0** uses application map **dcbx-iscsi-fcoe-app-map**, and interface **xe-0/0/11.0** uses application map **dcbx-iscsi-ptp-app-map**.

Verifying the PFC Configuration

Purpose

Verify that PFC has been enabled on the FCoE code point and applied to the correct interface.

Action

Display the PFC configuration to verify that PFC is enabled on the FCoE code point (**011**) in the congestion notification profile **fcoe-cnp** by using the configuration mode command **show class-of-service congestion-notification-profile**:

`user@switch#` **show class-of-service congestion-notification-profile**

```
fcoe-cnp {
    input {
        ieee-802.1 {
            code-point 011 {
                pfc;
            }
        }
    }
}
```

Display the class-of-service (CoS) interface information to verify that the correct interface has PFC enabled for the FCoE application by using the configuration mode command **show class-of-service interfaces**:

`user@switch#` **show class-of-service interfaces**

```
xe-0/0/10 {
    congestion-notification-profile fcoe-cnp;
}
```

NOTE: The sample output does not include all of the information this command can show. The output is abbreviated to focus on verifying the PFC configuration.

Meaning

The **show class-of-service congestion-notification-profile** configuration mode command lists the configured congestion notification profiles. The command output shows that the congestion notification profile **fcoe-cnp** has been configured and has enabled PFC on the IEEE 802.1p code point **011** (the default FCoE code point).

The **show class-of-service interfaces** configuration mode command shows the interface CoS configuration. The command output shows that the congestion notification profile **fcoe-cnp**, which enables PFC on the FCoE code point, is applied to interface **xe-0/0/10**.

Verifying the Classifier Configuration

Purpose

Verify that the classifiers have been configured and applied to the correct interfaces.

Action

Display the classifier configuration by using the configuration mode command **show class-of-service**:

`user@switch#` **show class-of-service**

```
classifiers {
    ieee-802.1 fcoe-iscsi-cl1 {
        import default;
        forwarding-class network-control {
            loss-priority high code-points 111;
        }
        forwarding-class fcoe {
            loss-priority high code-points 011;
        }
    }
    ieee-802.1 iscsi-ptp-cl2 {
        import default;
        forwarding-class network-control {
            loss-priority low code-points 111;
        }
        forwarding-class best-effort {
            loss-priority low code-points [ 001 101 ];
        }
    }
}
interfaces {
    xe-0/0/10 {
        congestion-notification-profile fcoe-cnp;
        unit 0 {
            classifiers {
                ieee-802.1 fcoe-iscsi-cl1;
            }
        }
    }
    xe-0/0/11 {
        unit 0 {
            classifiers {
                ieee-802.1 iscsi-ptp-cl2;
            }
        }
```

```
    }
}
```

NOTE: The sample output does not include all of the information this command can show. The output is abbreviated to focus on verifying the classifier configuration.

Meaning

The **show class-of-service** configuration mode command lists the classifier and CoS interface configuration, as well as other information not shown in this example. The command output shows that there are two classifiers configured, **fcoe-iscsi-cl1** and **iscsi-ptp-cl2**.

Classifier **fcoe-iscsi-cl1** uses the **default** classifier as a template and edits the template as follows:

- The forwarding class **network-control** is set to a loss priority of **high** and is mapped to code point **111** (the code point mapped to the iSCSI application).
- The forwarding class **fcoe** is set to a loss priority of **high** and is mapped to code point **011** (the code point mapped by default to the FCoE application).

Classifier **iscsi-ptp-cl2** uses the **default** classifier as a template and edits the template as follows:

- The forwarding class **network-control** is set to a loss priority of **low** and is mapped to IEEE 802.1p code point **111** (the code point mapped to the iSCSI application).
- The forwarding class **best-effort** is set to a loss priority of **low** and is mapped to IEEE 802.1p code points **001** and **101** (the code points mapped by default to the PTP application).

The command output also shows that classifier **fcoe-iscsi-cl1** is mapped to interface **xe-0/0/10.0** and that classifier **iscsi-ptp-cl2** is mapped to interface **xe-0/0/11.0**.

RELATED DOCUMENTATION

Defining an Application for DCBX Application Protocol TLV Exchange | 473
Configuring an Application Map for DCBX Application Protocol TLV Exchange | 474
Applying an Application Map to an Interface for DCBX Application Protocol TLV Exchange | 475
Configuring DCBX Autonegotiation | 465
show dcbx | 975
show dcbx neighbors | 977
Understanding DCBX Application Protocol TLV Exchange | 468

CHAPTER 17

Lossless FCoE

IN THIS CHAPTER

Example: Configuring CoS PFC for FCoE Traffic

IN THIS SECTION

Priority-based flow control (PFC, described in IEEE 802.1Qbb) is a link-level flow control mechanism that you apply at ingress interfaces. PFC enables you to divide traffic on one physical link into eight priorities. You can think of the eight priorities as eight “lanes” of traffic that correspond to queues (forwarding classes). Each priority is mapped to a 3-bit IEEE 802.1p CoS value in the VLAN header.

You can selectively apply PFC to the traffic in any queue without pausing the traffic in other queues on the same link. You must apply PFC to FCoE traffic to ensure lossless transport.

This example describes how to configure PFC for FCoE traffic:

Requirements

This example uses the following hardware and software components:

- One switch
- Junos OS Release 11.1 or later for the QFX Series

Overview

FCoE traffic requires PFC to ensure lossless packet transport. This example shows you how to configure PFC on FCoE traffic, use the default FCoE forwarding-class-to-queue mapping and:

- Configure a classifier that associates the FCoE forwarding class with FCoE traffic, which is identified by IEEE 802.1p code point 011 (priority 3).
- Configure a congestion notification profile to apply PFC to the FCoE traffic.
- Apply the classifier and the PFC configuration to ingress interfaces.

> **NOTE:** Configuring or changing PFC on an interface blocks the entire port until the PFC change is completed. After a PFC change is completed, the port is unblocked and traffic resumes. Blocking the port stops ingress and egress traffic, and causes packet loss on all queues on the port until the port is unblocked.

- Configure the CoS bandwidth scheduling for the FCoE forwarding class output queue.
- On switches that support enhanced transmission selection (ETS) hierarchical port scheduling, create a forwarding class set (priority group) that includes the FCoE forwarding class; this is required to configure enhanced transmission selection (ETS) and support data center bridging (DCB).
- For ETS, configure the bandwidth scheduling for the FCoE priority group.
- Apply the configuration to ingress and egress interfaces. How this is done differs depending on whether you use ETS or direct port scheduling for the CoS configuration.

  For direct port scheduling, you apply a scheduler map directly to the interface. A scheduler map maps schedulers to forwarding classes, and applies the CoS properties of the scheduler to the output queue mapped to the forwarding class.

For ETS hierarchical port scheduling, you apply the scheduler map to a traffic control profile, and then apply the traffic control profile to the interface. The scheduler map maps CoS properties to forwarding classes (and their associated output queues) just as it does for direct port scheduling. The traffic control profile maps CoS properties to the priority group (a group of forwarding classes defined in a forwarding class set) that contains the forwarding class, creating a CoS hierarchy that allocates port bandwidth to a group of forwarding classes (priority group), and then allocates the priority group bandwidth to the individual forwarding classes.

Each interface in this example acts as both an ingress interface and an egress interface, so the classifier, congestion notification profile, and scheduling are applied to all of the interfaces.

Topology

Table 91 on page 492 shows the configuration components for this example.

Table 91: Components of the PFC for FCoE Traffic Configuration Topology

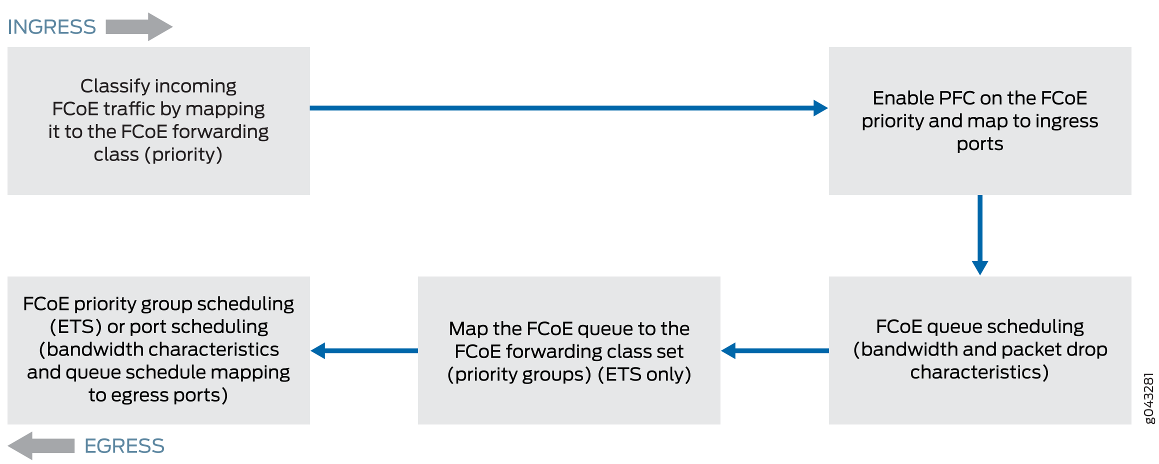
Component	Settings
Hardware	One switch
Behavior aggregate classifier (maps the FCoE forwarding class to incoming packets by IEEE 802.1 code point)	Code point **011** to forwarding class **fcoe** and loss priority **low** Ingress interfaces: **xe-0/0/31**, **xe-0/0/32**, **xe-0/0/33**, **xe-0/0/34**
PFC congestion notification profile	**fcoe-cnp**: Code point **011** Ingress interfaces: **xe-0/0/31**, **xe-0/0/32**, **xe-0/0/33**, **xe-0/0/34**
FCoE queue scheduler	**fcoe-sched**: Minimum bandwidth **3g** Maximum bandwidth **100%** Priority **low**
Forwarding class-to-scheduler mapping	Scheduler map **fcoe-map**: Forwarding class **fcoe** Scheduler **fcoe-sched** On switches that support direct port scheduling, if you use port scheduling, attach the scheduler map directly to interfaces **xe-0/0/31**, **xe-0/0/32**, **xe-0/0/33**, and **xe-0/0/34**.
ETS only: Forwarding class set (FCoE priority group)	**fcoe-pg**: Forwarding class **fcoe** Egress interfaces: **xe-0/0/31**, **xe-0/0/32**, **xe-0/0/33**, **xe-0/0/34**

Table 91: Components of the PFC for FCoE Traffic Configuration Topology *(continued)*

Component	Settings
ETS only: Traffic control profile	**fcoe-tcp**: Scheduler map **fcoe-map** Minimum bandwidth **3g** Maximum bandwidth **100%** For ETS hierarchical scheduling, attach the traffic control profile (using the **output-traffic-control-profile** keyword) to interfaces **xe-0/0/31**, **xe-0/0/32**, **xe-0/0/33**, and **xe-0/0/34**.

Figure 23 on page 493 shows a block diagram of the configuration components and the configuration flow of the CLI statements used in the example.

Figure 23: PFC for FCoE Traffic Configuration Components Block Diagram



Configuration

CLI Quick Configuration

To quickly configure PFC for FCoE traffic, 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.

The configuration is separated into the configuration common to ETS and direct port scheduling, and the portions of the configuration that apply only to ETS and only to port scheduling.

Common Configuration that applies to ETS Hierarchical Scheduling and to Port Scheduling:

```
[edit class-of-service]
```

set classifiers ieee-802.1 fcoe-classifier forwarding-class fcoe loss-priority low code-points 011
set congestion-notification-profile fcoe-cnp input ieee-802.1 code-point 011 pfc
set interfaces xe-0/0/31 unit 0 classifiers ieee-802.1 fcoe-classifier
set interfaces xe-0/0/32 unit 0 classifiers ieee-802.1 fcoe-classifier
set interfaces xe-0/0/33 unit 0 classifiers ieee-802.1 fcoe-classifier
set interfaces xe-0/0/34 unit 0 classifiers ieee-802.1 fcoe-classifier
set interfaces xe-0/0/31 congestion-notification-profile fcoe-cnp
set interfaces xe-0/0/32 congestion-notification-profile fcoe-cnp
set interfaces xe-0/0/33 congestion-notification-profile fcoe-cnp
set interfaces xe-0/0/34 congestion-notification-profile fcoe-cnp
set schedulers fcoe-sched priority low transmit-rate 3g
set schedulers fcoe-sched shaping-rate percent 100
set scheduler-maps fcoe-map forwarding-class fcoe scheduler fcoe-sched

Configuration for ETS hierarchical scheduling—the ETS-specific portion of this example configures forwarding class set (priority group) membership, priority group CoS settings (traffic control profile), and assigns the priority group and its CoS configuration to the interfaces:

```
[edit class-of-service]
```
set forwarding-class-sets fcoe-pg class fcoe
set traffic-control-profiles fcoe-tcp scheduler-map fcoe-map guaranteed-rate 3g
set traffic-control-profiles fcoe-tcp shaping-rate percent 100
set interfaces xe-0/0/31 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp
set interfaces xe-0/0/32 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp
set interfaces xe-0/0/33 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp
set interfaces xe-0/0/34 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp

Configuration for port scheduling—the port-scheduling-specific portion of this example assigns the scheduler map (which sets the CoS treatment of the forwarding classes in the scheduler map) to the interfaces:

```
[edit class-of-service]
```
set interfaces xe-0/0/31 scheduler-map fcoe-map
set interfaces xe-0/0/32 scheduler-map fcoe-map
set interfaces xe-0/0/33 scheduler-map fcoe-map
set interfaces xe-0/0/34 scheduler-map fcoe-map

Common Configuration (Applies to ETS Hierarchical Scheduling and to Port Scheduling)

Step-by-Step Procedure

To configure the ingress classifier for FCoE traffic, PFC on the FCoE traffic, apply the PFC and classifier configurations to interfaces, and configure queue scheduling, for both ETS hierarchical scheduling and port scheduling (common configuration):

1. Configure a classifier to set the loss priority and IEEE 802.1 code point assigned to the FCoE forwarding class at the ingress:

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

2. Configure PFC on the FCoE queue by applying FCoE to the IEEE 802.1 code point **011**:

   ```
   [edit class-of-service]
   user@switch# set congestion-notification-profile fcoe-cnp input ieee-802.1 code-point 011 pfc
   ```

3. Apply the PFC configuration to the ingress interfaces:

   ```
   [edit class-of-service]
   user@switch# set interfaces xe-0/0/31 congestion-notification-profile fcoe-cnp
   user@switch# set interfaces xe-0/0/32 congestion-notification-profile fcoe-cnp
   user@switch# set interfaces xe-0/0/33 congestion-notification-profile fcoe-cnp
   user@switch# set interfaces xe-0/0/34 congestion-notification-profile fcoe-cnp
   ```

4. Assign the classifier to the ingress interfaces:

   ```
   [edit class-of-service]
   user@switch# set interfaces xe-0/0/31 unit 0 classifiers ieee-802.1 fcoe-classifier
   user@switch# set interfaces xe-0/0/32 unit 0 classifiers ieee-802.1 fcoe-classifier
   user@switch# set interfaces xe-0/0/33 unit 0 classifiers ieee-802.1 fcoe-classifier
   user@switch# set interfaces xe-0/0/34 unit 0 classifiers ieee-802.1 fcoe-classifier
   ```

5. Configure output scheduling for the FCoE queue:

   ```
   [edit class-of-service]
   user@switch# set schedulers fcoe-sched priority low transmit-rate 3g
   user@switch# set schedulers fcoe-sched shaping-rate percent 100
   ```

6. Map the FCoE forwarding class to the FCoE scheduler:

   ```
   [edit class-of-service]
   ```

user@switch# **set scheduler-maps fcoe-map forwarding-class fcoe scheduler fcoe-sched**

ETS Hierarchical Scheduling Configuration

Step-by-Step Procedure

To configure the forwarding class set (priority group) and priority group scheduling (in a traffic control profile), and apply the ETS hierarchical scheduling for FCoE traffic to interfaces:

1. Configure the forwarding class set for the FCoE traffic:

   [edit class-of-service]
   user@switch# **set forwarding-class-sets fcoe-pg class fcoe**

2. Define the traffic control profile for the FCoE forwarding class set:

   [edit class-of-service]
   user@switch# **set traffic-control-profiles fcoe-tcp scheduler-map fcoe-map guaranteed-rate 3g**
   user@switch# **set traffic-control-profiles fcoe-tcp shaping-rate percent 100**

3. Apply the FCoE forwarding class set and traffic control profile to the egress ports:

   [edit class-of-service]
   user@switch# **set interfaces xe-0/0/31 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp**
   user@switch# **set interfaces xe-0/0/32 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp**
   user@switch# **set interfaces xe-0/0/33 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp**
   user@switch# **set interfaces xe-0/0/34 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp**

Port Scheduling Configuration

Step-by-Step Procedure

To apply port scheduling for FCoE traffic to interfaces:

1. Apply the scheduler map to the egress ports:

```
[edit class-of-service]
user@switch# set interfaces xe-0/0/31 scheduler-map fcoe-map
user@switch# set interfaces xe-0/0/32 scheduler-map fcoe-map
user@switch# set interfaces xe-0/0/33 scheduler-map fcoe-map
user@switch# set interfaces xe-0/0/34 scheduler-map fcoe-map
```

Results

Display the results of the configuration (the system shows only the explicitly configured parameters; it does not show default parameters such as the **fcoe** lossless forwarding class). The results are from the ETS hierarchical scheduling configuration to show the more complex configuration. Direct port scheduling results would not show the traffic control profile or forwarding class set portions of the configuration, and would display the name of the scheduler map under each interface (instead of the names of the forwarding class set and output traffic control profile), but is otherwise the same.

```
user@switch> show configuration class-of-service
classifiers {
  ieee-802.1 fcoe-classifier {
    forwarding-class fcoe {
      loss-priority low code-points 011;
    }
  }
  traffic-control-profiles {
    fcoe-tcp {
      scheduler-map fcoe-map;
      shaping-rate percent 100;
      guaranteed-rate 3000000000;
    }
  }
  forwarding-class-sets {
    fcoe-pg {
      class fcoe;
    }
  }
  congestion-notification-profile {
    fcoe-cnp {
      input {
        ieee-802.1 {
          code-point 011 {
```

```
                    pfc;
                }
            }
        }
    }
}
interfaces {
    xe-0/0/31 {
        congestion-notification-profile fcoe-cnp;
        forwarding-class-set {
            fcoe-pg {
                output-traffic-control-profile fcoe-tcp;
            }
        }
        unit 0 {
            classifiers {
                ieee-802.1 fcoe-classifier;
            }
        }
    }
    xe-0/0/32 {
        congestion-notification-profile fcoe-cnp;
        forwarding-class-set {
            fcoe-pg {
                output-traffic-control-profile fcoe-tcp;
            }
        }
        unit 0 {
            classifiers {
                ieee-802.1 fcoe-classifier;
            }
        }
    }
    xe-0/0/33 {
        congestion-notification-profile fcoe-cnp;
        forwarding-class-set {
            fcoe-pg {
                output-traffic-control-profile fcoe-tcp;
            }
        }
        unit 0 {
            classifiers {
                ieee-802.1 fcoe-classifier;
            }
```

```
        }
      }
      xe-0/0/34 {
        congestion-notification-profile fcoe-cnp;
        forwarding-class-set {
          fcoe-pg {
            output-traffic-control-profile fcoe-tcp;
          }
        }
        unit 0 {
          classifiers {
            ieee-802.1 fcoe-classifier;
          }
        }
      }
    }
    scheduler-maps {
      fcoe-map {
        forwarding-class fcoe scheduler fcoe-sched;
      }
    }
    schedulers {
      fcoe-sched {
        transmit-rate 3000000000;
        shaping-rate percent 100;
        priority low;
      }
    }
```

TIP: To quickly configure the interfaces, issue the **load merge terminal** command and then copy the hierarchy and paste it into the switch terminal window.

Verification

IN THIS SECTION

To verify that the PFC configuration for FCoE traffic components has been created and is operating properly, perform these tasks:

Verifying That Priority-Based Flow Control Has Been Enabled

Purpose

Verify that PFC is enabled on the FCoE queue to enable lossless transport.

Action

List the congestion notification profiles using the operational mode command **show class-of-service congestion-notification**:

`user@switch>` **show class-of-service congestion-notification**

```
Type: Input, Name: fcoe-cnp, Index: 51697
Cable Length: 100 m
  Priority     PFC          MRU
  000          Disabled
  001          Disabled
  010          Disabled
  011          Enabled      2500
  100          Disabled
  101          Disabled
  110          Disabled
  111          Disabled
Type: Output
  Priority     Flow-Control-Queues
  000
               0
  001
               1
  010
               2
  011
               3
  100
               4
  101
               5
  110
               6
  111
               7
```

Meaning

The **show class-of-service congestion-notification** operational command lists all of the congestion notification profiles and which IEEE 802.1p code points have PFC enabled. The command output shows that PFC is enabled on code point **011** for the **fcoe-cnp** congestion notification profile.

The command also shows the default cable length (100 meters), the default maximum receive unit (2500 bytes), and the default mapping of priorities to output queues because this example does not include configuring these options.

Verifying the Ingress Interface PFC Configuration

Purpose

Verify that the classifier **fcoe-classifier** and the congestion notification profile **fcoe-cnp** are configured on ingress interfaces **xe-0/0/31**, **xe-0/0/32**, **xe-0/0/33**, and **xe-0/0/34**.

Action

List the ingress interfaces using the operational mode command **show configuration class-of-service interfaces**:

user@switch> **show configuration class-of-service interfaces xe-0/0/31**

```
congestion-notification-profile fcoe-cnp;
unit 0 {
    classifiers {
        ieee-802.1 fcoe-classifier;
    }
}
```

user@switch> **show configuration class-of-service interfaces xe-0/0/32**

```
congestion-notification-profile fcoe-cnp;
unit 0 {
    classifiers {
        ieee-802.1 fcoe-classifier;
    }
}
```

user@switch> **show configuration class-of-service interfaces xe-0/0/33**

```
congestion-notification-profile fcoe-cnp;
unit 0 {
    classifiers {
        ieee-802.1 fcoe-classifier;
```

```
    }
}
```

user@switch> **show configuration class-of-service interfaces xe-0/0/34**

```
congestion-notification-profile fcoe-cnp;
unit 0 {
    classifiers {
        ieee-802.1 fcoe-classifier;
    }
}
```

Meaning

The **show configuration class-of-service interfaces** commands list the congestion notification profile that is mapped to the interface (**fcoe-cnp**) and the IEEE 802.1p classifier associated with the interface (**fcoe-classifier**).

RELATED DOCUMENTATION

Understanding CoS Flow Control (Ethernet PAUSE and PFC) | 197

Example: Configuring CoS for FCoE Transit Switch Traffic Across an MC-LAG

IN THIS SECTION

Multichassis link aggregation groups (MC-LAGs) provide redundancy and load balancing between two switches, multihoming support for client devices such as servers, and a loop-free Layer 2 network without running Spanning Tree Protocol (STP).

NOTE: This example uses Junos OS without support for the Enhanced Layer 2 Software (ELS) configuration style. If your switch runs software that does support ELS, see "Example: Configuring CoS Using ELS for FCoE Transit Switch Traffic Across an MC-LAG" on page 533. For ELS details, see *Using the Enhanced Layer 2 Software CLI*.

You can use an MC-LAG to provide a redundant aggregation layer for Fibre Channel over Ethernet (FCoE) traffic in an *inverted-U* topology. To support lossless transport of FCoE traffic across an MC-LAG, you must configure the appropriate class of service (CoS) on both of the switches with MC-LAG port members. The CoS configuration must be the same on both of the MC-LAG switches because an MC-LAG does not carry forwarding class and IEEE 802.1p priority information.

NOTE: This example describes how to configure CoS to provide lossless transport for FCoE traffic across an MC-LAG that connects two switches. It also describes how to configure CoS on the FCoE transit switches that connect FCoE hosts to the two switches that form the MC-LAG.

This example does *not* describe how to configure the MC-LAG itself. However, this example includes a subset of MC-LAG configuration that only shows how to configure interface membership in the MC-LAG.

Ports that are part of an FCoE-FC gateway configuration (a virtual FCoE-FC gateway fabric) do not support MC-LAGs. Ports that are members of an MC-LAG act as FCoE pass-through transit switch ports.

QFX Series switches and EX4600 switches support MC-LAGs. QFabric system Node devices do not support MC-LAGs.

Requirements

This example uses the following hardware and software components:

- Two Juniper Networks QFX3500 switches that form an MC-LAG for FCoE traffic.
- Two Juniper Networks QFX3500 switches that provide FCoE server access in transit switch mode and that connect to the MC-LAG switches. These switches can be standalone QFX3500 switches or they can be Node devices in a QFabric system.
- FCoE servers (or other FCoE hosts) connected to the transit switches.
- Junos OS Release 12.2 or later for the QFX Series.

Overview

FCoE traffic requires lossless transport. This example shows you how to:

- Configure CoS for FCoE traffic on the two QFX3500 switches that form the MC-LAG, including priority-based flow control (PFC) and enhanced transmission selection (ETS; hierarchical scheduling of resources for the FCoE forwarding class priority and for the forwarding class set priority group).

  > **NOTE:** Configuring or changing PFC on an interface blocks the entire port until the PFC change is completed. After a PFC change is completed, the port is unblocked and traffic resumes. Blocking the port stops ingress and egress traffic, and causes packet loss on all queues on the port until the port is unblocked.

- Configure CoS for FCoE on the two FCoE transit switches that connect FCoE hosts to the MC-LAG switches and enable FIP snooping on the FCoE VLAN at the FCoE transit switch access ports.
- Disable IGMP snooping on the FCoE VLAN.

  > **NOTE:** This is only necessary if IGMP snooping is enabled on the VLAN. Before Junos OS Release 13.2, IGMP snooping was enabled by default on VLANs. Beginning with Junos OS Release 13.2, IGMP snooping is enabled by default only on the default VLAN.

- Configure the appropriate port mode, MTU, and FCoE trusted or untrusted state for each interface to support lossless FCoE transport.

Topology

Switches that act as transit switches support MC-LAGs for FCoE traffic in an inverted-U network topology, as shown in Figure 24 on page 505.

Figure 24: Supported Topology for an MC-LAG on an FCoE Transit Switch

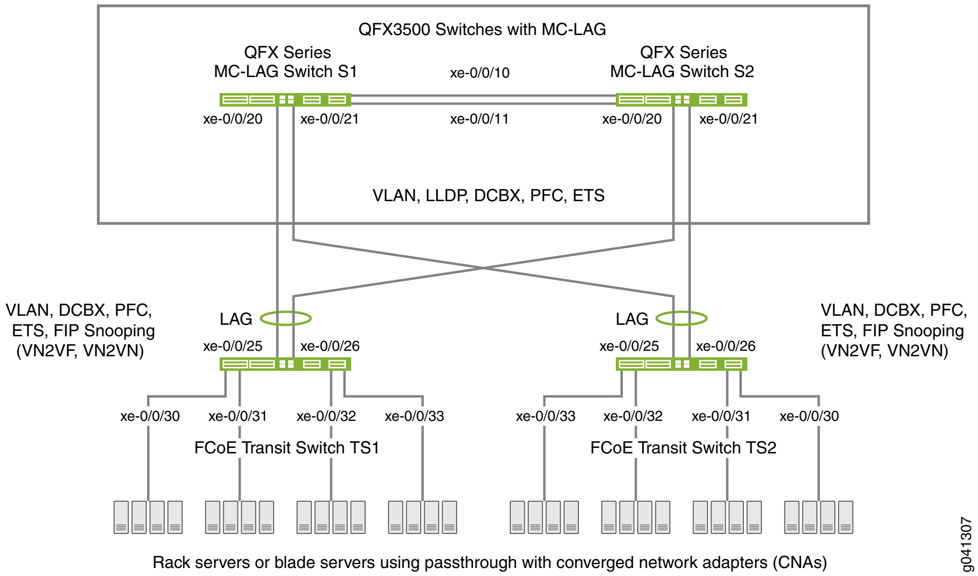


Table 92 on page 505 shows the configuration components for this example.

Table 92: Components of the CoS for FCoE Traffic Across an MC-LAG Configuration Topology

Component	Settings
Hardware	Four QFX3500 switches (two to form the MC-LAG as pass-through transit switches and two transit switches for FCoE access).
Forwarding class (all switches)	Default **fcoe** forwarding class.
Classifier (forwarding class mapping of incoming traffic to IEEE priority)	Default IEEE 802.1p trusted classifier on all FCoE interfaces.

Table 92: Components of the CoS for FCoE Traffic Across an MC-LAG Configuration Topology *(continued)*

Component	Settings
LAGs and MC-LAG	S1—Ports xe-0/0/10 and x-0/0/11 are members of LAG ae0, which connects Switch S1 to Switch S2. Ports xe-0/0/20 and xe-0/0/21 are members of MC-LAG ae1. All ports are configured in **trunk** port mode, as **fcoe-trusted**, and with an MTU of **2180**. S2—Ports xe-0/0/10 and x-0/0/11 are members of LAG ae0, which connects Switch S2 to Switch S1. Ports xe-0/0/20 and xe-0/0/21 are members of MC-LAG ae1. All ports are configured in **trunk** port mode, as **fcoe-trusted**, and with an MTU of **2180**. **NOTE:** Ports xe-0/0/20 and xe-0/0/21 on Switches S1 and S2 are the members of the MC-LAG. TS1—Ports xe-0/0/25 and x-0/0/26 are members of LAG ae1, configured in **trunk** port mode, as **fcoe-trusted**, and with an MTU of **2180**. Ports xe-0/0/30, xe-0/0/31, xe-0/0/32, and xe-0/0/33 are configured in **tagged-access** port mode, with an MTU of **2180**. TS2—Ports xe-0/0/25 and x-0/0/26 are members of LAG ae1, configured in **trunk** port mode, as **fcoe-trusted**, and with an MTU of **2180**. Ports xe-0/0/30, xe-0/0/31, xe-0/0/32, and xe-0/0/33 are configured in **tagged-access** port mode, with an MTU of **2180**.
FCoE queue scheduler (all switches)	**fcoe-sched**: Minimum bandwidth **3g** Maximum bandwidth **100%** Priority **low**
Forwarding class-to-scheduler mapping (all switches)	Scheduler map **fcoe-map**: Forwarding class **fcoe** Scheduler **fcoe-sched**

Table 92: Components of the CoS for FCoE Traffic Across an MC-LAG Configuration Topology *(continued)*

Component	Settings
Forwarding class set (FCoE priority group, all switches)	**fcoe-pg**: Forwarding class **fcoe** Egress interfaces: • S1—LAG ae0 and MC-LAG ae1 • S2—LAG ae0 and MC-LAG ae1 • TS1—LAG ae1, interfaces xe-0/0/30, xe-0/0/31, xe-0/0/32, and xe-0/0/33 • TS2—LAG ae1, interfaces xe-0/0/30, xe-0/0/31, xe-0/0/32, and xe-0/0/33
Traffic control profile (all switches)	**fcoe-tcp**: Scheduler map **fcoe-map** Minimum bandwidth **3g** Maximum bandwidth **100**%
PFC congestion notification profile (all switches)	**fcoe-cnp**: Code point **011** Ingress interfaces: • S1—LAG ae0 and MC-LAG ae1 • S2—LAG ae0 and MC-LAG ae1 • TS1—LAG ae1, interfaces xe-0/0/30, xe-0/0/31, xe-0/0/32, and xe-0/0/33 • TS2—LAG ae1, interfaces xe-0/0/30, xe-0/0/31, xe-0/0/32, and xe-0/0/33
FCoE VLAN name and tag ID	Name—**fcoe_vlan** ID—**100** Include the FCoE VLAN on the interfaces that carry FCoE traffic on all four switches. Disable IGMP snooping on the interfaces that belong to the FCoE VLAN on all four switches.

Table 92: Components of the CoS for FCoE Traffic Across an MC-LAG Configuration Topology *(continued)*

Component	Settings
FIP snooping	Enable FIP snooping on Transit Switches TS1 and TS2 on the FCoE VLAN. Configure the LAG interfaces that connect to the MC-LAG switches as FCoE trusted interfaces so that they do not perform FIP snooping. This example enables VN2VN_Port FIP snooping on the FCoE transit switch interfaces connected to the FCoE servers. The example is equally valid with VN2VF_Port FIP snooping enabled on the transit switch access ports. The method of FIP snooping you enable depends on your network configuration.

NOTE: This example uses the default IEEE 802.1p trusted BA classifier, which is automatically applied to trunk mode and tagged access mode ports if you do not apply an explicitly configured classifier.

To configure CoS for FCoE traffic across an MC-LAG:

- Use the default FCoE forwarding class and forwarding-class-to-queue mapping (do not explicitly configure the FCoE forwarding class or output queue). The default FCoE forwarding class is **fcoe**, and the default output queue is queue **3**.

  NOTE: In Junos OS Release 12.2, traffic mapped to explicitly configured forwarding classes, even lossless forwarding classes such as **fcoe**, is treated as lossy (**best-effort**) traffic and does *not* receive lossless treatment. To receive lossless treatment in Release 12.2, traffic must use one of the default lossless forwarding classes (**fcoe** or **no-loss**).

  In Junos OS Release 12.3 and later, you can include the *no-loss* packet drop attribute in the explicit forwarding class configuration to configure a lossless forwarding class.

- Use the default trusted BA classifier, which maps incoming packets to forwarding classes by the IEEE 802.1p code point (CoS priority) of the packet. The trusted classifier is the default classifier for interfaces in trunk and tagged-access port modes. The default trusted classifier maps incoming packets with the IEEE 802.1p code point 3 (**011**) to the FCoE forwarding class. If you choose to configure the BA classifier instead of using the default classifier, you must ensure that FCoE traffic is classified into forwarding classes in exactly the same way on both MC-LAG switches. Using the default classifier ensures consistent classifier configuration on the MC-LAG ports.

- Configure a congestion notification profile that enables PFC on the FCoE code point (code point **011** in this example). The congestion notification profile configuration must be the same on both MC-LAG switches.
- Apply the congestion notification profile to the interfaces.
- Configure enhanced transmission selection (ETS, also known as hierarchical scheduling) on the interfaces to provide the bandwidth required for lossless FCoE transport. Configuring ETS includes configuring bandwidth scheduling for the FCoE forwarding class, a forwarding class set (priority group) that includes the FCoE forwarding class, and a traffic control profile to assign bandwidth to the forwarding class set that includes FCoE traffic.
- Apply the ETS scheduling to the interfaces.
- Configure the port mode, MTU, and FCoE trusted or untrusted state for each interface to support lossless FCoE transport.

In addition, this example describes how to enable FIP snooping on the Transit Switch TS1 and TS2 ports that are connected to the FCoE servers and how to disable IGMP snooping on the FCoE VLAN. To provide secure access, FIP snooping must be enabled on the FCoE access ports.

This example focuses on the CoS configuration to support lossless FCoE transport across an MC-LAG. This example does not describe how to configure the properties of MC-LAGs and LAGs, although it does show you how to configure the port characteristics required to support lossless transport and how to assign interfaces to the MC-LAG and to the LAGs.

Before you configure CoS, configure:

- The MC-LAGs that connect Switches S1 and S2 to Switches TS1 and TS2.
- The LAGs that connect the Transit Switches TS1 and TS2 to MC-LAG Switches S1 and S2.
- The LAG that connects Switch S1 to Switch S2.

Configuration

IN THIS SECTION

To configure CoS for lossless FCoE transport across an MC-LAG, perform these tasks:

CLI Quick Configuration

To quickly configure CoS for lossless FCoE transport across an MC-LAG, 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 for MC-LAG Switch S1 and MC-LAG Switch S2 at the **[edit]** hierarchy level. The configurations on Switches S1 and S2 are identical because the CoS configuration must be identical, and because this example uses the same ports on both switches.

Switch S1 and Switch S2

```
set class-of-service schedulers fcoe-sched priority low transmit-rate 3g
set class-of-service schedulers fcoe-sched shaping-rate percent 100
set class-of-service scheduler-maps fcoe-map forwarding-class fcoe scheduler fcoe-sched
set class-of-service forwarding-class-sets fcoe-pg class fcoe
set class-of-service traffic-control-profiles fcoe-tcp scheduler-map fcoe-map guaranteed-rate 3g
set class-of-service traffic-control-profiles fcoe-tcp shaping-rate percent 100
set class-of-service interfaces ae0 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp
set class-of-service interfaces ae1 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp
set class-of-service congestion-notification-profile fcoe-cnp input ieee-802.1 code-point 011 pfc
set class-of-service interfaces ae0 congestion-notification-profile fcoe-cnp
set class-of-service interfaces ae1 congestion-notification-profile fcoe-cnp
set vlans fcoe_vlan vlan-id 100
set protocols igmp-snooping vlan fcoe_vlan disable
set interfaces xe-0/0/10 ether-options 802.3ad ae0
set interfaces xe-0/0/11 ether-options 802.3ad ae0
set interfaces xe-0/0/20 ether-options 802.3ad ae1
set interfaces xe-0/0/21 ether-options 802.3ad ae1
set interfaces ae0 unit 0 family ethernet-switching port-mode trunk vlan members fcoe_vlan
set interfaces ae1 unit 0 family ethernet-switching port-mode trunk vlan members fcoe_vlan
set interfaces ae0 mtu 2180
set interfaces ae1 mtu 2180
set ethernet-switching-options secure-access-port interface ae0 fcoe-trusted
set ethernet-switching-options secure-access-port interface ae1 fcoe-trusted
```

To quickly configure CoS for lossless FCoE transport across an MC-LAG, 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 for Transit Switch TS1 and Transit Switch TS2 at the **[edit]** hierarchy level. The configurations on Switches TS1 and TS2 are identical because the CoS configuration must be identical, and because this example uses the same ports on both switches.

Switch TS1 and Switch TS2

```
set class-of-service schedulers fcoe-sched priority low transmit-rate 3g
set class-of-service schedulers fcoe-sched shaping-rate percent 100
set class-of-service scheduler-maps fcoe-map forwarding-class fcoe scheduler fcoe-sched
set class-of-service forwarding-class-sets fcoe-pg class fcoe
set class-of-service traffic-control-profiles fcoe-tcp scheduler-map fcoe-map guaranteed-rate 3g
set class-of-service traffic-control-profiles fcoe-tcp shaping-rate percent 100
set class-of-service interfaces ae1 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp
set class-of-service interfaces xe-0/0/30 forwarding-class-set fcoe-pg output-traffic-control-profile
fcoe-tcp
set class-of-service interfaces xe-0/0/31 forwarding-class-set fcoe-pg output-traffic-control-profile
fcoe-tcp
set class-of-service interfaces xe-0/0/32 forwarding-class-set fcoe-pg output-traffic-control-profile
fcoe-tcp
set class-of-service interfaces xe-0/0/33 forwarding-class-set fcoe-pg output-traffic-control-profile
fcoe-tcp
set class-of-service congestion-notification-profile fcoe-cnp input ieee-802.1 code-point 011 pfc
set class-of-service interfaces ae1 congestion-notification-profile fcoe-cnp
set class-of-service interfaces xe-0/0/30 congestion-notification-profile fcoe-cnp
set class-of-service interfaces xe-0/0/31 congestion-notification-profile fcoe-cnp
set class-of-service interfaces xe-0/0/32 congestion-notification-profile fcoe-cnp
set class-of-service interfaces xe-0/0/33 congestion-notification-profile fcoe-cnp
set vlans fcoe_vlan vlan-id 100
set protocols igmp-snooping vlan fcoe_vlan disable
set interfaces xe-0/0/25 ether-options 802.3ad ae1
set interfaces xe-0/0/26 ether-options 802.3ad ae1
set interfaces ae1 unit 0 family ethernet-switching port-mode trunk vlan members fcoe_vlan
set interfaces xe-0/0/30 unit 0 family ethernet-switching port-mode tagged-access vlan members
fcoe_vlan
```

set interfaces xe-0/0/31 unit 0 family ethernet-switching port-mode tagged-access vlan members fcoe_vlan
set interfaces xe-0/0/32 unit 0 family ethernet-switching port-mode tagged-access vlan members fcoe_vlan
set interfaces xe-0/0/33 unit 0 family ethernet-switching port-mode tagged-access vlan members fcoe_vlan
set interfaces ae1 mtu 2180
set interfaces xe-0/0/30 mtu 2180
set interfaces xe-0/0/31 mtu 2180
set interfaces xe-0/0/32 mtu 2180
set interfaces xe-0/0/33 mtu 2180
set ethernet-switching-options secure-access-port interface ae1 fcoe-trusted
set ethernet-switching-options secure-access-port vlan fcoe_vlan examine-fip examine-vn2v2 beacon-period 90000

Configuring MC-LAG Switches S1 and S2

Step-by-Step Procedure

To configure CoS resource scheduling (ETS), PFC, the FCoE VLAN, and the LAG and MC-LAG interface membership and characteristics to support lossless FCoE transport across an MC-LAG (this example uses the default **fcoe** forwarding class and the default classifier to map incoming FCoE traffic to the FCoE IEEE 802.1p code point **011**, so you do not configure them):

1. Configure output scheduling for the FCoE queue.

   ```
   [edit class-of-service schedulers fcoe-sched]
   user@switch# set priority low transmit-rate 3g
   user@switch# set shaping-rate percent 100
   ```

2. Map the FCoE forwarding class to the FCoE scheduler (**fcoe-sched**).

   ```
   [edit class-of-service]
   user@switch# set scheduler-maps fcoe-map forwarding-class fcoe scheduler fcoe-sched
   ```

3. Configure the forwarding class set (**fcoe-pg**) for the FCoE traffic.

   ```
   [edit class-of-service]
   user@switch# set forwarding-class-sets fcoe-pg class fcoe
   ```

4. Define the traffic control profile (**fcoe-tcp**) to use on the FCoE forwarding class set.

```
[edit class-of-service traffic-control-profiles fcoe-tcp]
user@switch# set scheduler-map fcoe-map guaranteed-rate 3g
user@switch# set shaping-rate percent 100
```

5. Apply the FCoE forwarding class set and traffic control profile to the LAG and MC-LAG interfaces.

```
[edit class-of-service]
user@switch# set interfaces ae0 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp
user@switch# set interfaces ae1 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp
```

6. Enable PFC on the FCoE priority by creating a congestion notification profile (**fcoe-cnp**) that applies FCoE to the IEEE 802.1 code point **011**.

```
[edit class-of-service]
user@switch# set congestion-notification-profile fcoe-cnp input ieee-802.1 code-point 011 pfc
```

7. Apply the PFC configuration to the LAG and MC-LAG interfaces.

```
[edit class-of-service]
user@switch# set interfaces ae0 congestion-notification-profile fcoe-cnp
user@switch# set interfaces ae1 congestion-notification-profile fcoe-cnp
```

8. Configure the VLAN for FCoE traffic (**fcoe_vlan**).

```
[edit vlans]
user@switch# set fcoe_vlan vlan-id 100
```

9. Disable IGMP snooping on the FCoE VLAN.

```
[edit protocols]
user@switch# set igmp-snooping vlan fcoe_vlan disable
```

10. Add the member interfaces to the LAG between the two MC-LAG switches.

```
[edit interfaces]
user@switch# set xe-0/0/10 ether-options 802.3ad ae0
user@switch# set xe-0/0/11 ether-options 802.3ad ae0
```

11. Add the member interfaces to the MC-LAG.

```
[edit interfaces]
user@switch# set xe-0/0/20 ether-options 802.3ad ae1
user@switch# set xe-0/0/21 ether-options 802.3ad ae1
```

12. Configure the port mode as **trunk** and membership in the FCoE VLAN (**fcoe_vlan**)for the LAG (ae0) and for the MC-LAG (ae1).

```
[edit interfaces]
user@switch# set ae0 unit 0 family ethernet-switching port-mode trunk vlan members fcoe_vlan
user@switch# set ae1 unit 0 family ethernet-switching port-mode trunk vlan members fcoe_vlan
```

13. Set the MTU to **2180** for the LAG and MC-LAG interfaces.

    2180 bytes is the minimum size required to handle FCoE packets because of the payload and header sizes. You can configure the MTU to a higher number of bytes if desired, but not less than 2180 bytes.

```
[edit interfaces]
user@switch# set ae0 mtu 2180
user@switch# set ae1 mtu 2180
```

14. Set the LAG and MC-LAG interfaces as FCoE trusted ports.

    Ports that connect to other switches should be trusted and should not perform FIP snooping.

```
[edit ethernet-switching-options secure-access-port interface]
user@switch# set ae0 fcoe-trusted
user@switch# set ae1 fcoe-trusted
```

Configuring FCoE Transit Switches TS1 and TS2

Step-by-Step Procedure

The CoS configuration on FCoE Transit Switches TS1 and TS2 is similar to the CoS configuration on MC-LAG Switches S1 and S2. However, the port configurations differ, and you must enable FIP snooping on the Switch TS1 and Switch TS2 FCoE access ports.

To configure resource scheduling (ETS), PFC, the FCoE VLAN, and the LAG interface membership and characteristics to support lossless FCoE transport across the MC-LAG (this example uses the default **fcoe** forwarding class and the default classifier to map incoming FCoE traffic to the FCoE IEEE 802.1p code point **011**, so you do not configure them):

1. Configure output scheduling for the FCoE queue.

   ```
   [edit class-of-service schedulers fcoe-sched]
   user@switch# set priority low transmit-rate 3g
   user@switch# set shaping-rate percent 100
   ```

2. Map the FCoE forwarding class to the FCoE scheduler (**fcoe-sched**).

   ```
   [edit class-of-service]
   user@switch# set scheduler-maps fcoe-map forwarding-class fcoe scheduler fcoe-sched
   ```

3. Configure the forwarding class set (**fcoe-pg**) for the FCoE traffic.

   ```
   [edit class-of-service]
   user@switch# set forwarding-class-sets fcoe-pg class fcoe
   ```

4. Define the traffic control profile (**fcoe-tcp**) to use on the FCoE forwarding class set.

   ```
   [edit class-of-service]
   user@switch# set traffic-control-profiles fcoe-tcp scheduler-map fcoe-map guaranteed-rate 3g
   user@switch# set traffic-control-profiles fcoe-tcp shaping-rate percent 100
   ```

5. Apply the FCoE forwarding class set and traffic control profile to the LAG interface and to the FCoE access interfaces.

   ```
   [edit class-of-service]
   user@switch# set interfaces ae1 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp
   user@switch# set interfaces xe-0/0/30 forwarding-class-set fcoe-pg output-traffic-control-profile
   fcoe-tcp
   user@switch# set interfaces xe-0/0/31 forwarding-class-set fcoe-pg output-traffic-control-profile
   fcoe-tcp
   ```

user@switch# **set interfaces xe-0/0/32 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp**

user@switch# **set interfaces xe-0/0/33 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp**

6. Enable PFC on the FCoE priority by creating a congestion notification profile (**fcoe-cnp**) that applies FCoE to the IEEE 802.1 code point **011**.

   [edit class-of-service]
   user@switch# **set congestion-notification-profile fcoe-cnp input ieee-802.1 code-point 011 pfc**

7. Apply the PFC configuration to the LAG interface and to the FCoE access interfaces.

   [edit class-of-service]
   user@switch# **set interfaces ae1 congestion-notification-profile fcoe-cnp**
   user@switch# **set interfaces xe-0/0/30 congestion-notification-profile fcoe-cnp**
   user@switch# **set interfaces xe-0/0/31 congestion-notification-profile fcoe-cnp**
   user@switch# **set interfaces xe-0/0/32 congestion-notification-profile fcoe-cnp**
   user@switch# **set interfaces xe-0/0/33 congestion-notification-profile fcoe-cnp**

8. Configure the VLAN for FCoE traffic (**fcoe_vlan**).

   [edit vlans]
   user@switch# **set fcoe_vlan vlan-id 100**

9. Disable IGMP snooping on the FCoE VLAN.

   [edit protocols]
   user@switch# **set igmp-snooping vlan fcoe_vlan disable**

10. Add the member interfaces to the LAG.

   [edit interfaces]
   user@switch# **set xe-0/0/25 ether-options 802.3ad ae1**
   user@switch# **set xe-0/0/26 ether-options 802.3ad ae1**

11. On the LAG (**ae1**), configure the port mode as **trunk** and membership in the FCoE VLAN (**fcoe_vlan**).

   [edit interfaces]

`user@switch#` **set ae1 unit 0 family ethernet-switching port-mode trunk vlan members fcoe_vlan**

12. On the FCoE access interfaces (**xe-0/0/30**, **xe-0/0/31**, **xe-0/0/32**, **xe-0/0/33**), configure the port mode as **tagged-access** and membership in the FCoE VLAN (**fcoe_vlan**).

    `[edit interfaces]`
    `user@switch#` **set xe-0/0/30 unit 0 family ethernet-switching port-mode tagged-access vlan members fcoe_vlan**
    `user@switch#` **set xe-0/0/31 unit 0 family ethernet-switching port-mode tagged-access vlan members fcoe_vlan**
    `user@switch#` **set xe-0/0/32 unit 0 family ethernet-switching port-mode tagged-access vlan members fcoe_vlan**
    `user@switch#` **set xe-0/0/33 unit 0 family ethernet-switching port-mode tagged-access vlan members fcoe_vlan**

13. Set the MTU to **2180** for the LAG and FCoE access interfaces.

    2180 bytes is the minimum size required to handle FCoE packets because of the payload and header sizes; you can configure the MTU to a higher number of bytes if desired, but not less than 2180 bytes.

    `[edit interfaces]`
    `user@switch#` **set ae1 mtu 2180**
    `user@switch#` **set xe-0/0/30 mtu 2180**
    `user@switch#` **set xe-0/0/31 mtu 2180**
    `user@switch#` **set xe-0/0/32 mtu 2180**
    `user@switch#` **set xe-0/0/33 mtu 2180**

14. Set the LAG interface as an FCoE trusted port. Ports that connect to other switches should be trusted and should not perform FIP snooping:

    `[edit ethernet-switching-options]`
    `user@switch#` **set secure-access-port interface ae1 fcoe-trusted**

    NOTE: Access ports xe-0/0/30, xe-0/0/31, xe-0/0/32, and xe-0/0/33 are not configured as FCoE trusted ports. The access ports remain in the default state as untrusted ports because they connect directly to FCoE devices and must perform FIP snooping to ensure network security.

15. Enable FIP snooping on the FCoE VLAN to prevent unauthorized FCoE network access (this example uses VN2VN_Port FIP snooping; the example is equally valid if you use VN2VF_Port FIP snooping).

    ```
    [edit ethernet-switching-options]
    user@switch# set secure-access-port vlan fcoe_vlan examine-fip examine-vn2vn beacon-period 90000
    ```

Results

Display the results of the CoS configuration on MC-LAG Switch S1 and on MC-LAG Switch S2 (the results on both switches are the same).

```
user@switch> show configuration class-of-service
traffic-control-profiles {
  fcoe-tcp {
    scheduler-map fcoe-map;
    shaping-rate percent 100;
    guaranteed-rate 3g;
  }
}
forwarding-class-sets {
  fcoe-pg {
    class fcoe;
  }
}
congestion-notification-profile {
  fcoe-cnp {
    input {
      ieee-802.1 {
        code-point 011 {
          pfc;
        }
      }
    }
  }
}
interfaces {
  ae0 {
    forwarding-class-set {
      fcoe-pg {
        output-traffic-control-profile fcoe-tcp;
      }
    }
    congestion-notification-profile fcoe-cnp;
```

```
    }
    ae1 {
        forwarding-class-set {
            fcoe-pg {
                output-traffic-control-profile fcoe-tcp;
            }
        }
        congestion-notification-profile fcoe-cnp;
    }
}
scheduler-maps {
    fcoe-map {
        forwarding-class fcoe scheduler fcoe-sched;
    }
}
schedulers {
    fcoe-sched {
        transmit-rate 3g;
        shaping-rate percent 100;
        priority low;
    }
}
```

NOTE: The forwarding class and classifier configurations are not shown because the **show** command does not display default portions of the configuration.

Display the results of the CoS configuration on FCoE Transit Switch TS1 and on FCoE Transit Switch TS2 (the results on both transit switches are the same).

```
user@switch> show configuration class-of-service
traffic-control-profiles {
    fcoe-tcp {
        scheduler-map fcoe-map;
        shaping-rate percent 100;
        guaranteed-rate 3g;
    }
}
forwarding-class-sets {
    fcoe-pg {
        class fcoe;
    }
```

```
}
congestion-notification-profile {
    fcoe-cnp {
        input {
            ieee-802.1 {
                code-point 011 {
                    pfc;
                }
            }
        }
    }
}
interfaces {
    xe-0/0/30 {
        forwarding-class-set {
            fcoe-pg {
                output-traffic-control-profile fcoe-tcp;
            }
        }
        congestion-notification-profile fcoe-cnp;
    }
    xe-0/0/31 {
        forwarding-class-set {
            fcoe-pg {
                output-traffic-control-profile fcoe-tcp;
            }
        }
        congestion-notification-profile fcoe-cnp;
    }
    xe-0/0/32 {
        forwarding-class-set {
            fcoe-pg {
                output-traffic-control-profile fcoe-tcp;
            }
        }
        congestion-notification-profile fcoe-cnp;
    }
    xe-0/0/33 {
        forwarding-class-set {
            fcoe-pg {
                output-traffic-control-profile fcoe-tcp;
            }
        }
        congestion-notification-profile fcoe-cnp;
```

```
    }
    ae1 {
        forwarding-class-set {
            fcoe-pg {
                output-traffic-control-profile fcoe-tcp;
            }
        }
        congestion-notification-profile fcoe-cnp;
    }
}
scheduler-maps {
    fcoe-map {
        forwarding-class fcoe scheduler fcoe-sched;
    }
}
schedulers {
    fcoe-sched {
        transmit-rate 3g;
        shaping-rate percent 100;
        priority low;
    }
}
```

Verification

IN THIS SECTION

To verify that the CoS components and FIP snooping have been configured and are operating properly, perform these tasks. Because this example uses the default **fcoe** forwarding class and the default IEEE 802.1p trusted classifier, the verification of those configurations is not shown.

Verifying That the Output Queue Schedulers Have Been Created

Purpose

Verify that the output queue scheduler for FCoE traffic has the correct bandwidth parameters and priorities, and is mapped to the correct forwarding class (output queue). Queue scheduler verification is the same on each of the four switches.

Action

List the scheduler map using the operational mode command **show class-of-service scheduler-map fcoe-map**:

`user@switch>` **show class-of-service scheduler-map fcoe-map**

```
Scheduler map: fcoe-map, Index: 9023

  Scheduler: fcoe-sched, Forwarding class: fcoe, Index: 37289
    Transmit rate: 3000000000 bps, Rate Limit: none, Buffer size: remainder,
    Buffer Limit: none, Priority: low
    Excess Priority: unspecified
    Shaping rate: 100 percent,
    drop-profile-map-set-type: mark
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any             1    <default-drop-profile>
      Medium high     any             1    <default-drop-profile>
      High            any             1    <default-drop-profile>
```

Meaning

The **show class-of-service scheduler-map fcoe-map** command lists the properties of the scheduler map **fcoe-map**. The command output includes:

- The name of the scheduler map (**fcoe-map**)
- The name of the scheduler (**fcoe-sched**)
- The forwarding classes mapped to the scheduler (**fcoe**)
- The minimum guaranteed queue bandwidth (transmit rate **3000000000 bps**)
- The scheduling priority (**low**)

- The maximum bandwidth in the priority group the queue can consume (shaping rate **100 percent**)
- The drop profile loss priority for each drop profile name. This example does not include drop profiles because you do not apply drop profiles to FCoE traffic.

Verifying That the Priority Group Output Scheduler (Traffic Control Profile) Has Been Created

Purpose

Verify that the traffic control profile **fcoe-tcp** has been created with the correct bandwidth parameters and scheduler mapping. Priority group scheduler verification is the same on each of the four switches.

Action

List the FCoE traffic control profile properties using the operational mode command **show class-of-service traffic-control-profile fcoe-tcp**:

`user@switch>` **show class-of-service traffic-control-profile fcoe-tcp**

```
Traffic control profile: fcoe-tcp, Index: 18303
  Shaping rate: 100 percent
  Scheduler map: fcoe-map
  Guaranteed rate: 3000000000
```

Meaning

The **show class-of-service traffic-control-profile fcoe-tcp** command lists all of the configured traffic control profiles. For each traffic control profile, the command output includes:

- The name of the traffic control profile (**fcoe-tcp**)
- The maximum port bandwidth the priority group can consume (shaping rate **100 percent**)
- The scheduler map associated with the traffic control profile (**fcoe-map**)
- The minimum guaranteed priority group port bandwidth (guaranteed rate **3000000000** in bps)

Verifying That the Forwarding Class Set (Priority Group) Has Been Created

Purpose

Verify that the FCoE priority group has been created and that the **fcoe** priority (forwarding class) belongs to the FCoE priority group. Forwarding class set verification is the same on each of the four switches.

Action

List the forwarding class sets using the operational mode command **show class-of-service forwarding-class-set fcoe-pg**:

`user@switch>` **show class-of-service forwarding-class-set fcoe-pg**

```
Forwarding class set: fcoe-pg, Type: normal-type, Forwarding class set index: 31420

  Forwarding class                       Index
  fcoe                                     1
```

Meaning

The **show class-of-service forwarding-class-set fcoe-pg** command lists all of the forwarding classes (priorities) that belong to the **fcoe-pg** priority group, and the internal index number of the priority group. The command output shows that the forwarding class set **fcoe-pg** includes the forwarding class **fcoe**.

Verifying That Priority-Based Flow Control Has Been Enabled

Purpose

Verify that PFC is enabled on the FCoE code point. PFC verification is the same on each of the four switches.

Action

List the FCoE congestion notification profile using the operational mode command **show class-of-service congestion-notification fcoe-cnp**:

`user@switch>` **show class-of-service congestion-notification fcoe-cnp**

```
Type: Input, Name: fcoe-cnp, Index: 6879
Cable Length: 100 m
  Priority     PFC          MRU
  000          Disabled
  001          Disabled
  010          Disabled
  011          Enabled      2500
  100          Disabled
  101          Disabled
  110          Disabled
  111          Disabled
Type: Output
  Priority     Flow-Control-Queues
  000
               0
  001
               1
  010
               2
  011
               3
```

```
100
             4
101
             5
110
             6
111
             7
```

Meaning

The **show class-of-service congestion-notification fcoe-cnp** command lists all of the IEEE 802.1p code points in the congestion notification profile that have PFC enabled. The command output shows that PFC is enabled on code point **011** (**fcoe** queue) for the **fcoe-cnp** congestion notification profile.

The command also shows the default cable length (**100** meters), the default maximum receive unit (**2500** bytes), and the default mapping of priorities to output queues because this example does not include configuring these options.

Verifying That the Interface Class of Service Configuration Has Been Created

Purpose

Verify that the CoS properties of the interfaces are correct. The verification output on MC-LAG Switches S1 and S2 differs from the output on FCoE Transit Switches TS1 and TS2.

Action

List the interface CoS configuration on MC-LAG Switches S1 and S2 using the operational mode command **show configuration class-of-service interfaces**:

`user@switch>` **show configuration class-of-service interfaces**

```
ae0 {
    forwarding-class-set {
        fcoe-pg {
            output-traffic-control-profile fcoe-tcp;
        }
    }
    congestion-notification-profile fcoe-cnp;
}

ae1 {
    forwarding-class-set {
        fcoe-pg {
            output-traffic-control-profile fcoe-tcp;
```

```
        }
    }
    congestion-notification-profile fcoe-cnp;
}
```

List the interface CoS configuration on FCoE Transit Switches TS1 and TS2 using the operational mode command **show configuration class-of-service interfaces**:

`user@switch>` **show configuration class-of-service interfaces**

```
xe-0/0/30 {
    forwarding-class-set {
        fcoe-pg {
            output-traffic-control-profile fcoe-tcp;
        }
    }
    congestion-notification-profile fcoe-cnp;
}
xe-0/0/31 {
    forwarding-class-set {
        fcoe-pg {
            output-traffic-control-profile fcoe-tcp;
        }
    }
    congestion-notification-profile fcoe-cnp;
}
xe-0/0/32 {
    forwarding-class-set {
        fcoe-pg {
            output-traffic-control-profile fcoe-tcp;
        }
    }
    congestion-notification-profile fcoe-cnp;
}
xe-0/0/33 {
    forwarding-class-set {
        fcoe-pg {
            output-traffic-control-profile fcoe-tcp;
        }
    }
    congestion-notification-profile fcoe-cnp;
}
```

```
ae1 {
    forwarding-class-set {
        fcoe-pg {
            output-traffic-control-profile fcoe-tcp;
        }
    }
    congestion-notification-profile fcoe-cnp;
}
```

Meaning

The **show configuration class-of-service interfaces** command lists the class of service configuration for all interfaces. For each interface, the command output includes:

- The name of the interface (for example, **ae0** or **xe-0/0/30**)
- The name of the forwarding class set associated with the interface (**fcoe-pg**)
- The name of the traffic control profile associated with the interface (output traffic control profile, **fcoe-tcp**)
- The name of the congestion notification profile associated with the interface (**fcoe-cnp**)

> **NOTE:** Interfaces that are members of a LAG are not shown individually. The LAG or MC-LAG CoS configuration is applied to all interfaces that are members of the LAG or MC-LAG. For example, the interface CoS configuration output on MC-LAG Switches S1 and S2 shows the LAG CoS configuration but does not show the CoS configuration of the member interfaces separately. The interface CoS configuration output on FCoE Transit Switches TS1 and TS2 shows the LAG CoS configuration but also shows the configuration for interfaces xe-0/0/30, xe-0/0/31, xe-0/0/32, and xe-0/0/33, which are not members of a LAG.

Verifying That the Interfaces Are Correctly Configured

Purpose

Verify that the LAG membership, MTU, VLAN membership, and port mode of the interfaces are correct. The verification output on MC-LAG Switches S1 and S2 differs from the output on FCoE Transit Switches TS1 and TS2.

Action

List the interface configuration on MC-LAG Switches S1 and S2 using the operational mode command **show configuration interfaces**:

`user@switch>` **show configuration interfaces**

```
xe-0/0/10 {
    ether-options {
        802.3ad ae0;
    }
}
xe-0/0/11 {
    ether-options {
        802.3ad ae0;
    }
}
xe-0/0/20 {
    ether-options {
        802.3ad ae1;
    }
}
xe-0/0/21 {
    ether-options {
        802.3ad ae1;
    }
}
ae0 {
    mtu 2180;
    unit 0 {
        family ethernet-switching {
            port-mode trunk;
            vlan {
                members fcoe_vlan;
            }
        }
    }
}
ae1 {
    mtu 2180;
    unit 0 {
        family ethernet-switching {
            port-mode trunk;
            vlan {
                members fcoe_vlan;
            }
        }
    }
}
```

List the interface configuration on FCoE Transit Switches TS1 and TS2 using the operational mode command **show configuration interfaces**:

user@switch> **show configuration interfaces**

```
xe-0/0/25 {
    ether-options {
        802.3ad ae1;
    }
}
xe-0/0/26 {
    ether-options {
        802.3ad ae1;
    }
}
xe-0/0/30 {
    mtu 2180;
    unit 0 {
        family ethernet-switching {
            port-mode tagged-access;
            vlan {
                members fcoe_vlan;
            }
        }
    }
}
xe-0/0/31 {
    mtu 2180;
    unit 0 {
        family ethernet-switching {
            port-mode tagged-access;
            vlan {
                members fcoe_vlan;
            }
        }
    }
}
xe-0/0/32 {
    mtu 2180;
    unit 0 {
        family ethernet-switching {
            port-mode tagged-access;
            vlan {
                members fcoe_vlan;
            }
```

```
        }
    }
}
xe-0/0/33 {
    mtu 2180;
    unit 0 {
        family ethernet-switching {
            port-mode tagged-access;
            vlan {
                members fcoe_vlan;
            }
        }
    }
}

ae1 {
    mtu 2180;
    unit 0 {
        family ethernet-switching {
            port-mode trunk;
            vlan {
                members fcoe_vlan;
            }
        }
    }
}
```

Meaning

The **show configuration interfaces** command lists the configuration of each interface by interface name.

For each interface that is a member of a LAG, the command lists only the name of the LAG to which the interface belongs.

For each LAG interface and for each interface that is not a member of a LAG, the command output includes:

- The MTU (**2180**)
- The unit number of the interface (**0**)
- The port mode (**trunk** mode for interfaces that connect two switches, **tagged-access** mode for interfaces that connect to FCoE hosts)
- The name of the VLAN in which the interface is a member (**fcoe_vlan**)

Verifying That FIP Snooping Is Enabled on the FCoE VLAN on FCoE Transit Switches TS1 and TS2 Access Interfaces

Purpose

Verify that FIP snooping is enabled on the FCoE VLAN access interfaces. FIP snooping is enabled only on the FCoE access interfaces, so it is enabled only on FCoE Transit Switches TS1 and TS2. FIP snooping is not enabled on MC-LAG Switches S1 and S2 because FIP snooping is done at the Transit Switch TS1 and TS2 FCoE access ports.

Action

List the port security configuration on FCoE Transit Switches TS1 and TS2 using the operational mode command **show configuration ethernet-switching-options secure-access-port**:

`user@switch>` **show configuration ethernet-switching-options secure-access-port**

```
interface ae1.0 {
    fcoe-trusted;
}
vlan fcoe_vlan {
    examine-fip {
        examine-vn2vn {
            beacon-period 90000;
        }
    }
}
```

Meaning

The **show configuration ethernet-switching-options secure-access-port** command lists port security information, including whether a port is trusted. The command output shows that:

- LAG port **ae1.0**, which connects the FCoE transit switch to the MC-LAG switches, is configured as an FCoE trusted interface. FIP snooping is not performed on the member interfaces of the LAG (xe-0/0/25 and xe-0/0/26).
- FIP snooping is enabled (**examine-fip**) on the FCoE VLAN (**fcoe_vlan**), the type of FIP snooping is VN2VN_Port FIP snooping (**examine-vn2vn**), and the beacon period is set to **90000** milliseconds. On Transit Switches TS1 and TS2, all interface members of the FCoE VLAN perform FIP snooping unless the interface is configured as FCoE trusted. On Transit Switches TS1 and TS2, interfaces xe-0/0/30, xe-0/0/31, xe-0/0/32, and xe-0/0/33 perform FIP snooping because they are not configured as FCoE trusted. The interface members of LAG ae1 (xe-0/0/25 and xe-0/0/26) do not perform FIP snooping because the LAG is configured as FCoE trusted.

Verifying That the FIP Snooping Mode Is Correct on FCoE Transit Switches TS1 and TS2

Purpose

Verify that the FIP snooping mode is correct on the FCoE VLAN. FIP snooping is enabled only on the FCoE access interfaces, so it is enabled only on FCoE Transit Switches TS1 and TS2. FIP snooping is not enabled on MC-LAG Switches S1 and S2 because FIP snooping is done at the Transit Switch TS1 and TS2 FCoE access ports.

Action

List the FIP snooping configuration on FCoE Transit Switches TS1 and TS2 using the operational mode command **show fip snooping brief**:

user@switch> **show fip snooping brief**

```
VLAN: fcoe_vlan,    Mode: VN2VN Snooping
  FC-MAP: 0e:fd:00
…
```

NOTE: The output has been truncated to show only the relevant information.

Meaning

The **show fip snooping brief** command lists FIP snooping information, including the FIP snooping VLAN and the FIP snooping mode. The command output shows that:

- The VLAN on which FIP snooping is enabled is **fcoe_vlan**
- The FIP snooping mode is VN2VN_Port FIP snooping (**VN2VN Snooping**)

Verifying That IGMP Snooping Is Disabled on the FCoE VLAN

Purpose

Verify that IGMP snooping is disabled on the FCoE VLAN on all four switches.

Action

List the IGMP snooping protocol information on each of the four switches using the **show configuration protocols igmp-snooping** command:

user@switch> **show configuration protocols igmp-snooping**

```
vlan fcoe_vlan {
    disable;
}
```

Meaning

The **show configuration protocols igmp-snooping** command lists the IGMP snooping configuration for the VLANs configured on the switch. The command output shows that IGMP snooping is disabled on the FCoE VLAN (**fcoe_vlan**).

RELATED DOCUMENTATION

Example: Configuring CoS PFC for FCoE Traffic | **490**

Example: Configuring CoS Using ELS for FCoE Transit Switch Traffic Across an MC-LAG

IN THIS SECTION

Multichassis link aggregation groups (MC-LAGs) provide redundancy and load balancing between two QFX Series switches, multihoming support for client devices such as servers, and a loop-free Layer 2 network without running Spanning Tree Protocol (STP).

> **NOTE:** This example uses the Junos OS Enhanced Layer 2 Software (ELS) configuration style for QFX Series switches. If your switch runs software that does not support ELS, see *Example: Configuring CoS for FCoE Transit Switch Traffic Across an MC-LAG*. For ELS details, see *Using the Enhanced Layer 2 Software CLI*.

You can use an MC-LAG to provide a redundant aggregation layer for Fibre Channel over Ethernet (FCoE) traffic in an *inverted-U* topology. To support lossless transport of FCoE traffic across an MC-LAG, you must configure the appropriate class of service (CoS) on both of the QFX Series switches with MC-LAG port members. The CoS configuration must be the same on both of the MC-LAG switches because an MC-LAG does not carry forwarding class and IEEE 802.1p priority information.

Ports that are members of an MC-LAG act as FCoE passthrough transit switch ports.

NOTE: This example describes how to configure CoS to provide lossless transport for FCoE traffic across an MC-LAG that connects two QFX Series switches. It also describes how to configure CoS on the FCoE transit switches that connect FCoE hosts to the QFX Series switches that form the MC-LAG.

This example does not describe how to configure the MC-LAG itself; it includes a subset of MC-LAG configuration that only shows how to configure interface membership in the MC-LAG.

NOTE: Juniper Networks QFX10000 aggregation switches do not support FIP snooping, so they cannot be used as FIP snooping access switches (Transit Switches TS1 and TS2) in this example. However, QFX10000 switches can play the role of the MC-LAG switches (MC-LAG Switch S1 and MC-LAG Switch S2) in this example.

QFX3500 and QFX3600 Virtual Chassis switches do not support FCoE.

This topic describes:

Requirements

This example uses the following hardware and software components:

- Two Juniper Networks QFX5100 Switches running the ELS CLI that form an MC-LAG for FCoE traffic.
- Two Juniper Networks QFX5100 Switches running the ELS CLI that provide FCoE server access in transit switch mode and that connect to the MC-LAG switches.
- FCoE servers (or other FCoE hosts) connected to the transit switches.
- Junos OS Release 13.2 or later for the QFX Series.

Overview

FCoE traffic requires lossless transport. This example shows you how to:

- Configure CoS for FCoE traffic on the two QFX5100 switches that form the MC-LAG, including priority-based flow control (PFC). The example also includes configuration for both enhanced transmission selection (ETS) hierarchical scheduling of resources for the FCoE forwarding class priority and for the forwarding class set priority group, and also direct port scheduling. You can only use one of the scheduling methods on a port. Different switches support different scheduling methods.

NOTE: Configuring or changing PFC on an interface blocks the entire port until the PFC change is completed. After a PFC change is completed, the port is unblocked and traffic resumes. Blocking the port stops ingress and egress traffic, and causes packet loss on all queues on the port until the port is unblocked.

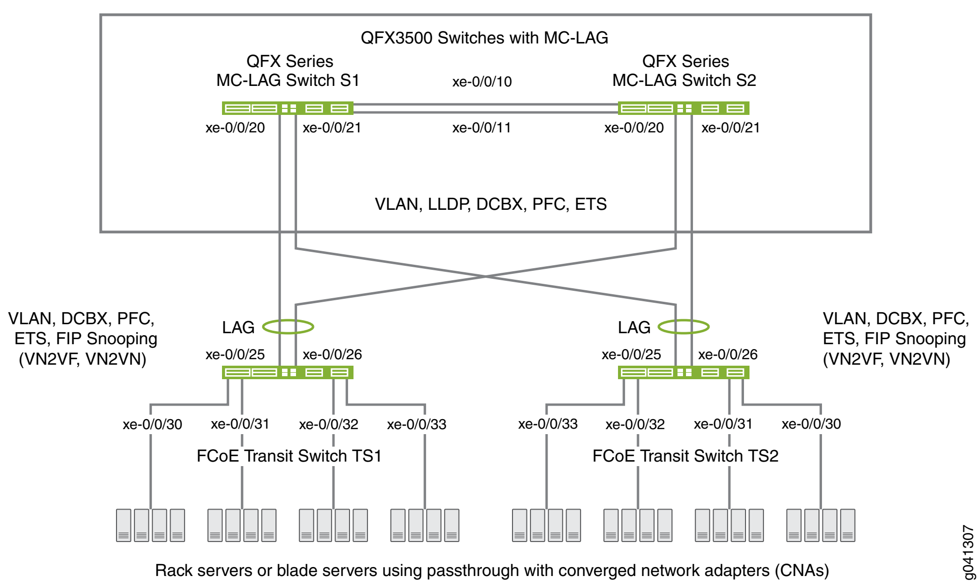
- Configure CoS for FCoE on the two FCoE transit switches that connect FCoE hosts to the MC-LAG switches and enable FIP snooping on the FCoE VLAN at the FCoE transit switch access ports.
- Configure the appropriate port mode, MTU, and FCoE trusted or untrusted state for each interface to support lossless FCoE transport.

NOTE: Do not enable IGMP snooping on the FCoE VLAN. (IGMP snooping is enabled on the default VLAN by default, but is disabled by default on all other VLANs.)

Topology

QFX5100 switches that act as transit switches support MC-LAGs for FCoE traffic in an inverted-U network topology, as shown in Figure 24 on page 505.

Figure 25: Supported Topology for an MC-LAG on an FCoE Transit Switch



NOTE: Juniper Networks QFX10000 aggregation switches do not support FIP snooping, so they cannot be used as FIP snooping access switches (Transit Switches TS1 and TS2) in this example. However, QFX10000 switches can play the role of the MC-LAG switches (MC-LAG Switch S1 and MC-LAG Switch S2) in this example.

Table 92 on page 505 shows the configuration components for this example.

Table 93: Components of the CoS for FCoE Traffic Across an MC-LAG Configuration Topology

Component	Settings
Hardware	Four QFX5100 switches running the ELS CLI (two to form the MC-LAG as passthrough transit switches and two transit switches for FCoE access).
Forwarding class (all switches)	Default **fcoe** forwarding class.
Classifier (forwarding class mapping of incoming traffic to IEEE priority)	Default IEEE 802.1p trusted classifier on all FCoE interfaces.

Table 93: Components of the CoS for FCoE Traffic Across an MC-LAG Configuration Topology *(continued)*

Component	Settings
LAGs and MC-LAG	S1—Ports xe-0/0/10 and x-0/0/11 are members of LAG **ae0**, which connects Switch S1 to Switch S2. Ports xe-0/0/20 and xe-0/0/21 are members of MC-LAG **ae1**. All ports are configured in **trunk** interface mode, as **fcoe-trusted**, and with an MTU of **2180**. S2—Ports xe-0/0/10 and x-0/0/11 are members of LAG **ae0**, which connects Switch S2 to Switch S1. Ports xe-0/0/20 and xe-0/0/21 are members of MC-LAG **ae1**. All ports are configured in **trunk** interface mode, as **fcoe-trusted**, and with an MTU of **2180**. **NOTE:** Ports xe-0/0/20 and xe-0/0/21 on Switches S1 and S2 are the members of the MC-LAG. TS1—Ports xe-0/0/25 and x-0/0/26 are members of LAG **ae1**, configured in **trunk** interface mode, as **fcoe-trusted**, and with an MTU of **2180**. Ports xe-0/0/30, xe-0/0/31, xe-0/0/32, and xe-0/0/33 are configured in **trunk** interface mode, with an MTU of **2180**. TS2—Ports xe-0/0/25 and x-0/0/26 are members of LAG **ae1**, configured in **trunk** interface mode, as **fcoe-trusted**, and with an MTU of **2180**. Ports xe-0/0/30, xe-0/0/31, xe-0/0/32, and xe-0/0/33 are configured in **trunk** interface mode, with an MTU of **2180**.
FCoE queue scheduler (all switches)	**fcoe-sched**: Minimum bandwidth **3g** Maximum bandwidth **100%** Priority **low**
Forwarding class-to-scheduler mapping (all switches)	Scheduler map **fcoe-map**: Forwarding class **fcoe** Scheduler **fcoe-sched**

Table 93: Components of the CoS for FCoE Traffic Across an MC-LAG Configuration Topology *(continued)*

Component	Settings
PFC congestion notification profile (all switches)	**fcoe-cnp**: Code point **011** Ingress interfaces: • S1—LAG **ae0** and MC-LAG **ae1** • S2—LAG **ae0** and MC-LAG **ae1** • TS1—LAG **ae1**, interfaces **xe-0/0/30**, **xe-0/0/31**, **xe-0/0/32**, and **xe-0/0/33** • TS2—LAG **ae1**, interfaces **xe-0/0/30**, **xe-0/0/31**, **xe-0/0/32**, and **xe-0/0/33**
FCoE VLAN name and tag ID	Name—**fcoe_vlan** ID—**100** Include the FCoE VLAN on the interfaces that carry FCoE traffic on all four switches.
ETS only—forwarding class set (FCoE priority group, all switches)	**fcoe-pg**: Forwarding class **fcoe** Egress interfaces: • S1—LAG **ae0** and MC-LAG **ae1** • S2—LAG **ae0** and MC-LAG **ae1** • TS1—LAG **ae1**, interfaces **xe-0/0/30**, **xe-0/0/31**, **xe-0/0/32**, and **xe-0/0/33** • TS2—LAG **ae1**, interfaces **xe-0/0/30**, **xe-0/0/31**, **xe-0/0/32**, and **xe-0/0/33**
ETS only—traffic control profile (all switches)	**fcoe-tcp**: Scheduler map **fcoe-map** Minimum bandwidth **3g** Maximum bandwidth **100%** The traffic control profile is applied to the same interfaces as the forwarding class set, using the same CLI statement. This applies ETS hierarchical scheduling to the interfaces.

Table 93: Components of the CoS for FCoE Traffic Across an MC-LAG Configuration Topology *(continued)*

Component	Settings
Port scheduling only—apply scheduling to interfaces	On switches that support direct port scheduling, if you use port scheduling, apply scheduling by attaching the scheduler map directly to interfaces: • S1—LAG **ae0** and MC-LAG **ae1** • S2—LAG **ae0** and MC-LAG **ae1** • TS1—LAG **ae1**, interfaces **xe-0/0/30**, **xe-0/0/31**, **xe-0/0/32**, and **xe-0/0/33** • TS2—LAG **ae1**, interfaces **xe-0/0/30**, **xe-0/0/31**, **xe-0/0/32**, and **xe-0/0/33**
FIP snooping	Enable FIP snooping on Transit Switches TS1 and TS2 on the FCoE VLAN. Configure the LAG interfaces that connect to the MC-LAG switches as FCoE trusted interfaces so that they do not perform FIP snooping. This example enables VN2VN_Port FIP snooping on the FCoE transit switch interfaces connected to the FCoE servers. The example is equally valid with VN2VF_Port FIP snooping enabled on the transit switch access ports. The method of FIP snooping you enable depends on your network configuration. **NOTE:** Juniper Networks QFX10000 aggregation switches do not support FIP snooping, so they cannot be used as FIP snooping access switches (Transit Switches TS1 and TS2) in this example.

NOTE: This example uses the default IEEE 802.1p trusted BA classifier, which is automatically applied to trunk mode interfaces if you do not apply an explicitly configured classifier.

To configure CoS for FCoE traffic across an MC-LAG:

- Use the default FCoE forwarding class and forwarding-class-to-queue mapping (do not explicitly configure the FCoE forwarding class or output queue). The default FCoE forwarding class is **fcoe**, and the default output queue is queue **3**.
- Use the default trusted BA classifier, which maps incoming packets to forwarding classes by the IEEE 802.1p code point (CoS priority) of the packet. The trusted classifier is the default classifier for interfaces in trunk interface mode. The default trusted classifier maps incoming packets with the IEEE 802.1p code

point 3 (**011**) to the FCoE forwarding class. If you choose to configure the BA classifier instead of using the default classifier, you must ensure that FCoE traffic is classified into forwarding classes in exactly the same way on both MC-LAG switches. Using the default classifier ensures consistent classifier configuration on the MC-LAG ports.

- Configure a congestion notification profile that enables PFC on the FCoE code point (code point **011** in this example). The congestion notification profile configuration must be the same on both MC-LAG switches.
- Apply the congestion notification profile to the interfaces.
- Configure the interface mode, MTU, and FCoE trusted or untrusted state for each interface to support lossless FCoE transport.
- For ETS hierarchical port scheduling, configure ETS on the interfaces to provide the bandwidth required for lossless FCoE transport. Configuring ETS includes configuring bandwidth scheduling for the FCoE forwarding class, a forwarding class set (priority group) that includes the FCoE forwarding class, and a traffic control profile to assign bandwidth to the forwarding class set that includes FCoE traffic, and applying the traffic control profile and forwarding class set to interfaces..

  On switches that support direct port scheduling, configure CoS properties on interfaces by applying scheduler maps directly to interfaces.

In addition, this example describes how to enable FIP snooping on the Transit Switch TS1 and TS2 ports that are connected to the FCoE servers. To provide secure access, FIP snooping must be enabled on the FCoE access ports.

This example focuses on the CoS configuration to support lossless FCoE transport across an MC-LAG. This example does not describe how to configure the properties of MC-LAGs and LAGs, although it does show you how to configure the port characteristics required to support lossless transport and how to assign interfaces to the MC-LAG and to the LAGs.

Before you configure CoS, configure:

- The MC-LAGs that connect Switches S1 and S2 to Switches TS1 and TS2.
- The LAGs that connect the Transit Switches TS1 and TS2 to MC-LAG Switches S1 and S2. (*Configuring Link Aggregation* describes how to configure LAGs.)
- The LAG that connects Switch S1 to Switch S2.

Configuration

IN THIS SECTION

To configure CoS for lossless FCoE transport across an MC-LAG, perform these tasks:

CLI Quick Configuration

To quickly configure CoS for lossless FCoE transport across an MC-LAG, 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 for the MC-LAG and FCoE transit switches at the **[edit]** hierarchy level.

The quick configuration shows the commands for the two MC-LAG switches and the two FCoE transit switches separately. The configurations on both of the MC-LAG switches are same and on both of the FCoE transit switches are the same because the CoS configuration must be identical, and because this example uses the same ports on each of these sets of switches.

NOTE: The CLI configurations for the MC-LAG switches and for the FCoE transit switches are each separated into three sections:

- Configuration common to all port scheduling methods
- Configuration specific to ETS hierarchical port scheduling
- Configuration specific to direct port scheduling

Quick configuration for MC-LAG Switch S1 and Switch S2:

MC-LAG Switches Configuration Common to ETS Hierarchical Port Scheduling and to Direct Port Scheduling

```
set class-of-service schedulers fcoe-sched priority low transmit-rate 3g
set class-of-service schedulers fcoe-sched shaping-rate percent 100
set class-of-service scheduler-maps fcoe-map forwarding-class fcoe scheduler fcoe-sched
set class-of-service congestion-notification-profile fcoe-cnp input ieee-802.1 code-point 011 pfc
set class-of-service interfaces ae0 congestion-notification-profile fcoe-cnp
set class-of-service interfaces ae1 congestion-notification-profile fcoe-cnp
```

set vlans fcoe_vlan vlan-id 100

set interfaces xe-0/0/10 ether-options 802.3ad ae0

set interfaces xe-0/0/11 ether-options 802.3ad ae0

set interfaces xe-0/0/20 ether-options 802.3ad ae1

set interfaces xe-0/0/21 ether-options 802.3ad ae1

set interfaces ae0 unit 0 family ethernet-switching interface-mode trunk vlan members fcoe_vlan

set interfaces ae1 unit 0 family ethernet-switching interface-mode trunk vlan members fcoe_vlan

set interfaces ae0 mtu 2180

set interfaces ae1 mtu 2180

set vlans fcoe_vlan forwarding-options fip-security interface ae0 fcoe-trusted

set vlans fcoe_vlan forwarding-options fip-security interface ae1 fcoe-trusted

MC-LAG Switches Configuration for ETS Hierarchical Port Scheduling

set class-of-service forwarding-class-sets fcoe-pg class fcoe

set class-of-service traffic-control-profiles fcoe-tcp scheduler-map fcoe-map guaranteed-rate 3g

set class-of-service traffic-control-profiles fcoe-tcp shaping-rate percent 100

set class-of-service interfaces ae0 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp

set class-of-service interfaces ae1 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp

MC-LAG Switches Configuration for Direct Port Scheduling

set class-of-service interfaces ae0 scheduler-map fcoe-map

set class-of-service interfaces ae1 scheduler-map fcoe-map

Quick configuration for FCoE Transit Switch TS1 and Switch TS2:

FCoE Transit Switches Configuration Common to ETS Hierarchical Port Scheduling and to Direct Port Scheduling

set class-of-service schedulers fcoe-sched priority low transmit-rate 3g

set class-of-service schedulers fcoe-sched shaping-rate percent 100

set class-of-service scheduler-maps fcoe-map forwarding-class fcoe scheduler fcoe-sched

set class-of-service congestion-notification-profile fcoe-cnp input ieee-802.1 code-point 011 pfc

set class-of-service interfaces ae1 congestion-notification-profile fcoe-cnp

set class-of-service interfaces xe-0/0/30 congestion-notification-profile fcoe-cnp

set class-of-service interfaces xe-0/0/31 congestion-notification-profile fcoe-cnp

set class-of-service interfaces xe-0/0/32 congestion-notification-profile fcoe-cnp

set class-of-service interfaces xe-0/0/33 congestion-notification-profile fcoe-cnp

set vlans fcoe_vlan vlan-id 100

set interfaces xe-0/0/25 ether-options 802.3ad ae1

```
set interfaces xe-0/0/26 ether-options 802.3ad ae1
set interfaces ae1 unit 0 family ethernet-switching interface-mode trunk vlan members fcoe_vlan
set interfaces xe-0/0/30 unit 0 family ethernet-switching interface-mode trunk vlan members fcoe_vlan
set interfaces xe-0/0/31 unit 0 family ethernet-switching interface-mode trunk vlan members fcoe_vlan
set interfaces xe-0/0/32 unit 0 family ethernet-switching interface-mode trunk vlan members fcoe_vlan
set interfaces xe-0/0/33 unit 0 family ethernet-switching interface-mode trunk vlan members fcoe_vlan
set interfaces ae1 mtu 2180
set interfaces xe-0/0/30 mtu 2180
set interfaces xe-0/0/31 mtu 2180
set interfaces xe-0/0/32 mtu 2180
set interfaces xe-0/0/33 mtu 2180
set vlans fcoe_vlan forwarding-options fip-security interface ae1 fcoe-trusted
set vlans fcoe_vlan forwarding-options fip-security examine-vn2v2 beacon-period 90000
```

FCoE Transit Switches Configuration for ETS Hierarchical Port Scheduling

```
set class-of-service forwarding-class-sets fcoe-pg class fcoe
set class-of-service traffic-control-profiles fcoe-tcp scheduler-map fcoe-map guaranteed-rate 3g
set class-of-service traffic-control-profiles fcoe-tcp shaping-rate percent 100
set class-of-service interfaces ae1 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp
set class-of-service interfaces xe-0/0/30 forwarding-class-set fcoe-pg output-traffic-control-profile
fcoe-tcp
set class-of-service interfaces xe-0/0/31 forwarding-class-set fcoe-pg output-traffic-control-profile
fcoe-tcp
set class-of-service interfaces xe-0/0/32 forwarding-class-set fcoe-pg output-traffic-control-profile
fcoe-tcp
set class-of-service interfaces xe-0/0/33 forwarding-class-set fcoe-pg output-traffic-control-profile
fcoe-tcp
```

FCoE Transit Switches Configuration for Direct Port Scheduling

```
set class-of-service interfaces ae1 scheduler-map fcoe-map
set class-of-service interfaces xe-0/0/30 scheduler-map fcoe-map
set class-of-service interfaces xe-0/0/31 scheduler-map fcoe-map
set class-of-service interfaces xe-0/0/32 scheduler-map fcoe-map
set class-of-service interfaces xe-0/0/33 scheduler-map fcoe-map
```

MC-LAG Switches S1 and S2 Common Configuration (Applies to ETS and Port Scheduling)

Step-by-Step Procedure

To configure queue scheduling, PFC, the FCoE VLAN, and LAG and MC-LAG interface membership and characteristics to support lossless FCoE transport across an MC-LAG (this example uses the default **fcoe** forwarding class and the default classifier to map incoming FCoE traffic to the FCoE IEEE 802.1p code point **011**), for both ETS hierarchical port scheduling and port scheduling (common configuration):

1. Configure output scheduling for the FCoE queue:

   ```
   [edit class-of-service]
   user@switch# set schedulers fcoe-sched priority low transmit-rate 3g
   user@switch# set schedulers fcoe-sched shaping-rate percent 100
   ```

2. Map the FCoE forwarding class to the FCoE scheduler (**fcoe-sched**):

   ```
   [edit class-of-service]
   user@switch# set scheduler-maps fcoe-map forwarding-class fcoe scheduler fcoe-sched
   ```

3. Enable PFC on the FCoE priority by creating a congestion notification profile (**fcoe-cnp**) that applies FCoE to the IEEE 802.1 code point **011**:

   ```
   [edit class-of-service]
   user@switch# set congestion-notification-profile fcoe-cnp input ieee-802.1 code-point 011 pfc
   ```

4. Apply the PFC configuration to the LAG and MC-LAG interfaces:

   ```
   [edit class-of-service]
   user@switch# set interfaces ae0 congestion-notification-profile fcoe-cnp
   user@switch# set interfaces ae1 congestion-notification-profile fcoe-cnp
   ```

5. Configure the VLAN for FCoE traffic (**fcoe_vlan**):

   ```
   [edit vlans]
   user@switch# set fcoe_vlan vlan-id 100
   ```

6. Add the member interfaces to the LAG between the two MC-LAG switches:

   ```
   [edit interfaces]
   user@switch# set xe-0/0/10 ether-options 802.3ad ae0
   user@switch# set xe-0/0/11 ether-options 802.3ad ae0
   ```

7. Add the member interfaces to the MC-LAG:

```
[edit interfaces]
user@switch# set xe-0/0/20 ether-options 802.3ad ae1
user@switch# set xe-0/0/21 ether-options 802.3ad ae1
```

8. Configure the interface mode as **trunk** and membership in the FCoE VLAN (**fcoe_vlan**)for the LAG (**ae0**) and for the MC-LAG (**ae1**):

```
[edit interfaces]
user@switch# set interfaces ae0 unit 0 family ethernet-switching interface-mode trunk vlan members fcoe_vlan
user@switch# set interfaces ae1 unit 0 family ethernet-switching interface-mode trunk vlan members fcoe_vlan
```

9. Set the MTU to **2180** for the LAG and MC-LAG interfaces. 2180 bytes is the minimum size required to handle FCoE packets because of the payload and header sizes; you can configure the MTU to a higher number of bytes if desired, but not less than 2180 bytes:

```
[edit interfaces]
user@switch# set ae0 mtu 2180
user@switch# set ae1 mtu 2180
```

10. Set the LAG and MC-LAG interfaces as FCoE trusted ports. Ports that connect to other switches should be trusted and should not perform FIP snooping:

```
[edit]
user@switch# set vlans fcoe_vlan forwarding-options fip-security interface ae0 fcoe-trusted
user@switch# set vlans fcoe_vlan forwarding-options fip-security interface ae1fcoe-trusted
```

MC-LAG Switches S1 and S2 ETS Hierarchical Scheduling Configuration

Step-by-Step Procedure

To configure the forwarding class set (priority group) and priority group scheduling (in a traffic control profile), and apply the ETS hierarchical scheduling for FCoE traffic to interfaces:

1. Configure the forwarding class set (**fcoe-pg**) for the FCoE traffic:

```
[edit class-of-service]
user@switch# set forwarding-class-sets fcoe-pg class fcoe
```

2. Define the traffic control profile (**fcoe-tcp**) to use on the FCoE forwarding class set:

```
[edit class-of-service]
user@switch# set traffic-control-profiles fcoe-tcp scheduler-map fcoe-map guaranteed-rate 3g
user@switch# set traffic-control-profiles fcoe-tcp shaping-rate percent 100
```

3. Apply the FCoE forwarding class set and traffic control profile to the LAG and MC-LAG interfaces:

```
[edit class-of-service]
user@switch# set interfaces ae0 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp
user@switch# set interfaces ae1 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp
```

MC-LAG Switches S1 and S2 Port Scheduling Configuration

Step-by-Step Procedure

To apply port scheduling for FCoE traffic to interfaces:

1. Apply the scheduler map to the egress ports:

set class-of-service interfaces ae0 scheduler-map fcoe-map
set class-of-service interfaces ae1 scheduler-map fcoe-map

FCoE Transit Switches TS1 and TS2 Common Configuration (Applies to ETS and Port Scheduling)

Step-by-Step Procedure

The CoS configuration on FCoE Transit Switches TS1 and TS2 is similar to the CoS configuration on MC-LAG Switches S1 and S2. However, the port configurations differ, and you must enable FIP snooping on the Switch TS1 and Switch TS2 FCoE access ports.

To configure queue scheduling, PFC, the FCoE VLAN, and LAG interface membership and characteristics to support lossless FCoE transport across the MC-LAG (this example uses the default **fcoe** forwarding class and the default classifier to map incoming FCoE traffic to the FCoE IEEE 802.1p code point **011**, so you do not configure them), or both ETS hierarchical scheduling and port scheduling (common configuration):

1. Configure output scheduling for the FCoE queue:

```
[edit class-of-service]
user@switch# set schedulers fcoe-sched priority low transmit-rate 3g
user@switch# set schedulers fcoe-sched shaping-rate percent 100
```

2. Map the FCoE forwarding class to the FCoE scheduler (**fcoe-sched**):

```
[edit class-of-service]
user@switch# set scheduler-maps fcoe-map forwarding-class fcoe scheduler fcoe-sched
```

3. Enable PFC on the FCoE priority by creating a congestion notification profile (**fcoe-cnp**) that applies FCoE to the IEEE 802.1 code point **011**:

   `[edit class-of-service]`
   `user@switch#` **set congestion-notification-profile fcoe-cnp input ieee-802.1 code-point 011 pfc**

4. Apply the PFC configuration to the LAG interface and to the FCoE access interfaces:

   `[edit class-of-service]`
   `user@switch#` **set interfaces ae1 congestion-notification-profile fcoe-cnp**
   `user@switch#` **set class-of-service interfaces xe-0/0/30 congestion-notification-profile fcoe-cnp**
   `user@switch#` **set class-of-service interfaces xe-0/0/31 congestion-notification-profile fcoe-cnp**
   `user@switch#` **set class-of-service interfaces xe-0/0/32 congestion-notification-profile fcoe-cnp**
   `user@switch#` **set class-of-service interfaces xe-0/0/33 congestion-notification-profile fcoe-cnp**

5. Configure the VLAN for FCoE traffic (**fcoe_vlan**):

   `[edit vlans]`
   `user@switch#` **set fcoe_vlan vlan-id 100**

6. Add the member interfaces to the LAG:

   `[edit interfaces]`
   `user@switch#` **set xe-0/0/25 ether-options 802.3ad ae1**
   `user@switch#` **set xe-0/0/26 ether-options 802.3ad ae1**

7. On the LAG (**ae1**), configure the interface mode as **trunk** and membership in the FCoE VLAN (**fcoe_vlan**):

   `[edit interfaces]`
   `user@switch#` **set interfaces ae1 unit 0 family ethernet-switching interface-mode trunk vlan members fcoe_vlan**

8. On the FCoE access interfaces (**xe-0/0/30**, **xe-0/0/31**, **xe-0/0/32**, **xe-0/0/33**), configure the interface mode as **trunk** and membership in the FCoE VLAN (**fcoe_vlan**):

   `[edit interfaces]`
   `user@switch#` **set interfaces xe-0/0/30 unit 0 family ethernet-switching interface-mode trunk vlan members fcoe_vlan**
   `user@switch#` **set interfaces xe-0/0/31 unit 0 family ethernet-switching interface-mode trunk vlan members fcoe_vlan**

```
user@switch# set interfaces xe-0/0/32 unit 0 family ethernet-switching interface-mode trunk vlan members fcoe_vlan
user@switch# set interfaces xe-0/0/33 unit 0 family ethernet-switching interface-mode trunk vlan members fcoe_vlan
```

9. Set the MTU to **2180** for the LAG and FCoE access interfaces. 2180 bytes is the minimum size required to handle FCoE packets because of the payload and header sizes; you can configure the MTU to a higher number of bytes if desired, but not less than 2180 bytes:

```
[edit interfaces]
user@switch# set ae1 mtu 2180
user@switch# set xe-0/0/30 mtu 2180
user@switch# set xe-0/0/31 mtu 2180
user@switch# set xe-0/0/32 mtu 2180
user@switch# set xe-0/0/33 mtu 2180
```

10. Set the LAG interface as an FCoE trusted port. Ports that connect to other switches should be trusted and should not perform FIP snooping:

```
[edit]
user@switch# set vlans fcoe_vlan forwarding-options fip-security interface ae1 fcoe-trusted
```

> **NOTE:** Access ports xe-0/0/30, xe-0/0/31, xe-0/0/32, and xe-0/0/33 are not configured as FCoE trusted ports. The access ports remain in the default state as untrusted ports because they connect directly to FCoE devices and must perform FIP snooping to ensure network security.

11. Enable FIP snooping on the FCoE VLAN to prevent unauthorized FCoE network access (this example uses VN2VN_Port FIP snooping; the example is equally valid if you use VN2VF_Port FIP snooping):

```
[edit]
user@switch# set vlans fcoe_vlan forwarding-options fip-security examine-vn2vn beacon-period 90000
```

> **NOTE:** QFX10000 switches do not support FIP snooping and cannot be used as FCoE access transit switches. (QFX10000 switches can be used as FCoE aggregation switches.)

FCoE Transit Switches TS1 and TS2 ETS Hierarchical Scheduling Configuration

Step-by-Step Procedure

To configure the forwarding class set (priority group) and priority group scheduling (in a traffic control profile), and apply the ETS hierarchical scheduling for FCoE traffic to interfaces:

1. Configure the forwarding class set (**fcoe-pg**) for the FCoE traffic:

   ```
   [edit class-of-service]
   user@switch# set forwarding-class-sets fcoe-pg class fcoe
   ```

2. Define the traffic control profile (**fcoe-tcp**) to use on the FCoE forwarding class set:

   ```
   [edit class-of-service]
   user@switch# set traffic-control-profiles fcoe-tcp scheduler-map fcoe-map guaranteed-rate 3g
   user@switch# set traffic-control-profiles fcoe-tcp shaping-rate percent 100
   ```

3. Apply the FCoE forwarding class set and traffic control profile to the LAG interface and to the FCoE access interfaces:

   ```
   [edit class-of-service]
   user@switch# set interfaces ae1 forwarding-class-set fcoe-pg output-traffic-control-profile fcoe-tcp
   user@switch# set class-of-service interfaces xe-0/0/30 forwarding-class-set fcoe-pg
   output-traffic-control-profile fcoe-tcp
   user@switch# set class-of-service interfaces xe-0/0/31 forwarding-class-set fcoe-pg
   output-traffic-control-profile fcoe-tcp
   user@switch# set class-of-service interfaces xe-0/0/32 forwarding-class-set fcoe-pg
   output-traffic-control-profile fcoe-tcp
   user@switch# set class-of-service interfaces xe-0/0/33 forwarding-class-set fcoe-pg
   output-traffic-control-profile fcoe-tcp
   ```

FCoE Transit Switches TS1 and TS2 Port Scheduling Configuration

Step-by-Step Procedure

To apply port scheduling for FCoE traffic to interfaces:

1. Apply the scheduler map to the egress ports:

   user@switch# **set class-of-service interfaces ae1 scheduler-map fcoe-map**

   user@switch# **set class-of-service interfaces xe-0/0/30 scheduler-map fcoe-map**

   user@switch# **set class-of-service interfaces xe-0/0/31 scheduler-map fcoe-map**

   user@switch# **set class-of-service interfaces xe-0/0/32 scheduler-map fcoe-map**

   user@switch# **set class-of-service interfaces xe-0/0/33 scheduler-map fcoe-map**

Results

Display the results of the CoS configuration on MC-LAG Switch S1 and on MC-LAG Switch S2 (the results on both switches are the same). The results are from the ETS hierarchical scheduling configuration, which shows the more complex configuration. Direct port scheduling results would not show the traffic control profile or forwarding class set portions of the configuration, but would display the name of the scheduler map under each interface (instead of the names of the forwarding class set and output traffic control profile). Other than that, they are the same.

```
user@switch> show configuration class-of-service
traffic-control-profiles {
  fcoe-tcp {
    scheduler-map fcoe-map;
    shaping-rate percent 100;
    guaranteed-rate 3000000000;
  }
}
forwarding-class-sets {
  fcoe-pg {
    class fcoe;
  }
}
congestion-notification-profile {
  fcoe-cnp {
    input {
      ieee-802.1 {
        code-point 011 {
          pfc;
        }
      }
    }
  }
}
interfaces {
```

```
    ae0 {
        forwarding-class-set {
            fcoe-pg {
                output-traffic-control-profile fcoe-tcp;
            }
        }
        congestion-notification-profile fcoe-cnp;
    }
    ae1 {
        forwarding-class-set {
            fcoe-pg {
                output-traffic-control-profile fcoe-tcp;
            }
        }
        congestion-notification-profile fcoe-cnp;
    }
}
scheduler-maps {
    fcoe-map {
        forwarding-class fcoe scheduler fcoe-sched;
    }
}
schedulers {
    fcoe-sched {
        transmit-rate 3000000000;
        shaping-rate percent 100;
        priority low;
    }
}
```

NOTE: The forwarding class and classifier configurations are not shown because the **show** command does not display default portions of the configuration.

Display the results of the CoS configuration on FCoE Transit Switch TS1 and on FCoE Transit Switch TS2 (the results on both transit switches are the same). The results are from the ETS hierarchical port scheduling configuration, which shows the more complex configuration. Direct port scheduling results would not show the traffic control profile or forwarding class set portions of the configuration, but would display the name of the scheduler map under each interface (instead of the names of the forwarding class set and output traffic control profile). Other than that, they are the same.

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

```
traffic-control-profiles {
    fcoe-tcp {
        scheduler-map fcoe-map;
        shaping-rate percent 100;
        guaranteed-rate 3000000000;
    }
}
forwarding-class-sets {
    fcoe-pg {
        class fcoe;
    }
}
congestion-notification-profile {
    fcoe-cnp {
        input {
            ieee-802.1 {
                code-point 011 {
                    pfc;
                }
            }
        }
    }
}
interfaces {
    xe-0/0/30 {
        forwarding-class-set {
            fcoe-pg {
                output-traffic-control-profile fcoe-tcp;
            }
        }
        congestion-notification-profile fcoe-cnp;
    }
    xe-0/0/31 {
        forwarding-class-set {
            fcoe-pg {
                output-traffic-control-profile fcoe-tcp;
            }
        }
        congestion-notification-profile fcoe-cnp;
    }
    xe-0/0/32 {
        forwarding-class-set {
            fcoe-pg {
                output-traffic-control-profile fcoe-tcp;
```

```
            }
        }
        congestion-notification-profile fcoe-cnp;
    }
    xe-0/0/33 {
        forwarding-class-set {
            fcoe-pg {
                output-traffic-control-profile fcoe-tcp;
            }
        }
        congestion-notification-profile fcoe-cnp;
    }
    ae1 {
        forwarding-class-set {
            fcoe-pg {
                output-traffic-control-profile fcoe-tcp;
            }
        }
        congestion-notification-profile fcoe-cnp;
    }
}
scheduler-maps {
    fcoe-map {
        forwarding-class fcoe scheduler fcoe-sched;
    }
}
schedulers {
    fcoe-sched {
        transmit-rate 3000000000;
        shaping-rate percent 100;
        priority low;
    }
}
```

NOTE: The forwarding class and classifier configurations are not shown because the **show** command does not display default portions of the configuration.

Verification

IN THIS SECTION

To verify that the CoS components and FIP snooping have been configured and are operating properly, perform these tasks. Because this example uses the default **fcoe** forwarding class and the default IEEE 802.1p trusted classifier, the verification of those configurations is not shown:

Verifying That the Output Queue Schedulers Have Been Created

Purpose

Verify that the output queue scheduler for FCoE traffic has the correct bandwidth parameters and priorities, and is mapped to the correct forwarding class (output queue). Queue scheduler verification is the same on each of the four switches.

Action

List the scheduler map using the operational mode command **show class-of-service scheduler-map fcoe-map**:

`user@switch>` **show class-of-service scheduler-map fcoe-map**

```
Scheduler map: fcoe-map, Index: 9023

  Scheduler: fcoe-sched, Forwarding class: fcoe, Index: 37289
    Transmit rate: 3000000000 bps, Rate Limit: none, Buffer size: remainder,
    Buffer Limit: none, Priority: low
    Excess Priority: unspecified
    Shaping rate: 100 percent,
```

```
    drop-profile-map-set-type: mark
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any             1    <default-drop-profile>
      Medium high     any             1    <default-drop-profile>
      High            any             1    <default-drop-profile>
```

Meaning

The **show class-of-service scheduler-map fcoe-map** command lists the properties of the scheduler map **fcoe-map**. The command output includes:

- The name of the scheduler map (**fcoe-map**)
- The name of the scheduler (**fcoe-sched**)
- The forwarding classes mapped to the scheduler (**fcoe**)
- The minimum guaranteed queue bandwidth (transmit rate **3000000000 bps**)
- The scheduling priority (**low**)
- The maximum bandwidth in the priority group the queue can consume (shaping rate **100 percent**)
- The drop profile loss priority for each drop profile name. This example does not include drop profiles because you do not apply drop profiles to FCoE traffic.

Verifying That the Priority Group Output Scheduler (Traffic Control Profile) Has Been Created (ETS Configuration Only)

Purpose

Verify that the traffic control profile **fcoe-tcp** has been created with the correct bandwidth parameters and scheduler mapping. Priority group scheduler verification is the same on each of the four switches.

Action

List the FCoE traffic control profile properties using the operational mode command **show class-of-service traffic-control-profile fcoe-tcp**:

`user@switch>` show class-of-service traffic-control-profile **fcoe-tcp**

```
Traffic control profile: fcoe-tcp, Index: 18303
  Shaping rate: 100 percent
  Scheduler map: fcoe-map
  Guaranteed rate: 3000000000
```

Meaning

The **show class-of-service traffic-control-profile fcoe-tcp** command lists all of the configured traffic control profiles. For each traffic control profile, the command output includes:

- The name of the traffic control profile (**fcoe-tcp**)
- The maximum port bandwidth the priority group can consume (shaping rate **100 percent**)
- The scheduler map associated with the traffic control profile (**fcoe-map**)
- The minimum guaranteed priority group port bandwidth (guaranteed rate **3000000000** in bps)

Verifying That the Forwarding Class Set (Priority Group) Has Been Created (ETS Configuration Only)

Purpose

Verify that the FCoE priority group has been created and that the **fcoe** priority (forwarding class) belongs to the FCoE priority group. Forwarding class set verification is the same on each of the four switches.

Action

List the forwarding class sets using the operational mode command **show class-of-service forwarding-class-set fcoe-pg**:

`user@switch>` show class-of-service forwarding-class-set **fcoe-pg**

```
Forwarding class set: fcoe-pg, Type: normal-type, Forwarding class set index: 31420

  Forwarding class                       Index
  fcoe                                     1
```

Meaning

The **show class-of-service forwarding-class-set fcoe-pg** command lists all of the forwarding classes (priorities) that belong to the **fcoe-pg** priority group, and the internal index number of the priority group. The command output shows that the forwarding class set **fcoe-pg** includes the forwarding class **fcoe**.

Verifying That Priority-Based Flow Control Has Been Enabled

Purpose

Verify that PFC is enabled on the FCoE code point. PFC verification is the same on each of the four switches.

Action

List the FCoE congestion notification profile using the operational mode command **show class-of-service congestion-notification fcoe-cnp**:

`user@switch>` show class-of-service congestion-notification **fcoe-cnp**

```
Type: Input, Name: fcoe-cnp, Index: 6879
Cable Length: 100 m
  Priority     PFC          MRU
  000          Disabled
  001          Disabled
  010          Disabled
  011          Enabled      2500
  100          Disabled
  101          Disabled
  110          Disabled
  111          Disabled
Type: Output
  Priority     Flow-Control-Queues
  000
               0
  001
               1
  010
               2
  011
               3
  100
               4
  101
               5
  110
               6
  111
               7
```

Meaning

The **show class-of-service congestion-notification fcoe-cnp** command lists all of the IEEE 802.1p code points in the congestion notification profile that have PFC enabled. The command output shows that PFC is enabled on code point **011** (**fcoe** queue) for the **fcoe-cnp** congestion notification profile.

The command also shows the default cable length (100 meters), the default maximum receive unit (2500 bytes), and the default mapping of priorities to output queues because this example does not include configuring these options.

Verifying That the Interface Class of Service Configuration Has Been Created

Purpose

Verify that the CoS properties of the interfaces are correct. The verification output on MC-LAG Switches S1 and S2 differs from the output on FCoE Transit Switches TS1 and TS2.

NOTE: The output is from the ETS hierarchical port scheduling configuration to show the more complex configuration. Direct port scheduling results do not show the traffic control profile or forwarding class sets because those elements are configured only for ETS. Instead, the name of the scheduler map is displayed under each interface.

Action

List the interface CoS configuration on MC-LAG Switches S1 and S2 using the operational mode command **show configuration class-of-service interfaces**:

`user@switch>` **show configuration class-of-service interfaces**

```
ae0 {
    forwarding-class-set {
        fcoe-pg {
            output-traffic-control-profile fcoe-tcp;
        }
    }
    congestion-notification-profile fcoe-cnp;
}

ae1 {
    forwarding-class-set {
        fcoe-pg {
            output-traffic-control-profile fcoe-tcp;
        }
    }
    congestion-notification-profile fcoe-cnp;
}
```

List the interface CoS configuration on FCoE Transit Switches TS1 and TS2 using the operational mode command **show configuration class-of-service interfaces**:

`user@switch>` **show configuration class-of-service interfaces**

```
xe-0/0/30 {
    forwarding-class-set {
        fcoe-pg {
            output-traffic-control-profile fcoe-tcp;
        }
```

```
    }
    congestion-notification-profile fcoe-cnp;
}
xe-0/0/31 {
    forwarding-class-set {
        fcoe-pg {
            output-traffic-control-profile fcoe-tcp;
        }
    }
    congestion-notification-profile fcoe-cnp;
}
xe-0/0/32 {
    forwarding-class-set {
        fcoe-pg {
            output-traffic-control-profile fcoe-tcp;
        }
    }
    congestion-notification-profile fcoe-cnp;
}
xe-0/0/33 {
    forwarding-class-set {
        fcoe-pg {
            output-traffic-control-profile fcoe-tcp;
        }
    }
    congestion-notification-profile fcoe-cnp;
}
ae1 {
    forwarding-class-set {
        fcoe-pg {
            output-traffic-control-profile fcoe-tcp;
        }
    }
    congestion-notification-profile fcoe-cnp;
}
```

Meaning

The **show configuration class-of-service interfaces** command lists the class of service configuration for all interfaces. For each interface, the command output includes:

- The name of the interface (for example, **ae0** or **xe-0/0/30**)
- The name of the forwarding class set associated with the interface (**fcoe-pg**)

- The name of the traffic control profile associated with the interface (output traffic control profile, **fcoe-tcp**)
- The name of the congestion notification profile associated with the interface (**fcoe-cnp**)

> **NOTE:** Interfaces that are members of a LAG are not shown individually. The LAG or MC-LAG CoS configuration is applied to all interfaces that are members of the LAG or MC-LAG. For example, the interface CoS configuration output on MC-LAG Switches S1 and S2 shows the LAG CoS configuration but does not show the CoS configuration of the member interfaces separately. The interface CoS configuration output on FCoE Transit Switches TS1 and TS2 shows the LAG CoS configuration but also shows the configuration for interfaces **xe-0/0/30**, **xe-0/0/31**, **xe-0/0/32**, and **xe-0/0/33**, which are not members of a LAG.

Verifying That the Interfaces Are Correctly Configured

Purpose

Verify that the LAG membership, MTU, VLAN membership, and port mode of the interfaces are correct. The verification output on MC-LAG Switches S1 and S2 differs from the output on FCoE Transit Switches T1 and T2.

Action

List the interface configuration on MC-LAG Switches S1 and S2 using the operational mode command **show configuration interfaces**:

`user@switch>` **show configuration interfaces**

```
xe-0/0/10 {
    ether-options {
        802.3ad ae0;
    }
}
xe-0/0/11 {
    ether-options {
        802.3ad ae0;
    }
}
xe-0/0/20 {
    ether-options {
        802.3ad ae1;
    }
}
xe-0/0/21 {
```

```
    ether-options {
        802.3ad ae1;
    }
}
ae0 {
    mtu 2180;
    unit 0 {
        family ethernet-switching {
            interface-mode trunk;
            vlan {
                members fcoe_vlan;
            }
        }
    }
}
ae1 {
    mtu 2180;
    unit 0 {
        family ethernet-switching {
            interface-mode trunk;
            vlan {
                members fcoe_vlan;
            }
        }
    }
}
```

List the interface configuration on FCoE Transit Switches TS1 and TS2 using the operational mode command **show configuration interfaces**:

`user@switch>` **show configuration interfaces**

```
xe-0/0/25 {
    ether-options {
        802.3ad ae1;
    }
}
xe-0/0/26 {
    ether-options {
        802.3ad ae1;
    }
}
```

```
xe-0/0/30 {
    mtu 2180;
    unit 0 {
        family ethernet-switching {
            interface-mode trunk;
            vlan {
                members fcoe_vlan;
            }
        }
    }
}
xe-0/0/31 {
    mtu 2180;
    unit 0 {
        family ethernet-switching {
            interface-mode trunk;
            vlan {
                members fcoe_vlan;
            }
        }
    }
}
xe-0/0/32 {
    mtu 2180;
    unit 0 {
        family ethernet-switching {
            interface-mode trunk;
            vlan {
                members fcoe_vlan;
            }
        }
    }
}
xe-0/0/33 {
    mtu 2180;
    unit 0 {
        family ethernet-switching {
            interface-mode trunk;
            vlan {
                members fcoe_vlan;
            }
        }
    }
}
```

```
ae1 {
    mtu 2180;
    unit 0 {
        family ethernet-switching {
            interface-mode trunk;
            vlan {
                members fcoe_vlan;
            }
        }
    }
}
```

Meaning

The **show configuration interfaces** command lists the configuration of each interface by interface name.

For each interface that is a member of a LAG, the command lists only the name of the LAG to which the interface belongs.

For each LAG interface and for each interface that is not a member of a LAG, the command output includes:

- The MTU (**2180**)
- The unit number of the interface (**0**)
- The interface mode (**trunk** mode both for interfaces that connect two switches and for interfaces that connect to FCoE hosts)
- The name of the VLAN in which the interface is a member (**fcoe_vlan**)

Verifying That FIP Snooping Is Enabled on the FCoE VLAN on FCoE Transit Switches TS1 and TS2 Access Interfaces

Purpose

Verify that FIP snooping is enabled on the FCoE VLAN access interfaces. FIP snooping is enabled only on the FCoE access interfaces, so it is enabled only on FCoE Transit Switches TS1 and TS2. FIP snooping is not enabled on MC-LAG Switches S1 and S2 because FIP snooping is done at the Transit Switch TS1 and TS2 FCoE access ports.

Action

List the port security configuration on FCoE Transit Switches TS1 and TS2 using the operational mode command **show configuration vlans fcoe_vlan forwarding-options fip-security**:

```
user@switch> show configuration vlans fcoe_vlan forwarding-options fip-security
```

```
interface ae1.0 {
    fcoe-trusted;
}
examine-vn2vn {
    beacon-period 90000;
}
```

Meaning

The **show configuration vlans fcoe_vlan forwarding-options fip-security** command lists VLAN FIP security information, including whether a port member of the VLAN is trusted. The command output shows that:

- LAG port **ae1.0**, which connects the FCoE transit switch to the MC-LAG switches, is configured as an FCoE trusted interface. FIP snooping is not performed on the member interfaces of the LAG (**xe-0/0/25** and **xe-0/0/26**).
- VN2VN_Port FIP snooping is enabled (**examine-vn2vn**) on the FCoE VLAN and the beacon period is set to 90000 milliseconds. On Transit Switches TS1 and TS2, all interface members of the FCoE VLAN perform FIP snooping unless the interface is configured as FCoE trusted. On Transit Switches TS1 and TS2, interfaces **xe-0/0/30**, **xe-0/0/31**, **xe-0/0/32**, and **xe-0/0/33** perform FIP snooping because they are not configured as FCoE trusted. The interface members of LAG **ae1** (**xe-0/0/25** and **xe-0/0/26**) do not perform FIP snooping because the LAG is configured as FCoE trusted.

Verifying That the FIP Snooping Mode Is Correct on FCoE Transit Switches TS1 and TS2

Purpose

Verify that the FIP snooping mode is correct on the FCoE VLAN. FIP snooping is enabled only on the FCoE access interfaces, so it is enabled only on FCoE Transit Switches TS1 and TS2. FIP snooping is not enabled on MC-LAG Switches S1 and S2 because FIP snooping is done at the Transit Switch TS1 and TS2 FCoE access ports.

Action

List the FIP snooping configuration on FCoE Transit Switches TS1 and TS2 using the operational mode command **show fip snooping brief**:

`user@switch>` **show fip snooping brief**

```
VLAN: fcoe_vlan,    Mode: VN2VN Snooping
  FC-MAP: 0e:fc:00
…
```

NOTE: The output has been truncated to show only the relevant information.

Meaning

The **show fip snooping brief** command lists FIP snooping information, including the FIP snooping VLAN and the FIP snooping mode. The command output shows that:

- The VLAN on which FIP snooping is enabled is **fcoe_vlan**
- The FIP snooping mode is VN2VN_Port FIP snooping (**VN2VN Snooping**)

RELATED DOCUMENTATION

Configuring Link Aggregation

Example: Configuring CoS PFC for FCoE Traffic | **490**

Example: Configuring CoS Hierarchical Port Scheduling (ETS) | **415**

Example: Configuring Queue Schedulers for Port Scheduling | **361**

Understanding MC-LAGs on an FCoE Transit Switch

Example: Configuring Lossless FCoE Traffic When the Converged Ethernet Network Does Not Use IEEE 802.1p Priority 3 for FCoE Traffic (FCoE Transit Switch)

IN THIS SECTION

The default system configuration supports FCoE traffic on priority 3 (IEEE 802.1p code point 011). If the FCoE traffic on your converged Ethernet network uses priority 3, the only user configuration required for lossless transport is to enable PFC on code point 011 on the FCoE ingress interfaces.

However, if your network uses a different priority than 3 for FCoE traffic, you need to configure lossless FCoE transport on that priority. This example shows you how to configure lossless FCoE transport on a converged Ethernet network that uses priority 5 (IEEE 802.1p code point 101) for FCoE traffic instead of using priority 3.

Requirements

This example uses the following hardware and software components:

- One switch used as an FCoE transit switch
- Junos OS Release 12.3 or later for the QFX Series

Overview

Although FCoE traffic typically uses IEEE 802.1p priority 3 on converged Ethernet networks, some networks use a different priority for FCoE traffic. Regardless of the priority used, FCoE traffic must receive lossless treatment. Supporting lossless behavior for FCoE traffic when your network does not use priority 3 requires configuring:

- A lossless forwarding class for FCoE traffic.
- A behavior aggregate (BA) classifier to map the FCoE forwarding class to the appropriate IEEE 802.1p priority.
- A congestion notification profile (CNP) to enable PFC on the FCoE code point at the interface ingress and to configure flow control on the interface egress. Flow control on the interface egress enables the interface to respond to PFC messages received from the connected peer and pause the correct IEEE 802.1p priority on the correct output queue.

  > **NOTE:** Configuring or changing PFC on an interface blocks the entire port until the PFC change is completed. After a PFC change is completed, the port is unblocked and traffic resumes. Blocking the port stops ingress and egress traffic, and causes packet loss on all queues on the port until the port is unblocked.

- A DCBX application and an application map to support DCBX application TLV exchange for the lossless FCoE traffic on the configured FCoE priority. By default, DCBX is enabled on all Ethernet interfaces, but only on priority 3 (IEEE 802.1p code point 011). To support DCBX application TLV exchange when you are not using the default configuration, you must configure all of the applications and map them to interfaces and priorities.

The priorities specified in the BA classifiers, CNP, and DCBX application map must match, or the configuration does not work. You must specify the same lossless FCoE forwarding class in each configuration and use the same IEEE 802.1p code point (priority) so that the FCoE traffic is properly classified into flows and so that those flows receive lossless treatment.

Topology

This example shows how to configure one lossless FCoE traffic class, map it to a priority other than priority 3, and configure flow control to ensure lossless behavior on the interfaces. This example uses two Ethernet interfaces, xe-0/0/25 and xe-0/0/26. The interfaces connect to a converged Ethernet network that uses IEEE 802.1p priority 5 (code point 101) for FCoE traffic.

The configuration on the two interfaces is the same. Both interfaces use the same explicitly configured lossless FCoE forwarding class and the same ingress classifier. Both interfaces enable PFC on priority 5 and enable flow control on the same output queue (which is mapped to the lossless FCoE forwarding class).

Table 94 on page 567 shows the configuration components for this example.

Table 94: Components of the Configuration Topology for FCoE Traffic That Does Not Use Priority 3

Component	Settings
Hardware	One switch
Forwarding class	Name—**fcoe1** Queue mapping—queue **5** Packet drop attribute—**no-loss** **NOTE:** A lossless forwarding class can be mapped to any output queue. However, because the **fcoe1** forwarding class uses priority 5 in this example, matching that traffic to a forwarding class that uses queue 5 creates a configuration that is logical and easy to map because the priority and the queue are identified by the same number.
BA classifier	Name—**fcoe_p5** FCoE priority mapping—Forwarding class **fcoe1** mapped to code point **101** (IEEE 802.1p priority 5) and a packet loss priority of **low**.

Table 94: Components of the Configuration Topology for FCoE Traffic That Does Not Use Priority 3 *(continued)*

Component	Settings
PFC configuration (CNPs)	CNP name—**fcoe_p5_cnp** Input CNP code point—**101** MRU—**2240** bytes Cable length—**100** meters Output CNP code point—**101** Output CNP flow control queue—**5** **NOTE:** When you apply a CNP with an explicit output queue flow control configuration to an interface, the explicit CNP overwrites the default output CNP. The output queues that are enabled for pause in the default configuration (queues 3 and 4) are not enabled for pause unless they are included in the explicitly configured output CNP.
DCBX application mapping	Application name—**fcoe_p5_app** Application EtherType—**0x8906** Application map name—**fcoe_p5_app_map** Application map code points—**101** **NOTE:** LLDP and DCBX must be enabled on the interface. By default, LLDP and DCBX are enabled on all Ethernet interfaces.

NOTE: This example does not include scheduling (bandwidth allocation) configuration or the FIP snooping configuration. This example focuses only on the lossless FCoE priority configuration.

QFX10000 switches do not support FIP snooping. For this reason, QFX10000 switches cannot be used as FCoE access transit switches. QFX10000 switches can be used as intermediate or aggregation transit switches in the FCoE path, between an FCoE access transit switch that performs FIP snooping and an FCF.

Configuration

CLI Quick Configuration

To quickly configure a lossless FCoE forwarding class that uses a different priority than IEEE 802.1p priority 3 for FCoE traffic on an FCoE transit switch, 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.

set class-of-service forwarding-classes class fcoe1 queue-num 5 no-loss
set class-of-service classifiers ieee-802.1 fcoe_p5 forwarding-class fcoe1 loss-priority low code-points 101
set class-of-service interfaces xe-0/0/25 unit 0 classifiers ieee-802.1 fcoe_p5
set class-of-service interfaces xe-0/0/26 unit 0 classifiers ieee-802.1 fcoe_p5
set class-of-service congestion-notification-profile fcoe_p5_cnp input ieee-802.1 code-point 101 pfc mru 2240
set class-of-service congestion-notification-profile fcoe_p5_cnp input cable-length 100
set class-of-service congestion-notification-profile fcoe_p5_cnp output ieee-802.1 code-point 101 pfc flow-control-queue 5
set class-of-service interfaces xe-0/0/25 congestion-notification-profile fcoe_p5_cnp
set class-of-service interfaces xe-0/0/26 congestion-notification-profile fcoe_p5_cnp
set applications application fcoe_p5_app ether-type 0x8906
set policy-options application-maps fcoe_p5_app_map application fcoe_p5_app code-points 101
set protocols dcbx interface xe-0/0/25 application-map fcoe_p5_app_map
set protocols dcbx interface xe-0/0/26 application-map fcoe_p5_app_map

Configuring A Lossless FCoE Forwarding Class On IEEE 802.1p Priority 5

Step-by-Step Procedure

To configure a lossless forwarding class for FCoE traffic on IEEE 802.1p priority 5 (code point 101), classify FCoE traffic into the lossless forwarding class, configure a congestion notification profile to enable PFC on the FCoE priority and output queue, and configure DCBX application protocol TLV exchange for traffic on the FCoE priority:

1. Configure the lossless forwarding class (named **fcoe1** and mapped to output queue **5**) for FCoE traffic on IEEE 802.1p priority 5:

```
[edit class-of-service]
user@switch# set forwarding-classes class fcoe1 queue-num 5 no-loss
```

2. Configure the ingress classifier (**fcoe_p5**). The classifier maps the FCoE priority (code point **101**) to the lossless FCoE forwarding class **fcoe1**:

```
[edit class-of-service classifiers]
user@switch# set ieee-802.1 fcoe_p5 forwarding-class fcoe1 loss-priority low code-points 101
```

3. Apply the classifier to interfaces **xe-0/0/25** and **xe-0/026**:

   [edit class-of-service]
   user@switch# **set interfaces xe-0/0/25 unit 0 classifiers ieee-802.1 fcoe_p5**
   user@switch# **set interfaces xe-0/0/26 unit 0 classifiers ieee-802.1 fcoe_p5**

4. Configure the CNP. The input stanza enables PFC on the FCoE priority (IEEE 802.1p code point 101), sets the MRU value (2240 bytes), and sets the cable length value (100 meters). The output stanza configures flow control on output queue 5 on the FCoE priority:

   [edit class-of-service]
   user@switch# **set congestion-notification-profile fcoe_p5_cnp input ieee-802.1 code-point 101 pfc mru 2240**
   user@switch# **set congestion-notification-profile fcoe_p5_cnp input cable-length 100**
   user@switch# **set congestion-notification-profile fcoe_p5_cnp output ieee-802.1 code-point 101 pfc flow-control-queue 5**

5. Apply the CNP to the interfaces:

   [edit class-of-service]
   user@switch# **set interfaces xe-0/0/25 congestion-notification-profile fcoe_p5_cnp**
   user@switch# **set interfaces xe-0/0/26 congestion-notification-profile fcoe_p5_cnp**

6. Configure the DCBX application for FCoE to map to the Ethernet interfaces, so that DCBX can exchange application protocol TLVs on the IEEE 802.1p priority 5 instead of on the default priority 3:

   [edit]
   user@switch# **set applications application fcoe_p5_app ether-type 0x8906**

7. Configure a DCBX application map to map the FCoE application to the correct IEEE 802.1p FCoE priority:

   [edit]
   user@switch# **set policy-options application-maps fcoe_p5_app_map application fcoe_p5_app code-points 101**

8. Apply the application map to the Ethernet interfaces so that DCBX exchanges FCoE application TLVs on the correct code point:

   [edit]

```
user@switch# set protocols dcbx interface xe-0/0/25 application-map fcoe_p5_app_map
user@switch# set protocols dcbx interface xe-0/0/26 application-map fcoe_p5_app_map
```

Verification

IN THIS SECTION

To verify the configuration and proper operation of the lossless forwarding class and IEEE 802.1p priority, perform these tasks:

Verifying the Forwarding Class Configuration

Purpose

Verify that the lossless forwarding class **fcoe1** has been created.

Action

Show the forwarding class configuration by using the operational command **show class-of-service forwarding class**:

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

```
Forwarding class                        ID       Queue  Policing priority  No-Loss
  best-effort                           0         0          normal         Disabled

  fcoe                                  1         3          normal         Enabled
  no-loss                               2         4          normal         Enabled
  network-control                       3         7          normal         Disabled
```

```
fcoe1                                4         5        normal      Enabled
mcast                                8         8        normal      Disabled
```

Meaning

The **show class-of-service forwarding-class** command shows all of the forwarding classes. The command output shows that the **fcoe1** forwarding class is configured on output queue **5** with the no-loss packet drop attribute enabled.

Because we did not explicitly configure the default forwarding classes, they remain in their default state, including the lossless configuration of the **fcoe** and **no-loss** default forwarding classes.

Verifying the Behavior Aggregate Classifier Configuration

Purpose

Verify that the classifier maps the forwarding classes to the correct IEEE 802.1p code points (priorities) and packet loss priorities.

Action

List the classifier configured to support lossless FCoE transport using the operational mode command **show class-of-service classifier**:

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

```
Classifier: fcoe_p5, Code point type: ieee-802.1, Index: 63065
  Code point          Forwarding class                    Loss priority
  101                 fcoe1                               low
```

Meaning

The **show class-of-service classifier** command shows the IEEE 802.1p code points and the loss priorities that are mapped to the forwarding classes in each classifier.

Classifier **fcoe_p5** maps code point **101** (priority 5) to explicitly configured lossless forwarding class **fcoe1** and a packet loss priority of **low**, and all other priorities to the **best-effort** forwarding class with a packet loss priority of **high**.

Verifying the PFC Flow Control Configuration (CNP)

Purpose

Verify that PFC is enabled on the correct input priority and that flow control is configured on the correct output queue in the CNP.

Action

Display the congestion notification profile using the operational mode command **show class-of-service congestion-notification**:

`user@switch>` **show class-of-service congestion-notification**

```
Name: fcoe_p5_cnp, Index: 12137
Type: Input
Cable Length: 100 m
  Priority     PFC          MRU
  000          Disabled
  001          Disabled
  010          Disabled
  011          Disabled
  100          Disabled
  101          Enabled      2240
  110          Disabled
  111          Disabled
Type: Output
  Priority     Flow-Control-Queues
  101
               5
```

Meaning

The **show class-of-service congestion-notification** command shows the input and output stanzas of the configured CNPs.

The **fcoe_p5_cnp** CNP input stanza shows that PFC is enabled on code point **101** (priority 5), the MRU is **2240** bytes, and the cable length is **100** meters. The CNP output stanza shows that output flow control is configured on queue **5** for code point **101** (priority 5).

Verifying the Interface Configuration

Purpose

Verify that the correct classifier and congestion notification profile are configured on the interfaces.

Action

List the ingress interfaces using the operational mode commands **show configuration class-of-service interfaces xe-0/0/25** and **show configuration class-of-service interfaces xe-0/0/26**:

`user@switch>` **show configuration class-of-service interfaces xe-0/0/25**

```
congestion-notification-profile fcoe_p5_cnp;
unit 0 {
```

```
    classifiers {
        ieee-802.1 fcoe_p5;
    }
}
```

`user@switch>` **show configuration class-of-service interfaces xe-0/0/26**

```
congestion-notification-profile fcoe_p5_cnp;
unit 0 {
    classifiers {
        ieee-802.1 fcoe_p5;
    }
}
```

Meaning

Both the **show configuration class-of-service interfaces xe-0/0/25** command and the **show configuration class-of-service interfaces xe-0/0/26** command show that the congestion notification profile **fcoe_p5_cnp** is configured on each interface, and that the IEEE 802.1p classifier associated with each interface is **fcoe_p5**.

Verifying the DCBX Application Configuration

Purpose

Verify that the DCBX application for FCoE is configured.

Action

List the DCBX applications by using the configuration mode command **show applications**:

`user@switch#` **show applications**

```
application fcoe_p5_app {
    ether-type 0x8906;
```

Meaning

The **show applications** configuration mode command shows all of the configured applications. The output shows that the application **fcoe_p5_app** is configured with an EtherType of **0x8906**.

Verifying the DCBX Application Map Configuration

Purpose

Verify that the application map is configured.

Action

List the application maps by using the configuration mode command **show policy-options application-maps**:

```
user@switch# show policy-options application-maps
```

```
fcoe_p5_app_map {
    application fcoe_p5_app code-points 101;
}
```

Meaning

The **show policy-options application-maps** configuration mode command lists all of the configured application maps and the applications that belong to each application map. The output shows that application map **fcoe_p5_app_map** consists of the application named **fcoe_p5_app**, which is mapped to IEEE 802.1p code point **101**.

Verifying the DCBX Application Protocol Exchange Interface Configuration

Purpose

Verify that the application map is applied to the correct interfaces.

Action

List the application maps on each interface using the configuration mode command **show protocols dcbx**:

```
user@switch# show protocols dcbx
```

```
interface xe-0/0/25.0 {
    application-map fcoe_p5_app_map;
}
interface xe-0/0/26.0 {
    application-map fcoe_p5_app_map;
}
```

Meaning

The **show protocols dcbx** configuration mode command lists the application map association with interfaces. The output shows that interfaces **xe-0/0/25.0** and **xe-0/0/26.0** use application map **fcoe_p5_app_map**.

RELATED DOCUMENTATION

Example: Configuring Two or More Lossless FCoE IEEE 802.1p Priorities on Different FCoE Transit Switch Interfaces | 587

Example: Configuring Two or More Lossless FCoE Priorities on the Same FCoE Transit Switch Interface

IN THIS SECTION

The default system configuration supports FCoE traffic on priority 3 (IEEE 802.1p code point 011). If the FCoE traffic on your converged Ethernet network uses priority 3, the only user configuration required for lossless transport is to enable PFC on code point 011 on the FCoE ingress interfaces.

However, if your converged Ethernet network uses more than one priority for FCoE traffic, you need to configure lossless transport for each FCoE priority. This example shows you how to configure lossless FCoE transport on a converged Ethernet network that uses both priority 3 (IEEE 802.1p code point 011) and priority 5 (IEEE 802.1p code point 101) for FCoE traffic.

Requirements

This example uses the following hardware and software components:

- One switch used as an FCoE transit switch
- Junos OS Release 12.3 or later for the QFX Series

Overview

Some network topologies support FCoE traffic on more than one IEEE 802.1p priority. For example, a converged Ethernet network might include two separate FCoE networks that use different priorities to identify traffic. Interfaces that carry traffic for both FCoE networks need to support lossless FCoE transport on both priorities.

Supporting lossless behavior for two FCoE traffic classes requires configuring:

- At least one lossless forwarding class for FCoE traffic (this example uses the default **fcoe** forwarding class as one of the lossless FCoE forwarding classes, so we need to explicitly configure only one FCoE forwarding class).
- A behavior aggregate (BA) classifier to map the FCoE forwarding classes to the appropriate IEEE 802.1p code points (priorities).
- A congestion notification profile (CNP) to enable PFC on the FCoE code points at the interface ingress and to configure PFC flow control on the interface egress so that the interface can respond to PFC messages received from the connected peer.

  > **NOTE:** Configuring or changing PFC on an interface blocks the entire port until the PFC change is completed. After a PFC change is completed, the port is unblocked and traffic resumes. Blocking the port stops ingress and egress traffic, and causes packet loss on all queues on the port until the port is unblocked.

- DCBX applications and an application map to support DCBX application TLV exchange for the lossless FCoE traffic on the configured FCoE priorities. By default, DCBX is enabled on all Ethernet interfaces, but only on priority 3 (IEEE 802.1p code point 011). To support DCBX application TLV exchange when you are not using the default configuration, you must configure all of the applications and map them to interfaces and priorities.

The priorities specified in the BA classifier, CNP, and DCBX application map must match, or the configuration does not work. You must specify the same lossless FCoE forwarding class in each configuration and use the same IEEE 802.1p code point (priority) so that the FCoE traffic is properly classified into flows and so that those flows receive lossless treatment.

Topology

This example shows how to configure two lossless FCoE traffic classes on an interface, map them to two different priorities, and configure flow control to ensure lossless behavior. This example uses two Ethernet interfaces, xe-0/0/20 and xe-0/0/21, that are connected to the converged Ethernet network. Both interfaces transport FCoE traffic on priorities 3 (011) and 5 (101), and must support lossless transport of that traffic.

Table 95 on page 578 shows the configuration components for this example.

Table 95: Components of the Two Lossless FCoE Priorities on an Interface Configuration Topology

Component	Settings
Hardware	One switch
Forwarding classes	Name—**fcoe1** Queue mapping—queue **5** Packet drop attribute—**no-loss** **NOTE:** A lossless forwarding class can be mapped to any output queue. However, because the **fcoe1** forwarding class uses priority 5 in this example, matching that traffic to a forwarding class that uses queue 5 creates a configuration that is logical and easy to map because the priority and the queue are identified by the same number. Name—**fcoe** This is the default lossless FCoE forwarding class, so no configuration required. The **fcoe** forwarding class is mapped to priority 3 (IEEE 802.1p code point 011) and to output queue 3 with a packet drop attribute of **no-loss**.
BA classifier	Name—**fcoe_classifier** FCoE priority mapping for forwarding class **fcoe**—mapped to code point **011** (IEEE 802.1p priority 3) and a packet loss priority of **low**. FCoE priority mapping for forwarding class **fcoe1**—mapped to code point **101** (IEEE 802.1p priority 5) and a packet loss priority of **low**.
PFC configuration (CNP)	CNP name—**fcoe_cnp** Input CNP code points—**011** and **101** MRU—2240 bytes Cable length—100 meters Output CNP code points—**011** and **101** Output CNP flow control queues—**3** and **5** **NOTE:** When you apply a CNP with an explicit output queue flow control configuration to an interface, the explicit CNP overwrites the default output CNP. The output queues that are enabled for PFC pause in the default configuration (queues 3 and 4) are not enabled for PFC pause unless they are included in the explicitly configured output CNP. In this example, because the explicit output CNP overwrites the default output CNP, we must explicitly configure flow control on queue 3.

Table 95: Components of the Two Lossless FCoE Priorities on an Interface Configuration Topology *(continued)*

Component	Settings
DCBX application mapping	Application name—**fcoe_app** Application EtherType—**0x8906** Application map name—**fcoe_app_map** Application map code points—**011** and **101** **NOTE:** LLDP and DCBX must be enabled on the interface. By default, LLDP and DCBX are enabled on all Ethernet interfaces.
Interfaces	Interfaces **xe-0/0/20** and **xe-0/0/21** use the same configuration: • Classifier—**fcoe_classifier** • CNP—**fcoe_cnp** • DCBX application map—**fcoe_app_map**

NOTE: This example does not include scheduling (bandwidth allocation) configuration or the FIP snooping configuration. This examples focuses only on the lossless FCoE priority configuration.

QFX10000 switches do not support FIP snooping. For this reason, QFX10000 switches cannot be used as FCoE access transit switches. QFX10000 switches can be used as intermediate or aggregation transit switches in the FCoE path, between an FCoE access transit switch that performs FIP snooping and an FCF.

Configuration

CLI Quick Configuration

To quickly configure two lossless FCoE forwarding classes that use different priorities on an FCoE transit switch interface, 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.

set class-of-service forwarding-classes class fcoe1 queue-num 5 no-loss
set class-of-service classifiers ieee-802.1 fcoe_classifier forwarding-class fcoe loss-priority low code-points 011
set class-of-service classifiers ieee-802.1 fcoe_classifier forwarding-class fcoe1 loss-priority low code-points 101set class-of-service interfaces xe-0/0/20 unit 0 classifiers ieee-802.1 fcoe_classifier

set class-of-service interfaces xe-0/0/21 unit 0 classifiers ieee-802.1 fcoe_classifier

set class-of-service congestion-notification-profile fcoe_cnp input ieee-802.1 code-point 011 pfc mru 2240

set class-of-service congestion-notification-profile fcoe_cnp input ieee-802.1 code-point 101 pfc mru 2240

set class-of-service congestion-notification-profile fcoe_cnp input cable-length 100

set class-of-service congestion-notification-profile fcoe_cnp output ieee-802.1 code-point 011 pfc flow-control-queue 3

set class-of-service congestion-notification-profile fcoe_cnp output ieee-802.1 code-point 101 pfc flow-control-queue 5

set class-of-service interfaces xe-0/0/20 congestion-notification-profile fcoe_cnp

set class-of-service interfaces xe-0/0/21 congestion-notification-profile fcoe_cnp

set applications application fcoe_app ether-type 0x8906

set policy-options application-maps fcoe_app_map application fcoe_app code-points [011 101]

set protocols dcbx interface xe-0/0/20 application-map fcoe_app_map

set protocols dcbx interface xe-0/0/21 application-map fcoe_app_map

Step-by-Step Procedure

To configure two lossless forwarding classes for FCoE traffic on the same interface, classify FCoE traffic into the forwarding classes, configure CNPs to enable PFC on the FCoE priorities and output queues, and configure DCBX application protocol TLV exchange for traffic on both FCoE priorities:

1. Configure lossless forwarding class **fcoe1** and map it to output queue **5** for FCoE traffic that uses IEEE 802.1p priority 5:

   ```
   [edit class-of-service]
   user@switch# set forwarding-classes class fcoe1 queue-num 5 no-loss
   ```

   > **NOTE:** This examples uses the default **fcoe** forwarding class as the other lossless FCoE forwarding class.

2. Configure the ingress classifier. The classifier maps the FCoE priorities (IEEE 802.1p code points **011** and **101**) to lossless FCoE forwarding classes **fcoe** and **fcoe1**, respectively:

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

```
user@switch# set ieee-802.1 fcoe_classifier forwarding-class fcoe1 loss-priority low code-points 101
```

3. Apply the classifier to the interfaces:

```
[edit class-of-service]
user@switch# set interfaces xe-0/0/20 unit 0 classifiers ieee-802.1 fcoe_classifier
user@switch# set interfaces xe-0/0/21 unit 0 classifiers ieee-802.1 fcoe_classifier
```

4. Configure the CNP. The input stanza enables PFC on the FCoE priorities (IEEE 802.1p code points 011 and 101), sets the MRU value (2240 bytes), and sets the cable length value (100 meters). The output stanza configures flow control on output queues 3 and 5 on the FCoE priorities:

```
[edit class-of-service]
user@switch# set congestion-notification-profile fcoe_cnp input ieee-802.1 code-point 011 pfc mru 2240
user@switch# set congestion-notification-profile fcoe_cnp input ieee-802.1 code-point 101 pfc mru 2240
user@switch# set congestion-notification-profile fcoe_cnp input cable-length 100
user@switch# set congestion-notification-profile fcoe_cnp output ieee-802.1 code-point 011 pfc flow-control-queue 3
user@switch# set congestion-notification-profile fcoe_cnp output ieee-802.1 code-point 101 pfc flow-control-queue 5
```

5. Apply the CNP to the interfaces:

```
[edit class-of-service]
user@switch# set interfaces xe-0/0/20 congestion-notification-profile fcoe_cnp
user@switch# set interfaces xe-0/0/21 congestion-notification-profile fcoe_cnp
```

6. Configure a DCBX application for FCoE to map to the Ethernet interfaces, so that DCBX can exchange application protocol TLVs on both of the IEEE 802.1p priorities used for FCoE transport:

```
[edit]
user@switch# set applications application fcoe_app ether-type 0x8906
```

7. Configure a DCBX application map to map the FCoE application to the correct IEEE 802.1p FCoE priorities:

```
[edit]
user@switch# set policy-options application-maps fcoe_app_map application fcoe_app code-points
[011 101]
```

8. Apply the application map to the interfaces so that DCBX exchanges FCoE application TLVs on the correct code points:

```
[edit]
user@switch# set protocols dcbx interface xe-0/0/20 application-map fcoe_app_map
user@switch# set protocols dcbx interface xe-0/0/21 application-map fcoe_app_map
```

Verification

IN THIS SECTION

To verify the configuration and proper operation of the lossless forwarding classes and IEEE 802.1p priorities, perform these tasks:

Verifying the Forwarding Class Configuration

Purpose

Verify that the lossless forwarding class **fcoe1** has been created.

Action

Show the forwarding class configuration by using the operational command **show class-of-service forwarding class**:

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

```
Forwarding class                       ID       Queue  Policing priority  No-Loss
  best-effort                          0          0         normal        Disabled

  fcoe                                 1          3         normal        Enabled
  no-loss                              2          4         normal        Enabled
  network-control                      3          7         normal        Disabled

  fcoe1                                4          5         normal        Enabled
  mcast                                8          8         normal        Disabled
```

Meaning

The **show class-of-service forwarding-class** command shows all of the forwarding classes. The command output shows that the **fcoe1** forwarding class is configured on output queue **5** with the no-loss packet drop attribute enabled.

Because we did not explicitly configure the default forwarding classes, they remain in their default state, including the lossless configuration of the **fcoe** and **no-loss** default forwarding classes.

Verifying the Behavior Aggregate Classifier Configuration

Purpose

Verify that the three classifiers map the forwarding classes to the correct IEEE 802.1p code points (priorities) and packet loss priorities.

Action

List the classifiers using the operational mode command **show class-of-service classifier**:

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

```
Classifier: fcoe_classifier, Code point type: ieee-802.1, Index: 10964
  Code point          Forwarding class                    Loss priority
  011                 fcoe                                low
  101                 fcoe1                               low
```

Meaning

The **show class-of-service classifier** command shows the IEEE 802.1p code points and the loss priorities that are mapped to the forwarding classes in each classifier.

Classifier **fcoe_classifier** maps code point **011** to default lossless forwarding class **fcoe** and a packet loss priority of **low**, and maps code point **101** to explicitly configured lossless forwarding class **fcoe1** and a packet loss priority of **low**.

Verifying the PFC Flow Control Configuration (CNP)

Purpose

Verify that PFC is enabled on the correct input priorities and that flow control is configured on the correct output queues and priorities.

Action

List the CNPs using the operational mode command **show class-of-service congestion-notification**:

```
user@switch> show class-of-service congestion-notification
```

```
Name: fcoe_cnp, Index: 46504
Type: Input
Cable Length: 100 m
  Priority     PFC          MRU
  000          Disabled
  001          Disabled
  010          Disabled
  011          Enabled      2240
  100          Disabled
  101          Enabled      2240
  110          Disabled
  111          Disabled
Type: Output
  Priority     Flow-Control-Queues
  011
               3
  101
               5
```

Meaning

The **show class-of-service congestion-notification** command shows the input and output stanzas of the CNP.

The CNP **fcoe_cnp** input stanza shows that PFC is enabled on code points **011** and **101**, the MRU is **2240** bytes on both priorities, and the interface cable length is **100** meters. The CNP output stanza shows that output flow control is configured on queues **3** and **5** for code points **011** and **101**, respectively.

Verifying the Interface Configuration

Purpose

Verify that the classifier and congestion notification profile are configured on the interfaces. Both interfaces should show the same configuration.

Action

List the ingress interfaces using the operational mode commands **show configuration class-of-service interfaces xe-0/0/20** and **show configuration class-of-service interfaces xe-0/0/21**:

`user@switch>` **show configuration class-of-service interfaces xe-0/0/20**

```
ccongestion-notification-profile fcoe_cnp;
unit 0 {
    classifiers {
        ieee-802.1 fcoe_classifier;
    }
}
```

`user@switch>` **show configuration class-of-service interfaces xe-0/0/21**

```
congestion-notification-profile fcoe_cnp;
unit 0 {
    classifiers {
        ieee-802.1 fcoe_classifier;
    }
}
```

Meaning

The **show configuration class-of-service interfaces xe-0/0/20** command shows that the congestion notification profile **fcoe_cnp** is configured on the interface, and that the IEEE 802.1p classifier associated with the interface is **fcoe_classifier**.

The **show configuration class-of-service interfaces xe-0/0/21** command shows that the congestion notification profile **fcoe_cnp** is configured on the interface, and that the IEEE 802.1p classifier associated with the interface is **fcoe_classifier**.

Verifying the DCBX Application Configuration

Purpose

Verify that the DCBX application for FCoE is configured.

Action

List the DCBX applications by using the configuration mode command **show applications**:

`user@switch#` **show applications**

```
application fcoe_app {
    ether-type 0x8906;
```

Meaning

The **show applications** configuration mode command shows all of the configured applications. The output shows that the application **fcoe_app** is configured with an EtherType of **0x8906**.

Verifying the DCBX Application Map Configuration

Purpose

Verify that the application map is configured.

Action

List the application maps by using the configuration mode command **show policy-options application-maps**:

`user@switch#` **show policy-options application-maps**

```
fcoe_app_map {
    application fcoe_app code-points [011 101];
}
```

Meaning

The **show policy-options application-maps** configuration mode command lists all of the configured application maps and the applications that belong to each application map. The output shows that application map **fcoe_app_map** consists of the application named **fcoe_app**, which is mapped to IEEE 802.1p code points **011** and **101** (priorities 3 and 5, respectively).

Verifying the DCBX Application Protocol Exchange Interface Configuration

Purpose

Verify that the application map is applied to the interfaces.

Action

List the application maps on each interface using the configuration mode command **show protocols dcbx**:

`user@switch#` **show protocols dcbx**

```
interface xe-0/0/20.0 {
    application-map fcoe_app_map;
}
interface xe-0/0/21.0 {
    application-map fcoe_app_map;
}
```

Meaning

The **show protocols dcbx** configuration mode command lists the application map association with interfaces. The output shows that interfaces **xe-0/0/20.0** and **xe-0/0/21.0** use application map **fcoe_app_map**.

RELATED DOCUMENTATION

Example: Configuring Two or More Lossless FCoE IEEE 802.1p Priorities on Different FCoE Transit Switch Interfaces

IN THIS SECTION

Although the default configuration provides two lossless forwarding classes mapped to two different IEEE 802.1p priorities (code points), you can explicitly configure up to six lossless forwarding classes and map them to different priorities. You can support up to six different types of lossless traffic, and you can support the same type of traffic if it uses different priorities in different parts of your converged network.

This example shows you how to configure two lossless forwarding classes for FCoE traffic and map them to two different priorities on an FCoE transit switch.

Requirements

This example uses the following hardware and software components:

- One switch used as an FCoE transit switch
- Junos OS Release 12.3 or later for the QFX Series

Overview

Some network topologies support FCoE traffic on more than one IEEE 802.1p priority. For example, when the switch acts as a transit switch, it could be connected to two QFX3500 switches in FCoE-FC gateway mode. Each of the gateway switches could connect a set of FCoE clients to a different SAN, and each set of FCoE clients could use a different priority for FCoE traffic to avoid fate sharing and maintain separation of the two FCoE networks. In this case, you need to configure two forwarding classes for FCoE traffic, each mapped to a different output queue and a different priority.

Supporting lossless behavior for two FCoE traffic classes requires configuring:

- At least one lossless forwarding class for FCoE traffic (this example uses the default **fcoe** forwarding class as one of the two lossless FCoE forwarding classes, so we need to explicitly configure only one FCoE forwarding class)
- Behavior aggregate (BA) classifiers to map the FCoE forwarding classes to the appropriate IEEE 802.1p code points (priorities) on each interface
- Congestion notification profiles (CNPs) for each interface to enable PFC on the FCoE code points at the interface ingress and to configure PFC flow control on the interface egress so that the interface can respond to PFC messages received from the connected peer

  > **NOTE:** Configuring or changing PFC on an interface blocks the entire port until the PFC change is completed. After a PFC change is completed, the port is unblocked and traffic resumes. Blocking the port stops ingress and egress traffic, and causes packet loss on all queues on the port until the port is unblocked.

- DCBX applications and an application map to support DCBX application TLV exchange for the lossless FCoE traffic on the configured FCoE priorities. By default, DCBX is enabled on all Ethernet interfaces, but only on priority 3 (IEEE 802.1p code point 011). To support DCBX application TLV exchange when you are not using the default configuration, you must configure all of the applications and map them to interfaces and priorities.

The priorities specified in the BA classifiers, CNPs, and DCBX application map must match, or the configuration does not work. You must specify the same lossless FCoE forwarding class in each configuration and use the same IEEE 802.1p code point (priority) so that the FCoE traffic is properly classified into flows and so that those flows receive lossless treatment.

Topology

This example shows how to configure two lossless FCoE traffic classes, map them to two different priorities, and configure flow control to ensure lossless behavior for those priorities on the interfaces. This example uses three Ethernet interfaces, xe-0/0/20, xe-0/0/21, and xe-0/0/22:

- Interface xe-0/0/20 connects to an FCoE-FC gateway that connects to Fibre Channel (FC) SAN 1. FCoE traffic to and from FC SAN 1 uses the default **fcoe** forwarding class and the default mapping to priority 3 (IEEE 802.1p code point 011) and output queue 3.
- Interface xe-0/0/21 connects to another FCoE-FC gateway that connects to Fibre Channel (FC) SAN 2. FCoE traffic to and from FC SAN-2 uses an explicitly configured FCoE forwarding class that is mapped to priority 5 (code point 101) and output queue 5.
- Interface xe-0/0/22 connects to FCoE devices on the converged Ethernet network and handles traffic destined for FC SAN 1 and FC SAN 2. Interface xe-0/0/22 must properly handle lossless FCoE traffic of both priorities (both FCoE forwarding classes), including pausing the traffic on ingress or egress as required.

Figure 26 on page 589 shows the topology for this example, and Table 96 on page 589 shows the configuration components for this example.

Figure 26: Topology of the Two Lossless FCoE Priorities Example

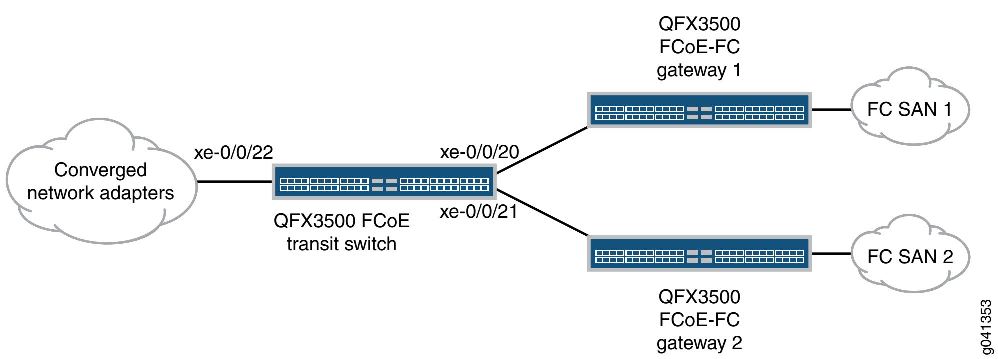


Table 96: Components of the Two Lossless FCoE Priorities Configuration Topology

Component	Settings
Hardware	One switch

Table 96: Components of the Two Lossless FCoE Priorities Configuration Topology *(continued)*

Component	Settings
Forwarding classes	Name—**fcoe1** Queue mapping—queue **5** Packet drop attribute—**no-loss** **NOTE:** A lossless forwarding class can be mapped to any output queue. However, because the **fcoe1** forwarding class uses priority 5 in this example, matching that traffic to a forwarding class that uses queue 5 creates a configuration that is logical and easy to map because the priority and the queue are identified by the same number. Name—**fcoe** This is the default lossless FCoE forwarding class, so no configuration required. The **fcoe** forwarding class is mapped to priority 3 (IEEE 802.1p code point 011) and to output queue 3 with a packet drop attribute of **no-loss**
BA classifiers	Each interface requires a different classifier because each interface handles a different subset of FCoE traffic. • Interface xe-0/0/20 classifier: Name—**fcoe_p3** FCoE priority mapping—Forwarding class **fcoe** mapped to code point **011** (IEEE 802.1p priority 3) and a packet loss priority of **low**. • Interface xe-0/0/21 classifier: Name—**fcoe_p5** FCoE priority mapping—Forwarding class **fcoe1** mapped to code point **101** (IEEE 802.1p priority 5) and a packet loss priority of **low**. • Interface xe-0/0/22 classifier: Name—**fcoe_p3_p5** FCoE priority mapping—Forwarding class **fcoe1** mapped to code point **101** and a packet loss priority of **low**, and forwarding class **fcoe** mapped to code point **011** and a packet loss priority of **low**.

Table 96: Components of the Two Lossless FCoE Priorities Configuration Topology *(continued)*

Component	Settings
PFC configuration (CNPs)	Each interface requires a different CNP because each interface handles a different subset of FCoE traffic and must pause that traffic on different priorities. • Interface xe-0/0/20 CNP: CNP name—**fcoe_p3_cnp** Input CNP code point—**011** MRU—2240 bytes Cable length—100 meters **NOTE:** Because interface xe-0/0/20 uses the default FCoE configuration, output queue 3 is paused by default and you do not need to configure the output stanza of the CNP. • Interface xe-0/0/21 CNP: CNP name—**fcoe_p5_cnp** Input CNP code point—**101** MRU—**2240** bytes Cable length—**150** meters Output CNP code point—**101** Output CNP flow control queue—**5** • Interface xe-0/0/22 CNP: CNP name—**fcoe_p3_p5_cnp** Input CNP code points—**011** and **101** MRU—**2240** bytes (both priorities) Cable length—**100** meters Output CNP code points—**011** (for queue **3**) and **101** (for queue **5**) Output CNP flow control queues—**3** for priority 3 (code point 011) and **5** for priority 5 (code point 101) **NOTE:** When you apply a CNP with an explicit output queue flow control configuration to an interface, the explicit CNP overwrites the default output CNP. The output queues that are enabled for pause in the default configuration (queues 3 and 4) are not enabled for pause unless they are included in the explicitly configured output CNP.

Table 96: Components of the Two Lossless FCoE Priorities Configuration Topology *(continued)*

Component	Settings
DCBX application mapping	Interface xe-0/0/20 does not need an application map because DCBX exchanges application protocol TLVs only on the default FCoE priority (priority 3). Interface xe-0/0/21 requires an application map that enables DCBX application protocol TLV exchange on priority 5 (code point 101) for FCoE traffic. Interface xe-0/0/22 requires an application map that enables DCBX application protocol TLV exchange both on priority 3 (code point 011) and on priority 5 (code point 101) for FCoE traffic. • Interface xe-0/0/21 DCBX application mapping: Application name—**fcoe_p5_app** Application ether-type—**0x8906** Application map name—**fcoe_p5_app_map** Application map code points—**101** • Interface xe-0/0/22 DCBX application mapping: Application name—**fcoe_all_app** Application ether-type—**0x8906** Application map name—**fcoe_all_app_map** Application map code points—**011** and **101** **NOTE:** LLDP and DCBX must be enabled on the interface. By default, LLDP and DCBX are enabled on all Ethernet interfaces.

NOTE: This example does not include scheduling (bandwidth allocation) configuration or the FIP snooping configuration. This examples focuses only on the lossless FCoE priority configuration.

QFX10000 switches do not support FIP snooping. For this reason, QFX10000 switches cannot be used as FCoE access transit switches. QFX10000 switches can be used as intermediate or aggregation transit switches in the FCoE path, between an FCoE access transit switch that performs FIP snooping and an FCF.

Configuration

CLI Quick Configuration

To quickly configure two lossless FCoE forwarding classes that use different priorities on an FCoE transit switch, 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.

set class-of-service forwarding-classes class fcoe1 queue-num 5 no-loss
set class-of-service classifiers ieee-802.1 fcoe_p3 forwarding-class fcoe loss-priority low code-points 011
set class-of-service classifiers ieee-802.1 fcoe_p5 forwarding-class fcoe1 loss-priority low code-points 101
set class-of-service classifiers ieee-802.1 fcoe_p3_p5 forwarding-class fcoe loss-priority low code-points 011
set class-of-service classifiers ieee-802.1 fcoe_p3_p5 forwarding-class fcoe1 loss-priority low code-points 101
set class-of-service interfaces xe-0/0/20 unit 0 classifiers ieee-802.1 fcoe_p3
set class-of-service interfaces xe-0/0/21 unit 0 classifiers ieee-802.1 fcoe_p5
set class-of-service interfaces xe-0/0/22 unit 0 classifiers ieee-802.1 fcoe_p3_p5
set class-of-service congestion-notification-profile fcoe_p3_cnp input ieee-802.1 code-point 011 pfc mru 2240
set class-of-service congestion-notification-profile fcoe_p3_cnp input cable-length 100
set class-of-service congestion-notification-profile fcoe_p5_cnp input ieee-802.1 code-point 101 pfc mru 2240
set class-of-service congestion-notification-profile fcoe_p5_cnp input cable-length 150
set class-of-service congestion-notification-profile fcoe_p5_cnp output ieee-802.1 code-point 101 pfc flow-control-queue 5
set class-of-service congestion-notification-profile fcoe_p3_p5_cnp input ieee-802.1 code-point 011 pfc mru 2240
set class-of-service congestion-notification-profile fcoe_p3_p5_cnp input ieee-802.1 code-point 101 pfc mru 2240
set class-of-service congestion-notification-profile fcoe_p3_p5_cnp input cable-length 100
set class-of-service congestion-notification-profile fcoe_p3_p5_cnp output ieee-802.1 code-point 011 pfc flow-control-queue 3
set class-of-service congestion-notification-profile fcoe_p3_p5_cnp output ieee-802.1 code-point 101 pfc flow-control-queue 5
set class-of-service interfaces xe-0/0/20 congestion-notification-profile fcoe_p3_cnp
set class-of-service interfaces xe-0/0/21 congestion-notification-profile fcoe_p5_cnp
set class-of-service interfaces xe-0/0/22 congestion-notification-profile fcoe_p3_p5_cnp
set applications application fcoe_p5_app ether-type 0x8906
set applications application fcoe_all_app ether-type 0x8906
set policy-options application-maps fcoe_p5_app_map application fcoe_p5_app code-points 101

set policy-options application-maps fcoe_all_app_map application fcoe_all_app code-points [011 101]

set protocols dcbx interface xe-0/0/21 application-map fcoe_p5_app_map

set protocols dcbx interface xe-0/0/22 application-map fcoe_all_app_map

Step-by-Step Procedure

To configure two lossless forwarding classes for FCoE traffic on different interfaces, classify FCoE traffic into the forwarding classes, configure congestion notification profiles to enable PFC on the FCoE priorities and output queues, and configure DCBX application protocol TLV exchange for traffic on both FCoE priorities:

1. Configure lossless forwarding class **fcoe1** and map it to output queue **5** for FCoE traffic that uses IEEE 802.1p priority 5:

   ```
   [edit class-of-service]
   user@switch# set forwarding-classes class fcoe1 queue-num 5 no-loss
   ```

   > **NOTE:** This examples uses the default **fcoe** forwarding class as the other lossless FCoE forwarding class.

2. Configure the ingress classifier (**fcoe_p3**) for interface **xe-0/0/20**. The classifier maps the FCoE priority (IEEE 802.1p code point **011**) to lossless FCoE forwarding class **fcoe**:

   ```
   [edit class-of-service classifiers]
   user@switch# set ieee-802.1 fcoe_p3 forwarding-class fcoe loss-priority low code-points 011
   ```

3. Configure the ingress classifier (**fcoe_p5**) for interface **xe-0/0/21**. The classifier maps the FCoE priority (IEEE 802.1p code point **101**) to lossless FCoE forwarding class **fcoe1**:

   ```
   [edit class-of-service classifiers]
   user@switch# set ieee-802.1 fcoe_p5 forwarding-class fcoe1 loss-priority low code-points 101
   ```

4. Configure the ingress classifier (**fcoe_p3_p5**) for interface **xe-0/0/22**. The classifier maps the two FCoE priorities (IEEE 802.1p code points **011** and **101**) to the two lossless FCoE forwarding classes **fcoe** and **fcoe1**, respectively:

   ```
   [edit class-of-service classifiers]
   user@switch# set ieee-802.1 fcoe_p3_p5 forwarding-class fcoe loss-priority low code-points 011
   user@switch# set ieee-802.1 fcoe_p3_p5 forwarding-class fcoe1 loss-priority low code-points 101
   ```

5. Apply each classifier to the appropriate interface:

   `[edit class-of-service]`

   `user@switch#` **set interfaces xe-0/0/20 unit 0 classifiers ieee-802.1 fcoe_p3**

   `user@switch#` **set interfaces xe-0/0/21 unit 0 classifiers ieee-802.1 fcoe_p5**

   `user@switch#` **set interfaces xe-0/0/22 unit 0 classifiers ieee-802.1 fcoe_p3_p5**

6. Configure the CNP input stanza for interface xe-0/0/20 to enable PFC on the FCoE priority (IEEE 802.1p code point 011), set the MRU value (2240 bytes), and set the cable length value (100 meters). No output stanza is needed because queue 3 is paused by default on priority 3, and we are not explicitly configuring output queue flow control for any other queues.

   `[edit class-of-service]`

   `user@switch#` **set congestion-notification-profile fcoe_p3_cnp input ieee-802.1 code-point 011 pfc mru 2240**

   `user@switch#` **set congestion-notification-profile fcoe_p3_cnp input cable-length 100**

7. Configure the CNP for interface xe-0/0/21. The input stanza enables PFC on the FCoE priority (IEEE 802.1p code point 101), sets the MRU value (2240 bytes), and sets the cable length value (150 meters). The output stanza configures flow control on output queue 5 on the FCoE priority:

   `[edit class-of-service]`

   `user@switch#` **set congestion-notification-profile fcoe_p5_cnp input ieee-802.1 code-point 101 pfc mru 2240**

   `user@switch#` **set congestion-notification-profile fcoe_p5_cnp input cable-length 150**

   `user@switch#` **set congestion-notification-profile fcoe_p5_cnp output ieee-802.1 code-point 101 pfc flow-control-queue 5**

8. Configure the CNP for interface xe-0/0/22. The input stanza enables PFC on the FCoE priorities (IEEE 802.1p code points 011 and 101), sets the MRU value (2240 bytes), and sets the cable length value (100 meters). The output stanza configures flow control on output queues 3 and 5 on the FCoE priorities:

   `[edit class-of-service]`

   `user@switch#` **set congestion-notification-profile fcoe_p3_p5_cnp input ieee-802.1 code-point 011 pfc mru 2240**

   `user@switch#` **set congestion-notification-profile fcoe_p3_p5_cnp input ieee-802.1 code-point 101 pfc mru 2240**

   `user@switch#` **set congestion-notification-profile fcoe_p3_p5_cnp input cable-length 100**

   `user@switch#` **set congestion-notification-profile fcoe_p3_p5_cnp output ieee-802.1 code-point 011 pfc flow-control-queue 3** user@switch# **set congestion-notification-profile fcoe_p3_p5_cnp output ieee-802.1 code-point 101 pfc flow-control-queue 5**

9. Apply each CNP to the appropriate interface:

   [edit class-of-service]
   user@switch# **set interfaces xe-0/0/20 congestion-notification-profile fcoe_p3_cnp**
   user@switch# **set interfaces xe-0/0/21 congestion-notification-profile fcoe_p5_cnp**
   user@switch# **set interfaces xe-0/0/22 congestion-notification-profile fcoe_p3_p5_cnp**

10. Configure the DCBX FCoE application and application map to apply to interface xe-0/0/21. Interface xe-0/0/21 uses priority 5 (IEEE 802.1p code point 101) for FCoE traffic, which requires DCBX to exchange FCoE application protocol TLVs on priority 5 on interface xe-0/0/21. Configure an application named **fcoe_p5_app** for FCoE traffic (EtherType **0x8906**) and configure an application map named **fcoe_p5_app_map** to map the application to code point 101:

    [edit]
    user@switch# **set applications application fcoe_p5_app ether-type 0x8906**
    user@switch# **set policy-options application-maps fcoe_p5_app_map application fcoe_p5_app code-points 101**

    NOTE: Interface xe-0/0/20 uses the default FCoE configuration (priority 3). DCBX exchanges protocol TLVs for the FCoE application by default, so you do not need to configure DCBX explicitly on interface xe-0/0/20.

11. Configure the DCBX FCoE application and application map to apply to interface xe-0/0/22. Interface xe-0/0/22 uses both priority 3 (IEEE 802.1p code point 011) and priority 5 for FCoE traffic, which requires DCBX to exchange FCoE application protocol TLVs on both priority 3 and priority 5. Configure an application named **fcoe_all_app** for FCoE traffic (EtherType **0x8906**) and configure an application map named **fcoe_all_app_map** to map the application to code points 011 and 101:

    [edit]
    user@switch# **set applications application fcoe_all_app ether-type 0x8906**
    user@switch# **set policy-options application-maps fcoe_all_app_map application fcoe_all_app code-points [011 101]**

12. Apply the application maps to the interfaces xe-0/0/21 and xe-0/0/22 so that DCBX exchanges FCoE application TLVs on the correct code points on each interface:

```
[edit]
user@switch# set protocols dcbx interface xe-0/0/21 application-map fcoe_p5_app_map
user@switch# set protocols dcbx interface xe-0/0/22 application-map fcoe_all_app_map
```

Verification

IN THIS SECTION

To verify the configuration and proper operation of the lossless forwarding classes and IEEE 802.1p priorities, perform these tasks:

Verifying the Forwarding Class Configuration

Purpose

Verify that the lossless forwarding class **fcoe1** has been created.

Action

Show the forwarding class configuration by using the operational command **show class-of-service forwarding class**:

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

```
Forwarding class                       ID       Queue  Policing priority  No-Loss
  best-effort                          0         0         normal         Disabled

  fcoe                                 1         3         normal         Enabled
  no-loss                              2         4         normal         Enabled
  network-control                      3         7         normal         Disabled
```

```
fcoe1                                    4          5          normal       Enabled
mcast                                    8          8          normal       Disabled
```

Meaning

The **show class-of-service forwarding-class** command shows all of the forwarding classes. The command output shows that the **fcoe1** forwarding class is configured on output queue **5** with the no-loss packet drop attribute enabled.

Because we did not explicitly configure the default forwarding classes, they remain in their default state, including the lossless configuration of the **fcoe** and **no-loss** default forwarding classes.

Verifying the Behavior Aggregate Classifier Configuration

Purpose

Verify that the three classifiers map the forwarding classes to the correct IEEE 802.1p code points (priorities) and packet loss priorities.

Action

List the classifiers configured to support lossless FCoE transport using the operational mode command **show class-of-service classifier**:

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

```
Classifier: fcoe_p3, Code point type: ieee-802.1, Index: 13913
  Code point          Forwarding class                    Loss priority
  011                 fcoe                                low

Classifier: fcoe_p5, Code point type: ieee-802.1, Index: 63065
  Code point          Forwarding class                    Loss priority
  101                 fcoe1                               low

Classifier: fcoe_p3_p5, Code point type: ieee-802.1, Index: 10964
  Code point          Forwarding class                    Loss priority
  011                 fcoe                                low
  101                 fcoe1                               low
```

Meaning

The **show class-of-service classifier** command shows the IEEE 802.1p code points and the loss priorities that are mapped to the forwarding classes in each classifier. The command output shows that there are three classifiers, **fcoe_p3**, **fcoe_p5**, and **fcoe_p3_p5**.

Classifier **fcoe_p3** maps code point **011** (priority 3) to default lossless forwarding class **fcoe** and a packet loss priority of **low**, and all other priorities to the **best-effort** forwarding class with a packet loss priority of **high**.

Classifier **fcoe_p5** maps code point **101** (priority 5) to explicitly configured lossless forwarding class **fcoe1** and a packet loss priority of **low**, and all other priorities to the **best-effort** forwarding class with a packet loss priority of **high**.

Classifier **fcoe_p3_p5** maps code point **011** to default lossless forwarding class **fcoe** and a packet loss priority of **low**, and maps code point **101** to explicitly configured lossless forwarding class **fcoe1** and a packet loss priority of **low**. The classifier maps all other priorities to the **best-effort** forwarding class with a packet loss priority of **high**.

Verifying the PFC Flow Control Configuration (CNP)

Purpose

Verify that PFC is enabled on the correct input priorities and that flow control is configured on the correct output queues and priorities in each CNP.

Action

List the congestion notification profiles using the operational mode command **show class-of-service congestion-notification**:

`user@switch>` **show class-of-service congestion-notification**

```
Name: fcoe_p3_cnp, Index: 12037
Type: Input
Cable Length: 100 m
  Priority      PFC          MRU
  000           Disabled
  001           Disabled
  010           Disabled
  011           Enabled      2240
  100           Disabled
  101           Disabled
  110           Disabled
  111           Disabled
Type: Output
  Priority      Flow-Control-Queues
  000
                0
  001
                1
  010
                2
```

```
  011
                3
  100
                4
  101
                5
  110
                6
  111
                7

Name: fcoe_p3_p5_cnp, Index: 46484
Type: Input
Cable Length: 100 m
  Priority     PFC          MRU
  000          Disabled
  001          Disabled
  010          Disabled
  011          Enabled      2240
  100          Disabled
  101          Enabled      2240
  110          Disabled
  111          Disabled
Type: Output
  Priority     Flow-Control-Queues
  011
                3
  101
                5

Name: fcoe_p5_cnp, Index: 12133
Type: Input
Cable Length: 150 m
  Priority     PFC          MRU
  000          Disabled
  001          Disabled
  010          Disabled
  011          Disabled
  100          Disabled
  101          Enabled      2240
  110          Disabled
  111          Disabled
Type: Output
  Priority     Flow-Control-Queues
```

```
101
                  5
```

Meaning

The **show class-of-service congestion-notification** command shows the input and output stanzas of the three CNPs. For CNP **fcoe_p3_cnp**, the input stanza shows that PFC is enabled on IEEE 802.1p code point **011** (priority 3), the MRU is **2240** bytes, and the cable length is **100** meters. The CNP output stanza shows the default mapping of priorities to output queues.

> **NOTE:** By default, only queues 3 and 4 are enabled to respond to pause messages from the connected peer. For queue 3 to respond to pause messages, priority 3 (code point 011) must be enabled for PFC in the input stanza. For queue 4 to respond to pause messages, priority 4 (code point 100) must be enabled for PFC in the input stanza. In this example, only queue 3 responds to pause messages from the connected peer on interfaces that use CNP **fcoe_p3_cnp**, because the input stanza enables PFC priority 3 only.

For CNP **fcoe_p3_p5_cnp**, the input stanza shows that PFC is enabled on code points **011** and **101**, the MRU is **2240** bytes on both priorities, and the cable length is **100** meters. The CNP output stanza shows that output flow control is configured on queues **3** and **5** for code points **011** and **101**, respectively.

For CNP **fcoe_p5_cnp**, the input stanza shows that PFC is enabled on code point **101** (priority 5), the MRU is **2240** bytes, and the cable length is **150** meters. The CNP output stanza shows that output flow control is configured on queue **5** for code point **101** (priority 5).

Verifying the Interface Configuration

Purpose

Verify that the correct classifiers and congestion notification profiles are configured on the correct interfaces.

Action

List the ingress interfaces using the operational mode commands **show configuration class-of-service interfaces xe-0/0/20**, **show configuration class-of-service interfaces xe-0/0/21**, and **show configuration class-of-service interfaces xe-0/0/22**:

`user@switch>` **show configuration class-of-service interfaces xe-0/0/20**

```
ccongestion-notification-profile fcoe_p3_cnp;
unit 0 {
    classifiers {
        ieee-802.1 fcoe_p3;
```

```
    }
}
```

`user@switch>` **show configuration class-of-service interfaces xe-0/0/21**

```
congestion-notification-profile fcoe_p5_cnp;
unit 0 {
    classifiers {
        ieee-802.1 fcoe_p5;
    }
}
```

`user@switch>` **show configuration class-of-service interfaces xe-0/0/22**

```
congestion-notification-profile fcoe_p3_p5_cnp;
unit 0 {
    classifiers {
        ieee-802.1 fcoe_p3_p5;
    }
}
```

Meaning

The **show configuration class-of-service interfaces xe-0/0/20** command shows that the congestion notification profile **fcoe_p3_cnp** is configured on the interface, and that the IEEE 802.1p classifier associated with the interface is **fcoe_p3**.

The **show configuration class-of-service interfaces xe-0/0/21** command shows that the congestion notification profile **fcoe_p5_cnp** is configured on the interface, and that the IEEE 802.1p classifier associated with the interface is **fcoe_p5**.

The **show configuration class-of-service interfaces xe-0/0/22** command shows that the congestion notification profile **fcoe_p3_p5_cnp** is configured on the interface, and that the IEEE 802.1p classifier associated with the interface is **fcoe_p3_p5**.

Verifying the DCBX Application Configuration

Purpose

Verify that the two DCBX applications for FCoE are configured.

Action

List the DCBX applications by using the configuration mode command **show applications**:

`user@switch#` **show applications**

```
application fcoe_all_app {
    ether-type 0x8906;

application fcoe_p5_app {
    ether-type 0x8906;
```

Meaning

The **show applications** configuration mode command shows all of the configured applications. The output shows that the application **fcoe_all_app** is configured with an EtherType of **0x8906** (the correct EtherType for FCoE traffic) and that the application **fcoe_p5_app** is also configured with an EtherType of **0x8906**.

Verifying the DCBX Application Map Configuration

Purpose

Verify that the application maps are configured.

Action

List the application maps by using the configuration mode command **show policy-options application-maps**:

`user@switch#` **show policy-options application-maps**

```
fcoe_all_app_map {
    application fcoe_all_app code-points [011 101];
}
fcoe_p5_app_map {
    application fcoe_p5_app code-points 101;
}
```

Meaning

The **show policy-options application-maps** configuration mode command lists all of the configured application maps and the applications that belong to each application map. The output shows that there are two application maps.

Application map **fcoe_all_app_map** consists of the application named **fcoe_all_app** mapped to IEEE 802.1p code points **011** (priority 3) and **101** (priority 5).

Application map **fcoe_p5_app_map** consists of the application named **fcoe_p5_app** mapped to IEEE 802.1p code point **101** (priority 5).

Verifying the DCBX Application Protocol Exchange Interface Configuration

Purpose

Verify that the application maps are applied to the correct interfaces.

Action

List the application maps on each interface using the configuration mode command **show protocols dcbx**:

`user@switch#` **show protocols dcbx**

```
interface xe-0/0/21.0 {
    application-map fcoe_p5_app_map;
}
interface xe-0/0/22.0 {
    application-map fcoe_all_app_map;
}
```

Meaning

The **show protocols dcbx** configuration mode command lists the application map association with interfaces. The output shows that interface **xe-0/0/21.0** uses application map **fcoe_p5_app_map** and interface **xe-0/0/22.0** uses application map **fcoe_all_app_map**.

> **NOTE:** Because interface xe-0/0/20 uses the default lossless FCoE configuration, you do not configure application mapping to interface xe-0/0/20. The default configuration automatically exchanges application protocol TLVs for the default FCoE configuration on priority 3 (IEEE 802.1p code point 011).

RELATED DOCUMENTATION

Example: Configuring Two or More Lossless FCoE Priorities on the Same FCoE Transit Switch Interface | 576

Example: Configuring Lossless FCoE Traffic When the Converged Ethernet Network Does Not Use IEEE 802.1p Priority 3 for FCoE Traffic (FCoE Transit Switch) | 565

Example: Configuring Lossless IEEE 802.1p Priorities on Ethernet Interfaces for Multiple Applications (FCoE and iSCSI) | 605

Example: Configuring DCBX Application Protocol TLV Exchange | 476

Configuring CoS PFC (Congestion Notification Profiles) | 194

Example: Configuring Lossless IEEE 802.1p Priorities on Ethernet Interfaces for Multiple Applications (FCoE and iSCSI)

IN THIS SECTION

Although the default configuration provides two lossless forwarding classes mapped to two different IEEE 802.1p priorities (code points), you can explicitly configure up to six lossless forwarding classes and map them to different priorities. You can support up to six different types of lossless traffic, and you can support the same type of traffic on different priorities in different parts of your converged network.

This example shows you how to configure two lossless forwarding classes for FCoE traffic and one lossless forwarding class for iSCSI traffic, and map the forwarding classes to three different priorities. (The converged Ethernet network includes two FCoE networks, each of which uses a different priority to identify FCoE traffic, and an iSCSI network.)

Requirements

This example uses the following hardware and software components:

- One switch used as an FCoE transit switch
- Junos OS Release 12.3 or later for the QFX Series

Overview

Some converged Ethernet networks support FCoE on more than one IEEE 802.1p priority and also require supporting other lossless traffic classes. Interfaces that carry multiple lossless forwarding classes need to support lossless behavior for the priorities mapped to those forwarding classes. To support the two FCoE forwarding classes and the iSCSI forwarding class used in this example, you need to configure:

- At least one lossless forwarding class for FCoE traffic (this example uses the default **fcoe** forwarding class as one of the two lossless FCoE forwarding classes, so we need to explicitly configure only one FCoE forwarding class)
- A lossless forwarding class for iSCSI traffic
- Behavior aggregate (BA) classifiers to map the lossless forwarding classes to the appropriate IEEE 802.1p code points (priorities) on each interface
- Congestion notification profiles (CNPs) for each interface to enable PFC on the FCoE and iSCSI code points at the interface ingress, and to configure PFC flow control on the interface egress so that the interface can respond to PFC messages received from the connected peer

  > **NOTE:** Configuring or changing PFC on an interface blocks the entire port until the PFC change is completed. After a PFC change is completed, the port is unblocked and traffic resumes. Blocking the port stops ingress and egress traffic, and causes packet loss on all queues on the port until the port is unblocked.

- DCBX applications and an application map to support DCBX application TLV exchange for the FCoE and iSCSI traffic on the configured lossless priorities. By default, DCBX is enabled on all Ethernet interfaces for FCoE, but only on priority 3 (IEEE 802.1p code point 011). To support DCBX application TLV exchange when you are not using the default configuration, you must configure all of the applications and map them to interfaces and priorities.

The priorities specified in the BA classifiers, CNPs, and DCBX application map must match, or the configuration does not work. You must specify the same lossless FCoE forwarding class in each configuration and use the same IEEE 802.1p code point (priority) so that the FCoE traffic is properly classified into flows and so that those flows receive lossless treatment.

Topology

This example shows how to configure two lossless FCoE traffic classes and one lossless iSCSI traffic class, map them to three different priorities, and configure flow control to ensure lossless behavior for those priorities on the interfaces. This example uses four Ethernet interfaces, xe-0/0/31, xe-0/0/32, xe-0/0/33, and xe-0/0/34:

- Interface xe-0/0/31 handles FCoE traffic on priority 3 (IEEE 802.1p code point 011) and iSCSI traffic on priority 4 (code point 100).
- Interface xe-0/0/32 handles FCoE traffic on priority 5 (code point 101) and iSCSI traffic on priority 4.
- Interface xe-0/0/33 handles FCoE traffic on priority 3 and priority 5.
- Interface xe-0/0/34 handles iSCSI traffic on priority 4.

Figure 27 on page 607 shows the topology for this example, and Table 97 on page 607 shows the configuration components for this example.

Figure 27: Topology of the Lossless FCoE and iSCSI Priorities Example

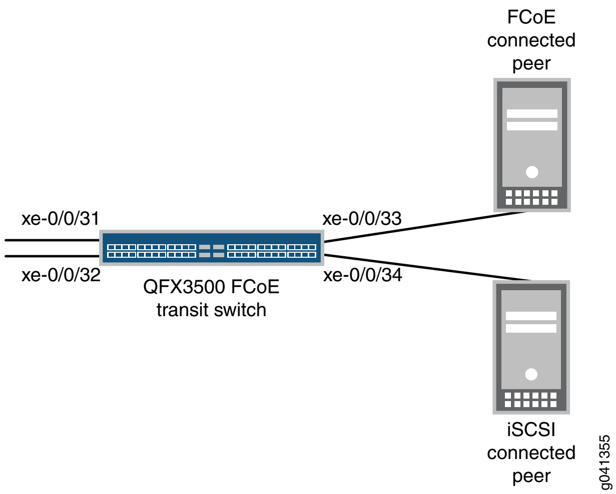


Table 97: Components of the Lossless FCoE and iSCSI Priorities Configuration Topology

Component	Settings
Hardware	One switch

Table 97: Components of the Lossless FCoE and iSCSI Priorities Configuration Topology *(continued)*

Component	Settings
Forwarding classes	This example uses one explicitly configured lossless FCoE forwarding class, the default lossless FCoE forwarding class, and one explicitly configured iSCSI forwarding class. • iSCSI forwarding class: Name—**iscsi** Queue mapping—queue **4** Packet drop attribute—**no-loss** • FCoE forwarding class (explicitly configured): Name—**fcoe1** Queue mapping—queue **5** Packet drop attribute—**no-loss** **NOTE:** A lossless forwarding class can be mapped to any output queue. However, because the **fcoe1** forwarding class uses priority 5 in this example, matching that traffic to a forwarding class that uses queue 5 creates a configuration that is logical and easy to map because the priority and the queue are identified by the same number. • FCoE forwarding class (default) Name—**fcoe** The default **fcoe** forwarding class is mapped to priority 3 (IEEE 802.1p code point 011) and to output queue 3 with a packet drop attribute of **no-loss**.

Table 97: Components of the Lossless FCoE and iSCSI Priorities Configuration Topology *(continued)*

Component	Settings
BA classifiers	Each interface requires a different classifier because each interface handles a different subset of FCoE traffic. • Interface xe-0/0/31 classifier: Name—**fcoe_p3_iscsi** FCoE priority mapping—Forwarding class **fcoe** mapped to code point **011** (IEEE 802.1p priority 3) and a packet loss priority of **low**. iSCSI priority mapping—Forwarding class **iscsi** mapped to code point **100** (priority 4) and a packet loss priority of **low**. All other priority mapping—All other forwarding classes are mapped to the **best-effort** forwarding class with packet loss priorities of **high**. • Interface xe-0/0/32 classifier: Name—**fcoe_p5_iscsi** FCoE priority mapping—Forwarding class **fcoe1** mapped to code point **101** (IEEE 802.1p priority 5) and a packet loss priority of **low**. iSCSI priority mapping—Forwarding class **iscsi** mapped to code point **100** (priority 4) and a packet loss priority of **low**. All other priority mapping—All other forwarding classes are mapped to the **best-effort** forwarding class with packet loss priorities of **high**. • Interface xe-0/0/33 classifier: Name—**fcoe_p3_p5** FCoE priority mapping—Forwarding class **fcoe1** mapped to code point **101** (priority 5) and a packet loss priority of **low**, and forwarding class **fcoe** mapped to code point **011** and a packet loss priority of **low**. All other priority mapping—All other forwarding classes are mapped to the **best-effort** forwarding class with packet loss priorities of **high**. • Interface xe-0/0/34 classifier: Name—**iscsi_classifier** iSCSI priority mapping—Forwarding class **iscsi** mapped to code point **100** (priority 4) and a packet loss priority of **low**. All other priority mapping—All other forwarding classes are mapped to the **best-effort** forwarding class with packet loss priorities of **high**.

Table 97: Components of the Lossless FCoE and iSCSI Priorities Configuration Topology *(continued)*

Component	Settings
PFC configuration (CNPs)	

Table 97: Components of the Lossless FCoE and iSCSI Priorities Configuration Topology *(continued)*

Component	Settings
	Each interface requires a different CNP because each interface handles a different subset of FCoE and iSCSI traffic, and must pause that traffic on different priorities. • Interface xe-0/0/31 CNP: CNP name—**fcoe_p3_cnp** Input CNP code points—**011** and **100** MRU—2240 bytes for code point **011**, default value (2500 bytes) for code point **100** Cable length—100 meters **NOTE:** On interface xe-0/0/31, the FCoE forwarding class is mapped to queue 3 and priority 3 (code point 011), and the iSCSI forwarding class is mapped to queue 4 and priority 4 (code point 100). Therefore, interface xe-0/0/31 does not require an output CNP configuration because queue 3 and queue 4 are enabled for PFC flow control by default on code points 011 and 100, respectively. • Interface xe-0/0/32 CNP: CNP name—**fcoe_p5_cnp** Input CNP code points—**100** and **101** MRU—Default value (2500 bytes) for code point **100**, **2240** bytes for code point **101** Cable length—**150** meters Output CNP code points—**100** and **101** Output CNP flow control queues—**4** and **5** • Interface xe-0/0/33 CNP: CNP name—**fcoe_p3_p5_cnp** Input CNP code points—**011** and **101** MRU—**2240** bytes (both priorities) Cable length—**100** meters Output CNP code points—**011** and **101** Output CNP flow control queues—**3** and **5** • Interface xe-0/0/34 CNP: CNP name—**iscsi_cnp** Input CNP code point—**100** MRU—**2500** bytes (default value) Cable length—**100** meters **NOTE:** On interface xe-0/0/34, the iSCSI forwarding class is mapped to queue 4 and priority 4 (code point 100). Interface xe-0/0/34 does not require an output CNP configuration because queue 4 is enabled for PFC flow control by default on code point 100.

Table 97: Components of the Lossless FCoE and iSCSI Priorities Configuration Topology *(continued)*

Component	Settings
	NOTE: When you apply a CNP with an explicit output queue flow control configuration to an interface, the explicit CNP overwrites the default output CNP. The output queues that are enabled for PFC pause in the default configuration (queues 3 and 4) are not enabled for pause unless they are included in the explicitly configured output CNP.
DCBX application mapping	This example requires configuring applications for FCoE and iSCSI, including them in the same application map, and applying the application map to all four interfaces. Application map name—**dcbx_iscsi_fcoe_app_map** • FCoE application name—**fcoe_app** Application ether-type—**0x8906** Application map code points—**011** and **101** • iSCSI application name—**iscsi_app** Application protocol type—**tcp** Application destination port—**3260** Application map code point—**100** **NOTE:** LLDP and DCBX must be enabled on the interface. By default, LLDP and DCBX are enabled on all Ethernet interfaces.

NOTE: This example does not include scheduling (bandwidth allocation) configuration or the FIP snooping configuration. This examples focuses only on the lossless FCoE priority configuration.

QFX10000 switches do not support FIP snooping. For this reason, QFX10000 switches cannot be used as FCoE access transit switches. QFX10000 switches can be used as intermediate or aggregation transit switches in the FCoE path, between an FCoE access transit switch that performs FIP snooping and an FCF.

Configuration

CLI Quick Configuration

To quickly configure two lossless FCoE forwarding classes and one lossless iSCSI forwarding class and map them to different priorities, 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.

set class-of-service forwarding-classes class iscsi queue-num 4 no-loss
set class-of-service forwarding-classes class fcoe1 queue-num 5 no-loss
set class-of-service classifiers ieee-802.1 fcoe_p3_iscsi forwarding-class fcoe loss-priority low code-points 011
set class-of-service classifiers ieee-802.1 fcoe_p3_iscsi forwarding-class iscsi loss-priority low code-points 100
set class-of-service classifiers ieee-802.1 fcoe_p3_iscsi forwarding-class best-effort loss-priority high code-points 000
set class-of-service classifiers ieee-802.1 fcoe_p3_iscsi forwarding-class best-effort loss-priority high code-points 001
set class-of-service classifiers ieee-802.1 fcoe_p3_iscsi forwarding-class best-effort loss-priority high code-points 010
set class-of-service classifiers ieee-802.1 fcoe_p3_iscsi forwarding-class best-effort loss-priority high code-points 101
set class-of-service classifiers ieee-802.1 fcoe_p3_iscsi forwarding-class best-effort loss-priority high code-points 110
set class-of-service classifiers ieee-802.1 fcoe_p3_iscsi forwarding-class best-effort loss-priority high code-points 111
set class-of-service classifiers ieee-802.1 fcoe_p5_iscsi forwarding-class iscsi loss-priority low code-points 100
set class-of-service classifiers ieee-802.1 fcoe_p5_iscsi forwarding-class fcoe1 loss-priority low code-points 101
set class-of-service classifiers ieee-802.1 fcoe_p5_iscsi forwarding-class best-effort loss-priority high code-points 000
set class-of-service classifiers ieee-802.1 fcoe_p5_iscsi forwarding-class best-effort loss-priority high code-points 001
set class-of-service classifiers ieee-802.1 fcoe_p5_iscsi forwarding-class best-effort loss-priority high code-points 010
set class-of-service classifiers ieee-802.1 fcoe_p5_iscsi forwarding-class best-effort loss-priority high code-points 011
set class-of-service classifiers ieee-802.1 fcoe_p5_iscsi forwarding-class best-effort loss-priority high code-points 110
set class-of-service classifiers ieee-802.1 fcoe_p5_iscsi forwarding-class best-effort loss-priority high code-points 111

set class-of-service classifiers ieee-802.1 fcoe_p3_p5 forwarding-class fcoe loss-priority low code-points 011
set class-of-service classifiers ieee-802.1 fcoe_p3_p5 forwarding-class fcoe1 loss-priority low code-points 101
set class-of-service classifiers ieee-802.1 fcoe_p3_p5 forwarding-class best-effort loss-priority high code-points 000
set class-of-service classifiers ieee-802.1 fcoe_p3_p5 forwarding-class best-effort loss-priority high code-points 001
set class-of-service classifiers ieee-802.1 fcoe_p3_p5 forwarding-class best-effort loss-priority high code-points 010
set class-of-service classifiers ieee-802.1 fcoe_p3_p5 forwarding-class best-effort loss-priority high code-points 100
set class-of-service classifiers ieee-802.1 fcoe_p3_p5 forwarding-class best-effort loss-priority high code-points 110
set class-of-service classifiers ieee-802.1 fcoe_p3_p5 forwarding-class best-effort loss-priority high code-points 111
set class-of-service classifiers ieee-802.1 iscsi_classifier forwarding-class iscsi loss-priority low code-points 100
set class-of-service classifiers ieee-802.1 iscsi_classifier forwarding-class best-effort loss-priority high code-points 000
set class-of-service classifiers ieee-802.1 iscsi_classifier forwarding-class best-effort loss-priority high code-points 001
set class-of-service classifiers ieee-802.1 iscsi_classifier forwarding-class best-effort loss-priority high code-points 010
set class-of-service classifiers ieee-802.1 iscsi_classifier forwarding-class best-effort loss-priority high code-points 011
set class-of-service classifiers ieee-802.1 iscsi_classifier forwarding-class best-effort loss-priority high code-points 101
set class-of-service classifiers ieee-802.1 iscsi_classifier forwarding-class best-effort loss-priority high code-points 110
set class-of-service classifiers ieee-802.1 iscsi_classifier forwarding-class best-effort loss-priority high code-points 111
set class-of-service interfaces xe-0/0/31 unit 0 classifiers ieee-802.1 fcoe_p3_iscsi
set class-of-service interfaces xe-0/0/32 unit 0 classifiers ieee-802.1 fcoe_p5_iscsi
set class-of-service interfaces xe-0/0/33 unit 0 classifiers ieee-802.1 fcoe_p3_p5set class-of-service interfaces xe-0/0/34 unit 0 classifiers ieee-802.1 iscsi_classifier

```
set class-of-service congestion-notification-profile fcoe_p3_cnp input ieee-802.1 code-point 011 pfc mru 2240
set class-of-service congestion-notification-profile fcoe_p3_cnp input ieee-802.1 code-point 100 pfc
set class-of-service congestion-notification-profile fcoe_p3_cnp input cable-length 100
set class-of-service congestion-notification-profile fcoe_p5_cnp input ieee-802.1 code-point 100 pfc
set class-of-service congestion-notification-profile fcoe_p5_cnp input ieee-802.1 code-point 101 pfc mru 2240
set class-of-service congestion-notification-profile fcoe_p5_cnp input cable-length 150
set class-of-service congestion-notification-profile fcoe_p5_cnp output ieee-802.1 code-point 100 pfc flow-control-queue 4
set class-of-service congestion-notification-profile fcoe_p5_cnp output ieee-802.1 code-point 101 pfc flow-control-queue 5
set class-of-service congestion-notification-profile fcoe_p3_p5_cnp input ieee-802.1 code-point 011 pfc mru 2240
set class-of-service congestion-notification-profile fcoe_p3_p5_cnp input ieee-802.1 code-point 101 pfc mru 2240
set class-of-service congestion-notification-profile fcoe_p3_p5_cnp input cable-length 100
set class-of-service congestion-notification-profile fcoe_p3_p5_cnp output ieee-802.1 code-point 011 pfc flow-control-queue 3
set class-of-service congestion-notification-profile fcoe_p3_p5_cnp output ieee-802.1 code-point 101 pfc flow-control-queue 5
set class-of-service congestion-notification-profile iscsi_cnp input ieee-802.1 code-point 100 pfc
set class-of-service congestion-notification-profile iscsi_cnp input cable-length 100
set class-of-service interfaces xe-0/0/31 congestion-notification-profile fcoe_p3_cnp
set class-of-service interfaces xe-0/0/32 congestion-notification-profile fcoe_p5_cnp
set class-of-service interfaces xe-0/0/33 congestion-notification-profile fcoe_p3_p5_cnp
set class-of-service interfaces xe-0/0/34 congestion-notification-profile iscsi_cnp
set applications application iscsi_app protocol tcp destination-port 3260
set applications application fcoe_app ether-type 0x8906
set policy-options application-maps dcbx_iscsi_fcoe_app_map application iscsi_app code-points 100
set policy-options application-maps dcbx_iscsi_fcoe_app_map application fcoe_app code-points [011 101]
set protocols dcbx interface xe-0/0/31 application-map dcbx_iscsi_fcoe_app_map
set protocols dcbx interface xe-0/0/32 application-map dcbx_iscsi_fcoe_app_map
set protocols dcbx interface xe-0/0/33 application-map dcbx_iscsi_fcoe_app_map
set protocols dcbx interface xe-0/0/34 application-map dcbx_iscsi_fcoe_app_map
```

Step-by-Step Procedure

To configure two lossless forwarding classes for FCoE traffic and one lossless forwarding class for iSCSI traffic, classify the traffic into the three forwarding classes, configure congestion notification profiles to enable PFC on the FCoE priorities and output queues, and configure DCBX application protocol TLV exchange for traffic on both FCoE priorities:

1. Configure lossless forwarding classes **iscsi** for iSCSI traffic and **fcoe1** for FCoE traffic (this example uses the default **fcoe** forwarding class as the other lossless FCoE forwarding class) and map them to output queues:

   ```
   [edit class-of-service]
   user@switch# set forwarding-classes class iscsi queue-num 4 no-loss
   user@switch# set forwarding-classes class fcoe1 queue-num 5 no-loss
   ```

2. Configure the ingress classifier (**fcoe_p3_iscsi**) for interface **xe-0/0/31**. The classifier maps the FCoE priority (code point **011**) to lossless FCoE forwarding class **fcoe** and the iSCSI priority (code point **100**) to lossless iSCSI forwarding class **iscsi**, and traffic of other priorities to the **best-effort** forwarding class with a packet loss priority of **high**:

   ```
   [edit class-of-service classifiers]
   user@switch# set ieee-802.1 fcoe_p3_iscsi forwarding-class fcoe loss-priority low code-points 011
   user@switch# set ieee-802.1 fcoe_p3_iscsi forwarding-class iscsi loss-priority low code-points 100
   user@switch# set ieee-802.1 fcoe_p3_iscsi forwarding-class best-effort loss-priority high code-points 000
   user@switch# set ieee-802.1 fcoe_p3_iscsi forwarding-class best-effort loss-priority high code-points 001
   user@switch# set ieee-802.1 fcoe_p3_iscsi forwarding-class best-effort loss-priority high code-points 010
   user@switch# set ieee-802.1 fcoe_p3_iscsi forwarding-class best-effort loss-priority high code-points 101
   user@switch# set ieee-802.1 fcoe_p3_iscsi forwarding-class best-effort loss-priority high code-points 110
   user@switch# set ieee-802.1 fcoe_p3_iscsi forwarding-class best-effort loss-priority high code-points 111
   ```

3. Configure the ingress classifier (**fcoe_p5_iscsi**) for interface **xe-0/0/32**. The classifier maps the FCoE priority (code point **101**) to lossless FCoE forwarding class **fcoe1** and the iSCSI priority (code point **100**) to lossless iSCSI forwarding class **iscsi**, and traffic of other priorities to the **best-effort** forwarding class with a packet loss priority of **high**:

```
[edit class-of-service classifiers]
user@switch# set ieee-802.1 fcoe_p5_iscsi forwarding-class iscsi loss-priority low code-points 100
user@switch# set ieee-802.1 fcoe_p5_iscsi forwarding-class fcoe1 loss-priority low code-points 101
user@switch# set ieee-802.1 fcoe_p5_iscsi forwarding-class best-effort loss-priority high code-points 000
user@switch# set ieee-802.1 fcoe_p5_iscsi forwarding-class best-effort loss-priority high code-points 001
user@switch# set ieee-802.1 fcoe_p5_iscsi forwarding-class best-effort loss-priority high code-points 010
user@switch# set ieee-802.1 fcoe_p5_iscsi forwarding-class best-effort loss-priority high code-points 011
user@switch# set ieee-802.1 fcoe_p5_iscsi forwarding-class best-effort loss-priority high code-points 110
user@switch# set ieee-802.1 fcoe_p5_iscsi forwarding-class best-effort loss-priority high code-points 111
```

4. Configure the ingress classifier (**fcoe_p3_p5**) for interface **xe-0/0/33**. The classifier maps the two FCoE priorities (code points **011** and **101**) to lossless FCoE forwarding classes **fcoe** and **fcoe1**, respectively, and traffic of other priorities to the **best-effort** forwarding class with a packet loss priority of **high**:

```
[edit class-of-service classifiers]
user@switch# set ieee-802.1 fcoe_p3_p5 forwarding-class fcoe loss-priority low code-points 011
user@switch# set ieee-802.1 fcoe_p3_p5 forwarding-class fcoe1 loss-priority low code-points 101
user@switch# set ieee-802.1 fcoe_p3_p5 forwarding-class best-effort loss-priority high code-points 000
user@switch# set ieee-802.1 fcoe_p3_p5 forwarding-class best-effort loss-priority high code-points 001
user@switch# set ieee-802.1 fcoe_p3_p5 forwarding-class best-effort loss-priority high code-points 010
user@switch# set ieee-802.1 fcoe_p3_p5 forwarding-class best-effort loss-priority high code-points 100
user@switch# set ieee-802.1 fcoe_p3_p5 forwarding-class best-effort loss-priority high code-points 110
user@switch# set ieee-802.1 fcoe_p3_p5 forwarding-class best-effort loss-priority high code-points 111
```

5. Configure the ingress classifier (**iscsi_classifier**) for interface **xe-0/0/34**. The classifier maps the iSCSI priority (code point **101**) to lossless iSCSI forwarding class **iscsi**, and traffic of other priorities to the **best-effort** forwarding class with a packet loss priority of **high**:

   ```
   [edit class-of-service classifiers]
   user@switch# set ieee-802.1 iscsi_classifier forwarding-class iscsi loss-priority low code-points 100
   user@switch# set ieee-802.1 iscsi_classifier forwarding-class best-effort loss-priority high code-points 000
   user@switch# set ieee-802.1 iscsi_classifier forwarding-class best-effort loss-priority high code-points 001
   user@switch# set ieee-802.1 iscsi_classifier forwarding-class best-effort loss-priority high code-points 010
   user@switch# set ieee-802.1 iscsi_classifier forwarding-class best-effort loss-priority high code-points 011
   user@switch# set ieee-802.1 iscsi_classifier forwarding-class best-effort loss-priority high code-points 101
   user@switch# set ieee-802.1 iscsi_classifier forwarding-class best-effort loss-priority high code-points 110
   user@switch# set ieee-802.1 iscsi_classifier forwarding-class best-effort loss-priority high code-points 111
   ```

6. Apply each classifier to the appropriate interface:

   ```
   [edit class-of-service]
   user@switch# set interfaces xe-0/0/31 unit 0 classifiers ieee-802.1 fcoe_p3_iscsi
   user@switch# set interfaces xe-0/0/32 unit 0 classifiers ieee-802.1 fcoe_p5_iscsi
   user@switch# set interfaces xe-0/0/33 unit 0 classifiers ieee-802.1 fcoe_p3_p5
   user@switch# set interfaces xe-0/0/34 unit 0 classifiers ieee-802.1 iscsi_classifier
   ```

7. Configure the CNP input stanza for interface xe-0/0/31 to enable PFC on the FCoE and iSCSI priorities that the interface handles (code points 011 and 100), set the MRU value for the FCoE traffic (2240 bytes), and set the cable length value (100 meters). No output stanza is needed because queues 3 and 4 are paused by default on priorities 3 and 4, respectively, and we are not explicitly configuring output queue flow control for any other queues.

   ```
   [edit class-of-service]
   user@switch# set congestion-notification-profile fcoe_p3_cnp input ieee-802.1 code-point 011 pfc mru 2240
   ```

user@switch# **set congestion-notification-profile fcoe_p3_cnp input ieee-802.1 code-point 100 pfc**

user@switch# **set congestion-notification-profile fcoe_p3_cnp input cable-length 100**

8. Configure the CNP for interface xe-0/0/32. The input stanza enables PFC on the FCoE priority (code point 101), sets the MRU value for FCoE traffic (2240 bytes), enables PFC on the iSCSI priority (code point 100), and sets the cable length value (150 meters). The output stanza configures flow control on output queue 5 on the FCoE priority and on output queue 4 on the iSCSI priority:

   [edit class-of-service]

   user@switch# **set congestion-notification-profile fcoe_p5_cnp input ieee-802.1 code-point 100 pfc**

   user@switch# **set congestion-notification-profile fcoe_p5_cnp input ieee-802.1 code-point 101 pfc mru 2240**

   user@switch# **set congestion-notification-profile fcoe_p5_cnp input cable-length 150**

   user@switch# **set congestion-notification-profile fcoe_p5_cnp output ieee-802.1 code-point 100 pfc flow-control-queue 4**

   user@switch# **set congestion-notification-profile fcoe_p5_cnp output ieee-802.1 code-point 101 pfc flow-control-queue 5**

9. Configure the CNP for interface xe-0/0/33. The input stanza enables PFC on the FCoE priorities (IEEE 802.1p code points 011 and 101), sets the MRU value (2240 bytes), and sets the cable length value (100 meters). The output stanza configures flow control on output queues 3 and 5 on the FCoE priorities:

   [edit class-of-service]

   user@switch# **set congestion-notification-profile fcoe_p3_p5_cnp input ieee-802.1 code-point 011 pfc mru 2240**

   user@switch# **set congestion-notification-profile fcoe_p3_p5_cnp input ieee-802.1 code-point 101 pfc mru 2240**

   user@switch# **set congestion-notification-profile fcoe_p3_p5_cnp input cable-length 100**

   user@switch# **set congestion-notification-profile fcoe_p3_p5_cnp output ieee-802.1 code-point 011 pfc flow-control-queue 3**

   user@switch# **set congestion-notification-profile fcoe_p3_p5_cnp output ieee-802.1 code-point 101 pfc flow-control-queue 5**

10. Configure the CNP input stanza for interface xe-0/0/34 to enable PFC on the iSCSI priority (code point 100) and set the cable length value (100 meters). No output stanza is needed because queue 4 is paused by default on priority 4, and we are not explicitly configuring output queue flow control for any other queues.

```
[edit class-of-service]
user@switch# set congestion-notification-profile iscsi_cnp input ieee-802.1 code-point 100 pfc
user@switch# set congestion-notification-profile iscsi_cnp input cable-length 100
```

11. Apply each CNP to the appropriate interface:

```
[edit class-of-service]
user@switch# set interfaces xe-0/0/31 congestion-notification-profile fcoe_p3_cnp
user@switch# set interfaces xe-0/0/32 congestion-notification-profile fcoe_p5_cnp
user@switch# set interfaces xe-0/0/33 congestion-notification-profile fcoe_p3_p5_cnp
user@switch# set interfaces xe-0/0/34 congestion-notification-profile iscsi_cnp
```

12. Configure the DCBX applications for FCoE and iSCSI to map to the interfaces so that DCBX can exchange application protocol TLVs on the IEEE 802.1p priorities used for FCoE and iSCSI traffic:

```
[edit]
user@switch# set applications application fcoe_app ether-type 0x8906
user@switch# set applications application iscsi_app protocol tcp destination-port 3260
```

13. Configure a DCBX application map to map the FCoE and iSCSI applications to the correct priorities:

```
[edit]
user@switch# set policy-options application-maps dcbx_iscsi_fcoe_app_map application fcoe_app
code-points [011 101]
user@switch# set policy-options application-maps dcbx_iscsi_fcoe_app_map application iscsi_app
code-points 100
```

14. Apply the application map to the interfaces so that DCBX exchanges FCoE application TLVs on the correct code points:

```
[edit]
user@switch# set protocols dcbx interface xe-0/0/31 application-map dcbx_iscsi_fcoe_app_map
user@switch# set protocols dcbx interface xe-0/0/32 application-map dcbx_iscsi_fcoe_app_map
user@switch# set protocols dcbx interface xe-0/0/33 application-map dcbx_iscsi_fcoe_app_map
user@switch# set protocols dcbx interface xe-0/0/34 application-map dcbx_iscsi_fcoe_app_map
```

Verification

IN THIS SECTION

To verify the configuration and proper operation of the lossless forwarding classes and IEEE 802.1p priorities, perform these tasks:

Verifying the Forwarding Class Configuration

Purpose

Verify that the lossless forwarding classes **iscsi** and **fcoe1** have been created and that the default lossless forwarding class **fcoe** is still enabled for lossless transport.

Action

Show the forwarding class configuration by using the operational command **show class-of-service forwarding class**:

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

```
Forwarding class                          ID       Queue  Policing priority  No-Loss
  best-effort                             0        0         normal         Disabled

  fcoe                                    1        3         normal         Enabled
  iscsi                                   2        4         normal         Enabled
  network-control                         3        7         normal         Disabled

  fcoe1                                   4        5         normal         Enabled
  mcast                                   8        8         normal         Disabled
```

Meaning

The **show class-of-service forwarding-class** command shows all of the forwarding classes. The command output shows that the **iscsi** and **fcoe1** forwarding classes are configured on output queues **4** and **5**, respectively, with the no-loss packet drop attribute enabled.

Because we did not explicitly configure the default **fcoe** forwarding class, it remains in its default state (lossless configuration).

Verifying the Behavior Aggregate Classifier Configuration

Purpose

Verify that the four classifiers map the forwarding classes to the correct IEEE 802.1p code points (priorities) and packet loss priorities.

Action

List the classifiers configured to support lossless FCoE transport using the operational mode command **show class-of-service classifier**:

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

```
Classifier: fcoe_p3_iscsi, Code point type: ieee-802.1, Index: 13915
  Code point         Forwarding class                    Loss priority
  011                fcoe                                low
  100                iscsi                               low

Classifier: fcoe_p5_iscsi, Code point type: ieee-802.1, Index: 62035
  Code point         Forwarding class                    Loss priority
  100                iscsi                               low
  101                fcoe1                               low

Classifier: fcoe_p3_p5, Code point type: ieee-802.1, Index: 17774
  Code point         Forwarding class                    Loss priority
  011                fcoe                                low
  101                fcoe1                               low

Classifier: iscsi_classifier, Code point type: ieee-802.1, Index: 31635
  Code point         Forwarding class                    Loss priority
  100                iscsi                               low
```

Meaning

The **show class-of-service classifier** command shows the IEEE 802.1p code points and the loss priorities that are mapped to the forwarding classes in each classifier. The command output shows that there are four classifiers, **fcoe_p3_iscsi**, **fcoe_p5_iscsi**, **fcoe_p3_p5**, and **iscsi_classifier**.

Classifier **fcoe_p3_iscsi** maps code point **011** (priority 3) to default lossless forwarding class **fcoe** and a packet loss priority of **low**, and code point **100** (priority 4) to explicitly configured lossless forwarding class **iscsi**, and all other priorities to the **best-effort** forwarding class with a packet loss priority of **high**.

Classifier **fcoe_p5_iscsi** maps code point **100** to explicitly configured forwarding class **iscsi** and a packet loss priority of **low**, and code point **101** (priority 5) to explicitly configured lossless forwarding class **fcoe1** and a packet loss priority of **low**, and all other priorities to the **best-effort** forwarding class with a packet loss priority of **high**.

Classifier **fcoe_p3_p5** maps code point **011** to default lossless forwarding class **fcoe** and a packet loss priority of **low**, and maps code point **101** to explicitly configured lossless forwarding class **fcoe1** and a packet loss priority of **low**.The classifier maps all other priorities to the **best-effort** forwarding class with a packet loss priority of **high**.

Classifier **iscsi_classifier** maps code point **100** to explicitly configured forwarding class **iscsi** and a packet loss priority of **low**, and all other priorities to the **best-effort** forwarding class with a packet loss priority of **high**.

Verifying the PFC Flow Control Configuration (CNP)

Purpose

Verify that PFC is enabled on the correct input priorities and that flow control is configured on the correct output queues and priorities in each CNP.

Action

List the congestion notification profiles using the operational mode command **show class-of-service congestion-notification**:

`user@switch>` **show class-of-service congestion-notification**

```
Name: fcoe_p3_cnp, Index: 12037
Type: Input
Cable Length: 100 m
  Priority     PFC          MRU
  000          Disabled
  001          Disabled
  010          Disabled
  011          Enabled      2240
  100          Enabled      9216
  101          Disabled
  110          Disabled
  111          Disabled
Type: Output
  Priority     Flow-Control-Queues
  000
```

```
             0
  001
             1
  010
             2
  011
             3
  100
             4
  101
             5
  110
             6
  111
             7

Name: fcoe_p3_p5_cnp, Index: 46484
Type: Input
Cable Length: 100 m
  Priority     PFC          MRU
  000          Disabled
  001          Disabled
  010          Disabled
  011          Enabled      2240
  100          Disabled
  101          Enabled      2240
  110          Disabled
  111          Disabled
Type: Output
  Priority     Flow-Control-Queues
  011
             3
  101
             5

Name: fcoe_p5_cnp, Index: 12133
Type: Input
Cable Length: 150 m
  Priority     PFC          MRU
  000          Disabled
  001          Disabled
  010          Disabled
  011          Disabled
  100          Enabled      9216
```

```
  101         Enabled      2240
  110         Disabled
  111         Disabled
Type: Output
  100
              4
  101
              5

Name: iscsi_cnp, Index: 19342
Type: Input
Cable Length: 100 m
  Priority     PFC          MRU
  000         Disabled
  001         Disabled
  010         Disabled
  011         Disabled
  100         Enabled      9216
  101         Disabled
  110         Disabled
  111         Disabled
Type: Output
  Priority     Flow-Control-Queues
  000
              0
  001
              1
  010
              2
  011
              3
  100
              4
  101
              5
  110
              6
  111
              7
```

Meaning

The **show class-of-service congestion-notification** command shows the input and output stanzas of the four CNPs.

For CNP **fcoe_p3_cnp**, the input stanza shows that PFC is enabled on IEEE 802.1p code point **011** (priority 3) with an MRU of **2240** bytes, and cable length of **100** meters. The input stanza also shows that PFC is enabled on code point **100** (priority 4) with the default MRU value of **9216** bytes. The CNP output stanza shows the default mapping of priorities to output queues because no explicit output CNP is configured.

> **NOTE:** By default, only queues 3 and 4 are enabled respond to pause messages from the connected peer. For queue 3 to respond to pause messages, priority 3 (code point 011) must be enabled for PFC in the input stanza. For queue 4 to respond to pause messages, priority 4 (code point 100) must be enabled for PFC in the input stanza. In this example, only queues 3 and 4 respond to pause messages from the connected peer on interfaces that use CNP **fcoe_p3_cnp** because the input stanza enables PFC only on priorities 3 and 4.

For CNP **fcoe_p3_p5_cnp**, the input stanza shows that PFC is enabled on code points **011** and **101** (priority 5), the MRU is **2240** bytes on both priorities, and the cable length is **100** meters. The CNP output stanza shows that output flow control is configured on queues **3** and **5** for code points **011** and **101**, respectively.

For CNP **fcoe_p5_cnp**, the input stanza shows that PFC is enabled on code points **100** and **101**. The MRU for code point **101** (FCoE traffic) is **2240** bytes and the MRU for code point **100** is **9216**. The interface cable length is **150** meters. The CNP output stanza shows that output flow control is configured on queue **4** for code point **100** and on queue **5** for code point **101**.

For CNP **iscsi_cnp**, the input stanza shows that PFC is enabled on code point **100**, the MRU value is **9216** bytes, and the interface cable length is **100** meters. The CNP output stanza shows the default mapping of priorities to output queues because no explicit output CNP is configured.

Verifying the Interface Configuration

Purpose

Verify that the correct classifiers and congestion notification profiles are configured on the correct interfaces.

Action

List the ingress interfaces using the operational mode commands **show configuration class-of-service interfaces xe-0/0/31**, **show configuration class-of-service interfaces xe-0/0/32**, **show configuration class-of-service interfaces xe-0/0/33**, and **show configuration class-of-service interfaces xe-0/0/34**:

`user@switch>` **show configuration class-of-service interfaces xe-0/0/31**

```
congestion-notification-profile fcoe_p3_cnp;
unit 0 {
    classifiers {
        ieee-802.1 fcoe_p3_iscsi;
```

```
    }
}
```

user@switch> **show configuration class-of-service interfaces xe-0/0/32**

```
congestion-notification-profile fcoe_p5_cnp;
unit 0 {
    classifiers {
        ieee-802.1 fcoe_p5_iscsi;
    }
}
```

user@switch> **show configuration class-of-service interfaces xe-0/0/33**

```
congestion-notification-profile fcoe_p3_p5_cnp;
unit 0 {
    classifiers {
        ieee-802.1 fcoe_p3_p5;
    }
}
```

user@switch> **show configuration class-of-service interfaces xe-0/0/34**

```
congestion-notification-profile iscsi_cnp;
unit 0 {
    classifiers {
        ieee-802.1 iscsi_classifier;
    }
}
```

Meaning

The **show configuration class-of-service interfaces xe-0/0/31** command shows that the congestion notification profile **fcoe_p3_cnp** is configured on the interface, and that the IEEE 802.1p classifier associated with the interface is **fcoe_p3_iscsi**.

The **show configuration class-of-service interfaces xe-0/0/32** command shows that the congestion notification profile **fcoe_p5_cnp** is configured on the interface, and that the IEEE 802.1p classifier associated with the interface is **fcoe_p5_iscsi**.

The **show configuration class-of-service interfaces xe-0/0/33** command shows that the congestion notification profile **fcoe_p3_p5_cnp** is configured on the interface, and that the IEEE 802.1p classifier associated with the interface is **fcoe_p3_p5**.

The **show configuration class-of-service interfaces xe-0/0/34** command shows that the congestion notification profile **iscsi_cnp** is configured on the interface, and that the IEEE 802.1p classifier associated with the interface is **iscsi_classifier**.

Verifying the DCBX Application Configuration

Purpose

Verify that the DCBX applications for FCoE and iSCSI are configured.

Action

List the DCBX applications by using the configuration mode command **show applications**:

`user@switch#` **show applications**

```
application iscsi_app {
    protocol tcp;
    destination-port 3260;
}
application fcoe_app {
    ether-type 0x8906;
```

Meaning

The **show applications** configuration mode command shows all of the configured applications. The output shows that the application **iscsi_app** is configured with a protocol value of **tcp** and a destination port value of **3260**, and that the application **fcoe_app** is configured with an EtherType of **0x8906** (the correct EtherType for FCoE traffic).

Verifying the DCBX Application Map Configuration

Purpose

Verify that the application map is configured.

Action

List the application maps by using the configuration mode command **show policy-options application-maps**:

`user@switch#` **show policy-options application-maps**

```
dcbx-iscsi-fcoe-app-map {
    application iscsi_app code-points 100;
```

```
    application fcoe_app code-points [011 101];
}
```

Meaning

The **show policy-options application-maps** configuration mode command lists all of the configured application maps and the applications that belong to each application map. The output shows that there is one application map named **dcbx-iscsi-fcoe_app_map**. It consists of the application **iscsi_app** mapped to code point **100** and the application **fcoe_app** mapped to code points **011** and **101**.

Verifying the DCBX Application Protocol Exchange Interface Configuration

Purpose

Verify that the application maps are applied to the correct interfaces.

Action

List the application maps on each interface using the configuration mode command **show protocols dcbx**:

`user@switch#` **show protocols dcbx**

```
interface xe-0/0/31.0 {
    application-map dcbx-iscsi-fcoe-app-map;
}
interface xe-0/0/32.0 {
    application-map dcbx-iscsi-fcoe-app-map;
}
interface xe-0/0/33.0 {
    application-map dcbx-iscsi-fcoe-app-map;
}
interface xe-0/0/34.0 {
    application-map dcbx-iscsi-fcoe-app-map;
}
```

Meaning

The **show protocols dcbx** configuration mode command lists the application map association with interfaces. The output shows that all four interfaces use the application map **dcbx-iscsi-fcoe-app-map**.

RELATED DOCUMENTATION

Example: Configuring Two or More Lossless FCoE Priorities on the Same FCoE Transit Switch Interface | 576

Troubleshooting Dropped FCoE Traffic

Problem

Description: Fibre Channel over Ethernet (FCoE) traffic for which you want guaranteed delivery is dropped.

Cause

There are several possible causes of dropped FCoE traffic (the list numbers of the possible causes correspond to the list numbers of the solutions in the *Solution* section.):

1. Priority-based flow control (PFC) is not enabled on the FCoE priority (IEEE 802.1p code point) in both the input and output stanzas of the congestion notification profile.

2. The FCoE traffic is not classified correctly at the ingress interface. FCoE traffic should either use the default **fcoe** forwarding class and classifier configuration (maps the **fcoe** forwarding class to IEEE 802.1p code point 011) or be mapped to a lossless forwarding class and to the code point enabled for PFC on the input and output interfaces.

3. The congestion notification profile that enables PFC on the FCoE priority is not attached to the interface.

4. The forwarding class set (priority group) used for guaranteed delivery traffic does not include the forwarding class used for FCoE traffic.

   NOTE: This issue can occur only on switches that support enhanced transmission selection (ETS) hierarchical port scheduling. (Direct port scheduling does not use forwarding class sets.)

5. Insufficient bandwidth has been allocated for the FCoE queue or for the forwarding class set to which the FCoE queue belongs.

NOTE: This issue can occur for forwarding class sets only on switches that support ETS hierarchical port scheduling. (Direct port scheduling does not use forwarding class sets.)

6. If you are using Junos OS Release 12.2, the **fcoe** forwarding class has been explicitly configured instead of using the default **fcoe** forwarding class configuration (forwarding-class-to-queue mapping).

   NOTE: If you are using Junos OS Release 12.2, 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.

7. If you are using Junos OS Release 12.3 or later and you are not using the default **fcoe** forwarding class configuration, the forwarding class used for FCoE is not configured with the **no-loss** packet drop attribute. In Junos OS 12.3 or later, explicit forwarding classes configurations must include the **no-loss** packet drop attribute to be treated as lossless forwarding classes.

Solution

The list numbers of the possible solutions correspond to the list numbers of the causes in the *Cause* section.

1. Check the congestion notification profile (CNP) to see if PFC is enabled on the FCoE priority (the correct IEEE 802.1p code point) on both input and output interfaces. Use the **show class-of-service congestion-notification** operational command to show the code points that are enabled for PFC in each CNP.

   If you are using the default configuration, FCoE traffic is mapped to code point 011 (priority 3). In this case, the input stanza of the CNP should show that PFC is enabled on code point 011, and the output stanza should show that priority 011 is mapped to flow control queue 3.

   If you explicitly configured a forwarding class for FCoE traffic, ensure that:

   - You specified the **no-loss** packet drop attribute in the forwarding class configuration
   - The code point mapped to the FCoE forwarding class in the ingress classifier is the code point enabled for PFC in the CNP input stanza
   - The code point and output queue used for FCoE traffic are mapped to each other in the CNP output stanza (if you are not using the default priority and queue, you must explicitly configure each output queue that you want to respond to PFC messages)

   For example, if you explicitly configure a forwarding class for FCoE traffic that is mapped to output queue 5 and to code point 101 (priority 5), the output of the **show class-of-service congestion-notification** looks like:

   ```
   Name: fcoe_p5_cnp, Index: 12183
   Type: Input
   Cable Length: 100 m
     Priority     PFC          MRU
     000          Disabled
     001          Disabled
     010          Disabled
     011          Disabled
     100          Disabled
     101          Enabled      2500
     110          Disabled
     111          Disabled
   Type: Output
     Priority     Flow-Control-Queues
     101
                  5
   ```

2. Use the **show class-of-service classifier type ieee-802.1p** operational command to check if the classifier maps the forwarding class used for FCoE traffic to the correct IEEE 802.1p code point.

3. Ensure that the congestion notification profile and classifier are attached to the correct ingress interface. Use the operational command **show configuration class-of-service interfaces *interface-name***.

4. Check that the forwarding class set includes the forwarding class used for FCoE traffic. Use the operational command **show configuration class-of-service forwarding-class-sets** to show the configured priority groups and their forwarding classes.

5. Verify the amount of bandwidth allocated to the queue mapped to the FCoE forwarding class and to the forwarding class set to which the FCoE traffic queue belongs. Use the **show configuration class-of-service schedulers** ***scheduler-name*** operational command (specify the scheduler for FCoE traffic as the ***scheduler-name***) to see the minimum guaranteed bandwidth (**transmit-rate**) and maximum bandwidth (**shaping-rate**) for the queue.

   Use the **show configuration class-of-service traffic-control-profiles** ***traffic-control-profile*** operational command (specify the traffic control profile used for FCoE traffic as the ***traffic-control-profile***) to see the minimum guaranteed bandwidth (**guaranteed-rate**) and maximum bandwidth (**shaping-rate**) for the forwarding class set.

6. Delete the explicit FCoE forwarding-class-to-queue mapping so that the system uses the default FCoE forwarding-class-to-queue mapping. Include the **delete forwarding-classes class fcoe queue-num 3** statement at the **[edit class-of-service]** hierarchy level to remove the explicit configuration. The system then uses the default configuration for the FCoE forwarding class and preserves the lossless treatment of FCoE traffic.

7. Use the **show class-of-service forwarding-class** operational command to display the configured forwarding classes. The *No-Loss* column shows whether lossless transport is enabled or disabled for each forwarding class. If the forwarding class used for FCoE traffic is not enabled for lossless transport, include the **no-loss** packet drop attribute in the forwarding class configuration (**set class-of-service forwarding-classes class** ***fcoe-forwarding-class-name*** **queue-num** ***queue-number*** **no-loss**).

See "Example: Configuring CoS PFC for FCoE Traffic" on page 490 for step-by-step instructions on how to configure PFC for FCoE traffic, including classifier, interface, congestion notification profile, PFC, and bandwidth scheduling configuration.

RELATED DOCUMENTATION

show class-of-service congestion-notification | 884

Configuring CoS PFC (Congestion Notification Profiles) | 194

Example: Configuring CoS PFC for FCoE Traffic | 490

Understanding CoS Flow Control (Ethernet PAUSE and PFC) | 197

5

PART

CoS Buffers and the Shared Buffer Pool

CHAPTER 18

CoS Buffers

IN THIS CHAPTER

Understanding CoS Buffer Configuration

IN THIS SECTION

Packet Forwarding Engine (PFE) wide common packet buffer memory is used to store packets on interface queues. The buffer memory has separate ingress and egress accounting to make accept, drop, or pause decisions. Because the switch has a single pool of memory with separate ingress and egress accounting, the full amount of buffer memory is available from both the ingress and the egress perspective. Packets are accounted for as they enter and leave the switch, but there is no concept of a packet arriving at an ingress buffer and then being moved to an egress buffer. Specific common buffer memory amounts for individual switches is listed in Table 98 on page 636.

Table 98: Common Packet Buffer Memory on Switches

Switch	Common Packet Buffer Memory
QFX3500, QFX3600	9 MB
QFX5100, EX4600, and OCX Series	12 MB
QFX5110, QFX5200-32C	16MB
QFX5200-48Y	22MB
QFX5120	32MB
QFX5210	42MB

NOTE: QFX10000 does not have a shared buffer.

The buffers are divided into two pools from both an ingress and an egress perspective:

1. *Shared buffers* are a global memory pool that the switch allocates dynamically to ports as needed, so the buffers are shared among the switch ports.
2. *Dedicated buffers* are a memory pool divided equally among the switch ports. Each port receives a minimum guaranteed amount of buffer space, dedicated to each port, not shared among ports.

NOTE: Lossless traffic is traffic on which you enable priority-based flow control (PFC) to ensure lossless transport. Lossless traffic does not refer to best-effort traffic on a link enabled for Ethernet PAUSE (IEEE 802.3x).

OCX Series switches do not support lossless transport or PFC. In this topic, references to lossless transport do not apply to OCX Series switches.

The switch reserves nonconfigurable buffer space to ensure that ports and queues receive a minimum memory allocation. You can configure how the system uses the rest of the buffer space to optimize the allocation for your mix of network traffic. You can configure the percentage of available buffer space used as shared buffer space versus dedicated buffer space. You can also configure how shared buffer space is allocated to different types of traffic. You can optimize the buffer settings for the traffic on your network.

The default buffer configuration is designed for networks that have a balance of best-effort and lossless traffic. Because OCX Series switches do not support lossless traffic, instead of using the default buffer configuration on OCX Series switches, consider configuring the buffers as recommended for networks with mostly best-effort traffic as shown in Table 117 on page 652 and Table 118 on page 652.

The default class-of-service configuration provides two lossless forwarding classes (**fcoe** and **no-loss**), a best-effort unicast forwarding class, a network control traffic forwarding class, and one multidestination (multicast, broadcast, and destination lookup fail) forwarding class.

> **NOTE:** On OCX Series switches, do not map traffic to the default lossless forwarding classes.

Each default forwarding class maps to a different default output queue. The default configuration allocates the buffers in a manner that supports a moderate amount of lossless traffic while still providing the ability to absorb bursts in best-effort traffic transmission.

Changing the buffer settings changes the abilities of the buffers to absorb traffic bursts and handle lossless traffic. For example, networks with mostly best-effort traffic require allocating most of the shared buffer space to best-effort buffers. This provides deep, flexible buffers that can absorb traffic bursts with minimal packet loss, at the expense of buffer availability for lossless traffic.

Conversely, networks with mostly lossless traffic require allocating most of the shared buffer space to lossless headroom buffers. This prevents packet loss on lossless flows at the expense of absorbing bursty best-effort traffic efficiently.

> **CAUTION:** Changing the buffer configuration is a disruptive event. Traffic stops on *all* ports until buffer reprogramming is complete.

This topic describes the buffer architecture and settings:

Buffer Pools

IN THIS SECTION

From both an ingress and an egress perspective, the PFE buffer is split into two main pools, a shared buffer pool and a dedicated buffer pool that ensures a minimum allocation to each port. You can configure the amount of buffer space allocated to each of the two pools. A portion of the buffer space is reserved so that there is always a minimum amount of shared and dedicated buffer space available to each port.

- Shared buffer pool—A global memory space that all of the ports on the switch share dynamically as they need buffers. The shared buffer pool is further partitioned into buffers for best-effort unicast, best-effort multidestination (broadcast, multicast, and destination lookup fail), and PFC (lossless) traffic types. You can allocate global shared memory space to buffer partitions to better support different mixes of network traffic. The larger the shared buffer pool, the better the switch can absorb traffic bursts because more shared memory is available for the traffic.
- Dedicated buffer pool—A reserved global memory space allocated equally to each port. The switch reserves a minimum dedicated buffer pool that is not user-configurable. You can divide the dedicated buffer allocation for a port among the port queues on a per-port, per-queue basis. (For example, this enables you to dedicate more buffer space to queues that transport lossless traffic.)

  A larger dedicated buffer pool means a larger amount of dedicated buffer space for each port, so congestion on one port is less likely to affect traffic on another port because the traffic does not need to use as much shared buffer space. However, the larger the dedicated buffer pool, the less bursty traffic the switch can handle because there is less dynamic shared buffer memory.

You can configure the way the available unreserved portion of the buffer space is allocated to the global shared buffer pool and to the dedicated shared buffer pool by configuring the ingress and egress shared buffer percentages.

By default, 100 percent of the available unreserved buffer space is allocated to the shared buffer pool. If you change the percentage of space allocated to the shared buffer, the available buffer space that is not allocated to the shared buffer is allocated to the dedicated buffer. For example, if you configure the ingress shared buffer pool as 80 percent, the remaining 20 percent of the available buffer space is allocated to the dedicated buffer pool and divided equally across the ports.

NOTE: When 100 percent of the available (user-configurable) buffers are allocated to the shared buffer pool, the switch still reserves a minimum dedicated buffer pool.

You can separately configure ingress and egress shared buffer pool allocations. You can also partition the ingress and egress shared buffer pool to allocate percentages of the shared buffer pool to specific types of traffic. If you do not use the default configuration or one of the recommended configurations, pay particular attention to the ingress configuration of the lossless headroom buffers (these buffers handle PFC pause during periods of congestion) and to the egress configuration of the best-effort buffers to handle incast congestion (multiple synchronized sources sending data to the same receiver in parallel).

In addition to the shared buffer pool and the dedicated buffer pool, there is also a small ingress global headroom buffer pool that is reserved and is not configurable.

When contention for buffer space occurs, the switch uses an internal algorithm to ensure that the buffer pools are distributed fairly among competing flows. When traffic for a given flow exceeds the amount of dedicated port buffer reserved for that flow, the flow begins to consume memory from the dynamic shared buffer pool. Competing flows compete for shared buffer memory with other flows that also have exhausted their dedicated buffers. When there is no congestion, there are no competing flows.

Buffer Handling of Lossless Flows (PFC) Versus Ethernet PAUSE

When we discuss lossless buffers in the following sections, we mean buffers that handle traffic on which you enable PFC to ensure lossless transport. The lossless buffers are not used for best-effort traffic on a link on which you enable Ethernet PAUSE (IEEE 802.3x). The lossless ingress and egress shared buffers, and the ingress lossless headroom shared buffer, are used only for traffic on which you enable PFC.

NOTE: To support lossless flows, you must configure the appropriate data center bridging capabilities (PFC, DCBX, and ETS) and scheduling properties.

NOTE: OCX Series switches do not support PFC or lossless transport. OCX Series switches support symmetric Ethernet PAUSE.

Shared Buffer Pool and Partitions

The shared buffer pool is a global memory space that all of the ports on the switch share dynamically as they need buffers. The switch uses the shared buffer pool to absorb traffic bursts after the dedicated buffer pool for a port is exhausted.

You can divide both the ingress shared buffer pool and the egress shared buffer pool into three partitions to allocate percentages of each buffer pool to different types of traffic. When you partition the ingress or egress shared buffer pool:

- If you explicitly configure one ingress shared buffer partition, you must explicitly configure all three ingress shared buffer partitions. (You either explicitly configure all three ingress partitions or you use the default setting for all three ingress partitions.)

  If you explicitly configure one egress shared buffer partition, you must explicitly configure all three egress shared buffer partitions. (You either explicitly configure all three egress partitions or you use the default setting for all three egress partitions.)

  The switch returns a commit error if you do not explicitly configure all three partitions when configuring the ingress or egress shared buffer partitions.

- The combined percentages of the three ingress shared buffer partitions must total exactly 100 percent.

  The combined percentages of the three egress shared buffer partitions must total exactly 100 percent.

  When you explicitly configure ingress or egress shared buffer partitions, the switch returns a commit error if the total percentage of the three partitions does not equal 100 percent.

- If you explicitly partition one set of shared buffers, you do not have to explicitly partition the other set of shared buffers. For example, you can explicitly configure the ingress shared buffer partitions and use the default egress shared buffer partitions. However, if you change the buffer partitions for the ingress buffer pool to match the expected types of traffic flows, you would probably also want to change the buffer partitions for the egress buffer pool to match those traffic flows.

You can configure the percentage of available unreserved buffer space allocated to the shared buffer pool. Space that you do not allocate to the shared buffer pool is added to the dedicated buffer pool and divided equally among the ports. The default configuration allocates 100 percent of the unreserved ingress and egress buffer space to the shared buffers.

Configuring the ingress and egress shared buffer pool partitions enables you to allocate more buffers to the types of traffic your network predominantly carries, and fewer buffers to other traffic.

Ingress Shared Buffer Pool Partitions

You can configure three ingress buffer pool partitions:

- Lossless buffers—Shared buffer pool for all lossless ingress traffic. We recommend 5 percent as the minimum value for lossless buffers.

- Lossless headroom buffers—Shared buffer pool for packets received while a pause is asserted. If PFC is enabled on priorities on a port, when the port sends a pause message to the connected peer, the port uses the headroom buffers to store the packets that arrive between the time the port sends the pause message and the time the last packet arrives after the peer pauses traffic. The minimum value for lossless headroom buffers is 0 (zero) percent. (Lossless headroom buffers are the only buffers for which the recommended value can be less than 5 percent.)

- Lossy buffers—Shared buffer pool for all best-effort ingress traffic (best-effort unicast, multidestination, and strict-high priority traffic). We recommend 5 percent as the minimum value for best-effort buffers.

The combined percentage values of the ingress lossless, lossless headroom, and best-effort buffer partitions must total exactly 100 percent. If the buffer percentages total more than 100 percent or less than 100 percent, the switch returns a commit error. If you explicitly configure an ingress shared buffer partition, you must explicitly configure all three ingress buffer partitions, even if the lossless headroom buffer partition has a value of 0 (zero) percent.

Egress Shared Buffer Pool Partitions

You can configure three egress buffer pool partitions:

- Lossless buffers—Shared buffer pool for all lossless egress queues. We recommend 5 percent as the minimum value for lossless buffers.
- Lossy buffers—Shared buffer pool for all best-effort egress queues (best-effort unicast, and strict-high priority queues). We recommend 5 percent as the minimum value for best-effort buffers.
- Multicast buffers—Shared buffer pool for all multidestination (multicast, broadcast, and destination lookup fail) egress queues. We recommend 5 percent as the minimum value for multicast buffers.

The combined percentage values of the egress lossless, lossy, and multicast buffer partitions must total exactly 100 percent. If the buffer percentages total more than 100 percent or less than 100 percent, the switch returns a commit error. All egress buffer partitions must be explicitly configured and should have a value of at least 5 percent. If you explicitly configure an egress shared buffer partition, you must explicitly configure all three egress buffer partitions, and each partition should have a value of at least 5 percent.

> **NOTE:** QFX5200-32C does not replicate all multicast streams when two or more downstream interface packet sizes are higher than ~6k and have an 1000pps packet ingress rate. This is because the number of working flows on QFX5200-32C is indirectly proportional to the packet size and directly proportional to available multicast shared buffers.

Dedicated Port Buffer Pool and Buffer Allocation to Queues

The global dedicated buffer pool is memory that is allocated equally to each port, so each port receives a guaranteed minimum amount of buffer space. Dedicated buffers are not shared among ports. Each port receives an equal proportion of the dedicated buffer pool.

When traffic enters and exits the switch, the switch ports use their dedicated buffers to store packets. If the dedicated buffers are not sufficient to handle the traffic, the switch uses shared buffers. The only way to increase the dedicated buffer pool is to decrease the shared buffer pool from its default value of 100 percent of available unreserved buffers.

The amount of dedicated buffer space is not user-configurable and depends on the percentage of available nonreserved buffers allocated to the shared buffers. (The dedicated buffer space is equal to the minimum reserved port buffers plus the remainder of the available nonreserved buffers that are not allocated to the shared buffer pool.)

NOTE: If 100 percent of the available unreserved buffers are allocated to the shared buffer pool, the switch still reserves a minimum dedicated buffer pool.

The larger the shared buffer pool, the better the burst absorption across the ports. The larger the dedicated buffer pool, the larger the amount of dedicated buffer space for each port. The greater the dedicated buffer space, the less likely that congestion on one port can affect traffic on another port, because the traffic does not need to use as much shared buffer space.

Allocating Dedicated Port Buffers to Queues

You can divide the dedicated buffer allocation for an egress port among the port queues by including the **buffer-size** statement in the scheduler configuration. This enables you to control the egress port dedicated buffer allocation on a per-port, per-queue basis. (For example, this enables you to dedicate more buffer space to queues that transport lossless traffic, or to stop the port from reserving buffers for queues that do not carry traffic.) Egress dedicated port buffer allocation is a hierarchical structure that allocates a global dedicated buffer pool evenly among ports, and then divides the allocation for each port among the port queues.

By default, ports divide their allocation of dedicated buffers among their egress queues in the same proportion as the default scheduler sets the minimum guaranteed transmission rates (the **transmit-rate** option) for traffic. Only the queues included in the default scheduler receive bandwidth and dedicated buffers, in the proportions shown in Table 99 on page 642:

Table 99: Default Dedicated Buffer Allocation to Egress Queues (Based on Default Scheduler)

Forwarding Class	Queue	Minimum Guaranteed Bandwidth (**transmit-rate**)	Proportion of Reserved Dedicated Port Buffers
best-effort	0	5%	5%
fcoe	3	35%	35%
no-loss	4	35%	35%
network-control	7	5%	5%
mcast	8	20%	20%

In the default configuration, no egress queues other than the ones shown in Table 99 on page 642 receive an allocation of dedicated port buffers.

NOTE: The switch uses hierarchical scheduling to control port and queue bandwidth allocation, as described in "Understanding CoS Hierarchical Port Scheduling (ETS)" on page 407 and shown in "Example: Configuring CoS Hierarchical Port Scheduling (ETS)" on page 415. For egress queue buffer size configuration, when you attach a traffic control profile (includes the queue scheduler information) to a port, the dedicated egress buffers on the port are divided among the queues as configured in the scheduler.

If you do not want to use the default allocation of dedicated port buffers to queues, use the **buffer-size** option in the scheduler that is attached to the port to configure the queue allocation. You can configure the dedicated buffer allocation to queues in two ways:

- As a percentage—The queue receives the specified percentage of dedicated port buffers when the queue is mapped to the scheduler and the scheduler is attached to a port.
- As a remainder—After the port services the queues that have an explicit percentage buffer size configuration, the remaining dedicated port buffer space is divided equally among the other queues to which a scheduler is attached. (No default or explicit scheduler for a queue means no dedicated buffer allocation for that queue.) If you configure a scheduler and you do not specify a buffer size as a percentage, *remainder* is the default setting.

NOTE: The total of all of the explicitly configured buffer size percentages for all of the queues on a port cannot exceed 100 percent.

Configuring Dedicated Port Buffer Allocation to Queues

In a port configuration that includes multiple forwarding class sets, with multiple forwarding classes mapped to multiple schedulers, the allocation of port dedicated buffers to queues depends on the mix of queues with buffer sizes configured as explicit percentages and queues configured with (or defaulted to) the **remainder** option.

The best way to demonstrate how using the percentage and remainder options affects dedicated port buffer allocation to queues is by showing an example of queue buffer allocation, and then showing how the queue buffer allocation changes when you add another forwarding class (queue) to the port.

Table 100 on page 644 shows an initial configuration that includes four forwarding class sets, the five default forwarding classes (mapped to the five default queues for those forwarding classes), the **buffer-size** option configuration, and the resulting buffer allocation for each queue. Table 101 on page 644 shows the same configuration after we add another forwarding class (best-effort-2, mapped to queue 1) to the best-effort forwarding class set. Comparing the buffer allocations in each table shows you how adding another queue affects buffer allocation when you use remainders and explicit percentages to configure the buffer allocation for different queues.

Table 100: Egress Queue Dedicated Buffer Allocation (Example 1)

Forwarding Class Set (Priority Group)	Forwarding Class	Queue	Scheduler Buffer Size Configuration	Buffer Allocation per Queue (Percentage)
fc-set-be	best-effort	0	10%	10%
fc-set-lossless	fcoe	3	20%	20%
	no-loss	4	40%	40%
fc-set-strict-high	network-control	7	remainder	15%
fc-set-mcast	mcast	8	remainder	15%

In this first example, 70 percent of the egress port dedicated buffer pool is explicitly allocated to the best-effort, fcoe, and no-loss queues. The remaining 30 percent of the port dedicated buffer pool is split between the two queues that use the **remainder** option (network-control and mcast), so each queue receives 15 percent of the dedicated buffer pool.

Now we add another forwarding class (queue) to the best-effort priority group (fc-set-be) and configure it with a buffer size of *remainder* instead of configuring a specific percentage. Because a third queue now shares the remaining dedicated buffers, the queues that share the remainder receive fewer dedicated buffers, as shown in Table 101 on page 644. The queues with explicitly configured percentages receive the configured percentage of dedicated buffers.

Table 101: Egress Queue Dedicated Buffer Allocation with Another Remainder Queue (Example 2)

Priority Group (fc-set)	Forwarding Class	Queue	Scheduler Buffer Size Configuration	Buffer Allocation per Queue (Percentage)
fc-set-be	best-effort	0	10%	10%
	best-effort-2	1	remainder	10%
fc-set-lossless	fcoe	3	20%	20%
	no-loss	4	40%	40%
fc-set-strict-high	network-control	7	remainder	10%
fc-set-mcast	mcast	8	remainder	10%

The two tables show how the port divides the dedicated buffer space that remains after servicing the queues that have an explicitly configured percentage of dedicated buffer space.

Trade-off Between Shared Buffer Space and Dedicated Buffer Space

The trade-off between shared buffer space and dedicated buffer space is:

- Shared buffers provide better absorption of traffic bursts because there is a larger pool of dynamic buffers that ports can use as needed to handle the bursts. However, all flows that exhaust their dedicated buffer space compete for the shared buffer pool. A larger shared buffer pool means a smaller dedicated buffer pool, and therefore more competition for the shared buffer pool because more flows exhaust their dedicated buffer allocation. Too much shared buffer space results in no single flow receiving very much shared buffer space, to maintain fairness when many flows contend for that space.
- Dedicated buffers provide guaranteed buffer space to each port. The larger the dedicated buffer pool, the less likely that congestion on one port affects traffic on another port, because the traffic does not need to use as much shared buffer space. However, less shared buffer space means less ability to dynamically absorb traffic bursts.

For optimal burst absorption, the switch needs enough dedicated buffer space to avoid persistent competition for the shared buffer space. When fewer flows compete for the shared buffers, the flows that need shared buffer space to absorb bursts receive more of the shared buffer because fewer flows exhaust their dedicated buffer space.

The default configuration and the configurations recommended for different traffic scenarios allocate 100 percent of the user-configurable memory space to the global shared buffer pool because the amount of space reserved for dedicated buffers provides enough space to avoid persistent competition for dynamic shared buffers. This results in fewer flows competing for the shared buffers, so the competing flows receive more of the buffer space.

Order of Buffer Consumption

The total buffer pool is divided into ingress and egress shared buffer pools and dedicated buffer pools. When traffic flows through the switch, the buffer space is used in a particular order that depends on the type of traffic.

On ingress, the order of buffer consumption is:

- Best-effort unicast traffic:
  1. Dedicated buffers
  2. Shared buffers
  3. Global headroom buffers (very small)
- Lossless unicast traffic:
  1. Dedicated buffers
  2. Shared buffers
  3. Lossless headroom buffers
  4. Global headroom buffers (very small)

- Multidestination traffic:
  1. Dedicated buffers
  2. Shared buffers
  3. Global headroom buffers (very small)

On egress, the order of buffer consumption is the same for unicast best-effort, lossless unicast, and multidestination traffic:

- Dedicated buffers
- Shared buffers

In all cases on all ports, the switch uses the dedicated buffer pool first and the shared buffer pool only after the dedicated buffer pool for the port or queue is exhausted. This reserves the maximum amount of dynamic shared buffer space to absorb traffic bursts.

Default Buffer Pool Values

IN THIS SECTION

You can view the default or configured ingress and egress buffer pool values in KB units using the **show class-of-service shared-buffer** operational command. You can view the configured shared buffer pool values in percent units using the **show configuration class-of-service shared-buffer** operational command.

This section provides the default total buffer, shared buffer, and dedicated buffer values.

Total Buffer Pool Size

The total buffer pool is common memory that has separate ingress and egress accounting, so the full buffer pool is available from both the ingress and egress perspective. The total buffer pool consists of the dedicated buffer space and the shared buffer space. The size of the total buffer pool is not user-configurable, but the allocation of buffer space to the dedicated and shared buffer pools is user-configurable.

On QFX3500 and QFX3600 switches, the combined total size of the ingress and egress buffer pools is approximately 9 MB (exactly 9360 KB).

On QFX5100, EX4600, and OCX Series switches, the combined total size of the ingress and egress buffer pools is approximately 12 MB (exactly 12480 KB).

On QFX5110 and QFX5200-32C switches, the combined total size of the ingress and egress buffer pools is approximately 16 MB.

On QFX5200-48Y switches, the combined total size of the ingress and egress buffer pools is approximately 22 MB.

On QFX5210 switches, the combined total size of the ingress and egress buffer pools is approximately 42 MB.

Shared Buffer Pool Default Values

IN THIS SECTION

Some switches have a larger shared buffer pool than other switches. However, the allocation of shared buffer space to the individual ingress and egress buffer pools is the same on a percentage basis, even though the absolute values are different. For example, the default ingress lossless buffer is 9 percent of the total shared ingress buffer space on all of the switches, even though the default absolute value of the ingress lossless buffer differs from switch to switch.

This section describes the default values in percent and in KB for the shared ingress and shared egress buffers.

Shared Ingress Buffer Default Values

Table 102 on page 647 shows the default ingress shared buffer allocation values in KB units for QFX5210 switches.

Table 102: QFX5210 Switch Default Shared Ingress Buffer Values (KB)

Total Shared Ingress Buffer	Lossless Buffer	Lossless-Headroom Buffer	Lossy Buffer
29224	2630.16	13150.80	13443.04

Table 103 on page 647 shows the default ingress shared buffer allocation values in KB units for QFX5200-48Y switches.

Table 103: QFX5200-48Y Switch Default Shared Ingress Buffer Values (KB)

Total Shared Ingress Buffer	Lossless Buffer	Lossless-Headroom Buffer	Lossy Buffer
19154.69	1723.92	8619.61	8811.16

Table 104 on page 648 shows the default ingress shared buffer allocation values in KB units for QFX5110 and QFX5200-32C switches.

Table 104: QFX5110 and QFX5200-32C Switch Default Shared Ingress Buffer Values (KB)

Total Shared Ingress Buffer	Lossless Buffer	Lossless-Headroom Buffer	Lossy Buffer
11779.62	1060.17	5300.83	5418.63

Table 105 on page 648 shows the default ingress shared buffer allocation values in KB units for QFX5100, EX4600, and OCX Series switches.

Table 105: QFX5100, EX4600, and OCX Series Switch Default Shared Ingress Buffer Values (KB)

Total Shared Ingress Buffer	Lossless Buffer	Lossless-Headroom Buffer	Lossy Buffer
9567.19 KB	861.05 KB	4305.23 KB	4400.91 KB

Table 106 on page 648 shows the default ingress shared buffer allocation values in KB units for QFX3500 and QFX3600 switches.

Table 106: QFX3500 and QFX3600 Switch Default Shared Ingress Buffer Values (KB)

Total Shared Ingress Buffer	Lossless Buffer	Lossless-Headroom Buffer	Lossy Buffer
7202 KB	648.18 KB	3240.9 KB	3312.92 KB

Table 107 on page 648 shows the default ingress shared buffer allocation values as percentages for all switches. (If you change the default shared buffer allocation, you configure the change as a percentage.)

Table 107: Default Shared Ingress Buffer Values (Percentage)

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

Shared Egress Buffer Default Values

Table 108 on page 648 shows the default egress shared buffer allocation values in KB units for QFX5210 switches.

Table 108: QFX5210 Switch Default Shared Egress Buffer Values (KB)

Total Shared Egress Buffer	Lossless Buffer	Lossy Buffer	Multicast Buffer
28080	14040	8704.80	5335.20

Table 109 on page 649 shows the default egress shared buffer allocation values in KB units for QFX5200-48Y switches.

Table 109: QFX5200-48Y Switch Default Shared Egress Buffer Values (KB)

Total Shared Egress Buffer	Lossless Buffer	Lossy Buffer	Multicast Buffer
19115.69	9557.84	5925.86	3631.98

Table 110 on page 649 shows the default egress shared buffer allocation values in KB units for QFX5110 and QFX5200-32C switches.

Table 110: QFX5110 and QFX5200-32C Switch Default Shared Egress Buffer Values (KB)

Total Shared Egress Buffer	Lossless Buffer	Lossy Buffer	Multicast Buffer
11232	5616	3481.92	2134

NOTE: QFX5200-32C does not replicate all multicast streams when two or more downstream interface packet sizes are higher than ~6k and have an 1000pps packet ingress rate. This is because the number of working flows on QFX5200-32C is indirectly proportional to the packet size and directly proportional to available multicast shared buffers.

Table 111 on page 649 shows the default egress shared buffer allocation values in KB units for QFX5100, EX4600, and OCX Series switches.

Table 111: QFX5100, EX4600, and OCX Series Switch Default Shared Egress Buffer Values (KB)

Total Shared Egress Buffer	Lossless Buffer	Lossy Buffer	Multicast Buffer
8736 KB	4368 KB	2708.16 KB	1659.84 KB

Table 112 on page 649 shows the default egress shared buffer allocation values in KB units.

Table 112: QFX3500 and QFX3600 Switch Default Shared Egress Buffer Values (KB)

Total Shared Egress Buffer	Lossless Buffer	Lossy Buffer	Multicast Buffer
6656 KB	3328 KB	2063.36 KB	1264.64 KB

Table 113 on page 650 shows the default egress shared buffer allocation values for all switches as percentages.

Table 113: Default Shared Egress Buffer Values (Percentage)

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

Dedicated Buffer Pool Default Values

The system reserves ingress and egress dedicated buffer pools that are divided equally among the switch ports. By default, the system allocates 100 percent of the available unreserved buffer space to the shared buffer pool. If you reduce the percentage of available unreserved buffer space allocated to the shared buffer pool, the remaining unreserved buffer space is added to the dedicated buffer pool allocation. You configure the amount of dedicated buffer pool space by reducing (or increasing) the percentage of buffer space allocated to the shared buffer pool. You do not directly configure the dedicated buffer pool allocation.

Table 114 on page 650 shows the default ingress and egress dedicated buffer pool values in KB units for QFX5210, QFX5200, QFX5110, QFX5100, QFX3500, QFX3600, EX4600, and OCX Series switches.

Table 114: Default Ingress and Egress Dedicated Buffer Pool Values KB) per Switch (

Dedicated Buffer Type	QFX5210	QFX5200-48Y	QFX5110, QFX5200-32C	QFX5100, EX4600, OCX Series	QFX3500, QFX3600
Ingress	14040	3373.50	4860.38	2912.81	2158
Egress	15184	3412.50	5408	3744	2704

Shared Buffer Configuration Recommendations for Different Network Traffic Scenarios

IN THIS SECTION

The way you configure the shared buffer pool depends on the mix of traffic on your network. This section provides shared buffer configuration recommendations for five basic network traffic scenarios:

- Balanced traffic—The network carries a balanced mix of unicast best-effort, lossless, and multicast traffic. (This is the default configuration.)
- Best-effort unicast traffic—The network carries mostly unicast best-effort traffic.
- Best-effort traffic with Ethernet PAUSE (IEEE 802.3X) enabled—The network carries mostly best-effort traffic with Ethernet PAUSE enabled on the links.
- Best-effort multicast traffic—The network carries mostly multicast best-effort traffic.
- Lossless traffic—The network carries mostly lossless traffic (traffic on which PFC is enabled).

> **NOTE:** Lossless traffic is defined as traffic on which you enable PFC to ensure lossless transport. Lossless traffic does not refer to best-effort traffic on a link on which you enable Ethernet PAUSE. Start with the recommended profiles for each network traffic scenario, and adjust them if necessary for your network traffic conditions.
>
> OCX Series switches do not support lossless transport or PFC. In this topic, references to lossless transport do not apply to OCX Series switches. OCX Series switches support symmetric Ethernet PAUSE.

> **CAUTION:** Changing the buffer configuration is a disruptive event. Traffic stops on *all* ports until buffer reprogramming is complete. This includes changing the default configuration to one of the recommended configurations.

Because you configure buffer allocations in percentages, the recommended allocations for each network traffic scenario are valid for all QFX Series switches, EX4600 switches, and OCX Series switches. Use one of the following recommended shared buffer configurations for your network traffic conditions. Start with a recommended configuration, then make small adjustments to the buffer allocations to fine-tune the buffers if necessary as described in "Optimizing Buffer Configuration" on page 654.

Balanced Traffic (Default Configuration)

The default shared buffer configuration is optimized for networks that carry a balanced mix of best-effort unicast, lossless, and multidestination (multicast, broadcast, and destination lookup fail) traffic. The default class-of-service (CoS) configuration is also optimized for networks that carry a balanced mix of traffic.

> **NOTE:** On OCX Series switches, the default CoS configuration optimization does not include lossless traffic because OCX Series switches do not support lossless transport.

Except on OCX Series switches, we recommend that you use the default shared buffer configuration for networks that carry a balanced mix of traffic, especially if you are using the default CoS settings. Table 115 on page 652 shows the default ingress shared buffer allocations:

Table 115: Default Ingress Shared Buffer Configuration

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

Table 116 on page 652 shows the default egress shared buffer allocations:

Table 116: Default Egress Shared Buffer Configuration

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

Best-Effort Unicast Traffic

If your network carries mostly best-effort (lossy) unicast traffic, then the default shared buffer configuration allocates too much buffer space to support lossless transport. Instead of wasting those buffers, we recommend that you use the following ingress shared buffer settings (see Table 117 on page 652) and egress shared buffer settings (see Table 118 on page 652):

Table 117: Recommended Ingress Shared Buffer Configuration for Networks with Mostly Best-Effort Unicast Traffic

Total Shared Ingress Buffer	Lossless Buffer	Lossless-Headroom Buffer	Lossy Buffer
100%	5%	0%	95%

Table 118: Recommended Egress Shared Buffer Configuration for Networks with Mostly Best-Effort Unicast Traffic

Total Shared Egress Buffer	Lossless Buffer	Lossy Buffer	Multicast Buffer
100%	5%	75%	20%

See "Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Best-Effort Unicast Traffic" on page 659 for an example that shows you how to configure the recommended buffer settings shown in Table 117 on page 652 and Table 118 on page 652.

Ethernet PAUSE Traffic

If your network carries mostly best-effort (lossy) traffic *and* enables Ethernet PAUSE on links, then the default shared buffer configuration allocates too much buffer space to the shared ingress buffer (Ethernet PAUSE traffic uses the dedicated buffers instead of shared buffers) and not enough space to the lossless-headroom buffers. We recommend that you use the following ingress shared buffer settings (see Table 119 on page 653) and egress shared buffer settings (see Table 120 on page 653):

Table 119: Recommended Ingress Shared Buffer Configuration for Networks with Mostly Best-Effort Traffic and Ethernet PAUSE Enabled

Total Shared Ingress Buffer	Lossless Buffer	Lossless-Headroom Buffer	Lossy Buffer
70%	5%	80%	15%

Table 120: Recommended Egress Shared Buffer Configuration for Networks with Mostly Best-Effort Traffic and Ethernet PAUSE Enabled

Total Shared Egress Buffer	Lossless Buffer	Lossy Buffer	Multicast Buffer
100%	5%	75%	20%

See "Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Best-Effort Traffic on Links with Ethernet PAUSE Enabled" on page 667 for an example that shows you how to configure the recommended buffer settings shown in Table 117 on page 652 and Table 118 on page 652.

Best-Effort Multicast (Multidestination) Traffic

If your network carries mostly best-effort (lossy) multicast traffic, then the default shared buffer configuration allocates too much buffer space to support lossless transport. Instead of wasting those buffers, we recommend that you use the following ingress shared buffer settings (see Table 121 on page 653) and egress shared buffer settings (see Table 122 on page 653):

Table 121: Recommended Ingress Shared Buffer Configuration for Networks with Mostly Best -Effort Multicast Traffic

Total Shared Ingress Buffer	Lossless Buffer	Lossless-Headroom Buffer	Lossy Buffer
100%	5%	0%	95%

Table 122: Recommended Egress Shared Buffer Configuration for Networks with Mostly Best-Effort Multicast Traffic

Total Shared Egress Buffer	Lossless Buffer	Lossy Buffer	Multicast Buffer
100%	5%	20%	75%

See "Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Multicast Traffic" on page 675 for an example that shows you how to configure the recommended buffer settings shown in Table 121 on page 653 and Table 122 on page 653.

Lossless Traffic

If your network carries mostly lossless traffic, then the default shared buffer configuration allocates too much buffer space to support best-effort traffic. Instead of wasting those buffers, we recommend that you use the following ingress shared buffer settings (see Table 123 on page 654) and egress shared buffer settings (see Table 124 on page 654):

Table 123: Recommended Ingress Shared Buffer Configuration for Networks with Mostly Lossless Traffic

Total Shared Ingress Buffer	Lossless Buffer	Lossless-Headroom Buffer	Lossy Buffer
100%	15%	80%	5%

Table 124: Recommended Egress Shared Buffer Configuration for Networks with Mostly Lossless Traffic

Total Shared Egress Buffer	Lossless Buffer	Lossy Buffer	Multicast Buffer
100%	90%	5%	5%

See "Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Lossless Traffic" on page 683 for an example that shows you how to configure the recommended buffer settings shown in Table 123 on page 654 and Table 124 on page 654.

Optimizing Buffer Configuration

Starting from the default configuration or from a recommended buffer configuration, you can further optimize the buffer allocation to best support the mix of traffic on your network. Adjust the settings gradually to fine-tune the shared buffer allocation. Use caution when adjusting the shared buffer configuration, not just when you fine-tune the ingress and egress buffer partitions, but also when you fine-tune the total ingress and egress shared buffer percentage. (Remember that if you allocate less than 100 percent of the available buffers to the shared buffers, the remaining buffers are added to the dedicated buffers). Tuning the buffers incorrectly can cause problems such as ingress port congestion.

> **CAUTION:** Changing the buffer configuration is a disruptive event. Traffic stops on *all* ports until buffer reprogramming is complete.

The relationship between the sizes of the ingress buffer pool and the egress buffer pool affects when and where packets are dropped. The buffer pool sizes include the shared buffers and the dedicated buffers. In general, if there are more ingress buffers than egress buffers, the switch can experience ingress port congestion because egress queues fill before ingress queues can empty.

Use the **show class-of-service shared-buffer** operational command to see the sizes in kilobytes (KB) of the dedicated and shared buffers and of the shared buffer partitions.

For best-effort traffic (unicast and multidestination), the combined ingress lossy shared buffer partition and ingress dedicated buffers must be *less than* the combined egress lossy and multicast shared buffer partitions plus the egress dedicated buffers. This prevents ingress port congestion by ensuring that egress best-effort buffers are deeper than ingress best-effort buffers, and ensures that if packets are dropped, they are dropped at the egress queues. (Packets dropping at the ingress prevents the egress schedulers from working properly.)

For lossless traffic (traffic on which you enable PFC), the combined ingress lossless shared buffer partition and a reasonable portion of the ingress headroom buffer partition, plus the dedicated buffers, must be *less than* the total egress lossless shared buffer partition and dedicated buffers. (A reasonable portion of the ingress headroom buffer is approximately 20 to 25 percent of the buffer space, but this varies depending on how much buffer headroom is required to support the lossless traffic.) When these conditions are met, if there is ingress port congestion, the ingress port congestion triggers PFC on the ingress port to prevent packet loss. If the total lossless ingress buffers exceed the total lossless egress buffers, packets could be dropped at the egress instead of PFC being applied at the ingress to prevent packet loss.

NOTE: If you commit a buffer configuration for which the switch does not have sufficient resources, the switch might log an error instead of returning a commit error. In that case, a syslog message is displayed on the console. For example:

```
user@host# commit
configuration check succeeds

Message from syslogd@host at Jun 13 11:11:10  ...
host dc-pfe: Not enough Ingress Lossless headroom.(Already allocated more).
 Dedicated : 14340 Lossy : 47100 Lossless 4239 Headroom 21195 Avail : 20781
commit complete
```

If the buffer configuration commits but you receive a syslog message that indicates the configuration cannot be implemented, you can:

- Reconfigure the buffers or reconfigure other parameters (for example, the PFC configuration, which affects the need for lossless headroom buffers and lossless buffers—the more priorities you pause, the more lossless and lossless headroom buffer space you need), then attempt the commit operation again.
- Roll back the switch to the last successful configuration.

If you receive a syslog message that says the buffer configuration cannot be implemented, you must take corrective action. If you do not fix the configuration or roll back to a previous successful configuration, the system behavior is unpredictable.

General Buffer Configuration Rules and Considerations

Keep the following rules and considerations in mind when you configure the buffers:

- Changing the buffer configuration is a disruptive event. Traffic stops on *all* ports until buffer reprogramming is complete.
- If you configure the ingress or egress shared buffer percentages as less than 100 percent, the remaining percentage of buffer space is added to the dedicated buffer pool.
- The sum of all of the ingress shared buffer partitions must equal 100 percent. Each partition must be configured with a value of at least 5 percent except the lossless headroom buffer, which can have a value of 0 percent.
- The sum of all of the egress shared buffer partitions must equal 100 percent. Each partition must be configured with a value of at least 5 percent.
- Lossless and lossless headroom shared buffers serve traffic on which you enable PFC, and do not serve traffic subject to Ethernet PAUSE.
- The switch uses the dedicated buffer pool first and the shared buffer pool only after the dedicated buffer pool for a port or queue is exhausted.
- Too little dedicated buffer space results in too much competition for shared buffer space.
- Too much dedicated buffer space results in poorer burst absorption because there is less available shared buffer space.
- Always check the syslog messages after you commit a new buffer configuration.
- The optimal buffer configuration for your network depends on the types of traffic on the network. If your network carries less traffic of a certain type (for example, lossless traffic), then you can reduce the size of the buffers allocated to that type of traffic (for example, you can reduce the sizes of the lossless and lossless headroom buffers).

RELATED DOCUMENTATION

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Best-Effort Unicast Traffic | 659

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Best-Effort Traffic on Links with Ethernet PAUSE Enabled | 667

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Multicast Traffic | 675

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Lossless Traffic | 683

Example: Configuring Queue Schedulers | 325

Configuring Global Ingress and Egress Shared Buffers

Although the switch reserves some buffer space to ensure a minimum memory allocation for ports and queues, you can configure how the system uses the rest of the buffer space to optimize the buffer allocation for your particular mix of network traffic. The global shared buffer pool is memory space that all of the ports on the switch share dynamically as they need buffers. You can allocate global shared memory space to different types of ingress and egress buffers to better support different mixes of network traffic.

> **CAUTION:** Changing the buffer configuration is a disruptive event. Traffic stops on *all* ports until buffer reprogramming is complete.

Use the default shared buffer settings (for a network with a balanced mix of lossless, best-effort, and multicast traffic) or one of the recommended shared buffer configurations for your mix of network traffic (mostly best-effort unicast traffic, mostly best-effort traffic on links enabled for Ethernet PAUSE, mostly multicast traffic, or mostly lossless traffic). Either the default configuration or one of the recommended configurations provides a buffer allocation that satisfies the needs of most networks.

After starting from one of the recommended configurations, you can fine-tune the shared buffer settings, but do so with caution to prevent traffic loss due to buffer misconfiguration.

You can configure the percentage of available (user-configurable) buffer space allocated to the global shared buffers. Any space that you do not allocate to the global shared buffer pool is added to the dedicated buffer pool. The default configuration allocates 100 percent of the available buffer space to the global shared buffers.

You can partition the ingress and egress shared buffer pools to allocate more buffers to the types of traffic your network predominantly carries, and fewer buffers to other traffic. From the buffer space allocated to the ingress shared buffer pool, you can allocate space to:

- Lossless buffers—Percentage of shared buffer pool for all lossless ingress traffic. The minimum value for the lossless buffers is 5 percent.
- Lossless headroom buffers—Percentage of shared buffer pool for packets received while a pause is asserted. If Ethernet PAUSE is configured on a port or if priority-based flow control (PFC) is configured on priorities on a port, when the port sends a pause message to the connected peer, the port uses the headroom buffers to store the packets that arrive between the time the port sends the pause message and the time the last packet arrives after the peer pauses traffic. The minimum value for the lossless headroom buffers is 0 (zero) percent. (Lossless headroom buffers are the only buffers that can have a minimum value of less than 5 percent.)

- Lossy buffers—Percentage of shared buffer pool for all best-effort ingress traffic (best-effort unicast, multidestination, and strict-high priority traffic). The minimum value for the lossy buffers is 5 percent.

The combined percentage values of the ingress lossless, lossless headroom, and lossy buffer partitions must total exactly 100 percent. If the buffer percentages total more than 100 percent or less than 100 percent, the switch returns a commit error. All ingress buffer partitions must be explicitly configured, even when the lossless headroom buffer partition has a value of 0 (zero) percent.

From the buffer space allocated to the egress shared buffer pool, you can allocate space to:

- Lossless buffers—Percentage of shared buffer pool for all lossless egress queues. The minimum value for the lossless buffers is 5 percent.
- Lossy buffers—Percentage of shared buffer pool for all best-effort egress queues (best-effort unicast and strict-high priority queues). The minimum value for the lossy buffers is 5 percent.
- Multicast buffers—Percentage of shared buffer pool for all multidestination (multicast, broadcast, and destination lookup fail) egress queues. The minimum value for the multicast buffers is 5 percent.

The combined percentage values of the egress lossless, lossy, and multicast buffer partitions must total exactly 100 percent. If the buffer percentages total more than 100 percent or less than 100 percent, the switch returns a commit error. All egress buffer partitions must be explicitly configured and must have a value of at least 5 percent.

To configure the shared buffer allocation and partitioning using the CLI:

1. Configure the percentage of available (nonreserved) buffers used for the ingress global shared buffer pool:

   ```
   [edit class-of-service shared-buffer]
   user@switch# set ingress percent percent
   ```

2. Configure the global ingress buffer partitions for lossless, lossless-headroom, and lossy traffic:

   ```
   [edit class-of-service shared-buffer]
   user@switch# set ingress buffer-partition lossless percent percent
   user@switch# set ingress buffer-partition lossless-headroom percent percent
   user@switch# set ingress buffer-partition lossy percent percent
   ```

3. Configure the percentage of available (nonreserved) buffers used for the egress global shared buffer pool:

   ```
   [edit class-of-service shared-buffer]
   user@switch# set egress percent percent
   ```

4. Configure the global egress buffer partitions for lossless, lossy, and multicast queues:

   ```
   [edit class-of-service shared-buffer]
   user@switch# set egress buffer-partition lossless percent percent
   user@switch# set egress buffer-partition lossy percent percent
   user@switch# set egress buffer-partition multicast percent percent
   ```

RELATED DOCUMENTATION

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Best-Effort Unicast Traffic

IN THIS SECTION

Although the switch reserves some buffer space to ensure a minimum memory allocation for ports and queues, you can configure how the system uses the rest of the buffer space to optimize the buffer allocation for your particular mix of network traffic.

This example shows you the recommended configuration of the global shared buffer pool to support a network that carries mostly best-effort (lossy) unicast traffic. The global shared buffer pool is memory space that all of the ports on the switch share dynamically as they need buffers. You can allocate global shared memory space to different types of buffers to better support different mixes of network traffic.

> **CAUTION:** Changing the buffer configuration is a disruptive event. Traffic stops on *all* ports until buffer reprogramming is complete.

Use the default shared buffer settings (for a network with a balanced mix of lossless, best effort, and multicast traffic) or one of the recommended shared buffer configurations for your mix of network traffic (mostly best-effort unicast traffic, mostly best-effort traffic on links enabled for Ethernet PAUSE, mostly multicast traffic, or mostly lossless traffic). Either the default configuration or one of the recommended configurations provides a buffer allocation that satisfies the needs of most networks.

> **NOTE:** OCX Series switches do not support lossless transport.

After starting from the recommended configuration, you can fine-tune the shared buffer settings, but do so with caution to prevent traffic loss due to buffer misconfiguration.

Requirements

This example uses the following hardware and software components:

- One switch (this example was tested on a Juniper Networks QFX3500 Switch)
- Junos OS Release 12.3 or later for the QFX Series or Junos OS Release 14.1X53-D20 or later for the OCX Series

Overview

You can configure the percentage of available (user-configurable) buffer space allocated to the global shared buffers. Any space that you do not allocate to the global shared buffer pool is added to the dedicated buffer pool. The default configuration allocates 100 percent of the available buffer space to the global shared buffers.

You can partition the ingress and egress shared buffer pools to allocate more buffers to the types of traffic your network predominantly carries, and fewer buffers to other traffic. From the buffer space allocated to the ingress shared buffer pool, you can allocate space to:

- Lossless buffers—Percentage of shared buffer pool for all lossless ingress traffic. The minimum value for the lossless buffers is 5 percent.

- Lossless headroom buffers—Percentage of shared buffer pool for packets received while a pause is asserted. If Ethernet PAUSE is configured on a port or if priority-based flow control (PFC) is configured on priorities on a port, when the port sends a pause message to the connected peer, the port uses the headroom buffers to store the packets that arrive between the time the port sends the pause message and the time the last packet arrives after the peer pauses traffic. The minimum value for the lossless headroom buffers is 0 (zero) percent. (Lossless headroom buffers are the only buffers that can have a minimum value of less than 5 percent.)
- Lossy buffers—Percentage of shared buffer pool for all best-effort ingress traffic (best-effort unicast, multidestination, and strict-high priority traffic). The minimum value for the lossy buffers is 5 percent.

The combined percentage values of the ingress lossless, lossless headroom, and lossy buffer partitions must total exactly 100 percent. If the buffer percentages total more than 100 percent or less than 100 percent, the switch returns a commit error. All ingress buffer partitions must be explicitly configured, even when the lossless headroom buffer partition has a value of 0 (zero) percent.

From the buffer space allocated to the egress shared buffer pool, you can allocate space to:

- Lossless buffers—Percentage of shared buffer pool for all lossless egress queues. The minimum value for the lossless buffers is 5 percent.
- Lossy buffers—Percentage of shared buffer pool for all best-effort egress queues (best-effort unicast, and strict-high priority queues). The minimum value for the lossy buffers is 5 percent.
- Multicast buffers—Percentage of shared buffer pool for all multidestination (multicast, broadcast, and destination lookup fail) egress queues. The minimum value for the multicast buffers is 5 percent.

The combined percentage values of the egress lossless, lossy, and multicast buffer partitions must total exactly 100 percent. If the buffer percentages total more than 100 percent or less than 100 percent, the switch returns a commit error. All egress buffer partitions must be explicitly configured and must have a value of at least 5 percent.

To configure the shared buffers to support a network that carries mostly best-effort unicast traffic, more buffer space needs to be allocated to lossy buffers, and less buffer space should be allocated to lossless buffers. This example shows you how to configure the global shared buffer pool allocation that we recommend to support a network that carries mostly unicast traffic.

Topology

Table 125 on page 661 shows the configuration components for this example.

Table 125: Components of the Recommended Shared Buffer Configuration for Best-Effort Unicast Network Topologies

Component	Settings
Hardware	QFX3500 switch

Table 125: Components of the Recommended Shared Buffer Configuration for Best-Effort Unicast Network Topologies *(continued)*

Component	Settings
Ingress shared buffer	Percentage of available ingress buffer space allocated to the ingress shared buffer: 100% Percentage of ingress buffer space allocated to lossless traffic (lossless buffer partition): 5% Percentage of ingress buffer space allocated to lossless headroom traffic (lossless-headroom buffer partition): 0% Percentage of ingress buffer space allocated to best-effort traffic (lossy buffer partition): 95%
Egress shared buffer	Percentage of available egress buffer space allocated to the egress shared buffer: 100% Percentage of egress buffer space allocated to lossless queues (lossless buffer partition): 5% Percentage of egress buffer space allocated to best-effort queues (lossy buffer partition): 75% Percentage of egress buffer space allocated to multicast traffic (multicast buffer partition): 20%

Configuration

CLI Quick Configuration

To quickly configure the recommended shared buffer settings for networks that carry mostly best-effort unicast traffic, 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 class-of-service shared-buffer]** hierarchy level:

```
[edit class-of-service shared-buffer]
set ingress percent 100
set ingress buffer-partition lossless percent 5
set ingress buffer-partition lossless-headroom percent 0
set ingress buffer-partition lossy percent 95
set egress percent 100
set egress buffer-partition lossless percent 5
set egress buffer-partition lossy percent 75
set egress buffer-partition multicast percent 20
```

Configuring the Global Shared Buffer Pool for Networks with Mostly Best-Effort Unicast Traffic

Step-by-Step Procedure

To configure the global ingress and egress shared buffer allocations and partitions for a network that carries mostly best-effort unicast traffic:

1. Configure the percentage of available (nonreserved) buffers used for the ingress global shared buffer pool:

   `[edit class-of-service shared-buffer]`
   `user@switch#` **set ingress percent 100**

2. Configure the global ingress buffer partitions for lossless, lossless-headroom, and lossy traffic:

   `[edit class-of-service shared-buffer]`
   `user@switch#` **set ingress buffer-partition lossless percent 5**
   `user@switch#` **set ingress buffer-partition lossless-headroom percent 0**
   `user@switch#` **set ingress buffer-partition lossy percent 95**

3. Configure the percentage of available (nonreserved) buffers used for the egress global shared buffer pool:

   `[edit class-of-service shared-buffer]`
   `user@switch#` **set egress percent 100**

4. Configure the global egress buffer partitions for lossless, lossy, and multicast queues:

   `[edit class-of-service shared-buffer]`
   `user@switch#` **set egress buffer-partition lossless percent 5**
   `user@switch#` **set egress buffer-partition lossy percent 75**
   `user@switch#` **set egress buffer-partition multicast percent 20**

Results

Display the results of the configuration:

```
root@dcbg-tp-pa-02> show configuration class-of-service shared-buffer
ingress {
    percent 100;
    buffer-partition lossless {
        percent 5;
```

```
    }
    buffer-partition lossy {
        percent 95;
    }
    buffer-partition lossless-headroom {
        percent 0;
    }
}
egress {
    percent 100;
    buffer-partition lossless {
        percent 5;
    }
    buffer-partition lossy {
        percent 75;
    }
    buffer-partition multicast {
        percent 20;
    }
}
```

Verification

Verify that you correctly configured the shared buffer.

Verifying the Shared Buffer Configuration

Purpose

Verify that the ingress and egress global shared buffer pools are correctly configured and partitioned among the shared buffer types.

Action

List the global shared buffer configuration using the operational mode command **show class-of-service shared-buffer**:

`user@switch>` **show class-of-service shared-buffer**

```
root@dcbg-tp-pa-02> show class-of-service shared-buffer
Ingress:
  Total Buffer     :  9360.00 KB
  Dedicated Buffer :  2158.00 KB
  Shared Buffer    :  7202.00 KB
    Lossless          :  360.10 KB
```

```
    Lossless Headroom :  0.00 KB
    Lossy             :  6841.90 KB

  Lossless Headroom Utilization:
  Node Device         Total          Used                 Free
  0                   0.00 KB        0.00 KB              0.00 KB

Egress:
  Total Buffer     :  9360.00 KB
  Dedicated Buffer :  2704.00 KB
  Shared Buffer    :  6656.00 KB
    Lossless          :  332.80 KB
    Multicast         :  1331.20 KB
    Lossy             :  4992.00 KB
```

Meaning

The **show class-of-service shared-buffer** operational command shows all of the ingress and egress global shared buffer settings, including the buffer partitioning.

For the ingress shared buffers, the command output shows:

- The total switch buffer pool is 9360 KB (9 MB).
- The dedicated buffer pool is 2158 KB. This is the size of the global ingress dedicated buffer pool when you configure the ingress shared buffer pool as 100 percent of the available (user-configurable) buffer space. This is the minimum size of the reserved, ingress dedicated ingress buffer pool (not user-configurable). If you configure the shared buffer as less than 100 percent of the available buffer pool, the remaining buffer space is added to the dedicated buffer pool.
- With the ingress shared buffer pool configured as 100 percent of the available buffers, the total size of the ingress shared buffer pool is 7202 KB.
- The ingress shared buffer pool is partitioned to allocate:
  - 360.10 KB to lossless traffic
  - No space to lossless headroom traffic
  - 6841.90 KB to lossy unicast traffic
- The Lossless Headroom Utilization field shows how much of the buffer space reserved for paused traffic is used. Because the lossless headroom buffer partition is set to 0 (zero) percent, the total amount of lossless headroom buffer space is 0 KB; therefore the amount of used and free lossless headroom buffer space is also 0 KB.

For the egress shared buffers, the command output shows:

- The total switch buffer pool is 9360 KB (9 MB).
- The dedicated buffer pool is 2704 KB. This is the size of the global egress dedicated buffer pool when you configure the egress shared buffer pool as 100 percent of the available (user-configurable) buffer space. This is the minimum size of the reserved, egress dedicated buffer pool (not user-configurable). If you configure the shared buffer as less than 100 percent of the available buffer pool, the remaining buffer space is added to the dedicated buffer pool.
- With the egress shared buffer pool configured as 100 percent of the available buffers, the total size of the egress shared buffer pool is 6656 KB. This is less than the ingress shared buffer pool because the switch reserves more egress dedicated buffer space than ingress dedicated buffer space. (More dedicated buffer space means less shared buffer space, and more shared buffer space means less dedicated buffer space.)
- The egress shared buffer pool is partitioned to allocate:
  - 332.80 KB to lossless traffic
  - 1331.20 KB to multicast traffic
  - 4992 KB to lossy unicast traffic

> **NOTE:** The output values are valid for QFX3500 and QFX3600 switches. QFX5100, EX4600, and OCX Series switches have larger buffers (12 MB instead of 9 MB), so the total buffer size and the sizes of each buffer partition are larger on those switches.

RELATED DOCUMENTATION

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Best-Effort Traffic on Links with Ethernet PAUSE Enabled | 667

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Multicast Traffic | 675

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Lossless Traffic | 683

Configuring Global Ingress and Egress Shared Buffers | 657

Understanding CoS Buffer Configuration | 635

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Best-Effort Traffic on Links with Ethernet PAUSE Enabled

IN THIS SECTION

Although the switch reserves some buffer space to ensure a minimum memory allocation for ports and queues, you can configure how the system uses the rest of the buffer space to optimize the buffer allocation for your particular mix of network traffic.

This example shows you the recommended configuration of the global shared buffer pool to support a network that carries mostly best-effort (lossy) traffic on links with Ethernet PAUSE (IEEE 802.3X) enabled.

NOTE: OCX Series switches support symmetric Ethernet PAUSE flow control, but do not support asymmetric Ethernet PAUSE flow control.

The global shared buffer pool is memory space that all of the ports on the switch share dynamically as they need buffers. You can allocate global shared memory space to different types of buffers to better support different mixes of network traffic.

CAUTION: Changing the buffer configuration is a disruptive event. Traffic stops on *all* ports until buffer reprogramming is complete.

Use the default shared buffer settings (for a network with a balanced mix of lossless, best effort, and multicast traffic) or one of the recommended shared buffer configurations for your mix of network traffic (mostly best-effort unicast traffic, mostly best-effort traffic on links enabled for Ethernet PAUSE, mostly multicast traffic, or mostly lossless traffic). Either the default configuration or one of the recommended configurations provides a buffer allocation that satisfies the needs of most networks.

After starting from the recommended configuration, you can fine-tune the shared buffer settings, but do so with caution to prevent traffic loss due to buffer misconfiguration.

Requirements

This example uses the following hardware and software components:

- One switch (this example was tested on a Juniper Networks QFX3500 Switch)
- Junos OS Release 12.3 or later for the QFX Series or Junos OS Release 14.1X53-D20 or later for the OCX Series

Overview

You can configure the percentage of available (user-configurable) buffer space allocated to the global shared buffers. Any space that you do not allocate to the global shared buffer pool is added to the dedicated buffer pool. The default configuration allocates 100 percent of the available buffer space to the global shared buffers.

You can partition the ingress and egress shared buffer pools to allocate more buffers to the types of traffic your network predominantly carries, and fewer buffers to other traffic. From the buffer space allocated to the ingress shared buffer pool, you can allocate space to:

- Lossless buffers—Percentage of shared buffer pool for all lossless ingress traffic. The minimum value for the lossless buffers is 5 percent.
- Lossless headroom buffers—Percentage of shared buffer pool for packets received while a pause is asserted. If Ethernet PAUSE is configured on a port or if priority-based flow control (PFC) is configured on priorities on a port, when the port sends a pause message to the connected peer, the port uses the headroom buffers to store the packets that arrive between the time the port sends the pause message and the time the last packet arrives after the peer pauses traffic. The minimum value for the lossless headroom buffers is 0 (zero) percent. (Lossless headroom buffers are the only buffers that can have a minimum value of less than 5 percent.)

  NOTE: OCX Series switches do not support PFC.

- Lossy buffers—Percentage of shared buffer pool for all best-effort ingress traffic (best-effort unicast, multidestination, and strict-high priority traffic). The minimum value for the lossy buffers is 5 percent.

The combined percentage values of the ingress lossless, lossless headroom, and lossy buffer partitions must total exactly 100 percent. If the buffer percentages total more than 100 percent or less than 100 percent, the switch returns a commit error. All ingress buffer partitions must be explicitly configured, even when the lossless headroom buffer partition has a value of 0 (zero) percent.

From the buffer space allocated to the egress shared buffer pool, you can allocate space to:

- Lossless buffers—Percentage of shared buffer pool for all lossless egress queues. The minimum value for the lossless buffers is 5 percent.
- Lossy buffers—Percentage of shared buffer pool for all best-effort egress queues (best-effort unicast and strict-high priority queues). The minimum value for the lossy buffers is 5 percent.
- Multicast buffers—Percentage of shared buffer pool for all multidestination (multicast, broadcast, and destination lookup fail) egress queues. The minimum value for the multicast buffers is 5 percent.

The combined percentage values of the egress lossless, lossy, and multicast buffer partitions must total exactly 100 percent. If the buffer percentages total more than 100 percent or less than 100 percent, the switch returns a commit error. All egress buffer partitions must be explicitly configured and must have a value of at least 5 percent.

To configure the shared buffers to support a network that carries mostly best-effort traffic on links enabled for Ethernet PAUSE, more buffer space needs to be allocated to ingress dedicated port buffers, and less buffer space should be allocated to ingress shared buffers. Also, more buffer space needs to be allocated to lossless-headroom buffers, and less space to ingress lossy buffers. This example shows you how to configure the global shared buffer pool allocation that we recommend to support a network that carries mostly best-effort traffic on links enabled for Ethernet PAUSE.

Topology

Table 126 on page 669 shows the configuration components for this example.

Table 126: Components of the Recommended Shared Buffer Configuration for Best-Effort Network Topologies with Links Enabled for Ethernet PAUSE

Component	Settings
Hardware	QFX3500 switch
Ingress shared buffer	Percentage of available ingress buffer space allocated to the ingress shared buffer: 70% Percentage of ingress buffer space allocated to lossless traffic (lossless buffer partition): 5% Percentage of ingress buffer space allocated to lossless headroom traffic (lossless-headroom buffer partition): 80% Percentage of ingress buffer space allocated to best-effort traffic (lossy buffer partition): 15%

Table 126: Components of the Recommended Shared Buffer Configuration for Best-Effort Network Topologies with Links Enabled for Ethernet PAUSE *(continued)*

Component	Settings
Egress shared buffer	Percentage of available egress buffer space allocated to the egress shared buffer: 100% Percentage of egress buffer space allocated to lossless queues (lossless buffer partition): 5% Percentage of egress buffer space allocated to best-effort queues (lossy buffer partition): 75% Percentage of egress buffer space allocated to multicast traffic (multicast buffer partition): 20%

Configuration

CLI Quick Configuration

To quickly configure the recommended shared buffer settings for networks that carry mostly best-effort unicast traffic, 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 class-of-service shared-buffer]** hierarchy level:

```
[edit class-of-service shared-buffer]
set ingress percent 70
set ingress buffer-partition lossless percent 5
set ingress buffer-partition lossless-headroom percent 80
set ingress buffer-partition lossy percent 15
set egress percent 100
set egress buffer-partition lossless percent 5
set egress buffer-partition lossy percent 75
set egress buffer-partition multicast percent 20
```

Configuring the Global Shared Buffer Pool for Networks with Mostly Best-Effort Traffic on Links Enabled for Ethernet PAUSE

Step-by-Step Procedure

To configure the global ingress and egress shared buffer allocations and partitions:

1. Configure the percentage of available (nonreserved) buffers used for the ingress global shared buffer pool:

   ```
   [edit class-of-service shared-buffer]
   user@switch# set ingress percent 70
   ```

2. Configure the global ingress buffer partitions for lossless, lossless-headroom, and lossy traffic:

   ```
   [edit class-of-service shared-buffer]
   user@switch# set ingress buffer-partition lossless percent 5
   user@switch# set ingress buffer-partition lossless-headroom percent 80
   user@switch# set ingress buffer-partition lossy percent 15
   ```

3. Configure the percentage of available (nonreserved) buffers used for the egress global shared buffer pool:

   ```
   [edit class-of-service shared-buffer]
   user@switch# set egress percent 100
   ```

4. Configure the global egress buffer partitions for lossless, lossy, and multicast queues:

   ```
   [edit class-of-service shared-buffer]
   user@switch# set egress buffer-partition lossless percent 5
   user@switch# set egress buffer-partition lossy percent 75
   user@switch# set egress buffer-partition multicast percent 20
   ```

Results

Display the results of the configuration:

```
root@dcbg-tp-pa-02> show configuration class-of-service shared-buffer
ingress {
    percent 70;
    buffer-partition lossless {
        percent 5;
    }
    buffer-partition lossy {
        percent 15;
```

```
    }
    buffer-partition lossless-headroom {
        percent 80;
    }
}
egress {
    percent 100;
    buffer-partition lossless {
        percent 5;
    }
    buffer-partition lossy {
        percent 75;
    }
    buffer-partition multicast {
        percent 20;
    }
}
```

Verification

Verify that you correctly configured the shared buffer.

Verifying the Shared Buffer Configuration

Purpose

Verify that the ingress and egress global shared buffer pools are correctly configured and partitioned among the shared buffer types.

Action

List the global shared buffer configuration using the operational mode command **show class-of-service shared-buffer**:

`user@switch>` **show class-of-service shared-buffer**

```
root@dcbg-tp-pa-02> show class-of-service shared-buffer
Ingress:
  Total Buffer     :  9360.00 KB
  Dedicated Buffer :  4318.60 KB
  Shared Buffer    :  5041.40 KB
    Lossless          :  252.07 KB
    Lossless Headroom :  4033.12 KB
    Lossy             :  756.21 KB
```

```
Egress:
  Total Buffer     :  9360.00 KB
  Dedicated Buffer :  2704.00 KB
  Shared Buffer    :  6656.00 KB
    Lossless          :  332.80 KB
    Multicast         :  1331.20 KB
    Lossy             :  4992.00 KB
```

Meaning

The **show class-of-service shared-buffer** operational command shows all of the ingress and egress global shared buffer settings, including the buffer partitioning.

For the ingress shared buffers, the command output shows:

- The total switch buffer pool is 9360 KB (9 MB).
- The dedicated buffer pool is 4318.6 KB. This is the size of the global ingress dedicated buffer pool when you configure the ingress shared buffer pool as 70 percent of the available (user-configurable) buffer space.
- With the ingress shared buffer pool configured as 70 percent of the available buffers, the total size of the ingress shared buffer pool is 5041.4 KB.
- The ingress shared buffer pool is partitioned to allocate:
  - 252.07 KB to lossless traffic
  - 4033.12 KB to lossless headroom traffic
  - 756.21 KB to lossy unicast traffic

For the egress shared buffers, the command output shows:

- The total switch buffer pool is 9360 KB (9 MB).
- The dedicated buffer pool is 2704 KB. This is the size of the global egress dedicated buffer pool when you configure the egress shared buffer pool as 100 percent of the available (user-configurable) buffer space. This is the minimum size of the reserved, egress dedicated buffer pool (not user-configurable). If you configure the shared buffer as less than 100 percent of the available buffer pool, the remaining buffer space is added to the dedicated buffer pool.
- With the egress shared buffer pool configured as 100 percent of the available buffers, the total size of the egress shared buffer pool is 6656 KB. This is less than the ingress shared buffer pool because the switch reserves more egress dedicated buffer space than ingress dedicated buffer space. (More dedicated buffer space means less shared buffer space, and more shared buffer space means less dedicated buffer space.)
- The egress shared buffer pool is partitioned to allocate:

- 332.80 KB to lossless traffic
- 1331.20 KB to multicast traffic
- 4992 KB to lossy unicast traffic

> **NOTE:** The output values are valid for QFX3500 and QFX3600 switches. QFX5100, EX4600, and OCX Series switches have larger buffers (12 MB instead of 9 MB), so the total buffer size and the sizes of each buffer partition are larger on those switches.

RELATED DOCUMENTATION

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Best-Effort Unicast Traffic | 659

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Multicast Traffic | 675

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Lossless Traffic | 683

Configuring Global Ingress and Egress Shared Buffers | 657

Understanding CoS Buffer Configuration | 635

CHAPTER 19

Shared Buffer Pool

IN THIS CHAPTER

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Multicast Traffic

IN THIS SECTION

Although the switch reserves some buffer space to ensure a minimum memory allocation for ports and queues, you can configure how the system uses the rest of the buffer space to optimize the buffer allocation for your particular mix of network traffic.

This example shows you the recommended configuration of the global shared buffer pool to support a network that carries mostly multicast traffic. The global shared buffer pool is memory space that all of the ports on the switch share dynamically as they need buffers. You can allocate global shared memory space to different types of buffers to better support different mixes of network traffic.

> **CAUTION:** Changing the buffer configuration is a disruptive event. Traffic stops on *all* ports until buffer reprogramming is complete.

Use the default shared buffer settings (for a network with a balanced mix of lossless, best effort, and multicast traffic) or one of the recommended shared buffer configurations for your mix of network traffic (mostly best-effort unicast traffic, mostly best-effort traffic on links enabled for Ethernet PAUSE, mostly multicast traffic, or mostly lossless traffic). Either the default configuration or one of the recommended configurations provides a buffer allocation that satisfies the needs of most networks.

After starting from the recommended configuration, you can fine-tune the shared buffer settings, but do so with caution to prevent traffic loss due to buffer misconfiguration.

Requirements

This example uses the following hardware and software components:

- One switch (this example was tested on a Juniper Networks QFX3500 Switch)
- Junos OS Release 12.3 or later for the QFX Series or Junos OS Release 14.1X53-D20 or later for the OCX Series

Overview

You can configure the percentage of available (user-configurable) buffer space allocated to the global shared buffers. Any space that you do not allocate to the global shared buffer pool is added to the dedicated buffer pool. The default configuration allocates 100 percent of the available buffer space to the global shared buffers.

You can partition the ingress and egress shared buffer pools to allocate more buffers to the types of traffic your network predominantly carries, and fewer buffers to other traffic. From the buffer space allocated to the ingress shared buffer pool, you can allocate space to:

- Lossless buffers—Percentage of shared buffer pool for all lossless ingress traffic. The minimum value for the lossless buffers is 5 percent.
- Lossless headroom buffers—Percentage of shared buffer pool for packets received while a pause is asserted. If Ethernet PAUSE is configured on a port or if priority-based flow control (PFC) is configured on priorities on a port, when the port sends a pause message to the connected peer, the port uses the headroom buffers to store the packets that arrive between the time the port sends the pause message and the time the last packet arrives after the peer pauses traffic. The minimum value for the lossless headroom buffers is 0 (zero) percent. (Lossless headroom buffers are the only buffers that can have a minimum value of less than 5 percent.)

- Lossy buffers—Percentage of shared buffer pool for all best-effort ingress traffic (best-effort unicast, multidestination, and strict-high priority traffic). The minimum value for the lossy buffers is 5 percent.

The combined percentage values of the ingress lossless, lossless headroom, and lossy buffer partitions must total exactly 100 percent. If the buffer percentages total more than 100 percent or less than 100 percent, the switch returns a commit error. All ingress buffer partitions must be explicitly configured, even when the lossless headroom buffer partition has a value of 0 (zero) percent.

From the buffer space allocated to the egress shared buffer pool, you can allocate space to:

- Lossless buffers—Percentage of shared buffer pool for all lossless egress queues. The minimum value for the lossless buffers is 5 percent.
- Lossy buffers—Percentage of shared buffer pool for all best-effort egress queues (best-effort unicast, and strict-high priority queues). The minimum value for the lossy buffers is 5 percent.
- Multicast buffers—Percentage of shared buffer pool for all multidestination (multicast, broadcast, and destination lookup fail) egress queues. The minimum value for the multicast buffers is 5 percent.

The combined percentage values of the egress lossless, lossy, and multicast buffer partitions must total exactly 100 percent. If the buffer percentages total more than 100 percent or less than 100 percent, the switch returns a commit error. All egress buffer partitions must be explicitly configured and must have a value of at least 5 percent.

To configure the shared buffers to support a network that carries mostly multicast traffic, more buffer space needs to be allocated to lossy buffers, less buffer space should be allocated to lossless buffers, and more space needs to be allocated to egress multicast buffers. This example shows you how to configure the global shared buffer pool allocation that we recommend to support a network that carries mostly multicast traffic.

Topology

Table 127 on page 677 shows the configuration components for this example.

Table 127: Components of the Recommended Shared Buffer Configuration for Multicast Network Topologies

Component	Settings
Hardware	QFX3500 switch

Table 127: Components of the Recommended Shared Buffer Configuration for Multicast Network Topologies *(continued)*

Component	Settings
Ingress shared buffer	Percentage of available ingress buffer space allocated to the ingress shared buffer: 100% Percentage of ingress buffer space allocated to lossless traffic (lossless buffer partition): 5% Percentage of ingress buffer space allocated to lossless headroom traffic (lossless-headroom buffer partition): 0% Percentage of ingress buffer space allocated to best-effort traffic (lossy buffer partition): 95%
Egress shared buffer	Percentage of available egress buffer space allocated to the egress shared buffer: 100% Percentage of egress buffer space allocated to lossless queues (lossless buffer partition): 5% Percentage of egress buffer space allocated to best-effort queues (lossy buffer partition): 20% Percentage of egress buffer space allocated to multicast traffic (multicast buffer partition): 75%

Configuration

CLI Quick Configuration

To quickly configure the recommended shared buffer settings for networks that carry mostly multicast traffic, 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 class-of-service shared-buffer]** hierarchy level:

```
[edit class-of-service shared-buffer]
```

set ingress percent 100

set ingress buffer-partition lossless percent 5

set ingress buffer-partition lossless-headroom percent 0

set ingress buffer-partition lossy percent 95

set egress percent 100

set egress buffer-partition lossless percent 5

set egress buffer-partition lossy percent 20

set egress buffer-partition multicast percent 75

Configuring the Global Shared Buffer Pool for Networks with Mostly Multicast Traffic

Step-by-Step Procedure

To configure the global ingress and egress shared buffer allocations and partitions for a network that carries mostly multicast traffic:

1. Configure the percentage of available (nonreserved) buffers used for the ingress global shared buffer pool:

   ```
   [edit class-of-service shared-buffer]
   user@switch# set ingress percent 100
   ```

2. Configure the global ingress buffer partitions for lossless, lossless-headroom, and lossy traffic:

   ```
   [edit class-of-service shared-buffer]
   user@switch# set ingress buffer-partition lossless percent 5
   user@switch# set ingress buffer-partition lossless-headroom percent 0
   user@switch# set ingress buffer-partition lossy percent 95
   ```

3. Configure the percentage of available (nonreserved) buffers used for the egress global shared buffer pool:

   ```
   [edit class-of-service shared-buffer]
   user@switch# set egress percent 100
   ```

4. Configure the global egress buffer partitions for lossless, lossy, and multicast queues:

   ```
   [edit class-of-service shared-buffer]
   user@switch# set egress buffer-partition lossless percent 5
   user@switch# set egress buffer-partition lossy percent 20
   user@switch# set egress buffer-partition multicast percent 75
   ```

Results

Display the results of the configuration:

```
root@dcbg-tp-pa-02> show configuration class-of-service shared-buffer
ingress {
    percent 100;
    buffer-partition lossless {
        percent 5;
```

```
    }
    buffer-partition lossy {
        percent 95;
    }
    buffer-partition lossless-headroom {
        percent 0;
    }
}
egress {
    percent 100;
    buffer-partition lossless {
        percent 5;
    }
    buffer-partition lossy {
        percent 20;
    }
    buffer-partition multicast {
        percent 75;
    }
}
```

Verification

Verify that you correctly configured the shared buffer.

Verifying the Shared Buffer Configuration

Purpose

Verify that you correctly configured the ingress and egress global shared buffer pools and that you correctly partitioned the buffer among the shared buffer types.

Action

List the global shared buffer configuration using the operational mode command **show class-of-service shared-buffer**:

`user@switch>` **show class-of-service shared-buffer**

```
root@dcbg-tp-pa-02> show class-of-service shared-buffer
Ingress:
  Total Buffer     :  9360.00 KB
  Dedicated Buffer :  2158.00 KB
  Shared Buffer    :  7202.00 KB
    Lossless          :  360.10 KB
```

```
    Lossless Headroom :  0.00 KB
    Lossy             :  6841.90 KB

  Lossless Headroom Utilization:
  Node Device         Total          Used                 Free
  0                   0.00 KB        0.00 KB              0.00 KB

Egress:
  Total Buffer     :  9360.00 KB
  Dedicated Buffer :  2704.00 KB
  Shared Buffer    :  6656.00 KB
    Lossless          :  332.80 KB
    Multicast         :  4992.00 KB
    Lossy             :  1331.20 KB
```

Meaning

The **show class-of-service shared-buffer** operational command shows all of the ingress and egress global shared buffer settings, including the buffer partitioning.

For the ingress shared buffers, the command output shows:

- The total switch buffer pool is 9360 KB (9 MB).
- The dedicated buffer pool is 2158 KB. This is the size of the global ingress dedicated buffer pool when you configure the ingress shared buffer pool as 100 percent of the available (user-configurable) buffer space. This is the minimum size of the reserved, ingress dedicated ingress buffer pool (not user-configurable). If you configure the shared buffer as less than 100 percent of the available buffer pool, the remaining buffer space is added to the dedicated buffer pool.
- With the ingress shared buffer pool configured as 100 percent of the available buffers, the total size of the ingress shared buffer pool is 7202 KB.
- The ingress shared buffer pool is partitioned to allocate:
  - 360.10 KB to lossless traffic
  - No space to lossless headroom traffic
  - 6841.90 KB to lossy unicast traffic
- The Lossless Headroom Utilization field shows how much of the buffer space reserved for paused traffic is used. Because the lossless headroom buffer partition is set to 0 (zero) percent, the total amount of lossless headroom buffer space is 0 KB; therefore the amount of used and free lossless headroom buffer space is also 0 KB.

For the egress shared buffers, the command output shows:

- The total switch buffer pool is 9360 KB (9 MB).
- The dedicated buffer pool is 2704 KB. This is the size of the global egress dedicated buffer pool when you configure the egress shared buffer pool as 100 percent of the available (user-configurable) buffer space. This is the minimum size of the reserved, egress dedicated buffer pool (not user-configurable). If you configure the shared buffer as less than 100 percent of the available buffer pool, the remaining buffer space is added to the dedicated buffer pool.
- With the egress shared buffer pool configured as 100 percent of the available buffers, the total size of the egress shared buffer pool is 6656 KB. This is less than the ingress shared buffer pool because the switch reserves more egress dedicated buffer space than ingress dedicated buffer space. (More dedicated buffer space means less shared buffer space, and more shared buffer space means less dedicated buffer space.)
- The egress shared buffer pool is partitioned to allocate:
  - 332.80 KB to lossless traffic
  - 4992 KB to multicast traffic
  - 1331.20 KB to lossy unicast traffic

> **NOTE:** The output values are valid for QFX3500 and QFX3600 switches. QFX5100, EX4600, and OCX Series switches have larger buffers (12 MB instead of 9 MB), so the total buffer size and the sizes of each buffer partition are larger on those switches.

RELATED DOCUMENTATION

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Best-Effort Unicast Traffic | 659

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Best-Effort Traffic on Links with Ethernet PAUSE Enabled | 667

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Lossless Traffic | 683

Configuring Global Ingress and Egress Shared Buffers | 657

Understanding CoS Buffer Configuration | 635

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Lossless Traffic

IN THIS SECTION

Although the switch reserves some buffer space to ensure a minimum memory allocation for ports and queues, you can configure how the system uses the rest of the buffer space to optimize the buffer allocation for your particular mix of network traffic.

This example shows you the recommended configuration of the global shared buffer pool to support a network that carries mostly lossless traffic. The global shared buffer pool is memory space that all of the ports on the switch share dynamically as they need buffers. You can allocate global shared memory space to different types of buffers to better support different mixes of network traffic.

CAUTION: Changing the buffer configuration is a disruptive event. Traffic stops on *all* ports until buffer reprogramming is complete.

Use the default shared buffer settings (for a network with a balanced mix of lossless, best effort, and multicast traffic) or one of the recommended shared buffer configurations for your mix of network traffic (mostly best-effort unicast traffic, mostly best-effort traffic on links enabled for Ethernet PAUSE, mostly multicast traffic, or mostly lossless traffic). Either the default configuration or one of the recommended configurations provides a buffer allocation that satisfies the needs of most networks.

NOTE: When we discuss lossless buffers, we mean buffers that handle traffic on which you enable priority-based flow control (PFC) to ensure lossless transport. The lossless buffers are not used for best-effort traffic on a link on which you enable Ethernet PAUSE (IEEE 802.3x).

After starting from the recommended configuration, you can fine-tune the shared buffer settings, but do so with caution to prevent traffic loss due to buffer misconfiguration.

Requirements

This example uses the following hardware and software components:

- Juniper Networks QFX3500 Switch
- Junos OS Release 12.3 or later for the QFX Series

Overview

You can configure the percentage of available (user-configurable) buffer space allocated to the global shared buffers. Any space that you do not allocate to the global shared buffer pool is added to the dedicated buffer pool. The default configuration allocates 100 percent of the available buffer space to the global shared buffers.

You can partition the ingress and egress shared buffer pools to allocate more buffers to the types of traffic your network predominantly carries, and fewer buffers to other traffic. From the buffer space allocated to the ingress shared buffer pool, you can allocate space to:

- Lossless buffers—Percentage of shared buffer pool for all lossless ingress traffic. The minimum value for the lossless buffers is 5 percent.
- Lossless headroom buffers—Percentage of shared buffer pool for packets received while a pause is asserted. If Ethernet PAUSE is configured on a port or if priority-based flow control (PFC) is configured on priorities on a port, when the port sends a pause message to the connected peer, the port uses the headroom buffers to store the packets that arrive between the time the port sends the pause message and the time the last packet arrives after the peer pauses traffic. The minimum value for the lossless headroom buffers is 0 (zero) percent. (Lossless headroom buffers are the only buffers that can have a minimum value of less than 5 percent.)
- Lossy buffers—Percentage of shared buffer pool for all best-effort ingress traffic (best-effort unicast, multidestination, and strict-high priority traffic). The minimum value for the lossy buffers is 5 percent.

The combined percentage values of the ingress lossless, lossless headroom, and lossy buffer partitions must total exactly 100 percent. If the buffer percentages total more than 100 percent or less than 100 percent, the switch returns a commit error. All ingress buffer partitions must be explicitly configured, even when the lossless headroom buffer partition has a value of 0 (zero) percent.

NOTE: If you commit a buffer configuration for which the switch does not have sufficient resources, the switch might log an error instead of returning a commit error. In that case, a syslog message is displayed on the console. For example:

```
user@host# commit
configuration check succeeds

Message from syslogd@host at Jun 13 11:11:10  ...
host dc-pfe: Not enough Ingress Lossless headroom.(Already allocated more).
 Dedicated : 14340 Lossy : 47100 Lossless 4239 Headroom 21195 Avail : 20781
commit complete
```

From the buffer space allocated to the egress shared buffer pool, you can allocate space to:

- Lossless buffers—Percentage of shared buffer pool for all lossless egress queues. The minimum value for the lossless buffers is 5 percent.
- Lossy buffers—Percentage of shared buffer pool for all best-effort egress queues (best-effort unicast, and strict-high priority queues). The minimum value for the lossy buffers is 5 percent.
- Multicast buffers—Percentage of shared buffer pool for all multidestination (multicast, broadcast, and destination lookup fail) egress queues. The minimum value for the multicast buffers is 5 percent.

The combined percentage values of the egress lossless, lossy, and multicast buffer partitions must total exactly 100 percent. If the buffer percentages total more than 100 percent or less than 100 percent, the switch returns a commit error. All egress buffer partitions must be explicitly configured and must have a value of at least 5 percent.

To configure the shared buffers to support a network that carries mostly lossless traffic, more buffer space needs to be allocated to lossless buffers, and less buffer space should be allocated to lossy buffers. This example shows you how to configure the global shared buffer pool allocation that we recommend to support a network that carries mostly lossless traffic.

Topology

Table 128 on page 685 shows the configuration components for this example.

Table 128: Components of the Recommended Shared Buffer Configuration for Lossless Network Topologies

Component	Settings
Hardware	QFX3500 switch

Table 128: Components of the Recommended Shared Buffer Configuration for Lossless Network Topologies *(continued)*

Component	Settings
Ingress shared buffer	Percentage of available ingress buffer space allocated to the ingress shared buffer: 100% Percentage of ingress buffer space allocated to lossless traffic (lossless buffer partition): 15% Percentage of ingress buffer space allocated to lossless headroom traffic (lossless headroom buffer partition): 80% Percentage of ingress buffer space allocated to best-effort traffic (lossy buffer partition): 5%
Egress shared buffer	Percentage of available egress buffer space allocated to the egress shared buffer: 100% Percentage of egress buffer space allocated to lossless queues (lossless buffer partition): 90% Percentage of egress buffer space allocated to best-effort queues (lossy buffer partition): 5% Percentage of egress buffer space allocated to multicast traffic (multicast buffer partition): 5%

Configuration

CLI Quick Configuration

To quickly configure the recommended shared buffer settings for networks that carry mostly lossless traffic, 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 shared-buffer]
set ingress percent 100
set ingress buffer-partition lossless percent 15
set ingress buffer-partition lossless-headroom percent 80
set ingress buffer-partition lossy percent 5
set egress percent 100
set egress buffer-partition lossless percent 90
set egress buffer-partition lossy percent 5
set egress buffer-partition multicast percent 5
```

Configuring the Global Shared Buffer Pool for Networks with Mostly Lossless Traffic

Step-by-Step Procedure

To configure the global ingress and egress shared buffer allocations and partitions for a network that carries mostly lossless traffic:

1. Configure the percentage of available (nonreserved) buffers used for the ingress global shared buffer pool:

   ```
   [edit class-of-service shared-buffer]
   user@switch# set ingress percent 100
   ```

2. Configure the global ingress buffer partitions for lossless, lossless-headroom, and lossy traffic:

   ```
   [edit class-of-service shared-buffer]
   user@switch# set ingress buffer-partition lossless percent 15
   user@switch# set ingress buffer-partition lossless-headroom percent 80
   user@switch# set ingress buffer-partition lossy percent 5
   ```

3. Configure the percentage of available (nonreserved) buffers used for the egress global shared buffer pool:

   ```
   [edit class-of-service shared-buffer]
   user@switch# set egress percent 100
   ```

4. Configure the global egress buffer partitions for lossless, lossy, and multicast queues:

   ```
   [edit class-of-service shared-buffer]
   user@switch# set egress buffer-partition lossless percent 90
   user@switch# set egress buffer-partition lossy percent 5
   user@switch# set egress buffer-partition multicast percent 5
   ```

Results

Display the results of the configuration:

```
rroot@dcbg-tp-pa-02> show configuration class-of-service shared-buffer
ingress {
    percent 100;
    buffer-partition lossless {
        percent 15;
```

```
    }
    buffer-partition lossy {
        percent 5;
    }
    buffer-partition lossless-headroom {
        percent 80;
    }
}
egress {
    percent 100;
    buffer-partition lossless {
        percent 90;
    }
    buffer-partition lossy {
        percent 5;
    }
    buffer-partition multicast {
        percent 5;
    }
}
```

Verification

Verify that the shared buffer configuration has been created properly.

Verifying the Shared Buffer Configuration

Purpose

Verify that the ingress and egress global shared buffer pools are correctly configured and partitioned among the shared buffer types.

Action

List the global shared buffer configuration using the operational mode command **show class-of-service shared-buffer**:

`user@switch>` **show class-of-service shared-buffer**

```
root@dcbg-tp-pa-02> show class-of-service shared-buffer
Ingress:
  Total Buffer     :  9360.00 KB
  Dedicated Buffer :  2158.00 KB
  Shared Buffer    :  7202.00 KB
    Lossless          :  1080.30 KB
```

```
    Lossless Headroom :  5761.60 KB
    Lossy             :  360.10 KB

  Lossless Headroom Utilization:
  Node Device         Total          Used                    Free
  0                   5761.60 KB     0.00 KB                 5761.60 KB

Egress:
  Total Buffer     :  9360.00 KB
  Dedicated Buffer :  2704.00 KB
  Shared Buffer    :  6656.00 KB
    Lossless          :  5990.40 KB
    Multicast         :  332.80 KB
    Lossy             :  332.80 KB
```

Meaning

The **show class-of-service shared-buffer** operational command shows all of the ingress and egress global shared buffer settings, including the buffer partitioning.

For the ingress shared buffers, the command output shows:

- The total switch buffer pool is 9360 KB (9 MB).
- The dedicated buffer pool is 2158 KB. This is the size of the global ingress dedicated buffer pool when you configure the ingress shared buffer pool as 100 percent of the available (user-configurable) buffer space. This is the minimum size of the reserved, ingress dedicated ingress buffer pool (not user-configurable). If you configure the shared buffer as less than 100 percent of the available buffer pool, the remaining buffer space is added to the dedicated buffer pool.
- With the ingress shared buffer pool configured as 100 percent of the available buffers, the total size of the ingress shared buffer pool is 7202 KB.
- The ingress shared buffer pool is partitioned to allocate:
  - 1080 KB to lossless traffic
  - 5761.60 KB to lossless headroom traffic
  - 360.10 KB to lossy unicast traffic
- The Lossless Headroom Utilization field shows how much of the buffer space reserved for paused traffic is used. Of the total available lossless headroom buffer space of 5761.60 KB, currently no buffer space is being used, so all 5761.60 KB of buffer space is free.

For the egress shared buffers, the command output shows:

- The total switch buffer pool is 9360 KB (9 MB).
- The dedicated buffer pool is 2704 KB. This is the size of the global egress dedicated buffer pool when you configure the egress shared buffer pool as 100 percent of the available (user-configurable) buffer space. This is the minimum size of the reserved, egress dedicated buffer pool (not user-configurable). If you configure the shared buffer as less than 100 percent of the available buffer pool, the remaining buffer space is added to the dedicated buffer pool.
- With the egress shared buffer pool configured as 100 percent of the available buffers, the total size of the egress shared buffer pool is 6656 KB. This is less than the ingress shared buffer pool because the switch reserves more egress dedicated buffer space than ingress dedicated buffer space. (More dedicated buffer space means less shared buffer space, and more shared buffer space means less dedicated buffer space.)
- The egress shared buffer pool is partitioned to allocate:
  - 5990.40 KB to lossless traffic
  - 332.80 KB to multicast traffic
  - 332.80 KB to lossy unicast traffic

> **NOTE:** The output values are valid for QFX3500 and QFX3600 switches. QFX5100 and EX4600 switches have larger buffers (12MB instead of 9MB), so the total buffer size and the sizes of each buffer partition are larger on QFX5100 and EX4600 switches.

RELATED DOCUMENTATION

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Best-Effort Unicast Traffic | 659

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Best-Effort Traffic on Links with Ethernet PAUSE Enabled | 667

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Multicast Traffic | 675

Configuring Global Ingress and Egress Shared Buffers | 657

Understanding CoS Buffer Configuration | 635

6

PART

Configuration Statements and Operational Commands

CHAPTER 20

Configuration Statements

IN THIS CHAPTER

application (Application Maps)

Syntax

```
application application-name {
    code-points [ aliases ] [ bit-patterns ];
}
```

Hierarchy Level

```
[edit policy-options application-maps application-map-name]
```

Release Information

Statement introduced in Junos OS Release 12.1 for EX Series switches.
Statement introduced in Junos OS Release 12.1 for the QFX Series.

Description

Add an application to an application map and define the application's code points.

Options

application-name—Name of the application.

The remaining statement is explained separately. See CLI Explorer.

Required Privilege Level

routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring an Application Map for DCBX Application Protocol TLV Exchange | 474
Example: Configuring DCBX Application Protocol TLV Exchange | 476
Example: Configuring DCBX to Support an iSCSI Application
Understanding DCBX Application Protocol TLV Exchange | 468
Understanding DCBX Application Protocol TLV Exchange on EX Series Switches

application (Applications)

Syntax

```
application application-name {
    destination-port port-value;
    protocol (tcp | udp);
    ether-type type;
}
```

Hierarchy Level

```
[edit applications]
```

Release Information

Statement introduced in Junos OS Release 12.1 for EX Series switches.
Statement introduced in Junos OS Release 12.1 for the QFX Series.

Description

Configure properties to define an application.

Options

application-name—Name of the application.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Defining an Application for DCBX Application Protocol TLV Exchange | 473
Example: Configuring DCBX Application Protocol TLV Exchange | 476
Example: Configuring DCBX to Support an iSCSI Application
Understanding DCBX Application Protocol TLV Exchange | 468
Understanding DCBX Application Protocol TLV Exchange on EX Series Switches

application-map

Syntax

```
application-map application-map-name;
```

Hierarchy Level

```
[edit protocols dcbx interface interface-name]
```

Release Information

Statement introduced in Junos OS Release 12.1 for EX Series switches.
Statement introduced in Junos OS Release 12.1 for the QFX Series.

Description

Specify an application map to apply to an interface.

Options

application-map-name—Name of the application map.

Required Privilege Level

routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

RELATED DOCUMENTATION

show dcbx neighbors | 977
Applying an Application Map to an Interface for DCBX Application Protocol TLV Exchange | 475
Example: Configuring DCBX Application Protocol TLV Exchange | 476
Example: Configuring DCBX to Support an iSCSI Application
Understanding DCBX Application Protocol TLV Exchange | 468
Understanding DCBX Application Protocol TLV Exchange on EX Series Switches

application-maps

Syntax

```
application-maps application-map-name {
  application application-name {
    code-points [ aliases ] [ bit-patterns ];
  }
}
```

Hierarchy Level

```
[edit policy-options]
```

Release Information

Statement introduced in Junos OS Release 12.1 for EX Series switches.
Statement introduced in Junos OS Release 12.1 for the QFX Series.

Description

Define an application map by specifying the applications that belong to the application map.

Options

application-map-name—Name of the application map.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring an Application Map for DCBX Application Protocol TLV Exchange | 474
Example: Configuring DCBX Application Protocol TLV Exchange | 476
Example: Configuring DCBX to Support an iSCSI Application
Understanding DCBX Application Protocol TLV Exchange | 468
Understanding DCBX Application Protocol TLV Exchange on EX Series Switches

applications (Applications)

Syntax

```
applications {
    application application-name {
        destination-port port-value;
        protocol (tcp | udp);
        ether-type type;
    }
}
```

Hierarchy Level

```
[edit]
```

Release Information

Statement introduced in Junos OS Release 12.1 for EX Series switches.
Statement introduced in Junos OS Release 12.1 for the QFX Series.

Description

Define applications that DCBX advertises.

Options

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Defining an Application for DCBX Application Protocol TLV Exchange | 473
Example: Configuring DCBX Application Protocol TLV Exchange | 476
Example: Configuring DCBX to Support an iSCSI Application
Understanding DCBX Application Protocol TLV Exchange | 468
Understanding DCBX Application Protocol TLV Exchange on EX Series Switches

applications (DCBX)

Syntax

```
applications {
    fcoe {
        no-auto-negotiation;
    }
}
```

Hierarchy Level

[edit protocols **dcbx interface** *interface-name*]

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 12.1 for the EX Series

Description

Configure Data Center Bridging Capability Exchange protocol (DCBX) applications on an interface.

Options

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

RELATED DOCUMENTATION

show dcbx neighbors | 977
Understanding DCB Features and Requirements | 450

buffer-partition (Egress)

Syntax

```
buffer-partition (lossless | lossy | multicast) {
    percent percent;
    dynamic-threshold value;
}
```

Hierarchy Level

```
[edit class-of-service shared-buffer egress]
```

Release Information

Statement introduced in Junos OS Release 12.3 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.
dynamic-threshold option introduced in Junos OS Release 19.1R1 for the QFX Series.

Description

The egress shared buffer pool is divided into three partitions. Each partition reserves a percentage of the available shared buffer pool for a type of traffic, so that the switch provides enough resources to support a mix of best-effort, lossless, and multicast traffic (multicast also includes broadcast and destination lookup fail traffic). To better support the mix of traffic on your network, you can optimize the allocation of egress shared buffers to different types of traffic by fine-tuning the shared buffer partitions.

> **NOTE:** OCX Series switches do not support lossless transport.

The percentages you configure for the three egress shared buffer partitions must total exactly 100 percent. If the total of the three shared buffer percentages is not 100 percent, the system returns a commit error and does not commit the configuration. You can configure any partition to 0 (zero) percent as long as the allocation to other partitions totals 100 percent.

This is a global allocation that applies to all ports. All ports on the switch receive the same allocation of egress shared buffers.

If you do not configure buffer partitions, the switch uses the default partitioning.

> **CAUTION:** Changing the buffer configuration is a disruptive event. Traffic stops on *all* ports until buffer reprogramming is complete.

Default

The default egress buffer partition shown in Table 129 on page 702 supports networks with a balanced mix of best-effort, multicast, and lossless traffic. It is the recommended configuration if you are using the default configuration with two lossless forwarding classes.

Table 129: Default Egress Shared Buffer Partitioning

Lossless Partition	Lossy Partition	Multicast Partition
50%	31%	19%

The sum of the default percentages configured for each partition is 100 percent. The sum of the partition percentages must always total 100 percent.

Options

dynamic-threshold *value*—Threshold for maximum buffer share for a queue at the egress buffer partition.

lossless—Shared buffer space reserved for all lossless egress traffic.

lossy—Shared buffer space for best-effort unicast egress traffic.

multicast—Shared buffer space reserved for all multicast (including broadcast and destination lookup fail) egress traffic.

percent *percent*—The percentage of buffer space to allocate to the specified buffer partition (lossless, lossy, or multicast buffers). The sum of the percentages for the three buffer partitions must total 100 percent.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Best-Effort Unicast Traffic | 659

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Multicast Traffic | 675

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Lossless Traffic | 683

Configuring Global Ingress and Egress Shared Buffers | 657

Understanding CoS Buffer Configuration | 635

Aggregated Ethernet Interfaces

buffer-partition (Ingress)

Syntax

```
buffer-partition (lossless | lossless-headroom | lossy) {
    percent percentage;
    dynamic-threshold value;
}
```

Hierarchy Level

```
[edit class-of-service shared-buffer ingress]
```

Release Information

Statement introduced in Junos OS Release 12.3 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.
dynamic-threshold option introduced in Junos OS Release 19.1R1 for the QFX Series.

Description

The ingress shared buffer pool is divided into three partitions. Each partition reserves a percentage of the available shared buffer pool for a type of traffic, so that the switch provides enough resources to support a mix of best effort (best-effort unicast and multicast) and lossless traffic. To better support the mix of traffic on your network, you can optimize the allocation of ingress shared buffers to different types of traffic by fine-tuning the shared buffer partitions.

> **NOTE:** OCX Series switches do not support lossless transport.

The percentages you configure for the three ingress shared buffer partitions must total exactly 100 percent. If the total of the three shared buffer percentages is not 100 percent, the system returns a commit error and does not commit the configuration. You can configure any partition to 0 (zero) percent as long as the allocation to other partitions totals 100 percent.

This is a global allocation that applies to all ingress traffic. All ports on the switch receive the same allocation of ingress shared buffers.

If you do not configure buffer partitions, the switch uses the default partitioning.

> **CAUTION:** Changing the buffer configuration is a disruptive event. Traffic stops on *all* ports until buffer reprogramming is complete.

Default

The default ingress buffer partition shown in Table 130 on page 704 supports networks with a balanced mix of best-effort, multicast, and lossless traffic. It is the recommended configuration if you are using the default configuration with two lossless forwarding classes.

Table 130: Default Ingress Shared Buffer Partitioning

Lossless Partition	Lossless-Headroom Partition	Lossy Partition
9%	45%	46%

The sum of the default percentages configured for each partition is 100 percent. The sum of the partition percentages always must total 100 percent.

Options

dynamic-threshold *value*—Threshold for maximum buffer share for a queue at the ingress buffer partition.

lossless—Shared buffer space reserved for all lossless ingress traffic.

lossless-headroom—Shared buffer space reserved to store packets received while either an 802.3x Ethernet PAUSE or a priority-based flow control (PFC) pause is asserted. (When an ingress interface pauses traffic, it must have the buffer space to store all of the packets currently in the buffer, and also all of the packets received before the connected peer stops sending traffic and the wire is cleared of packets.)

lossy—Shared buffer space for best-effort ingress traffic.

percent *percent*—The percentage of buffer space to allocate to the specified buffer partition (lossless, lossless-headroom, or lossy buffers). The sum of the percentages for the three buffer partitions must total 100 percent.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

buffer-size

Syntax

```
buffer-size (percent percent | remainder);
```

```
buffer-size (exact | percent percentage | remainder | temporal);
```

Hierarchy Level

```
[edit class-of-service schedulers scheduler-name]
```

```
[edit class-of-service schedulers scheduler-name]
```

Release Information

Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 12.3 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Specify buffer size in a scheduler configuration.

On all switches, you configure the proportion of port buffers allocated to a particular output queue using the following process:

1. Configure a scheduler and set the **buffer-size** option.

2. Use a scheduler map to map the scheduler to the forwarding class that is mapped to the queue to which you want to apply the buffer size.

   For example, suppose that you want to change the dedicated buffer allocation for FCoE traffic. FCoE traffic is mapped to the fcoe forwarding class, and the fcoe forwarding class is mapped to queue 3 (this is the default configuration). To use default FCoE traffic mapping, in the scheduler map configuration, map the scheduler to the **fcoe** forwarding class.

3. If you are using enhanced transmission selection (ETS) hierarchical scheduling, associate the scheduler map with the traffic control profile you want to use on the egress ports that carry FCoE traffic. If you are using direct port scheduling, skip this step.

4. If you are using ETS, associate the traffic control profile that includes the scheduler map with the desired egress ports. For this example, you associate the traffic control profile with the ports that carry FCoE traffic. If you are using port scheduling, associate the scheduler map with the desired egress ports.

Queue 3, which is mapped to the fcoe forwarding class and therefore to the FCoE traffic, receives the dedicated buffer allocation specified in the **buffer-size** statement.

NOTE: The total of all of the explicitly configured buffer size percentages for all of the queues on a port cannot exceed 100 percent.

QFX10000 Switches

On QFX10000 switches, the buffer size is the amount of time in milliseconds of port bandwidth that a queue can use to continue to transmit packets during periods of congestion, before the buffer runs out and packets begin to drop.

The switch can use up to 100 ms total (combined) buffer space for all queues on a port. A buffer-size configured as one percent is equal to 1 ms of buffer usage. A buffer-size of 15 percent (the default value for the best effort and network control queues) is equal to 15 ms of buffer usage.

The total buffer size of the switch is 4 GB. A 40-Gigabit port can use up to 500 MB of buffer space, which is equivalent to 100 ms of port bandwidth on a 40-Gigabit port. A 10-Gigabit port can use up to 125 MB of buffer space, which is equivalent to 100 ms of port bandwidth on a 10-Gigabit port. The total buffer sizes of the eight output queues on a port cannot exceed 100 percent, which is equal to the full 100 ms total buffer available to a port. The maximum amount of buffer space any queue can use is also 100 ms (which equates to a 100 percent buffer-size configuration), but if one queue uses all of the buffer, then no other queue receives buffer space.

There is no minimum buffer allocation, so you can set the buffer-size to zero (0) for a queue. However, we recommend that on queues on which you enable PFC to support lossless transport, you allocate a minimum of 5 ms (a minimum buffer-size of 5 percent). The two default lossless queues, fcoe and no-loss, have buffer-size default values of 35 ms (35 percent).

Queue buffer allocation is dynamic, shared among ports as needed. However, a queue cannot use more than its configured amount of buffer space. For example, if you are using the default CoS configuration, the best-effort queue receives a maximum of 15 ms of buffer space because the default transmit rate for the best-effort queue is 15 percent.

If a switch experiences congestion, queues continue to receives their full buffer allocation until 90 percent of the 4 GB buffer space is consumed. When 90 percent of the buffer space is in use, the amount of buffer space per port, per queue, is reduced in proportion to the configured buffer size for each queue. As the percentage of consumed buffer space rises above 90 percent, the amount of buffer space per port, per queue, continues to be reduced.

On 40-Gigabit ports, because the total buffer is 4 GB and the maximum buffer a port can use is 500 MB, up to seven 40-Gigabit ports can consume their full 100 ms allocation of buffer space. However, if an eighth 40-Gigabit port requires the full 500 MB of buffer space, then the buffer allocations are proportionally reduced because the buffer consumption is above 90 percent.

On 10-Gigabit ports, because the total buffer is 4 GB and the maximum buffer a port can use is 125 MB, up to 28 10-Gigabit ports can consume their full 100 ms allocation of buffer space. However, if a 29th 10-Gigabit port requires the full 125 MB of buffer space, then the buffer allocations are proportionally reduced because the buffer consumption is above 90 percent.

QFX5100, EX4600, QFX3500, and QFX3600 Switches, and QFabric Systems

Set the dedicated buffer size of the egress queue that you bind the scheduler to in the scheduler map configuration. The switch allocates space from the global dedicated buffer pool to ports and queues in a hierarchical manner. The switch allocates an equal number of dedicated buffers to each egress port, so each egress port receives the same amount of dedicated buffer space. The amount of dedicated buffer space per port is not configurable.

However, the **buffer-size** statement allows you to control the way each port allocates its share of dedicated buffers to its queues. For example, if a port only uses two queues to forward traffic, you can configure the port to allocate all of its dedicated buffer space to those two ports and avoid wasting buffer space on queues that are not in use. We recommend that the buffer size should be the same size as the minimum guaranteed transmission rate (the **transmit-rate**).

Default

The default behavior differs on different switches.

QFX10000 Switches

If you do not configure buffer-size and you do not explicitly configure a queue scheduler, the default buffer-size is the default transmit rate of the queue. If you explicitly configure a queue scheduler, the default buffer allocations are not used. If you explicitly configure a queue scheduler, configure the buffer-size for each queue in the scheduler, keeping in mind that the total buffer-size of the queues cannot exceed 100 percent (100 ms).

Table 131 on page 709 shows the default queue buffer sizes on QFX10000 switches. The default buffer size is the same as the default transmit rate for each default queue:

Table 131: Default Output Queue Buffer Sizes (QFX10000 Switches)

Queue Number	Forwarding Class	Transmit Rate	Buffer Size
0	best-effort	15%	15%
3	fcoe	35%	35%
4	no-loss	35%	35%
7	network-control	15%	15%

By default, only the queues mapped to the default forwarding classes receive buffer space from the port buffer pool. (Buffers are not wasted on queues that do not carry traffic.)

QFX5100, EX4600, QFX3500, and QFX3600 Switches, and QFabric Systems

The port allocates dedicated buffers to queues that have an explicitly configured scheduler buffer size. If you do not explicitly configure a scheduler buffer size for a queue, the port serves the explicitly configured queues first. Then the port divides the remaining dedicated buffers equally among the queues that have an explicitly attached scheduler *without* an explicitly configured buffer size configuration. (If you configure a scheduler, but you do not configure the buffer size parameter, the default is equivalent to configuring the buffer size with the **remainder** option.)

If you use the default scheduler and scheduler map on a port (no explicit scheduler configuration), then the port allocates its dedicated buffer pool to queues based on the default scheduling. Table 132 on page 710 shows the default queue buffer sizes. The default buffer size is the same as the default transmit rate for each default queue:

Table 132: Default Output Queue Buffer Sizes (QFX5100, EX4600, QFX3500, and QFX3600 Switches, and QFabric Systems)

Queue Number	Forwarding Class	Transmit Rate	Buffer Size
0	best-effort	5%	5%
3	fcoe	35%	35%
4	no-loss	35%	35%
7	network-control	5%	5%
8	mcast	20%	20%

By default, only the queues mapped to the default forwarding classes receive buffer space from the port buffer pool. (Buffers are not wasted on queues that do not carry traffic.)

NOTE: OCX Series switches do not support lossless transport. On OCX Series switches, do not map traffic to the lossless default fcoe and no-loss forwarding classes. OCX Series default DSCP classification does not map traffic to the fcoe and no-loss forwarding classes, so by default, the OCX system does not classify traffic into those forwarding classes. (On other switches, the fcoe and no-loss forwarding classes provide lossless transport for Layer 2 traffic. OCX Series switches do not support lossless Layer 2 transport.) The active forwarding classes (**best-effort**, **network-control**, and **mcast**) share the unused bandwidth assigned to the **fcoe** and **no-loss** forwarding classes.

On EX Series switches except EX4300 switches, the default scheduler transmission rate and buffer size percentages for queues 0 through 7 are 95, 0, 0, 0, 0, 0, 0, and 5 percent, respectively. On EX4300 switches, the default scheduler transmission rate and buffer size for queues 0 through 11 are 75, 0, 0, 5, 0, 0, 0, 0, 15, 0, 0 and 5 percent, respectively, of the total available buffer.

Options

percent ***percent***—Percentage of the port dedicated buffer pool allocated to the queue (or queues) mapped to the scheduler.

remainder—Remaining dedicated buffer pool after the port satisfies the needs of the explicitly configured buffers. The port divides the remaining buffers equally among the queues that are explicitly attached to a scheduler but that do not have an explicit buffer size configuration (or are configured with **remainder** as the buffer size).

exact—(Except on EX8200 standalone switches and EX8200 Virtual Chassis) Enforce the exact buffer size. When this option is configured, sharing is disabled on the queue, restricting the usage to guaranteed buffers only.

percentage —Buffer size as a percentage of the total buffer.

remainder—Remaining buffer available.

temporal—(EX4200 standalone switches, EX4200 Virtual Chassis, EX4300 standalong switches, EX4300 Virtual Chassis, EX8200 standalone switches, and EX8200 Virtual Chassis only) Buffer size as a temporal value.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Example: Configuring CoS on EX Series Switches

Defining CoS Schedulers and Scheduler Maps (CLI Procedure) or *Defining CoS Schedulers (J-Web Procedure)*

Understanding CoS Schedulers

cable-length (Congestion Notification)

Syntax

```
cable-length cable-length-value;
```

Hierarchy Level

```
[edit class-of-service congestion-notification-profile profile-name input]
```

Release Information

Statement introduced in Junos OS Release 12.3 for the QFX Series.

Description

Specify the length of the cable between the interface and its peer interface in meters. The system uses the cable length and the maximum receive unit (MRU) to calculate the amount of buffer headroom reserved to support priority-based flow control (PFC). The the shorter the cable length and lower the MRU, the less headroom buffer space is required for PFC.

> **NOTE:** You can also set a maximum transmission unit (MTU) value (the largest packet size the interface sends) for interfaces by including the **mtu** statement at the **[edit interfaces *interface-name*]** hierarchy level.

Default

The default cable length value is 100 meters (approximately 328 feet).

Options

cable-length-value—Length of the cable in meters. (Generally from 1 to 300 meters, but there is no configuration restriction.)

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring CoS PFC (Congestion Notification Profiles) | 194

class (Forwarding Classes)

List of Syntax

Syntax for QFX Series and OCX Series Routers on page 714
Syntax for M120, M320, MX Series routers, T Series routers, and EX Series switches on page 714

Syntax for QFX Series and OCX Series Routers

```
class {
   class-name {
      pfc-priority pfc-priority;
      queue-num queue-number <no-loss>;
   }
}
```

Syntax for M120, M320, MX Series routers, T Series routers, and EX Series switches

```
class {
   class-name {
      queue-num queue-number ;
      priority (high | low) ;
   }
}
```

Hierarchy Level

```
[edit class-of-service forwarding-classes]
```

Release Information

Statement introduced in Junos OS Release 8.1 for M120, M320, MX Series routers, T Series routers, and EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.
No-loss option introduced in Junos OS Release 12.3 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.
PFC-priority statement introduced in Junos OS Release 17.4R1 for the QFX Series.

Description

On M120 , M320, MX Series routers, T Series routers and EX Series switches only, specify the output transmission queue to which to map all input from an associated forwarding class.

This statement enables you to configure up to 16 forwarding classes with multiple forwarding classes mapped to single queues. If you want to configure up to eight forwarding classes with one-to-one mapping to output queues, use the *queue* statement instead of the **class** statement at the **[edit class-of-service forwarding-classes]** hierarchy level.

Map one or more forwarding classes to a single queue. Also, when configuring DSCP-based PFC, map a forwarding class to a PFC priority value to use in pause frames when traffic on a DSCP value becomes congested (see "Configuring DSCP-based PFC for Layer 3 Untagged Traffic" on page 245 for details).

You can map unicast forwarding classes to a unicast queue (0 through 7) and multidestination forwarding classes to a multicast queue (8 through 11). The queue to which you map a forwarding class determines if the forwarding class is a unicast or multicast forwarding class.

NOTE: On systems that do not use the ELS CLI, if you are using Junos OS Release 12.2, 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.

NOTE: On systems that do not use the ELS CLI, if you are using Junos OS Release 12.3 or later, the default configuration is the same as the default configuration for Junos OS Release 12.2, and the default behavior is the same (the **fcoe** and **no-loss** forwarding classes receive lossless treatment). However, if you explicitly configure lossless forwarding classes, you can configure up to six lossless forwarding classes by specifying the **no-loss** option. If you do not specify the **no-loss** option in an explicit forwarding class configuration, the forwarding class is lossy. For example, if you explicitly configure the **fcoe** forwarding class and you do not include the **no-loss** option, the **fcoe** forwarding class is lossy, not lossless.

Options

class-name—Name of the forwarding class.

queue-number—Output queue number.

Range: 0 through 7. Some T Series router PICs are restricted to 0 through 3.

The remaining statements are explained separately. See CLI Explorer for details.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Example: Configuring Forwarding Classes | 157
Understanding CoS Forwarding Classes | 139
Understanding CoS Forwarding Classes
Configuring a Custom Forwarding Class for Each Queue
queue (Global Queues)

class (Forwarding Class Sets)

Syntax

```
class class-name;
```

Hierarchy Level

```
[edit class-of-service forwarding-class-sets forwarding-class-set-name]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Group forwarding classes into sets of forwarding classes (priority groups). You can group some or all of the configured forwarding classes into up to three unicast forwarding class sets and one multidestination forwarding class set.

Options

class-name —Name of the forwarding class.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Example: Configuring CoS Hierarchical Port Scheduling (ETS) | 415
Example: Configuring Forwarding Class Sets | 165
Understanding CoS Forwarding Class Sets (Priority Groups) | 162

class-of-service

Syntax

```
class-of-service {
    classifiers {
        (dscp | dscp-ipv6 | ieee-802.1 | exp) classifier-name {
            import (classifier-name | default);
            forwarding-class class-name {
                loss-priority level {
                    code-points [ aliases ] [ bit-patterns ];
                }
            }
        }
    }
    code-point-aliases {
        (dscp| dscp-ipv6 | ieee-802.1) {
            alias-name bits;
        }
    }
    congestion-notification-profile profile-name {
        input {
            (dscp | ieee-802.1) {
                code-point [code-point-bits] {
                    pfc {
                        mru mru-value;
                    }
                }
            }
            cable-length cable-length-value;
        }
        output {
            ieee-802.1 {
                code-point [code-point-bits] {
                    flow-control-queue [queue | list-of-queues];
                }
            }
        }
    }
    drop-profiles {
        profile-name {
            interpolate {
                fill-level low-value fill-level high-value drop-probability 0 drop-probability high-value;
            }
        }
```

```
}
forwarding-class class-name {
    scheduler scheduler-name;
    loss-priority level {
        code-points [ aliases ] [ bit-patterns ];
    }
}
forwarding-class-sets forwarding-class-set-name {
    class class-name;
}
forwarding-classes {
    class class-name {
        pfc-priority pfc-priority;
        no-loss;
        queue-num queue-number <no-loss>;
    }
}
host-outbound-traffic{
    forwarding-class class-name;
    dscp-code-point code-point;
}
```

```
interfaces interface-name {
    classifiers {
      (dscp | dscp-ipv6 | ieee-802.1 | exp) (classifier-name | default);
      }
    congestion-notification-profile profile-name;
    forwarding-class lossless-forwarding-class-name;
    forwarding-class-set forwarding-class-set-name {
      output-traffic-control-profile profile-name;
    }
    rewrite-value {
      input {
        ieee-802.1{
          code-point code-point-bits;
        }
      }
    }
    scheduler-map scheduler-map-name;
    unit logical-unit-number {
      classifiers {
        (dscp | dscp-ipv6 | ieee-802.1 | exp) (classifier-name | default);
      }
      forwarding-class class-name;
      rewrite-rules {
        (dscp | dscp-ipv6 | ieee-802.1 | exp) (classifier-name | default);
      }
    }
}
multi-destination {
  classifiers {
    (dscp | ieee-802.1) classifier-name;
  }
}
rewrite-rules {
  (dscp | dscp-ipv6 | ieee-802.1 | exp) classifier-name {
    import (rewrite-name | default);
    forwarding-class class-name {
      loss-priority priority code-point (alias | bits);
    }
  }
}
```

```
    scheduler-map-forwarding-class-sets {
        fabric-scheduler-map-name {
            forwarding-class-set fabric-forwarding-class-set-name scheduler scheduler-name;
        }
    }
    scheduler-maps {
        map-name {
            forwarding-class class-name scheduler scheduler-name;
        }
    }
    schedulers {
        scheduler-name  {
            buffer-size (percent percentage | remainder);
            drop-profile-map loss-priority (low | medium-high | high) protocol protocol drop-profile drop-profile-name;
            excess-rate percent percentage;
            explicit-congestion-notification;
            priority priority;
            shaping-rate (rate | percent percentage);
            transmit-rate (percent percentage) <exact>;
        }
    }
    shared-buffer {
        egress {
            percent percent;
            buffer-partition (lossless | lossy | multicast) {
                percent percent
            }
        }
        ingress {
            percent percent;
            buffer-partition (lossless | lossless-headroom | lossy) {
                percent percent
            }
        }
    }
    system-defaults {
        classifiers exp classifier-name;
    }
    traffic-control-profiles profile-name {
        guaranteed-rate(rate| percent percentage);
        scheduler-map map-name;
        shaping-rate (rate| percent percentage);
    }
}
```

Hierarchy Level

```
[edit]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

> **NOTE:** Not all switches support all portions of the class of service hierarchy. For example, some switches use the same classifiers for unicast and multidestination traffic, and those switches do not support the **multi-destination** classifier hierarchy, and some switches do not support shared buffer configuration, and those switches do not support the **shared-buffer** hierarchy.

> **NOTE:** OCX Series switches do not support MPLS exp classifiers and rewrite rules (including MPLS system defaults), and they do not support congestion notification profiles.

Description

Configure class-of-service parameters on the switch.

The remaining statements are explained separately. Search for a statement in CLI Explorer or click a linked statement in the Syntax section for details.

Default

If you do not configure any CoS features, the default CoS settings are used.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Assigning CoS Components to Interfaces | 77
Overview of Junos OS CoS | 2

classifiers

List of Syntax

Syntax (BA Classifiers)

```
classifiers {
  (dscp | dscp-ipv6 | ieee-802.1) classifier-name {
    import (classifier-name | default);
    forwarding-class class-name {
      loss-priority level {
        code-points [ aliases ] [ bit-patterns ];
      }
    }
  }
}
```

Syntax (Multidestination BA Classifiers)

Multidestination BA Classifiers

```
classifiers {
  (dscp | ieee-802.1) classifier-name;
}
```

Syntax (Interface Classifier Association: DSCP, DSCP IPv6, IEEE)

Interface Classifier Association

```
classifiers {
  (dscp | dscp-ipv6 | ieee-802.1) (default | classifier-name);
}
```

Syntax (Global EXP Interface Classifier Association with Interfaces)

```
classifiers {
    exp classifier-name;
}
```

Hierarchy Level (BA Classifiers)

```
[edit class-of-service],
```

Hierarchy Level (Multidestination BA Classifiers)

```
[edit class-of-service multi-destination],
```

Hierarchy Level (Interface Classifier Association: DSCP, DSCP IPv6, IEEE)

```
[edit class-of-service interfaces interface-name unit logical-unit-number]
```

Hierarchy Level (Global EXP Classifier)

```
[edit class-of-service system-defaults]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
EXP statement introduced in Junos OS Release 12.3 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Define a unicast or multidestination CoS behavior aggregate (BA) classifier.

> **NOTE:** OCX Series switches do not support MPLS, so they do not support EXP classifier configuration.

Options

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Defining CoS BA Classifiers (DSCP, DSCP IPv6, IEEE 802.1p) | **94**
Configuring a Global MPLS EXP Classifier | **108**
Example: Configuring Unicast Classifiers | **100**
Example: Configuring Multidestination (Multicast, Broadcast, DLF) Classifiers | **104**
Understanding CoS Classifiers | **84**
Understanding CoS Classifiers
Understanding CoS MPLS EXP Classifiers and Rewrite Rules | **132**

code-point-aliases

Syntax

```
code-point-aliases {
  (dscp| dscp-ipv6 | ieee-802.1 | exp) {
    alias-name bits;
  }
}
```

Hierarchy Level

```
[edit class-of-service]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Define an alias for a CoS marker. You can use the alias instead of the bit pattern when you specify the code point during configuration.

> **NOTE:** OCX Series switches do not support MPLS, so they do not support EXP code-point aliases.

Options

(dcsp | dscp-ipv6 | ieee-802.1 | exp)—Set the type of classifier for which you are creating an alias.

alias-name—Name of the code-point alias.

bits —Value of the code-point bits, in decimal form.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Defining CoS Code-Point Aliases | 82

code-point (Input Congestion Notification)

Syntax

```
code-point [code-point-bits] {
  pfc {
    mru mru-value;
  }
}
```

Hierarchy Level

```
[edit class-of-service congestion-notification-profile profile-name input dscp]
[edit class-of-service congestion-notification-profile profile-name input ieee-802.1]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Support for DSCP values introduced in Junos OS Release 17.4R1 for the QFX Series.

Description

Enable priority-based flow control (PFC) on an IEEE 802.1p code point (priority) or a Differentiated Services code point (DSCP) value (to implement DSCP-based PFC at Layer 3).

Use this statement to configure PFC to operate based on an IEEE 802.1p code point in the VLAN-tagged packet header at Layer 2, or on a DSCP value from the Layer 3 IP header to support PFC with untagged traffic.

The **pfc** statement is mandatory when setting this statement.

Options

code-point-bits—Code point bit pattern corresponding to an IEEE 802.1p 3-bit priority value when used in the **ieee-802.1** hierarchy, or a 6-bit DSCP value when used in the **dscp** hierarchy.

The remaining statements are described separately.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Example: Configuring CoS PFC for FCoE Traffic | 490

code-point (Output Congestion Notification)

Syntax

```
code-point [ code-point-bits ] {
    flow-control-queue [ queue | list-of-queues ];
}
```

Hierarchy Level

```
[edit class-of-service congestion-notification-profile profile-name output ieee-802.1]
```

Release Information

Statement introduced in Junos OS Release 12.3 for the QFX Series.

Description

Specify the IEEE 802.1p code point bits that identify the traffic you want to enable for priority-based flow control (PFC) pause.

Default

By default, IEEE 802.1p priorities 3 and 4 (code points 011 and 100, respectively) are enabled for PFC pause on all Ethernet interfaces. If you explicitly configure priorities to pause and the output queues on which to enable pause, the explicit configuration overrides the default configuration. When you apply an explicit output congestion notification profile to an interface, only the priorities and queues specified in the output congestion notification profile are enabled for pause on that interface.

For example, if you configure an output congestion notification profile that specifies priority 2 (code point 010), then traffic with IEEE 802.1p priority 2 is paused on the configured output queue during periods of congestion. However, traffic with priority 3 and priority 4 is not programmed to pause, because the explicit configuration overwrites the default configuration, and the explicit configuration does not pause priority 3 and priority 4. If you configure an explicit output congestion notification profile, all of the priorities you want to enable for PFC and all of the output queues you want to pause must be explicitly configured.

Options

code-point-bits—3-bit value in decimal form.

The remaining statements are described separately.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring CoS PFC (Congestion Notification Profiles) | **194**

Example: Configuring Two or More Lossless FCoE IEEE 802.1p Priorities on Different FCoE Transit Switch Interfaces | **587**

Example: Configuring Two or More Lossless FCoE Priorities on the Same FCoE Transit Switch Interface | **576**

Example: Configuring Lossless FCoE Traffic When the Converged Ethernet Network Does Not Use IEEE 802.1p Priority 3 for FCoE Traffic (FCoE Transit Switch) | **565**

Example: Configuring Lossless IEEE 802.1p Priorities on Ethernet Interfaces for Multiple Applications (FCoE and iSCSI) | **605**

Understanding CoS IEEE 802.1p Priorities for Lossless Traffic Flows | **173**

code-point (Rewrite Rules)

Syntax

```
code-point [ alias ] [ bit-pattern ];
```

Hierarchy Level

```
[edit class-of-service rewrite-rules (dscp | ieee-802.1) forwarding-class class-name loss-priority level]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Configure a code-point alias or bit set to apply to a forwarding class for a rewrite rule.

> **NOTE:** OCX Series switches do not support MPLS, so they do not support EXP rewrite rules.

Options

alias—Name of the alias.

bit-pattern—Value of the code-point bits, in decimal form.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Defining CoS Rewrite Rules | 114
Understanding CoS Classifiers | 84
Understanding CoS Classifiers

code-points (Application Maps)

Syntax

```
code-points [ aliases ] [ bit-patterns ];
```

Hierarchy Level

```
[edit policy-options application-maps application-map-name application application-name]
```

Release Information

Statement introduced in Junos OS Release 12.1 for EX Series switches.
Statement introduced in Junos OS Release 12.1 for the QFX Series.

Description

Define one or more code-point aliases or bit sets for an application.

Options

aliases—Name of the alias or aliases.

bit-patterns—Value of the code-point bits, in decimal form.

Required Privilege Level

routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring an Application Map for DCBX Application Protocol TLV Exchange | 474
Example: Configuring DCBX Application Protocol TLV Exchange | 476
Example: Configuring DCBX to Support an iSCSI Application
Understanding DCBX Application Protocol TLV Exchange | 468
Understanding DCBX Application Protocol TLV Exchange on EX Series Switches

code-points (CoS)

Syntax

```
code-points ([ aliases ] | [ bit-patterns ]);
```

Hierarchy Level

```
[edit class-of-service classifiers type classifier-name forwarding-class class-name loss-priority level]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.2 for SRX Series devices.
Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 12.1X44 for the SRX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.
Statement introduced in Junos OS Release 14.2 for PTX Series Packet Transport Routers.

Description

Specify one or more DSCP code-point aliases or bit sets to apply to a forwarding class..

> **NOTE:** OCX Series switches do not support MPLS, and therefore, do not support EXP code points or code point aliases.

Options

aliases—Name of the DSCP alias.

bit-patterns—Value of the code-point bits, in six-bit binary form.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Understanding Interfaces

Understanding How Behavior Aggregate Classifiers Prioritize Trusted Traffic

Example: Configuring Behavior Aggregate Classifiers

Example: Configuring BA Classifiers on Transparent Mode Security Devices

configured-flow-control

Syntax

```
configured-flow-control {
    rx-buffers (on | off);
    tx-buffers (on | off);
}
```

Hierarchy Level

```
[edit interfaces interface-name ether-options]
```

Release Information

Statement introduced in Junos OS Release 12.1 for the QFX Series.

Description

Configure Ethernet PAUSE asymmetric flow control on an interface. You can set an interface to generate and send PAUSE messages, and you can set an interface to respond to PAUSE messages sent by the connected peer. You must set both the **rx-buffers** and the **tx-buffers** values when you configure asymmetric flow control.

Use the **flow-control** and **no-flow-control** statements to enable and disable symmetric PAUSE on an interface. Symmetric flow control and asymmetric flow control are mutually exclusive features. If you attempt to configure both, the switch returns a commit error.

> **NOTE:** Ethernet PAUSE temporarily stops transmitting all traffic on a link when the buffers fill to a certain threshold. To temporarily pause traffic on individual “lanes” of traffic (each lane contains the traffic associated with a particular IEEE 802.1p code point, so there can be eight lanes of traffic on a link), use priority-based flow control (PFC) by applying a congestion notification profile to the interface.
>
> Ethernet PAUSE and PFC are mutually exclusive features, so you cannot configure both of them on the same interface. If you attempt to configure both Ethernet PAUSE and PFC on an interface, the switch returns a commit error.

Default

Flow control is disabled. You must explicitly configure Ethernet PAUSE flow control on interfaces.

Options

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

congestion-notification-profile | **737**

flow-control | **771**

Configuring CoS Asymmetric Ethernet PAUSE Flow Control | **212**

Enabling and Disabling CoS Symmetric Ethernet PAUSE Flow Control | **211**

Understanding CoS Flow Control (Ethernet PAUSE and PFC) | **197**

congestion-notification-profile

Syntax

```
congestion-notification-profile profile-name {
  input {
    (dscp | ieee-802.1) {
      code-point [code-point-bits] {
        pfc {
          mru mru-value;
        }
      }
    }
    cable-length cable-length-value;
  }
  output {
    ieee-802.1 {
      code-point [code-point-bits] {
        flow-control-queue [queue | list-of-queues];
      }
    }
  }
}
```

Interface Congestion Notification Profile Association

```
congestion-notification-profile profile-name {
```

Hierarchy Level

```
[edit class-of-service],
[edit class-of-service interfaces interface-name]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Support for DSCP values introduced in Junos OS Release 17.4R1 for the QFX Series.

Description

Configure a congestion notification profile to enable priority-based flow control (PFC) on traffic and apply the profile to an interface.

A congestion notification profile can be configured to enable PFC on incoming traffic (**input** stanza) that matches the following:

- A Differentiated Services code point (DSCP) value in the Layer 3 IP header (for traffic that is not VLAN-tagged).
- An IEEE 802.1 code point at Layer 2 in the VLAN header (for VLAN-tagged traffic).

A congestion notification profile can be configured to enable PFC on outgoing traffic (**output** stanza) specified only by an IEEE 802.1 code point at Layer 2 in the VLAN header.

> **NOTE:** You must configure PFC for FCoE traffic. Each interface that carries FCoE traffic should be configured for PFC on the FCoE code point (usually **011**).

There is no limit to the total number of congestion notification profiles you can create. However:

- You can attach a maximum of one congestion notification profile to an interface.
- DSCP-based PFC and IEEE 802.1p PFC cannot be configured under the same congestion notification profile.

> **NOTE:** Configuring or changing PFC on an interface blocks the entire port until the PFC change is completed. After a PFC change is completed, the port is unblocked and traffic resumes. Blocking the port stops ingress and egress traffic, and causes packet loss on all queues on the port until the port is unblocked.

Options

profile-name—Name of the congestion notification profile.

The remaining statements are explained separately. Search for a statement in CLI Explorer or click a linked statement in the Syntax section for details.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

dcbx

Syntax

```
dcbx {
    disable;
    interface (interface-name | all) {
        disable;
        application-map application-map-name;
        applications {
            no-auto-negotiation;
        }
        enhanced-transmission-selection {
            no-auto-negotiation;
            no-recommendation-tlv;
            recommendation-tlv {
                no-auto-negotiation;
            }
        }
        dcbx-version (auto-negotiate | ieee-dcbx | dcbx-version-1.01);
        priority-flow-control {
            no-auto-negotiation;
        }
    }
}
```

Hierarchy Level

```
[edit protocols]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 11.3 for EX Series switches.
mode and **recommendation-tlv** statements introduced in Junos OS Release 12.2 for the QFX Series.

Description

Configure DCBX properties. DCBX is an extension of Link Layer Discovery Protocol (LLDP), and LLDP must remain enabled on every interface for which you want to use DCBX. If you attempt to enable DCBX on an interface on which LLDP is disabled, the configuration commit fails.

Options

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

RELATED DOCUMENTATION

show dcbx neighbors | 977
Understanding DCB Features and Requirements | 450
Configuring DCBX Autonegotiation | 465

dcbx-version

Syntax

```
dcbx-version (auto-negotiate | ieee-dcbx | dcbx-version-1.01);
```

Hierarchy Level

[edit protocols **dcbx interface** (all | *interface-name*)]

Release Information

Statement introduced in Junos OS Release 12.2 for the QFX Series.

Description

Set the DCBX version for the specified interface or interfaces.

QFX3500 switches come up in IEEE DCBX mode and then autonegotiate with the connected peer to set the DCBX version.

QFabric system Node devices come up using DCBX version 1.01, and then autonegotiate with the connected peer to set the DCBX mode.

Default

The default DCBX mode is autonegotation.

Options

auto-negotiate—Automatically negotiate the DCBX version with the connected peer.

ieee-dcbx—Force the interface to use IEEE DCBX mode, regardless of the peer configuration.

dcbx-version-1.01—Force the interface to use version 1.01 DCBX mode, regardless of the peer configuration.

Required Privilege Level

routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

RELATED DOCUMENTATION

show dcbx neighbors | 977
Configuring DCBX Autonegotiation | 465
Understanding DCBX | 454

destination-port (Applications)

Syntax

```
destination-port port-value;
```

Hierarchy Level

```
[edit applications application application-name]
```

Release Information

Statement introduced in Junos OS Release 12.1 for EX Series switches.
Statement introduced in Junos OS Release 12.1 for the QFX Series.

Description

Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) destination port number, which combines with **protocol** to identify an application type. The Internet Assigned Numbers Authority (IANA) assigns port numbers. See the IANA *Service Name and Transport Protocol Port Number Registry* at http://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xml for a list of assigned port numbers.

> **NOTE:** To create an application for iSCSI, use the protocol **tcp** with the destination port number **3260**.

Options

port-value—Identifier for the port.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Defining an Application for DCBX Application Protocol TLV Exchange | 473
Example: Configuring DCBX Application Protocol TLV Exchange | 476
Example: Configuring DCBX to Support an iSCSI Application
Understanding DCBX Application Protocol TLV Exchange | 468
Understanding DCBX Application Protocol TLV Exchange on EX Series Switches

disable (DCBX)

Syntax

```
disable
```

Hierarchy Level

```
[edit protocols dcbx]
[edit protocols dcbx interface interface-name]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 11.3 for EX Series switches.

Description

Disable Data Center Bridging Capability Exchange protocol (DCBX) on one or more 10-Gigabit Ethernet interfaces.

Default

DCBX is enabled by default on all 10-Gigabit or higher Ethernet interfaces.

DCBX is enabled by default on all 10-Gigabit Ethernet interfaces on EX4500 CEE-enabled switches.

Required Privilege Level

routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring DCBX Autonegotiation | 465

Disabling DCBX to Disable PFC Autonegotiation on EX Series Switches (CLI Procedure)

Understanding DCB Features and Requirements | 450

Understanding DCB Features and Requirements on EX Series Switches

drop-probability

List of Syntax

QFX5100, EX4600, QFX3500, and QFX3600, Switches, QFabric Systems

```
drop-probability 0 drop-probability high-value;
```

QFX10000 Switches

```
drop-probability percentage1 percentage2 ... percentage32;
```

Hierarchy Level

```
[edit class-of-service drop-profiles profile-name interpolate]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

When configuring WRED, map the packet **drop-probability** to the fullness of a queue (**fill-level**). You configure the **fill-level** and **drop-probability** statements in related pairs. The pairs of fill level and drop probability values set a probability of dropping packets at a specified queue fullness value.

On switches that support only two fill level/drop probability pairs, the first drop probability is always zero. The first fill level/drop probability pair sets the drop start point, and the second fill level/drop probability pair sets the drop end point.

On switches that support 32 fill level/drop probability pairs, the first fill level/drop probability pair sets the drop start point, and the last fill level/drop probability pair sets the drop end point.

As the queue fills from the drop start point to the drop end point, the rate of packet drop increases in a curve pattern. The higher the queue fill level, the higher the probability of dropping packets.

Options

0 (switches that support only two fill level/drop probability pairs)—Probability that packets will drop at the lowest **fill-level** value. This is always zero, because until the queue reaches the specified low **fill-level** value, no packets are scheduled to drop.

Range: 0

high-value (switches that support only two fill level/drop probability pairs)—The maximum probability that packets will drop before queue fullness exceeds the high value of the queue **fill-level**, expressed as a percentage. If the queue fills beyond the high **fill-level** value, all packets drop.

Range: 0 through 100 percent

percentage1 percentage2 ... *percentage32* (switches that support 32 fill level/drop probability pairs)—The probability that packets will drop before the queue fullness exceeds the **fill-level** value, expressed as a percentage. Each drop probability pairs with a queue fill level to define the probability of a packet dropping at a specified queue fullness.

Range: 0 through 100 percent

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

drop-profile

Syntax

```
drop-profile profile-name;
```

Hierarchy Level

```
[edit class-of-service schedulers scheduler-name drop-profile-map loss-priority (low | medium-high | high) protocol
    protocol]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Define drop profiles for weighted random early detection (WRED). When a packet arrives, WRED checks the queue fill level specified in the drop profile. If the fill level corresponds to a nonzero drop probability, the WRED algorithm determines whether to drop the arriving packet.

Options

profile-name—Name of the drop profile.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Example: Configuring Drop Profile Maps | **271**

drop-profile-map

Syntax

```
drop-profile-map loss-priority (low | medium-high | high) protocol protocol drop-profile drop-profile-name;
```

Hierarchy Level

```
[edit class-of-service schedulers scheduler-name]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Map a drop profile to a loss priority and protocol for weighted random early detection (WRED). When a packet arrives, WRED checks the queue fill level. If the fill level corresponds to a nonzero drop probability, the WRED algorithm determines whether to drop the arriving packet.

Options

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Example: Configuring Drop Profile Maps | **271**

drop-profiles

List of Syntax

Switches Except QFX10000, QFabric Systems on page 749
QFX10000 Switches on page 749

Switches Except QFX10000, QFabric Systems

```
drop-profiles {
    profile-name {
        interpolate {
            fill-level low-value fill-level high-value drop-probability 0 drop-probability high-value;
        }
    }
}
```

QFX10000 Switches

```
drop-profiles {
    profile-name {
        interpolate {
            fill-level level1 level2 ... level32 drop-probability percent1 percent2 ... percent32;
        }
    }
}
```

Hierarchy Level

```
[edit class-of-service]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Define drop profiles for weighted random early detection (WRED).

For a packet to be dropped, it must match the drop profile. When a packet arrives, WRED checks the queue fill level. If the fill level corresponds to a nonzero drop probability, the WRED algorithm determines whether to drop the arriving packet.

Options

profile-name—Name of the drop profile.

The remaining statements are explained separately.

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

dscp

List of Syntax

Syntax (Classifier)

```
dscp classifier-name {
    import (classifier-name | default);
    forwarding-class class-name {
        loss-priority level {
            code-points [ aliases ] [ bit-patterns ];
        }
    }
}
```

Syntax (Code-Point Alias)

Code-Point Alias Configuration

```
dscp alias-name bit-pattern;
```

Syntax (Multidestination Classifier)

Multidestination Classifier Configuration

```
dscp classifier-name;
```

Syntax (Interface Classifier Association)

Interface Classifier Association

```
dscp (classifier-name | default);
```

Syntax (Rewrite Rule)

Rewrite Rule Configuration

```
dscp rewrite-name {
    import (rewrite-name | default);
    forwarding-class class-name {
        loss-priority level {
            code-point [ aliases ] [ bit-patterns ];
        }
    }
}
```

Hierarchy Level (Classifier)

```
[edit class-of-service classifiers],
```

Hierarchy Level (Code-Point Aliases)

```
[edit class-of-service code-point-aliases],
```

Hierarchy Level (Multidestination Classifier)

```
[edit class-of-service multi-destination classifiers],
```

Hierarchy Level (Interface Classifier Association)

```
[edit class-of-service interfaces interface-name unit logical-unit-number classifiers],
[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules],
```

Hierarchy Level (Rewrite Rule)

```
[edit class-of-service rewrite-rules]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.

Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Define the Differentiated Services code point (DSCP) mapping that is applied to the packets.

Options

classifier-name—Name of the classifier.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Example: Configuring Unicast Classifiers | 100
Example: Configuring Classifiers | 96
Defining CoS Code-Point Aliases | 82
Defining CoS Rewrite Rules | 114
Assigning CoS Components to Interfaces | 77
Understanding CoS Classifiers | 84
Understanding CoS Classifiers
Understanding CoS Rewrite Rules | 111
Understanding Applying CoS Classifiers and Rewrite Rules to Interfaces | 116
Understanding Applying CoS Classifiers and Rewrite Rules to Interfaces

dscp (Input Congestion Notification)

Syntax

```
dscp {
    code-point [code-point-bits] {
        pfc {
            mru mru-value;
        }
    }
}
```

Hierarchy Level

```
[edit class-of-service congestion-notification-profile name input]
```

Release Information

Statement introduced in Junos OS Release 17.4R1 for the QFX Series.

Description

Configure a Differentiated Services code point (DSCP) value and apply priority-based flow control (PFC) to packets with that code point.

When this statement is configured, DSCP-based PFC can be invoked for untagged traffic by matching specified 6-bit DSCP values in the Layer 3 IP header of incoming packets instead of an IEEE 802.1p priority in the Layer 2 VLAN header. Additional configuration parameters associate configured DSCP values with a PFC priority to use in the Layer 2 pause frames sent to peers when the link becomes congested.

DSCP-based PFC can be used to support Remote Direct Memory Access (RDMA) over converged Ethernet version 2 (RoCEv2).

To enable DSCP-based PFC:

- Use this statement to define a congestion notification profile to enable PFC on traffic specified by a DSCP value.
- Use the **[edit class-of-service forwarding-classes class *class-name*]** pfc-priority statement to map a lossless forwarding class to a PFC priority value to use in the PFC pause frames.
- Use the **[edit class-of-service classifiers]** dscp statement to set up a DSCP classifier for the desired DSCP value and forwarding class mapped to a PFC priority above.

The remaining statements are explained separately. Search for a statement in CLI Explorer or click a linked statement in the Syntax section for details.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Understanding PFC Using DSCP at Layer 3 for Untagged Traffic | 242
Configuring DSCP-based PFC for Layer 3 Untagged Traffic | 245
Configuring CoS PFC (Congestion Notification Profiles) | 194
Understanding CoS Flow Control (Ethernet PAUSE and PFC) | 197

dscp-ipv6

List of Syntax

Syntax (Classifier) on page 756
Syntax (Code-Point Alias) on page 756
Syntax (Interface Classifier Association) on page 756
Syntax (Rewrite Rule) on page 756

Syntax (Classifier)

```
dscp-ipv6 classifier-name {
    import (classifier-name | default);
    forwarding-class class-name {
        loss-priority level {
            code-points [ aliases ] [ bit-patterns ];
        }
    }
}
```

Syntax (Code-Point Alias)

```
dscp-ipv6 alias-name bit-pattern;
```

Syntax (Interface Classifier Association)

```
dscp-ipv6 (classifier-name | default);
```

Syntax (Rewrite Rule)

```
dscp-ipv6 rewrite-name {
    import (rewrite-name | default);
    forwarding-class class-name {
        loss-priority level {
            code-point [ aliases ] [ bit-patterns ];
        }
    }
}
```

Hierarchy (Classifier)

```
[edit class-of-service classifiers],
```

Hierarchy (Code-Point Alias)

```
[edit class-of-service code-point-aliases],
```

Hierarchy (Interface Classifier Association)

```
[edit class-of-service interfaces interface-name unit logical-unit-number classifiers],
[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules],
```

Hierarchy (Rewrite Rule)

```
[edit class-of-service rewrite-rules]
```

Release Information

Statement introduced in Junos OS Release 12.2 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series

Description

Define the Differentiated Services code point (DSCP) IPv6 mapping that is applied to the packets.

> **NOTE:** On switches that use different classifiers for unicast and multidestination (multicast, broadcast, and destination lookup fail) traffic, there is no DSCP IPv6 classifier for multidestination (multicast, broadcast, and destination lookup fail) traffic. Multidestination IPv6 traffic uses the multidestination DSCP classifier.

Options

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Defining CoS Code-Point Aliases | 82
Understanding Applying CoS Classifiers and Rewrite Rules to Interfaces | 116
Understanding Applying CoS Classifiers and Rewrite Rules to Interfaces

dscp-code-point

Syntax

```
dscp-code-point code-point;
```

Hierarchy Level

```
[edit class-of-service host-outbound-traffic]
```

Release Information

Statement introduced in Junos OS Release 11.3 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Set the value of the DSCP code point in the type of service (ToS) field of the packet generated by the Routing Engine (host).

Options

code-point—Six-bit DSCP code point value.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Changing the Host Outbound Traffic Default Queue Mapping | 250
Understanding Host Routing Engine Outbound Traffic Queues and Defaults | 248

egress (Buffer Configuration)

Syntax

```
egress {
    percent percent;
    buffer-partition (lossless | lossy | multicast) {
        percent percent;
    }
}
```

Hierarchy Level

```
[edit class-of-service shared-buffer]
```

Release Information

Statement introduced in Junos OS Release 12.3 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Configure the global shared buffer pool allocation for egress traffic. The system allocates the shared buffer pool dynamically across its ports as the ports require memory space. Some buffer space is reserved for other buffers such as dedicated buffers (buffers allocated permanently to ports).

The percentage you specify is the percentage of available (user-configurable) buffer space allocated to the global shared egress buffer pool. If you allocate less than 100 percent of the available buffer space to the shared buffer pool, the remaining buffer space is added to the dedicated buffer pool. (You cannot directly configure the dedicated buffer pool for each port; dedicated buffers are allocated evenly across all the ports. However, on a port, you can configure the portion of dedicated port buffer space allocated to each queue in the scheduler configuration using the **buffer-size** option.)

> **CAUTION:** Changing the buffer configuration is a disruptive event. Traffic stops on *all* ports until buffer reprogramming is complete.

You can also partition the shared buffer pool to adjust the egress buffer allocations for different mixes of network traffic using the **buffer-partition** statement.

Default

The default shared buffer percentage is 100 percent. (All available buffer space is allocated to the shared buffer pool.)

Options

percent *percent*—Percentage of available egress buffer space allocated to the shared buffer pool. If the percentage is less than 100 percent, the remaining buffer space is allocated to the dedicated buffer pool.

The remaining statement is explained separately. See CLI Explorer.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Best-Effort Unicast Traffic | **659**

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Multicast Traffic | **675**

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Lossless Traffic | **683**

Configuring Global Ingress and Egress Shared Buffers | **657**

Understanding CoS Buffer Configuration | **635**

enhanced-transmission-selection

Syntax

```
enhanced-transmission-selection {
    no-auto-negotiation;
    no-recommendation-tlv;
    recommendation-tlv {
        no-auto-negotiation;
    }
}
```

Hierarchy Level

```
[edit protocols dcbx interface interface-name]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description

Disable advertising the enhanced transmission selection (ETS) state of the interface to the peer. To disable ETS on the interface, do not enable ETS on the interface in the class-of-service (CoS) configuration.

Disabling ETS autonegotiation stops the QFX Series from advertising the ETS Configuration TLV and the ETS Recommendation TLV.

Disabling the ETS recommendation TLV stops the QFX Series from advertising the ETS Recommendation TLV, but the ETS Configuration TLV is still advertised.

Options

no-auto-negotiation—Disable automatic negotiation of ETS (Configuration TLV and Recommendation TLV)

no-recommendation-tlv—Disable automatic negotiation of the ETS Recommendation TLV

recommendation-tlv—Enable automatic negotiation of ETS Recommendation TLV

Required Privilege Level

routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

RELATED DOCUMENTATION

show dcbx neighbors | 977

Configuring DCBX Autonegotiation | 465

Example: Configuring CoS Hierarchical Port Scheduling (ETS) | 415

Understanding DCB Features and Requirements | 450

ether-type

Syntax

```
ether-type ether-type;
```

Hierarchy Level

```
[edit applications application application-name]
```

Release Information

Statement introduced in Junos OS Release 12.1 for EX Series switches.
Statement introduced in Junos OS Release 12.1 for the QFX Series.

Description

Two-octet field in an Ethernet frame that defines the protocol encapsulated in the frame payload. See http://standards.ieee.org/develop/regauth/ethertype/eth.txt for a list of Institute of Electrical and Electronics Engineers (IEEE) EtherTypes.

> **NOTE:** To create a FIP application, use the EtherType 0x8914.

Options

type—Identifier for the EtherType.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Defining an Application for DCBX Application Protocol TLV Exchange | 473
Example: Configuring DCBX Application Protocol TLV Exchange | 476
Understanding DCBX Application Protocol TLV Exchange | 468

excess-rate

Syntax

```
excess-rate percent percentage;
```

Hierarchy Level

```
[edit class-of-service traffic-control-profiles profile-name],
[edit class-of-service schedulers scheduler-name]
```

Release Information

Statement introduced in Junos OS Release 15.1X53-D10 for the QFX Series.

Description

Determine the percentage of excess port bandwidth for which a queue (forwarding class) that is not a strict-high priority queue or forwarding class set (priority group) can contend. Excess bandwidth is the extra port bandwidth left after strict-high priority queues and the guaranteed minimum bandwidth requirements of other queues (as determined by each queue's transmit rate) are satisfied. With the exception of strict-high priority queues, the switch allocates extra port bandwidth to queues or to priority groups based on the configured excess rate. If you do not configure an excess rate for a queue, the default excess rate is the same as the transmit rate.

You cannot configure an excess rate on strict-high priority queues. Strict-high priority queues receive extra bandwidth based on an extra bandwidth sharing weight of "1", which is not configurable. However, the switch serves traffic on strict-high priority queues up to the configured transmit rate before it serves any other queues, so by configuring an appropriate transmit rate on a strict-high priority queue, you can guarantee strict-high priority traffic on that queue is treated in the manner you want.

Options

percent *percentage*—Percentage of the excess bandwidth to share.

Range: 0 through 100 percent

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Defining CoS Queue Schedulers for Port Scheduling | 357
Example: Configuring Traffic Control Profiles (Priority Group Scheduling) | 385

exp

Syntax

```
exp classifier-name {
    import (classifier-name | default);
    forwarding-class class-name {
        loss-priority level {
            code-points [ aliases ] [ bit-patterns ];
        }
    }
}
```

Rewrite Rule Configuration

```
exp rewrite-name {
    import (rewrite-name | default);
    forwarding-class class-name {
        loss-priority level {
            code-point [ aliases ] [ bit-patterns ];
        }
    }
}
```

Global Classifier Association with Interfaces

```
exp classifier-name;
```

Hierarchy Level

```
[edit class-of-service classifiers]
[edit class-of-service rewrite-rules]
[edit class-of-service system-defaults classifiers]
```

Release Information

Statement introduced in Junos OS Release 12.3X50 for the QFX Series.

Description

Define the EXP code point mapping that is applied to MPLS packets. EXP classifiers are not applied to any traffic except MPLS traffic. EXP classifiers are applied only to interfaces that are configured as **family mpls** (for example, **set interfaces xe-0/0/35 unit 0 family mpls**.)

There are no default EXP classifiers. You can configure up to 64 EXP classifiers.

On QFX10000 switches, you can configure and apply EXP classifiers to interfaces in the same way that you configure and apply DSCP, DSCP IPv6, and IEEE classifiers to interfaces. Different interfaces can have different EXP classifiers. QFX10000 switches do not support global EXP classifiers.

However, QFX5100, EX4600, QFX3500, and QFX3600 switches, and on QFabric systems, the switch uses only one EXP classifier as a global MPLS classifier on all interfaces. You specify the global EXP classifier in the **[edit class-of-service system-defaults]** hierarchy.

Options

classifier-name—Name of the EXP classifier.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Rewrite Rules for MPLS EXP Classifiers | **136**
Understanding CoS MPLS EXP Classifiers and Rewrite Rules | **132**
Understanding Applying CoS Classifiers and Rewrite Rules to Interfaces | **116**

explicit-congestion-notification

Syntax

```
explicit-congestion-notification;
```

Hierarchy Level

```
[edit class-of-service schedulers scheduler-name]
```

Release Information

Statement introduced in Junos OS Release 13.2X51 for EX Series switches.
Statement introduced in Junos OS Release 13.2X51-D20 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Enable explicit congestion notification (ECN) on the output queue (forwarding class) or output queues (forwarding classes) mapped to the scheduler. 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 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.

A weighted random early detection (WRED) packet drop profile must be applied to the output queues on which ECN is enabled. ECN uses the WRED drop profile thresholds to mark packets when the output queue experiences congestion.

ECN reduces packet loss by forwarding ECN-capable packets during periods of network congestion instead of dropping those packets. (TCP notifies the network about congestion by dropping packets.) During periods of congestion, ECN marks ECN-capable packets that egress from congested queues. When the receiver receives an ECN packet that is marked as experiencing congestion, the receiver echoes the congestion state back to the sender. The sender then reduces its transmission rate to clear the congestion.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Example: Configuring ECN | 284
Understanding CoS Explicit Congestion Notification | 274

fill-level

List of Syntax

QFX5100, EX4600, QFX3500, and QFX3600, Switches, QFabric Systems on page 769
QFX10000 Switches on page 769

QFX5100, EX4600, QFX3500, and QFX3600, Switches, QFabric Systems

```
fill-level low-value fill-level high-value;
```

QFX10000 Switches

```
fill-level level1 level2 ... level32;
```

Hierarchy Level

```
[edit class-of-service drop-profiles profile-name interpolate]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

When configuring weighted random early detection (WRED), map the fullness of a queue to a packet **drop-probability** value. You configure the **fill-level** and **drop-probability** statements in related pairs. The pairs of fill level and drop probability values set a probability of dropping packets at a specified queue fullness value.

The first fill level is the packet drop start point. Packets do not drop until the queue fullness reaches the first fill level. The last fill level is the packet drop end point. After the queue exceeds the fullness set by the drop end point, all non-ECN packets are dropped. As the queue fills from the drop start point to the drop end point, the rate of packet drop increases in a curve pattern. The higher the queue fill level, the higher the probability of dropping packets.

On switches that support only two fill level/drop probability pairs, the two pairs are the drop start point and the drop end point. On switches that support up to 32 fill level/drop probability pairs, you can configure intermediate interpolations between the drop start point and the drop end point, which provides greater flexibility in controlling the packet drop curve.

> **NOTE:** Do not configure the last fill level as 100 percent.

Options

low-value (switches that support only two fill level/drop probability pairs)—Fullness of the queue before packets begin to drop, expressed as a percentage. The low value must be less than the high value.

Range: 0 through 100

high-value (switches that support only two fill level/drop probability pairs)—Fullness of the queue before it reaches the maximum drop probability. If the queue fills beyond the fill level high value, all packets drop. The high value must be greater than the low value.

Range: 0 through 100

level1 level2 ... level32 (switches that support 32 fill level/drop probability pairs)—The queue fullness level, expressed as a percentage. Each fill level pairs with a drop probability to define the probability of a packet dropping at a specified queue fullness.

Range: 0 through 100

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

flow-control

Syntax

```
(flow-control | no-flow-control);
```

Hierarchy Level

```
[edit interfaces interface-name ether-options]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Explicitly enable or disable symmetric Ethernet PAUSE flow control, which regulates the flow of packets from the switch to the remote side of the connection by pausing all traffic flows on a link during periods of network congestion. Symmetric flow control means that Ethernet PAUSE is enabled in both directions. The interface generates and sends Ethernet PAUSE messages when the receive buffers fill to a certain threshold and the interface responds to PAUSE messages received from the connected peer. By default, flow control is disabled.

You can configure asymmetric flow control by including the **configured-flow-control** statement at the **[edit interfaces *interface-name* ether-options** hierarchy level. Symmetric flow control and asymmetric flow control are mutually exclusive features. If you attempt to configure both, the switch returns a commit error.

> **NOTE:** Ethernet PAUSE temporarily stops transmitting all traffic on a link when the buffers fill to a certain threshold. To temporarily pause traffic on individual “lanes” of traffic (each lane contains the traffic associated with a particular IEEE 802.1p code point, so there can be eight lanes of traffic on a link), use priority-based flow control (PFC).
>
> Ethernet PAUSE and PFC are mutually exclusive features, so you cannot configure both of them on the same interface. If you attempt to configure both Ethernet PAUSE and PFC on an interface, the switch returns a commit error.
>
> OCX Series switches do not support PFC.

- **flow-control**—Enable flow control; flow control is useful when the remote device is a Gigabit Ethernet switch.
- **no-flow-control**—Disable flow control.

Default

Flow control is disabled.

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

configured-flow-control | 735

Configuring Gigabit and 10-Gigabit Ethernet Interfaces for EX4600 and QFX Series Switches

Understanding CoS Flow Control (Ethernet PAUSE and PFC) | 197

Junos OS Network Interfaces Library for Routing Devices

flow-control-queue (Output Congestion Notification)

Syntax

```
flow-control-queue [ queue | list-of-queues ];
```

Hierarchy Level

```
[edit class-of-service congestion-notification-profile profile-name output ieee-802.1 code-point code-point-bits]
```

Release Information

Statement introduced in Junos OS Release 12.3 for the QFX Series.

Description

Specify one or more output queues to pause, to support priority-based flow control (PFC). The specified queues pause when the interface receives a PFC frame with a matching IEEE 802.1p code point.

Default

Queue 3 (mapped to the fcoe forwarding class) and queue 4 (mapped to the no-loss forwarding class) are programmed as flow control queues to pause. No other output queues are programmed to pause by default.

If you configure flow control queues explicitly, only the queues that you specify are programmed to pause. The explicit flow control queue to pause configuration overrides the default setting, so the queues paused in the default configuration are no longer paused by default.

For example, if you configure queue 2 as a flow control queue, then queue 2 pauses when congestion occurs, but queues 3 and 4 do not pause because they were not explicitly specified. To enable pause on output queues 2, 3, and 4, you must explicitly configure all three of the queues as flow control queues.

The same behavior applies to the IEEE 802.1p code points (priorities) on which PFC is enabled. By default, priorities 3 (011) and 4 (100) are enabled for PFC pause. If you explicitly configure flow control queues to pause, you must also explicitly configure pause for each priority (code point) that you want to pause, because the explicit configuration overrides the default configuration.

Options

[***queue*** | ***list-of-queues***]—The output queue or a list of output queues to pause.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

forwarding-class

List of Syntax

Classifier

```
forwarding-class class-name {
    loss-priority level {
        code-points [ aliases ] [ bit-patterns ];
    }
}
```

Classifier (EX Series switches)

```
forwarding-class class-name {
    loss-priority level {
        code-points [aliases] [6–bit-patterns];
    }
}
```

Rewrite Rule

```
forwarding-class class-name {
    loss-priority level {
        code-point [ aliases ] [ bit-patterns ];
    }
}
```

Scheduler Map

```
forwarding-class class-name {
    scheduler scheduler-name;
}
```

Interface

```
forwarding-class class-name;
```

Classifier Hierarchy Level

```
[edit class-of-service classifiers (dscp | dscp-ipv6 | ieee-802.1 | exp) classifier-name]
```

Classifier Hierarchy Level (EX Series Switches)

```
[edit class-of-service classifiers (dscp | ieee-802.1 | inet-precedence) classifier-name]
```

Rewrite Rule Hierarchy Level

```
[edit class-of-service rewrite-rules] (dscp | dscp-ipv6 | ieee-802.1) rewrite-name | exp]
```

Rewrite Rule Hierarchy Level (EX Series Switches)

```
[edit class-of-service rewrite-rules] (dscp | ieee-802.1 | inet-precedence) rewrite-rule-name]
```

Scheduler Map Hierarchy Level

```
[edit class-of-service scheduler-maps map-name]
```

Interface Hierarchy Level

```
[edit class-of-service interfaces interface-name unit logical-unit-number]
```

Host-Outbound Traffic Hierarchy Level (EX Series Switches)

```
[edit class-of-service host-outbound-traffic]
```

Release Information

Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Define forwarding class name and option values.

- Classifiers—Assign incoming traffic to the specified forwarding class based on the specified code point values and assign that traffic the specified loss priority.
- Rewrite rules—At the egress interface, change (rewrite) the value of the code point bits and the loss priority to specified new values for traffic assigned to the specified forwarding class, before forwarding the traffic to the next hop.
- Scheduler maps—Apply the specified scheduler to the specified forwarding class.
- Interfaces—Assign the specified forwarding class to the interface to use as a fixed classifier (all incoming traffic on the interface is classified into that forwarding class).

> **NOTE:** OCX Series switches do not support MPLS, so they do not support EXP classifiers or rewrite rules.

Options

class-name—Name of the forwarding class.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Example: Configuring CoS on EX Series Switches

Defining CoS Forwarding Classes (CLI Procedure) or *Defining CoS Forwarding Classes (J-Web Procedure)*

Understanding CoS Forwarding Classes

forwarding-class (Forwarding Policy)

Syntax

```
forwarding-class class-name {
    discard;
    lsp-next-hop [ lsp-regular-expression ];
    next-hop [ next-hop-name];
    non-labelled-next-hop;
    non-lsp-next-hop;
}
```

Hierarchy Level

```
[edit class-of-service forwarding-policy next-hop-map map-name]
[edit class-of-service forwarding-policy class class-name classification-override]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced for QFX10000 Series switches in Junos OS Release 17.1R1.
non-labelled-next-hop option introduced in Junos OS Release 19.1R1 for all platforms.

Description

Define forwarding class name and associated next hops.

Options

class-name—Name of the forwarding class.

non-labelled-next-hop—Match any non-labelled next hop.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Overriding the Input Classification

forwarding-class-default (Forwarding Policy) | 780

forwarding-class (Host Outbound Traffic)

Syntax

```
forwarding-class class-name;
```

Hierarchy Level

```
[edit class-of-service host-outbound-traffic ]
```

Release Information

Statement introduced in Junos OS Release 11.3 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Define forwarding class name for outbound host traffic (traffic generated by the Routing Engine).

Options

class-name—Name of the forwarding class.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Changing the Host Outbound Traffic Default Queue Mapping | **250**
Understanding Host Routing Engine Outbound Traffic Queues and Defaults | **248**

forwarding-class-default (Forwarding Policy)

Syntax

```
forwarding-class-default class-name {
    discard;
    lsp-next-hop [ lsp-regular-expression ];
    next-hop [ next-hop-name];
    non-lsp-next-hop;
}
```

Hierarchy Level

```
[edit class-of-service forwarding-policy next-hop-map map-name]
```

Release Information

Statement introduced in Junos OS Release 14.1.
Statement introduced for QFX10000 Series switches in Junos OS Release 17.1R1.

Description

Define the next hop for traffic that does not meet any forwarding class in the next-hop map.

Options

class-name—Name of the forwarding class.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

forwarding-class (Forwarding Policy) | 778

Load Balancing VPLS Non-Unicast Traffic Across Member Links of an Aggregate Interface

forwarding-classes

List of Syntax

SRX Series

```
forwarding-classes {
    class class-name {
        priority (high | low);
        queue-num number;
        spu-priority (high | low | medium);
    }
    queue queue-number {
        class class-name {
            priority (high | low);
        }
    }
}
```

QFX Series and OCX Series

```
forwarding-classes {
    class class-name {
        pfc-priority pfc-priority;
        no-loss;
        queue-num queue-number <no-loss>;
    }
}
```

EX Series (Except EX4300)

```
forwarding-classes {
    class class-name {
        queue-num queue-number;
        priority (high | low);
    }
}
```

EX4300

```
forwarding-classes {
  class class-name ;
    queue-num queue-number;
  }
}
```

M320, MX Series, T Series, and PTX Series

```
  forwarding-classes {
    class class-name {
      queue queue-number;
      priority (high | low);
    }
    queue queue-number {
      class class-name {
        priority (high | low) [policing-priority (premium | normal)];
        }
      }
    }
```

Hierarchy Level

```
[edit class-of-service]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 8.5.
Statement introduced in Junos OS Release 9.0 on EX Series switches.
The **policing-priority** option was introduced in Junos OS Release 9.5.
Statement introduced in Junos OS Release 11.1 on QFX Series switches.
Statement updated in Junos OS Release 11.4.
The **spu-priority** option was introduced in Junos OS Release 11.4R2.
Statement introduced in Junos OS Release 12.1 on PTX Series routers.
The **no-loss** option was introduced in Junos OS Release 12.3 on QFX Series switches.
Statement introduced in Junos OS Release 14.1X53-D20 on OCX Series switches.
Change from two to four queues made in Junos OS Release 12.3X48-D40 and in Junos OS Release 15.1X49-D70.
The **pfc-priority** statement was introduced in Junos OS Release 17.4R1 on QFX Series switches.
The **medium-high** and **medium-low** priorities for **spu-priority** were deprecated and **medium** priority was added in Junos OS Release 19.1R1.

Description

Command used to associate forwarding classes with class names and queues with queue numbers.

SRX Series Devices

All traffic traversing the SRX Series device is passed to an SPC to have service processing applied. Junos OS provides a configuration option to enable packets with specific Differentiated Services (DiffServ) code points (DSCP) precedence bits to enter a high-priority queue, a medium-priority queue, or a low-priority queue on the SPC. The Services Processing Unit (SPU) draws packets from the highest priority queue first, then from the medium priority queue, and last from the low priority queue. The processing of the queue is weighted-based not strict-priority-based. This feature can reduce overall latency for real-time traffic, such as voice traffic.

Initially, the spu-priority queue options were "high" and "low". Then, these options (depending on the devices) were expanded to "high", "medium-high", "medium-low", and "low". The two middle options ("medium-high" and "medium-low") have now been deprecated (again, depending on the devices) and replaced with "medium". So, the available options for spu-priority queue are "high", "medium", and "low".

We recommend that the high-priority queue be selected for real-time and high-value traffic. The other options would be selected based on user judgement on the value or sensitivity of the traffic.

M320, MX Series, and T Series Routers and EX Series Switches

For M320, MX Series, and T Series routers, and EX Series switches only, you can configure fabric priority queuing by including the **priority** statement. For Enhanced IQ PICs, you can include the **policing-priority** option.

> **NOTE:** The **priority** and **policing-priority** options are not supported on PTX Series routers.

EX Series Switches

For the EX Series switches, this statement associates the forwarding class with a class name and queue number. It can define the fabric queuing priority as high, medium-high, medium-low, or low.

Map one or more forwarding classes to a single output queue. Also, when configuring DSCP-based priority-based flow control (PFC), map a forwarding class to a PFC priority value to use in pause frames when traffic on a DSCP value becomes congested (see "Configuring DSCP-based PFC for Layer 3 Untagged Traffic" on page 245 for details).

Switches that use different forwarding classes for unicast and multidestination (multicast, broadcast, and destination lookup fail) traffic support 12 forwarding classes and 12 output queues (0 through 11). You map unicast forwarding classes to a unicast queue (0 through 7) and multidestination forwarding classes to a multidestination queue (8 through 11). The queue to which you map a forwarding class determines if the forwarding class is a unicast or multidestination forwarding class.

Switches that use the same forwarding classes for unicast and multidestination traffic support eight forwarding classes and eight output queues (0 through 7). You map forwarding classes to output queues. All traffic classified into one forwarding class (unicast and multidestination) uses the same output queue.

You cannot configure weighted random early detection (WRED) packet drop on forwarding classes configured with the **no-loss** packet drop attribute. Do not associate a drop profile with lossless forwarding classes.

NOTE: 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 (all of the forwarding classes must be lossy, or all of the forwarding classes mapped to a queue must be lossless).

OCX Series Switches

OCX Series switches do not support the **no-loss** packet drop attribute and do not support lossless forwarding classes. On OCX Series switches, do not configure the **no-loss** packet drop attribute on forwarding classes, and do not map traffic to the default **fcoe** and **no-loss** forwarding classes (both of these default forwarding classes carry the **no-loss** packet drop attribute).

NOTE: On switches that do not use the Enhanced Layer 2 Software (ELS) CLI, if you are using Junos OS Release 12.2, 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.

NOTE: On switches that do not use the ELS CLI, if you are using Junos OS Release 12.3 or later, the default configuration is the same as the default configuration for Junos OS Release 12.2, and the default behavior is the same (the **fcoe** and **no-loss** forwarding classes receive lossless treatment). However, if you explicitly configure lossless forwarding classes, you can configure up to six lossless forwarding classes by specifying the **no-loss** option. If you do not specify the **no-loss** option in an explicit forwarding class configuration, the forwarding class is lossy. For example, if you explicitly configure the **fcoe** forwarding class and you do not include the **no-loss** option, the **fcoe** forwarding class is lossy, not lossless.

Options

queue-num *queue-number*—Output queue number to associate with forwarding class.

Range: 0 through 7.

class *class-name*—Displays the forwarding class name assigned to the internal queue number.

> **NOTE:** This option is supported only on SRX5400, SRX5600, and SRX5800 devices.

> **NOTE:** AppQoS forwarding classes must be different from those defined for interface-based rewriters.

priority—Fabric priority value:

high—Forwarding class fabric queuing has high priority.

low—Forwarding class fabric queuing has low priority.

The **default** priority is **low**.

spu-priority—SPU priority queue, **high**, **medium**, or **low**. The default **spu-priority** is **low**.

> **NOTE:** The **spu-priority** option is supported only on the SRX5000 line of devices.

The remaining statements are explained separately. See CLI Explorer for details.

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Example: Configuring CoS on EX Series Switches

Defining CoS Forwarding Classes (CLI Procedure) or *Defining CoS Forwarding Classes (J-Web Procedure)*

Understanding CoS Forwarding Classes

Example: Configuring AppQoS

Configuring a Custom Forwarding Class for Each Queue

Forwarding Classes and Fabric Priority Queues

Configuring Hierarchical Layer 2 Policers on IQE PICs

Classifying Packets by Egress Interface

forwarding-class-set

Syntax

```
forwarding-class-set forwarding-class-set-name {
   output-traffic-control-profile profile-name;
}
```

Hierarchy Level

```
[edit class-of-service interfaces interface-name]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Apply a previously defined forwarding class set to an output traffic control profile.

Options

fowarding-class-set-name—Name of the forwarding class set.

The remaining statement is explained separately. See CLI Explorer.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Example: Configuring CoS Hierarchical Port Scheduling (ETS) | **415**
Assigning CoS Components to Interfaces | **77**
Understanding CoS Forwarding Class Sets (Priority Groups) | **162**

forwarding-class-sets

Syntax

```
forwarding-class-sets forwarding-class-set-name {
    class class-name;
}
```

Hierarchy Level

```
[edit class-of-service]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series

Description

Assign forwarding classes to forwarding class sets (priority groups).

Options

fowarding-class-set-name—Name of the forwarding class set.

The remaining statement is explained separately. See CLI Explorer.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Example: Configuring CoS Hierarchical Port Scheduling (ETS) | 415
Example: Configuring Forwarding Class Sets | 165
Understanding CoS Forwarding Class Sets (Priority Groups) | 162

forwarding-policy

Syntax

```
forwarding-policy {
    next-hop-map map-name {
        forwarding-class class-name {
            discard;
            lsp-next-hop [ lsp-regular-expression ];
            next-hop [ next-hop-name ];
            non-lsp-next-hop;
        }
        forwarding-class-default {
            discard;
            lsp-next-hop [ lsp-regular-expression ];
            next-hop [next-hop-name];
            non-lsp-next-hop;
        }
    }
    class class-name {
        classification-override {
            forwarding-class class-name;
        }
    }
}
```

Hierarchy Level

```
[edit class-of-service]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced for QFX10000 Series switches in Junos OS Release 17.1R1 to support CoS-based forwarding (CBF). **[set class-of-service forwarding-policy class]** is not supported on QFX10000 Series switches.

Description

Define CoS-based forwarding policy options.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring CoS-Based Forwarding | 150

guaranteed-rate

Syntax

```
guaranteed-rate (rate| percent percentage);
```

Hierarchy Level

```
[edit class-of-service traffic-control-profiles traffic-control-profile-name]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Configure a guaranteed minimum rate of transmission for a traffic control profile. The sum of the guaranteed rates of all of the forwarding class sets (priority groups) on a port should not exceed the total port bandwidth. The guaranteed rate also determines the amount of excess (extra) port bandwidth that the priority group (forwarding class set) 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 cannot configure a guaranteed rate for a forwarding class set (priority group) that includes strict-high priority queues. If the traffic control profile is for a forwarding class set that contains strict-high priority queues, do not configure a guaranteed rate.

Default

If you do not specify a guaranteed rate, the guaranteed rate is zero (0) and there is no minimum guaranteed bandwidth.

> **NOTE:** If you do not configure a guaranteed rate for a traffic control profile, the queues that belong to any forwarding class set (priority group) that uses that traffic control profile cannot have a configured transmit rate. The result is that there is no minimum guaranteed bandwidth for those queues and that those queues can be starved during periods of congestion.

Options

percent *percentage*—Minimum percentage of transmission capacity allocated to the forwarding class set or logical interface.

Range: 1 through 100 percent

rate—Minimum transmission rate allocated to the forwarding class set or logical interface, in bits per second (bps). You can specify a value in bits per second either as a complete decimal number or as a decimal number followed by the abbreviation **k** (1000), **m** (1,000,000), or **g** (1,000,000,000).

Range: 1000 through 10,000,000,000 bps

Required Privilege Level

interfaces—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Example: Configuring CoS Hierarchical Port Scheduling (ETS) | **415**

Example: Configuring Traffic Control Profiles (Priority Group Scheduling) | **385**

Example: Configuring Minimum Guaranteed Output Bandwidth | **392**

Understanding CoS Traffic Control Profiles | **372**

output-traffic-control-profile | **819**

host-outbound-traffic

Syntax

```
host-outbound-traffic {
    forwarding-class class-name;
    dscp-code-point code-point;
}
```

Hierarchy Level

```
[edit class-of-service]
```

Release Information

Statement introduced in Junos OS Release 11.3 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Allow queue selection for traffic generated by the Routing Engine (host). The selected queue must be configured properly. You can also configure specific DSCP code point bits for the type of service (ToS) field of the generated packets. This configuration does not affect transit packets or incoming packets. This is a global configuration that only affects packets originating on the Routing Engine. If you do not configure an output queue for host outbound traffic, the switch uses the default queue mapping.

Options

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Changing the Host Outbound Traffic Default Queue Mapping | 250
Understanding Host Routing Engine Outbound Traffic Queues and Defaults | 248

ieee-802.1

List of Syntax

Syntax (Classifier)

```
ieee-802.1 classifier-name {
    import (classifier-name | default);
    forwarding-class class-name {
        loss-priority level {
            code-points [ aliases ] [ bit-patterns ];
        }
    }
}
```

Syntax (Code-Point Alias)

Code-Point Alias Configuration

```
ieee-802.1 alias-name bit-pattern;
```

Syntax (Multidestination Classifier)

Multidestination Classifier Configuration

```
ieee-802.1 classifier-name;
```

Syntax (Interface Classifier Association)

Interface Classifier Association

```
ieee-802.1 (classifier-name | default);
```

Syntax (Rewrite Rule)

Rewrite Rule Configuration

```
ieee-802.1 rewrite-name {
    import (rewrite-name | default);
    forwarding-class class-name {
        loss-priority level {
            code-point [ aliases ] [ bit-patterns ];
        }
    }
}
```

Hierarchy Level (Classifier)

```
[edit class-of-service classifiers],
```

Hierarchy Level (Code-Point Alias)

```
[edit class-of-service code-point-aliases],
```

Hierarchy Level (Multidestination Classifier)

```
[edit class-of-service multi-destination classifiers ],
```

Hierarchy Level (Interface Classifier Association)

```
[edit class-of-service interfaces interface-name unit logical-unit-number classifiers],
[edit class-of-service interfaces interface-name unit logical-unit-number rewrite-rules],
```

Hierarchy Level (Rewrite Rule)

```
[edit class-of-service rewrite-rules]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.

Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Configure an IEEE 802.1 classifier, configure an IEEE 802.1 code-point alias, apply a fixed IEEE 802.1 classifier to an interface, or apply an IEEE-802.1 rewrite rule.

Options

classifier-name—Name of the classifier.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Example: Configuring Unicast Classifiers | 100
Defining CoS Code-Point Aliases | 82
Defining CoS Rewrite Rules | 114
Assigning CoS Components to Interfaces | 77
Understanding Applying CoS Classifiers and Rewrite Rules to Interfaces | 116
Understanding Applying CoS Classifiers and Rewrite Rules to Interfaces
Understanding CoS Classifiers | 84
Understanding CoS Classifiers
Understanding CoS Rewrite Rules | 111

ieee-802.1 (Input Congestion Notification)

Syntax

```
ieee-802.1 {
    code-point [code-point-bits] {
        pfc {
            mru mru-value;
        }
    }
}
```

Hierarchy Level

```
[edit class-of-service congestion-notification-profile profile-name input]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description

Configure an IEEE 802.1 code point and apply priority-based flow control (PFC) to packets with that code point.

Options

The statements are described separately.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Example: Configuring CoS PFC for FCoE Traffic | 490
Configuring CoS PFC (Congestion Notification Profiles) | 194
Understanding CoS Flow Control (Ethernet PAUSE and PFC) | 197

ieee-802.1 (Output Congestion Notification)

Syntax

```
ieee-802.1 {
    code-point [ code-point-bits ] {
        flow-control-queue [ queue | list-of-queues ];
    }
}
```

Hierarchy Level

```
[edit class-of-service congestion-notification-profile profile-name output]
```

Release Information

Statement introduced in Junos OS Release 12.3 for the QFX Series.

Description

Configure an IEEE 802.1 code point and apply priority-based flow control (PFC) to packets with that code point on output queues.

Options

The statements are described separately.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring CoS PFC (Congestion Notification Profiles) | 194
Example: Configuring Two or More Lossless FCoE IEEE 802.1p Priorities on Different FCoE Transit Switch Interfaces | 587
Example: Configuring Two or More Lossless FCoE Priorities on the Same FCoE Transit Switch Interface | 576
Example: Configuring Lossless FCoE Traffic When the Converged Ethernet Network Does Not Use IEEE 802.1p Priority 3 for FCoE Traffic (FCoE Transit Switch) | 565
Example: Configuring Lossless IEEE 802.1p Priorities on Ethernet Interfaces for Multiple Applications (FCoE and iSCSI) | 605
Understanding CoS IEEE 802.1p Priorities for Lossless Traffic Flows | 173

import

Syntax

```
import (import | default);
```

Hierarchy Level

```
[edit class-of-service classifiers (dscp | dscp-ipv6 | ieee-802.1 | exp) classifier-name],
[edit class-of-service rewrite-rules (dscp | dscp-ipv6 | ieee-802.1 | exp) classifier-name]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Specify a default or previously defined classifier.

> **NOTE:** OCX Series switches do not support MPLS, so they do not support EXP classifiers and rewrite rules.

Options

import —Name of the classifier mapping configured at the **[edit class-of-service classifiers]** hierarchy level.

default—Default classifier mapping.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Example: Configuring Unicast Classifiers | 100
Defining CoS BA Classifiers (DSCP, DSCP IPv6, IEEE 802.1p) | 94
Defining CoS Rewrite Rules | 114

Understanding CoS Classifiers | 84

Understanding CoS Classifiers

Understanding CoS Classifiers | 84

Understanding CoS Rewrite Rules | 111

ingress (Buffer Configuration)

Syntax

```
ingress {
  buffer-partition (lossless | lossless-headroom | lossy) {
    percent percent;
  }
  percent percent;
}
```

Hierarchy Level

```
[edit class-of-service shared-buffer]
```

Release Information

Statement introduced in Junos OS Release 12.3 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Configure the global shared buffer pool allocation for ingress traffic. The system allocates the shared buffer pool dynamically across its ports as the ports require memory space. Some buffer space is reserved for buffers such as dedicated buffers (buffers allocated permanently to ports) and headroom buffers (buffers that help prevent packet loss on lossless flows).

The percentage you specify is the percentage of available (user-configurable) buffer space allocated to the global shared ingress buffer pool. If you allocate less than 100 percent of the available buffer space to the shared buffer pool, the remaining buffer space is added to the dedicated buffer pool. (You cannot directly configure the dedicated buffer pool for each port; dedicated buffers are allocated evenly across all the ports.)

> **CAUTION:** Changing the buffer configuration is a disruptive event. Traffic stops on *all* ports until buffer reprogramming is complete.

You can also partition the shared buffer pool to adjust the ingress buffer allocations for different mixes of network traffic using the **buffer-partition** statement.

Default

The default shared buffer percentage is 100 percent. (All available buffer space is allocated to the shared buffer pool.)

Options

percent *percent*—Percentage of available ingress buffer space allocated to the shared buffer pool. If the percentage is less than 100 percent, the remaining buffer space is allocated to the dedicated buffer pool.

The remaining statement is explained separately. See CLI Explorer.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Best-Effort Unicast Traffic | **659**

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Multicast Traffic | **675**

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Lossless Traffic | **683**

Configuring Global Ingress and Egress Shared Buffers | **657**

Understanding CoS Buffer Configuration | **635**

input (Congestion Notification)

Syntax

```
input {
  (dscp | ieee-802.1) {
    code-point [code-point-bits] {
      pfc {
        mru mru-value;
      }
    }
  }
  cable-length cable-length-value;
}
```

Hierarchy Level

```
[edit class-of-service congestion-notification-profile profile-name]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Support for DSCP values introduced in Junos OS Release 17.4R1 for the QFX Series.

Description

Configure priority-based flow control (PFC) on incoming traffic based on either IEEE 802.1p priorities in the Layer 2 VLAN header or Differentiated Services code point (DSCP) values in the Layer 3 IP header.

DSCP-based PFC and IEEE 802.1p PFC cannot be configured under the same congestion notification profile.

Options

The remaining statements are explained separately. Search for a statement in CLI Explorer or click a linked statement in the Syntax section for details.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Example: Configuring CoS PFC for FCoE Traffic | 490
Configuring CoS PFC (Congestion Notification Profiles) | 194

interface (DCBX)

Syntax

```
interface (interface-name | all) {
    disable;
    application-map application-map-name;
    applications {
        no-auto-negotiation;
    }
    enhanced-transmission-selection {
        no-auto-negotiation;
        no-recommendation-tlv;
        recommendation-tlv {
            no-auto-negotiation;
        }
    }
    dcbx-version (auto-negotiate | ieee-dcbx | dcbx-version-1.01);
    priority-flow-control {
        no-auto-negotiation;
    }
}
```

Hierarchy Level

```
[edit protocols dcbx]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 11.3 for the EX Series switches.
Mode and **recommendation-tlv** statements introduced in Junos OS Release 12.2 for the QFX Series.

Description

Configure DCBX properties on an interface.

Options

interface-name—Name of the interface.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

RELATED DOCUMENTATION

show dcbx neighbors | 977

Configuring DCBX Autonegotiation | 465

Example: Configuring DCBX to Support an iSCSI Application

Understanding DCB Features and Requirements | 450

Understanding DCB Features and Requirements on EX Series Switches

Understanding DCBX Application Protocol TLV Exchange on EX Series Switches

interfaces (Class of Service)

Syntax

```
interfaces interface-name {
   classifiers {
      (dscp | dscp-ipv6 | ieee-802.1 | exp) (classifier-name | default);
   }
congestion-notification-profile profile-name;
   forwarding-class forwarding-class-name;
   forwarding-class-set forwarding-class-set-name {
      output-traffic-control-profile profile-name;
   }
   rewrite-value {
      input {
         ieee-802.1{
            code-point code-point-bits;
         }
      }
   }
   scheduler-map scheduler-map-name
   unit logical-unit-number {
      classifiers {
         (dscp | dscp-ipv6 | ieee-802.1 | exp) (classifier-name | default);
      }
      forwarding-class class-name;
      rewrite-rules {
         (dscp | dscp-ipv6 | ieee-802.1 | exp) (classifier-name | default);
      }
   }
}
```

Hierarchy Level

```
[edit class-of-service]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series

Description

Configure interface-specific CoS properties for incoming packets.

NOTE: Only switches that support direct port scheduling also support applying a scheduler map directly to an interface. When using enhanced transmission selection (ETS) hierarchical port scheduling, you cannot apply a scheduler map directly to an interface; instead, you associate the scheduler map with a traffic control profile and apply the traffic control profile to the interface.

NOTE: Only switches that support native Fibre Channel interfaces support the **rewrite-value** statement, which enables you to rewrite the IEEE 802.1p code points on native Fibre Channel interfaces.

NOTE: OCX Series switches do not support MPLS, so they do not support EXP classifiers or rewrite rules. OCX Series switches do not support the congestion-notification-profile configuration statement, which applies priority-based flow control (PFC) to interface output queues.

Options

interface-name—Name of the interface.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Assigning CoS Components to Interfaces | 77

Interfaces Overview for Switches

interpolate

List of Syntax

QFX5100, EX4600, QFX3500, and QFX3600, Switches, QFabric Systems on page 809
QFX10000 Switches on page 809

QFX5100, EX4600, QFX3500, and QFX3600, Switches, QFabric Systems

```
interpolate {
    fill-level low-value fill-level high-value;
    drop-probability 0 drop-probability high-value;
}
```

QFX10000 Switches

```
interpolate {
    fill-level level1 level2 ... level32 drop-probability percent1 percent2 ... percent32;
}
```

Hierarchy Level

```
[edit class-of-service drop-profiles profile-name]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Specify values for interpolating the relationship between queue fill level and drop probability for weighted random early detection (WRED) drop profiles.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

loss-priority (Classifiers)

Syntax

```
loss-priority level {
    code-points [ aliases ] [ bit-patterns ];
}
```

Hierarchy Level

```
[edit class-of-service classifiers (dscp | dscp-ipv6 | ieee-802.1) classifier-name forwarding-class class-name]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Configure packet loss priority value for a specific set of code-point aliases and bit patterns.

> **NOTE:** OCX Series switches do not support MPLS, so they do not support EXP classifiers.

Options

level—Can be one of the following:

- **low**—Packet has low loss priority.
- **medium-high**—Packet has medium-high loss priority.
- **high**—Packet has high loss priority.

The remaining statement is explained separately. See CLI Explorer.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Example: Configuring Unicast Classifiers | 100
Defining CoS BA Classifiers (DSCP, DSCP IPv6, IEEE 802.1p) | 94

Understanding CoS Classifiers | 84
Understanding CoS Classifiers

loss-priority (Drop Profiles)

Syntax

loss-priority *level* **protocol** *protocol* **drop-profile** *profile-name*;

Hierarchy Level

[edit **class-of-service schedulers** *scheduler-name* **drop-profile-map**]

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Configure packet loss priority value for a weighted random early detection (WRED) drop profile mapped to a system drop profile.

Options

level—Can be one of the following:

- **low**—Packet has low loss priority.
- **medium-high**—Packet has medium-high loss priority.
- **high**—Packet has high loss priority.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

loss-priority (Rewrite Rules)

Syntax

```
loss-priority level {
    code-point (alias | bit-pattern);
}
```

Hierarchy Level

```
[edit class-of-service rewrite-rules (dscp | ieee-802.1) rewrite-name forwarding-class class-name]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Specify a loss priority to which to apply a rewrite rule. The rewrite rule sets the code-point aliases and bit patterns for a specific forwarding class and loss priority. Packets that match the forwarding class and loss priority are rewritten with the rewrite code-point alias or bit pattern.

> **NOTE:** OCX Series switches do not support MPLS, so they do not support EXP rewrite rules.

Options

level—Can be one of the following:

- **low**—Packet has low loss priority.
- **medium-high**—Packet has medium-high loss priority.
- **high**—Packet has high loss priority.

The remaining statement is explained separately. See CLI Explorer.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Defining CoS Rewrite Rules | 114

mru

Syntax

```
mru mru-value;
```

Hierarchy Level

```
[edit class-of-service congestion-notification-profile profile-name input dscp code-point code-point-bits pfc]
[edit class-of-service congestion-notification-profile profile-name input ieee-802.1 code-point code-point-bits pfc]
```

Release Information

Statement introduced in Junos OS Release 12.3 for the QFX Series.
Support added in **dscp** statement hierarchy in Junos OS Release 17.4R1 for the QFX Series.

Description

Configure the maximum receive unit (MRU) of the interface in bytes. The system uses the MRU and the cable length to calculate the amount of buffer headroom reserved to support priority-based flow control (PFC). The lower the MRU and the shorter the cable length, the less headroom buffer space is required for PFC.

> **NOTE:** You can also set a maximum transmission unit (MTU) value (the largest packet size the interface sends) for interfaces by including the **mtu** statement at the **[edit interfaces *interface-name*]** hierarchy level.

Default

For priority 3 traffic, the default MRU value is 2500 bytes.

For priority 4 traffic, the default MRU value is 9216 bytes.

For user-configured priorities, the default MRU value is 2500 bytes.

Options

mru-value—Value of the maximum packet receive unit size in bytes (generally from 1500 to 9216 bytes, but there is no configuration restriction).

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring CoS PFC (Congestion Notification Profiles) | **194**

Example: Configuring Two or More Lossless FCoE IEEE 802.1p Priorities on Different FCoE Transit Switch Interfaces | **587**

Example: Configuring Two or More Lossless FCoE Priorities on the Same FCoE Transit Switch Interface | **576**

Example: Configuring Lossless FCoE Traffic When the Converged Ethernet Network Does Not Use IEEE 802.1p Priority 3 for FCoE Traffic (FCoE Transit Switch) | **565**

Example: Configuring Lossless IEEE 802.1p Priorities on Ethernet Interfaces for Multiple Applications (FCoE and iSCSI) | **605**

Understanding CoS Flow Control (Ethernet PAUSE and PFC) | **197**

Understanding CoS IEEE 802.1p Priorities for Lossless Traffic Flows | **173**

Understanding PFC Using DSCP at Layer 3 for Untagged Traffic | **242**

Configuring DSCP-based PFC for Layer 3 Untagged Traffic | **245**

multi-destination

Syntax

```
multi-destination {
    classifiers {
        (dscp | dscp-ipv6 | ieee-802.1 | inet-precedence) classifier-name;
    }
}
```

Hierarchy Level

```
[edit class-of-service]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Define a multicast CoS behavior aggregate (BA) classifier.

Options

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Defining CoS BA Classifiers (DSCP, DSCP IPv6, IEEE 802.1p) | 94
Example: Configuring Multidestination (Multicast, Broadcast, DLF) Classifiers | 104
Assigning CoS Components to Interfaces | 77
Understanding CoS Classifiers | 84
Understanding CoS Classifiers

next-hop-map

Syntax

```
next-hop-map map-name {
    forwarding-class class-name {
        discard;
        lsp-next-hop [ lsp-regular-expression ];
        next-hop [next-hop-name];
        non-lsp-next-hop;
    }
    forwarding-class-default {
        discard;
        lsp-next-hop [ lsp-regular-expression ];
        next-hop [next-hop-name];
        non-lsp-next-hop;
    }
}
```

Hierarchy Level

```
[edit class-of-service forwarding-policy]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced for QFX10000 Series switches in Junos OS Release 17.1R1.

Description

Specify the map for CoS forwarding routes.

Options

map-name—Map that defines next-hop routes.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring CoS-Based Forwarding | 150

output (Congestion Notification)

Syntax

```
output {
  ieee-802.1 {
    code-point [code-point-bits] {
      flow-control-queue [queue | list-of-queues];
    }
  }
}
```

Hierarchy Level

```
[edit class-of-service congestion-notification-profile profile-name]
```

Release Information

Statement introduced in Junos OS Release 12.3 for the QFX Series.

Description

Configure priority-based flow control (PFC) on output queues.

Options

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring CoS PFC (Congestion Notification Profiles) | 194
Example: Configuring Two or More Lossless FCoE IEEE 802.1p Priorities on Different FCoE Transit Switch Interfaces | 587
Example: Configuring Two or More Lossless FCoE Priorities on the Same FCoE Transit Switch Interface | 576
Example: Configuring Lossless FCoE Traffic When the Converged Ethernet Network Does Not Use IEEE 802.1p Priority 3 for FCoE Traffic (FCoE Transit Switch) | 565
Example: Configuring Lossless IEEE 802.1p Priorities on Ethernet Interfaces for Multiple Applications (FCoE and iSCSI) | 605

output-traffic-control-profile

Syntax

```
output-traffic-control-profile profile-name;
```

Hierarchy Level

```
[edit class-of-service interfaces interface-name forwarding-class-set forwarding-class-set-name]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Apply an output traffic scheduling and shaping profile to a forwarding class set (priority group).

Options

profile-name—Name of the traffic-control profile to apply to the specified forwarding class set.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

pfc (Input Congestion Notification)

Syntax

```
pfc {
    mru mru-value;
}
```

Hierarchy Level

```
[edit class-of-service congestion-notification-profile profile-name input dscp code-point code-point-bits]
[edit class-of-service congestion-notification-profile profile-name input ieee-802.1 code-point code-point-bits]
```

Release Information

Statement introduced in Junos OS Release 12.3 for the QFX Series.
Support for DSCP-based PFC introduced in Junos OS Release 17.4R1 for the QFX Series.

Description

Enable and configure ingress interface priority-based flow control (PFC) for a configured IEEE 802.1p code point in the VLAN-tagged packet header at Layer 2, or on a DSCP value from the Layer 3 IP header to support PFC with untagged traffic.

Options

The remaining statements are explained separately. Search for a statement in CLI Explorer or click a linked statement in the Syntax section for details.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring CoS PFC (Congestion Notification Profiles) | 194

Example: Configuring Two or More Lossless FCoE IEEE 802.1p Priorities on Different FCoE Transit Switch Interfaces | 587

Example: Configuring Two or More Lossless FCoE Priorities on the Same FCoE Transit Switch Interface | 576

Example: Configuring Lossless FCoE Traffic When the Converged Ethernet Network Does Not Use IEEE 802.1p Priority 3 for FCoE Traffic (FCoE Transit Switch) | 565

pfc-priority

Syntax

```
pfc-priority pfc-priority;
```

Hierarchy Level

```
[edit class-of-service forwarding-classes class class-name]
```

Release Information

Statement introduced in Junos OS Release 17.4R1 for the QFX Series.

Description

Explicitly map a forwarding class to a priority-based flow control (PFC) priority value.

To support lossless behavior for untagged traffic across Layer 3 connections to Layer 2 subnetworks, you can configure PFC to be invoked based on a configured 6-bit Distributed Services code point (DSCP) value in the Layer 3 IP header, rather than an IEEE 802.1p code point in a VLAN header at Layer 2. However, because PFC sends Layer 2 pause frames specifying a PFC priority on which to notify the peer about congestion, this statement defines the PFC priority to use in pause frames when PFC is triggered by a configured DSCP value.

DSCP-based PFC is used to support Remote Direct Memory Access (RDMA) over converged Ethernet version 2 (RoCEv2).

To enable DSCP-based PFC:

- Use this statement to map a lossless forwarding class to a PFC priority value to use in the PFC pause frames.
- Use the **[edit class-of-service congestion-notification-profile *name* input]** dscp statement to define an input congestion notification profile to enable PFC on traffic specified by a desired DSCP value.
- Use the **[edit class-of-service classifiers]** dscp statement to set up a DSCP classifier for the desired DSCP value and forwarding class mapped to a PFC priority above.

Options

pfc-priority—3-bit PFC priority value in decimal form (0-7).

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Understanding PFC Using DSCP at Layer 3 for Untagged Traffic | **242**

Configuring DSCP-based PFC for Layer 3 Untagged Traffic | **245**

Understanding CoS Flow Control (Ethernet PAUSE and PFC) | **197**

policy-options

Syntax

```
policy-options
    application-maps application-map-name {
        application application-name {
            code-points [ aliases ] [ bit-patterns ];
        }
    }
    policy-statement policy-name {
        term term-name {
            from {
                family family-name;
                match-conditions;
                policy subroutine-policy-name;
                prefix-list prefix-list-name;
                prefix-list-filter prefix-list-name match-type <actions>;
                route-filter destination-prefix match-type <actions>;
                source-address-filter source-prefix match-type <actions>;
            }
            to {
                match-conditions;
                policy subroutine-policy-name;
            }
            then actions;
        }
    }
```

Hierarchy Level

```
[edit]
```

Release Information

Statement introduced in Junos OS Release 12.1 for the QFX Series.
Statement introduced in Junos OS Release 12.1 for the EX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Configure options such as application maps for DCBX application protocol exchange and policy statements.

Required Privilege Level

storage—To view this statement in the configuration.
storage-control—To add this statement to the configuration.

RELATED DOCUMENTATION

priority (Schedulers)

Syntax

```
priority priority;
```

Hierarchy Level

```
[edit class-of-service schedulers scheduler-name]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.
medium-low and **medium-low** options introduced for QFX10000 Series switches in Junos OS 19.2R3.

Description

Specify the packet bandwidth-scheduling priority value.

> **NOTE:** On QFabric systems, the **priority** statement is valid only for Node device queue scheduling. The **priority** statement is notallowed for Interconnect device queue scheduling. If you map a schedulerthat includes a **priority** configuration to a fabric forwardingclass at the [edit class-of-service scheduler-map-fcset] hierarchylevel, the system generates a commit error. (On the Interconnect device,fabric fc-sets are not user-definable. Only the **fabric_fcset_strict_high** fabric fc-set is configured with high priority, and this configurationcannot be changed.)

Options

priority—It can be one of the following:

- **low**—Scheduler has low priority.
- **medium-low**—Scheduler has medium-low priority. (QFX10000 Series switches only)
- **medium-high**—Scheduler has medium-high priority. (QFX10000 Series switches only)
- **high**—Scheduler has high priority. Assigning high priority to a queue prevents the queue from being underserved. (QFX10000 Series switches only)
- **strict-high**—Scheduler has strict high priority. On QFX5100, EX4600, QFX3500, and QFX3600 switches, and on QFabric systems, you can configure only one queue as a strict-high priority queue. On QFX10000 switches, you can configure as many strict-high priority queues as you want. However, because strict-high priority traffic takes precedence over all other traffic, too much strict-high priority traffic can starve the other output queues.

Strict-high priority allocates the scheduled bandwidth to the packets on the queue before any other queue receives bandwidth. Other queues receive the bandwidth that remains after the strict-high queue has been serviced.

> **NOTE:** On QFX10000 switches, we strongly recommend that you apply a transmit rate to strict-high priority queues to prevent them from starving other queues. A transmit rate configured on a strict-high priority queue limits the amount of traffic that receives strict-high priority treatment to the amount or percentage set by the transmit rate. 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.
>
> On QFX5100, EX4600, QFX3500, and QFX3600 switches, and on QFabric systems, we recommend that you always apply a shaping rate to strict-high priority queues to prevent them from starving other queues. A shaping rate (shaper) sets the maximum amount of bandwidth a queue can consume. (Unlike using the transmit rate on a QFX10000 switch to limit traffic that receives strict-high priority treatment, traffic that exceeds the shaping rate is dropped, and is not treated as best-effort traffic that shares in excess bandwidth.) 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.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

priority-flow-control

Syntax

```
priority-flow-control {
    no-auto-negotiation;
}
```

Hierarchy Level

```
[edit protocols dcbx interface (all | interface-name)]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 11.3 for EX Series switches.

Description

Disable autonegotiation of priority-based flow control (PFC) on one or more Ethernet interfaces. Autonegotiation enables PFC on an interface only if the switch and the peer device connected to the switch both support PFC and have the same PFC configuration. Disabling autonegotiation on an interface forces the interface to use the PFC state (enabled or disabled) that is configured on the switch by the configuration and assignment of the congestion notification profile.

Options

no-auto-negotiation—Disable automatic negotiation of PFC.

Required Privilege Level

routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

RELATED DOCUMENTATION

show dcbx neighbors | 977

Configuring CoS PFC (Congestion Notification Profiles) | 194

Configuring Priority-Based Flow Control for an EX Series Switch (CLI Procedure)

Configuring DCBX Autonegotiation | 465

Example: Configuring CoS PFC for FCoE Traffic | 490

Understanding Data Center Bridging Capability Exchange Protocol for EX Series Switches

Understanding Priority-Based Flow Control

Understanding DCB Features and Requirements | 450

protocol (Applications)

Syntax

```
protocol (tcp | udp);
```

Hierarchy Level

```
[edit applications application application-name]
```

Release Information

Statement introduced in Junos OS Release 12.1 for EX Series switches.
Statement introduced in Junos OS Release 12.1 for the QFX Series.

Description

Networking protocol type, which combines with **destination-port** to identify an application type.

> **NOTE:** To create an application for iSCSI, use the protocol **tcp** with the destination port number **3260**.

Options

tcp—Transmission Control Protocol

udp—User Datagram Protocol

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Defining an Application for DCBX Application Protocol TLV Exchange | 473
Example: Configuring DCBX Application Protocol TLV Exchange | 476
Example: Configuring DCBX to Support an iSCSI Application
Understanding DCBX Application Protocol TLV Exchange | 468
Understanding DCBX Application Protocol TLV Exchange on EX Series Switches

protocol (Drop Profile Map)

Syntax

```
protocol protocol drop-profile profile-name;
```

Hierarchy Level

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

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Configure the protocol type for the specified weighted random early detection (WRED) drop profile.

Options

protocol—Type of protocol. The protocol can be:

- **any**—Accept any protocol type.

The remaining statement is explained separately. See CLI Explorer.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

queue-num

Syntax

```
queue-num queue-number <no-loss>;
```

Hierarchy Level

```
[edit class-of-service forwarding-classes class class-name]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
No-loss option introduced in Junos OS Release 12.3 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Map a forwarding class to an output queue number. Optionally, configure the forwarding class as a lossless forwarding class. Each switch provides enough output queues so that you can map forwarding classes to queues on a one-to-one basis, so each forwarding class can have a dedicated output queue.

On switches that use different forwarding classes and output queues for unicast and multidestination (multicast, broadcast, destination lookup fail) traffic, the switch supports 12 forwarding classes and 12 output queues, eight of each for unicast traffic and four of each for multidestination traffic. You can map some or all of the eight unicast forwarding classes to a unicast queue (0 through 7) and some or all of the four multidestination forwarding classes to the a multidestination queue (8 through 11). You cannot map a forwarding class to more than one queue (each forwarding class maps to one and only one queue), but you can map multiple forwarding classes to one queue. The queue to which you map a forwarding class determines if the forwarding class is a unicast or multidestination forwarding class.

On switches that use the same forwarding classes and output queues for unicast and multidestination traffic, the switch supports eight forwarding classes and eight output queues. You can map some or all of the eight of the forwarding classes to queues (0 through 7). You cannot map a forwarding class to more than one queue (each forwarding class maps to one and only one queue), but you can map multiple forwarding classes to one queue.

You cannot configure weighted random early detection (WRED) packet drop on forwarding classes configured with the no-loss packet drop attribute. Do not associate a drop profile with lossless forwarding classes. Instead, use priority-based flow control (PFC) to prevent frame drop on lossless forwarding classes.

NOTE: 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 (all of the forwarding classes must be lossy, or all of the forwarding classes mapped to a queue must be lossless).

OCX Series switches do not support the no-loss packet drop attribute and do not support lossless forwarding classes. On OCX Series switches, do not configure the no-loss packet drop attribute on forwarding classes, and do not map traffic to the default **fcoe** and **no-loss** forwarding classes (both of these default forwarding classes carry the no-loss packet drop attribute).

NOTE: On systems that do not use the ELS CLI, if you are using Junos OS Release 12.2, use the default forwarding-class-to-queue mapping for the lossless **fcoe** and **no-loss** forwarding classes. If you explicitly configure lossless forwarding classes, the traffic mapped to those forwarding classes is treated as lossy (best effort) traffic and does *not* receive lossless treatment.

NOTE: On systems that do not use the ELS CLI, if you are using Junos OS Release 12.3 or later, the default configuration is the same as the default configuration for Junos OS Release 12.2, and the default behavior is the same (the **fcoe** and **no-loss** forwarding classes receive lossless treatment). However, if you explicitly configure lossless forwarding classes, you can configure up to six lossless forwarding classes by specifying the **no-loss** option. If you do not specify the **no-loss** option in an explicit forwarding class configuration, the forwarding class is lossy. For example, if you explicitly configure the **fcoe** forwarding class and you do not include the **no-loss** option, the **fcoe** forwarding class is lossy, not lossless.

Options

queue-number—(Switches that use different output queues for unicast and multidestination traffic) Number of the CoS unicast queue (0 through 7) or the CoS multidestination queue (8 through 11).

queue-number—(Switches that use the same output queues for unicast and multidestination traffic) Number of the CoS queue (0 through 7).

no-loss—Optional packet drop attribute keyword to configure the forwarding class as lossless.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

recommendation-tlv

Syntax

```
recommendation-tlv {
    no-auto-negotiation;
}
```

Hierarchy Level

```
[edit protocols dcbx interface interface-name enhanced-transmission-selection]
```

Release Information

Statement introduced in Junos OS Release 12.2 for the QFX Series.

Description

Enable DCBX to send the ETS Recommendation TLV (also known as the Information TLV) on egress. This feature is valid only if the interface DCBX mode is IEEE DCBX. If the interface DCBX mode is DCBX version 1.01, this statement has no effect. (DCBX version 1.01 does not advertise separate TLVs for individual attributes.)

Default

DCBX-enabled interfaces send the ETS recommendation TLV unless it is disabled.

Options

no-auto-negotiation—Disable sending of the ETS recommendation TLV.

Required Privilege Level

routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

RELATED DOCUMENTATION

show dcbx neighbors | 977
Configuring DCBX Autonegotiation | 465

rewrite-rules

List of Syntax

Syntax (Rewrite Rule Configuration) on page 834
Syntax (Rewrite Rule Association with Interface) on page 834

Syntax (Rewrite Rule Configuration)

```
rewrite-rules {
  (dscp | dscp-ipv6 | ieee-802.1 | exp) rewrite-name {
    import (rewrite-name | default);
    forwarding-class class-name {
      loss-priority priority code-point (alias | bits);
    }
  }
}
```

Syntax (Rewrite Rule Association with Interface)

```
rewrite-rules {
  (dscp | dscp-ipv6 | ieee-802.1 | exp) rewrite-name;
}
```

Hierarchy Level (Rewrite Rule Configuration)

```
[edit class-of-service],
```

Hierarchy Level (Rewrite Rule Association with Interface)

```
[edit class-of-service interfaces interface-name unit logical-unit-number]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
EXP statement introduced in Junos OS Release 12.3 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Configure rewrite rules that map traffic to code points when traffic exits the system, and apply the rewrite rules to a specific interface.

MPLS EXP rewrite rules can only be bound to logical interfaces, not to physical interfaces. You can configure up to 64 EXP rewrite rules, but you can use only 16 EXP rewrite rules on switch interfaces at any given time.

NOTE: OCX Series switches do not support MPLS, so they do not support EXP rewrite rules.

Options

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Defining CoS Rewrite Rules | 114
Configuring Rewrite Rules for MPLS EXP Classifiers | 136
Understanding CoS Rewrite Rules | 111
Understanding CoS MPLS EXP Classifiers and Rewrite Rules | 132

rx-buffers

Syntax

```
rx-buffers (on | off);
```

Hierarchy Level

[edit interfaces *interface-name* ether-options **configured-flow-control**]

Release Information

Statement introduced in Junos OS Release 12.1 for the QFX Series.

Description

Enable or disable an interface to generate and send Ethernet PAUSE messages. If you enable the receive buffers to generate and send PAUSE messages, when the receive buffers reach a certain level of fullness, the interface sends a PAUSE message to the connected peer. If the connected peer is properly configured, it stops transmitting frames to the interface on the entire link. When the interface receive buffer empties below a certain threshold, the interface sends a message to the connected peer to resume sending frames.

Ethernet PAUSE prevents buffers from overflowing and dropping packets during periods of network congestion. If the other devices in the network are also configured to support PAUSE, PAUSE supports lossless operation. Use the **rx-buffers** statement with the **tx-buffers** statement to configure asymmetric Ethernet PAUSE on an interface. (Use the **flow-control** statement to enable symmetric PAUSE and the **no-flow-control** statement to disable symmetric PAUSE on an interface. Symmetric flow control and asymmetric flow control are mutually exclusive features. If you attempt to configure both, the switch returns a commit error.)

NOTE: Ethernet PAUSE temporarily stops transmitting all traffic on a link when the buffers fill to a certain threshold. To temporarily pause traffic on individual "lanes" of traffic (each lane contains the traffic associated with a particular IEEE 802.1p code point, so there can be eight lanes of traffic on a link), use priority-based flow control (PFC).

Ethernet PAUSE and PFC are mutually exclusive features, so you cannot configure both of them on the same interface. If you attempt to configure both Ethernet PAUSE and PFC on an interface, the switch returns a commit error.

Default

Flow control is disabled. You must explicitly configure Ethernet PAUSE flow control on interfaces.

Options

on | off—Enable or disable an interface to generate and send Ethernet PAUSE messages.

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

flow-control | **771**

tx-buffers | **868**

Configuring CoS Asymmetric Ethernet PAUSE Flow Control | **212**

Enabling and Disabling CoS Symmetric Ethernet PAUSE Flow Control | **211**

Understanding CoS Flow Control (Ethernet PAUSE and PFC) | **197**

scheduler

Syntax

```
scheduler scheduler-name;
```

Hierarchy Level

```
[edit class-of-service scheduler-maps map-name forwarding-class class-name]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Map a scheduler to a forwarding class using a scheduler map.

> **NOTE:** On QFX5200 only, absolute CoS rate limits for transmit rate and shaping rate do not reflect 50g and 100g interfaces. Therefore this statement does not affect those interfaces for QFX5200 in release 15.1X53-D30.

Options

scheduler-name—Name of the scheduler to map to the forwarding class.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

scheduler-map

Syntax

```
scheduler-map map-name;
```

Enhanced Transmission Selection (ETS) Hierarchical Scheduling

```
[edit class-of-service traffic-control-profiles traffic-control-profile-name]
```

Port Scheduling

```
[edit class-of-service interfaces interface-name]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Associate a scheduler map with a traffic control profile.

Options

map-name—Name of the scheduler map.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

scheduler-maps

Syntax

```
scheduler-maps {
    map-name {
        forwarding-class class-name scheduler scheduler-name;
    }
}
```

Hierarchy Level

```
[edit class-of-service]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Specify a scheduler map name to map a scheduler configuration to a forwarding class.

Options

map-name—Name of the scheduler map.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

schedulers

List of Syntax

QFX5100, EX4600, QFX3500, and QFX3600, Switches, QFabric Systems on page 841
QFX10000 Switches on page 841

QFX5100, EX4600, QFX3500, and QFX3600, Switches, QFabric Systems

```
schedulers {
   scheduler-name  {
      buffer-size (percent percentage | remainder);
      drop-profile-map loss-priority (low | medium-high | high) protocol protocol drop-profile drop-profile-name;
      explicit-congestion-notification;
      priority priority;
      shaping-rate (rate | percent percentage);
      transmit-rate (percent percentage);
   }
}
```

QFX10000 Switches

```
schedulers {
   scheduler-name  {
      buffer-size (percent percentage | remainder);
      drop-profile-map loss-priority (low | medium-high | high) protocol protocol drop-profile drop-profile-name;
      excess-rate;
      explicit-congestion-notification;
      priority priority;
      shaping-rate (rate | percent percentage);
      transmit-rate (percent percentage) <exact>;
   }
}
```

Hierarchy Level

```
[edit class-of-service]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Specify scheduler name and parameter values such as minimum bandwidth (**transmit-rate**), maximum bandwidth (**shaping-rate**), and priority (**priority**).

Options

scheduler-name —Name of the scheduler.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

shaping-rate

Syntax

```
shaping-rate (rate | percent percentage);
```

Hierarchy Level

```
[edit class-of-service schedulers scheduler-name],
[edit class-of-service traffic-control-profiles profile-name]
```

NOTE: Only switches that support enhanced transmission selection (ETS) hierarchical scheduling support the **traffic-control-profiles** hierarchy.

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Configure the shaping rate. The shaping rate throttles the rate of packet transmission by setting a maximum bandwidth (rate in bits per second) or a maximum percentage of bandwidth for a queue or a forwarding class set. You specify the maximum bandwidth for a queue by using a scheduler map to associate a forwarding class (queue) with a scheduler that has a configured shaping rate.

For ETS configuration, you specify the maximum bandwidth for a forwarding class set by setting the shaping rate for a traffic control profile, then you associate the scheduler map with the traffic control profile, and then you apply the traffic control profile and a forwarding class set to an interface.

For simple port scheduling configuration, you apply the scheduler map directly to an interface (instead of indirectly through the traffic control profile as in ETS).

We recommend that you configure the shaping rate as an absolute maximum usage and not as additional usage beyond the configured transmit rate (the minimum guaranteed bandwidth for a queue) or the configured guaranteed rate (the minimum guaranteed bandwidth for a forwarding class set).

NOTE: When you set the maximum bandwidth (**shaping-rate** value) for a queue or for a priority group at 100 Kbps or less, the traffic shaping behavior is accurate only within +/– 20 percent of the configured **shaping-rate** value.

NOTE: On QFX5200, QFX5100, EX4600, QFX3500, and QFX3600 switches, and on QFabric systems, 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.

NOTE: On QFX5200 Series switches, a granularity of 64kbps is supported for the shaping rate. Therefore, the shaping rate on queues for 100g interfaces might not be applied correctly.

NOTE: QFX10000 Series switches do not support the shaping-rate statement. However, you can configure the transmit-rate **exact** option to prevent a queue from consuming more bandwidth than you want the queue to consume.

On QFX10000 Series switches, we recommend that you use the transmit rate to set a limit on the amount of bandwidth that receives strict-high priority treatment on a strict-high priority queue. Traffic up to the transmit rate receives strict-high priority treatment. Traffic in excess of the transmit rate is treated as best-effort traffic that receives the strict-high priority queue excess rate weight of “1”. Do not use a shaping rate to set a maximum bandwidth limit on strict-high priority queues on QFX10000 Series switches.

Default

If you do not configure a shaping rate, the default shaping rate is 100 percent (all of the available bandwidth), which is the equivalent of no rate shaping.

Options

percent *percentage*—Shaping rate as a percentage of the available interface bandwidth.

Range: 1 through 100 percent

rate—Peak (maximum) rate, in bits per second (bps). You can specify a value in bits per second either as a complete decimal number or as a decimal number followed by the abbreviation k (1000), m (1,000,000), or g (1,000,000,000).

Range: 1000 through 10,000,000,000 bps

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

shared-buffer

Syntax

```
shared-buffer {
  egress {
    buffer-partition (lossless | lossy | multicast) {
      percent percent;
      dynamic-threshold threshold-value;
    }
    percent percent;
  }
  ingress {
    percent percent;
    buffer-partition (lossless | lossless-headroom | lossy) {
      percent percent;
      dynamic-threshold threshold-value;
    }
  }
}
```

Hierarchy Level

```
[edit class-of-service]
```

Release Information

Statement introduced in Junos OS Release 12.3 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Configure the global shared buffer pool allocation to ports. Shared buffers are a pool of buffer space that the system can allocate dynamically across all of its ports as memory space is needed. Some buffer space is reserved for dedicated buffers (buffers allocated permanently to ports), headroom buffers (buffers that help prevent packet loss on lossless flows), and other buffers.

The switch uses the shared-buffer pool to absorb traffic bursts after the dedicated-buffer-pool is exhausted. The shared pool threshold is dynamically calculated based on a factor called "*alpha*".

Configure the way the system uses the available (user-configurable) buffer space by setting the **shared-buffer** percentage for the ingress buffer pool and for the egress buffer pool.

The percentage you specify is the percentage of available buffer space allocated to the global shared ingress buffer pool or to the global shared egress buffer pool. If you allocate less than 100 percent of the available buffer space to the shared buffer pool, the remaining buffer space is added to the dedicated buffer pool. (You cannot directly configure the dedicated buffer pool for each port; dedicated buffers are allocated evenly across all the ports.)

You can adjust the maximum size of the shared-buffer pool by configuring the dynamic-threshold values:

- By adjusting the value for the egress partition (the calculation includes the alpha value and the number of competing queues).

[edit class-of-service shared-buffer egress buffer-partition dynamic-threshold *threshold-value*]

- By adjusting the value for the ingress partition (the calculation includes the alpha value and the number of competing queues).

[edit class-of-service shared-buffer ingress buffer-partition dynamic-threshold *threshold-value*

> **CAUTION:** Changing the buffer configuration is a disruptive event. Traffic stops on *all* ports until the buffer reprogramming is complete.

You can also partition the ingress shared buffer pool and the egress shared buffer pool to adjust the buffer allocations for different mixes of network traffic (best-effort, lossless, multicast) using the **buffer-partition** statement.

NOTE: If you commit a buffer configuration for which the switch does not have sufficient resources, the switch might log an error instead of returning a commit error. In that case, a syslog message is displayed on the console. For example:

```
user@host# commit
configuration check succeeds

Message from syslogd@host at Jun 13 11:11:10  ...
host dc-pfe: Not enough Ingress Lossless headroom.(Already allocated more).
 Dedicated : 14340 Lossy : 47100 Lossless 4239 Headroom 21195 Avail : 20781
commit complete
```

If the buffer configuration commits but you receive a syslog message that indicates the configuration cannot be implemented, you can:

- Reconfigure the buffers or reconfigure other parameters (for example, the PFC configuration, which affects the need for lossless headroom buffers and lossless buffers—the more priorities you pause, the more lossless and lossless headroom buffer space you need), then attempt the commit operation again.
- Roll back the switch to the last successful configuration.

If you receive a syslog message that says the buffer configuration cannot be implemented, you must take corrective action. If you do not fix the configuration or roll back to a previous successful configuration, the system behavior is unpredictable.

Options

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Best-Effort Unicast Traffic | **659**

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Multicast Traffic | **675**

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Lossless Traffic | **683**

Configuring Global Ingress and Egress Shared Buffers | **657**

Understanding CoS Buffer Configuration | **635**

system-defaults

Syntax

```
system-defaults {
    classifiers exp classifier-name;
}
```

Hierarchy Level

```
[edit class-of-service]
```

Release Information

Statement introduced in Junos OS Release 12.3X50 for the QFX Series.

Description

Configure the global EXP classifier used on all interfaces to classify MPLS traffic. You can configure up to 64 EXP classifiers. However, the switch uses only one EXP classifier as a global MPLS classifier on all interfaces. If you configure a global system default EXP classifier, then all switch interfaces use that EXP classifier to classify MPLS traffic.

On switches that have a default EXP classifier, if you do not configure a global system default classifier, interfaces configured as **family mpls** use the default EXP classifier.

On switches that do not have a default EXP classifier (QFX5100, QFX3500, QFX3600, QFabric systems, EX4600), if you do not configure a global system default EXP classifier, 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 are applied to the interface, MPLS traffic is treated as best-effort traffic using the IEEE 802.1 default untrusted classifier. 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.

Options

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

traceoptions (Class of Service)

Syntax

```
traceoptions {
    file filename <size size> <files number>
    <world-readable | no-world-readable>;
    flag flag <flag-modifier>;
    no-remote-trace
}
```

Hierarchy Level

```
[edit class-of-service]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Set class-of-service (CoS) tracing options.

> **NOTE:** The **traceoptions** statement is not supported on the QFabric system.

Default

Traceoptions is disabled.

Options

file *filename*—Name of the file to receive the tracing operation output. Enclose the name in quotation marks. Traceoption output files are located in the **/var/log/** directory.

files *number*—(Optional) Maximum number of trace files. When a trace file named ***trace-file*** reaches its maximum size, it is renamed ***trace-file*.0**. The traceoption output continues in a second trace file named ***trace-file*.1**. When ***trace-file*.1** reaches its maximum size, output continues in a third file named ***trace-file*.2**, and so on. When the maximum number of trace files is reached, the oldest trace file is overwritten.

If you specify a maximum number of files, you must also specify a maximum file size with the size option.

Range: 2 through 1000 files

Default: 1 trace file

flag—Tracing operation to perform. To specify more than one tracing operation, include multiple **flag** statements:

- **all**—Trace all operations.
- **asynch**—Trace asynchronous configuration processing.
- **chassis-scheduler**—Trace chassis stream scheduler processing.
- **cos-adjustment**—Trace CoS rate adjustments.
- **dynamic**—Trace dynamic CoS functions.
- **hardware-database**—Trace the chassis hardware database related processing.
- **init**—Trace initialization events.
- **performance-monitor**—Trace performance monitor countors.
- **process**—Trace configuration processing.
- **restart**—Trace restart processing.
- **route-socket**—Trace route-socket events.
- **show**—Trace show command servicing.
- **snmp**—Trace SNMP-related processing.
- **util**—Trace utilities.

The following are the global tracing options:

- **all**—Perform all tracing operations
- **parse**—Trace parser processing.

no-remote-trace—(Optional) Disable remote tracing.

no-world-readable—(Optional) Prevent any user from reading the log file.

size *size*—(Optional) Maximum size of each trace file, in kilobytes (KB), megabytes (MB), or gigabytes (GB). When a trace file named ***trace-file*** reaches its maximum size, it is renamed ***trace-file*.0.** Incoming tracefile data is logged in the now empty ***trace-file***. When ***trace-file*** again reaches its maximum size, ***trace-file*.0** is renamed ***trace-file*.1** and ***trace-file*** is renamed ***trace-file*.0**. This renaming scheme continues until the maximum number of trace files is reached. Then the oldest trace file is overwritten.

If you specify a maximum file size, you must also specify a maximum number of trace files with the **files** option.

Syntax: ***x*k** to specify KB, ***x*m** to specify MB, or ***x*g** to specify GB

Range: 10 KB through the maximum file size of 4 GB (maximum is lower if 4 GB is not supported on your system)

Default: 1 MB

world-readable—(Optional) Allow any user to read the log file.

Required Privilege Level

routing—To view this statement in the configuration.

routing-control—To add this statement to the configuration.

traffic-control-profiles

Syntax

EX Series (Except EX4600), M Series, MX Series, PTX Series, T Series

```
traffic-control-profiles profile-name {
    adjust-minimum rate;
    atm-service (cbr | rtvbr | nrtvbr);
    delay-buffer-rate (percent percentage | rate);
    excess-rate (percent percentage  | proportion value );
    excess-rate-high (percent percentage | proportion value);
    excess-rate-low (percent percentage | proportion value);
    guaranteed-rate (percent percentage | rate) <burst-size bytes>;
    max-burst-size cells;
    overhead-accounting (frame-mode | cell-mode | frame-mode-bytes | cell-mode-bytes) <bytes
        (byte-value)>;
    peak-rate rate;
    scheduler-map map-name;
    shaping-rate (percent percentage | rate) <burst-size bytes>;
    shaping-rate-excess-high (percent percentage | rate) <burst-size bytes>;
    shaping-rate-excess-medium-high (percent percentage | rate) <burst-size bytes>;
    shaping-rate-excess-medium-low (percent percentage | rate) <burst-size bytes>;
    shaping-rate-excess-low (percent percentage | rate) <burst-size bytes>;
    shaping-rate-priority-high (percent percentage | rate) <burst-size bytes>;
    shaping-rate-priority-low (percent percentage | rate) <burst-size bytes>;
    shaping-rate-priority-medium (percent percentage | rate) <burst-size bytes>;
    shaping-rate-priority-medium-low (percent percentage | rate) <burst-size bytes>;
    shaping-rate-priority-strict-high (percent percentage | rate) <burst-size bytes>;
    strict-priority-scheduler;
    sustained-rate rate;
}
```

QFX Series including QFabric, OCX OCX1100, EX4600, NFX Series

```
traffic-control-profiles profile-name {
    guaranteed-rate (rate| percent percentage);
    scheduler-map map-name;
    shaping-rate (rate| percent percentage);
}
```

ACX Series

```
traffic-control-profiles profile-name {
    atm-service (cbr | nrtvbr | rtvbr);
    delay-buffer-rate cps;
    max-burst-size max-burst-size;
    peak-rate peak-rate;
    sustained-rate sustained-rate;
}
```

Hierarchy Level

```
[edit class-of-service]
```

Release Information

Statement was introduced in Junos OS Release 7.6 (EX series, M series, MX series, T series, and PTX series devices).
Statement was introduced in Junos OS Release 11.1 for the QFX Series.
Statement was introduced in Junos OS Release 12.3 for ACX series routers.
Statement was introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

ACX Series Routers

Configure traffic-shaping profiles.

> **NOTE:** For CoS on ACX6360-OR, see the documentation for the PTX1000.

EX Series (Except EX4600), M Series, MX Series, T Series, and PTX Series Routers

For Gigabit Ethernet IQ, Channelized IQ PICs, FRF.15 and FRF.16 LSQ interfaces, Enhanced Queuing (EQ) DPCs, and PTX Series routers only, configure traffic shaping and scheduling profiles. For Enhanced EQ PICs, EQ DPCs, and PTX Series routers only, you can include the **excess-rate** statement.

QFX Series QFabric, OCX1100, EX4600, NFX Series

Configure traffic shaping and scheduling profiles for forwarding class sets (priority groups) to implement enhanced transmission selection (ETS) or for logical interfaces.

Options

profile-name—Name of the traffic-control profile. This name is also used to specify an output traffic control profile.

The remaining statements are explained separately. See CLI Explorer or click a linked statement in the Syntax section for details.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Oversubscribing Interface Bandwidth

Understanding Scheduling on PTX Series Routers

Example: Configuring CoS Hierarchical Port Scheduling (ETS) | **415**

Example: Configuring Traffic Control Profiles (Priority Group Scheduling) | **385**

Example: Configuring Forwarding Class Sets | **165**

Assigning CoS Components to Interfaces | **77**

output-traffic-control-profile | **819**

Understanding CoS Traffic Control Profiles | **372**

traffic-manager

List of Syntax

Syntax (MX Series, PTX Series) Configure Queue Monitoring

```
traffic-manager {
    egress-shaping-overhead number;
    ingress-shaping-overhead number;
    mode {
        egress-only;
        ingress-and-egress;
        session-shaping;
    }
    enhanced-priority-mode;
    no-enhanced-priority-mode;
    packet-timestamp {
        enable;
    }
    queue-threshold {
        fabric-queue {
            priority high/low {
                threshold threshold-percentage;
            }
        }
        wan-queue {
            priority high/medium-high/medium-low/low {
                threshold threshold-percentage;
            }
        }
    }
}
```

Syntax (MX Series, T Series)

```
traffic-manager {
    egress-shaping-overhead number;
    ingress-shaping-overhead number;
    mode {
```

```
      egress-only;
      ingress-and-egress;
    }
}
```

Syntax (M Series)

```
traffic-manager {
    egress-shaping-overhead number;
    ingress-shaping-overhead number;
    mode {
        egress-only;
        ingress-and-egress;
        session-shaping;
    }
}
```

Syntax (QFX Series)

```
traffic-manager {
    buffer-monitor-enable;
    packet-timestamp {
        enable;
    }
    queue-threshold {
        fabric-queue {
            priority high/low {
                threshold threshold-percentage;
            }
        }
        wan-queue {
            priority high/medium-high/medium-low/low {
                threshold threshold-percentage;
            }
        }
    }
}
```

Syntax (vSRX)

```
traffic-manager {
    egress-shaping-overhead number;
}
```

Hierarchy Level

```
[edit chassis fpc slot-number],
[edit chassis fpc slot-number pic pic-number],
[edit chassis lcc number fpc slot-number pic pic-number] (Routing Matrix)
```

Release Information

Statement introduced in Junos OS Release 8.3.

Description

Enable CoS queuing, scheduling, and shaping on an L2TP session.

> **NOTE:** Committing changes to **traffic-manager** automatically restarts any necessary components (PICs, DPCs, or FPCs).

Options

buffer-monitor-enable—QFX5000 Series only. Enable port buffer monitoring. Buffer utilization data is collected in one-second intervals and compared with the data from the previous interval. The larger value is kept to keep track of peak buffer occupancy for each queue or priority group.

queue-threshold—Enable monitoring of Fabric and WAN queues. When the **fabric-queue** statement is configured, an SNMP trap is generated whenever the fabric power utilization exceeds the configured threshold value.

When **wan-queue** is configured, an SNMP trap is generated whenever the WAN queue depth exceeds the configured threshold value.

egress-shaping-overhead ***number***—When traffic management (queueing and scheduling) is configured on the egress side, the number of CoS shaping overhead bytes to add to the packets on the egress interface.

Replace ***number*** with a value from **-63** through **192** bytes.

For vSRX, replace ***number*** with a value from **-62** through **192** bytes.

> **NOTE:** The L2 headers (DA/SA + VLAN tags) are automatically a part of the shaping calculation.

ingress-shaping-overhead ***number***—When L2TP session shaping is configured, the number of CoS shaping overhead bytes to add to the packets on the ingress side of the L2TP tunnel to determine the shaped session packet length.

When session shaping is not configured and traffic management (queueing and scheduling) is configured on the ingress side, the number of CoS shaping overhead bytes to add to the packets on the ingress interface.

Replace ***number*** with a value from **—63** through **192** bytes.

mode—Configure CoS traffic manager mode of operation. This option has the following suboptions:

- **egress-only**—Enable CoS queuing and scheduling on the egress side for the PIC that houses the interface. This is the default mode for an Enhanced Queueing (EQ) DPC on MX Series routers.

  > **NOTE:** If ingress packet drops are observed at a high rate for an IQ2 or IQ2E PIC, configure the **traffic-manager** statement to work in the **egress-only** mode.

- **ingress-and-egress**—Enable CoS queueing and scheduling on both the egress and ingress sides for the PIC. This is the default mode for IQ2 and IQ2E PICs on M Series and T Series routers.

> **NOTE:**
> - For EQ DPCs, you must configure the **traffic-manager** statement with **ingress-and-egress** mode to enable ingress CoS on the EQ DPC.
> - EQ DPCs have 250 ms of buffering, with only egress queueing (default mode). When **ingress-and-egress** is configured, the buffer is partitioned as 50 ms for the ingress direction and 200 ms for the egress direction.

- **session-shaping**—(M Series routers only) Configure the IQ2 PIC mode for session-aware traffic shaping to enable L2TP session shaping.

enhanced-priority-mode—Enable the enhanced priority mode. When you enable the enhanced priority mode, the scheduler supports four additional per-priority shaping rates and two additional excess priorities at the interface and interface set level. The four additional per-priority shaping rates are: Guaranteed Strict-high, Guaranteed Medium-low, Excess medium-high, and Excess medium-low. The two additional excess priorities are: Excess-rate Medium- high and Excess-rate Medium-low. This is the default mode for PTX Series routers.

no-enhanced-priority-mode—Disable the enhanced priority mode. This is the default mode for MX Series routers.

> **NOTE:** The line card reboots when you enable or disable the enhanced priority mode feature.

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring CoS for L2TP Tunnels on ATM Interfaces

Enabling a Timestamp for Ingress and Egress Queue Packets

show interfaces queue | 1010

transmit-rate

List of Syntax

QFX5100, EX4600, QFX3500, and QFX3600, Switches, QFabric Systems

```
transmit-rate (rate | percent percentage);
```

QFX10000 Switches

```
transmit-rate (rate | percent percentage) <exact>;
```

Hierarchy Level

```
[edit class-of-service schedulers scheduler-name]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.
Exact option introduced in Junos OS Release 15.1X53-D10 for the QFX Series.

Description

On QFX5100, EX4600, QFX3500, and QFX3600 switches, and on QFabric systems, the transmit rate specifies the minimum guaranteed transmission rate or percentage for a queue (forwarding class) scheduler. The queue transmit rate also determines the amount of excess (extra) priority group bandwidth that the queue can share on switches that support enhanced transmission selection (ETS) hierarchical scheduling.

On QFX10000 switches, the transmit rate specifies the minimum guaranteed transmission rate or percentage for a queue (forwarding class) scheduler. The queue transmit rate also determines the amount of excess (extra) port bandwidth the queue can share if you do not explicitly configure an excess rate in the scheduler. The transmit rate also determines the amount of excess (extra) priority group bandwidth that the queue can share on switches that support enhanced transmission selection (ETS) hierarchical scheduling.

On QFX10000 switch strict-high priority queues, the transmit rate limits the amount of traffic the switch treats as strict-high priority traffic. Traffic up to the transmit rate receives strict-high priority treatment. The switch treats traffic that exceeds the transmit rate as best-effort traffic that receives an excess bandwidth sharing weight of “1”; you cannot configure an excess rate on a strict-high priority queue, and unlike queues with other scheduling priorities, the switch does not use the transmit rate to determine extra bandwidth sharing for strict-high priority queues.

CAUTION: We strongly recommend that you configure a transmit rate on strict-high priority queues to limit the amount of traffic the switch treats as strict-high priority traffic on those queues. This is especially important if you configure more than one strict-high priority queue on a port. To prevent a strict-high priority queue from starving the other queues on a port, we recommend that you always configure a transmit rate, even if you only configure one strict-high priority queue.

NOTE: For ETS, the **transmit-rate** setting works only if you also configure the **guaranteed-rate** in the traffic control profile that is attached to the forwarding class set to which the queue belongs. If you do not configure the guaranteed rate, the minimum guaranteed rate for individual queues that you set using the **transmit-rate** statement does not work. The sum of all queue transmit rates in a forwarding class set should not exceed the traffic control profile guaranteed rate.

NOTE: On QFX5100, EX4600, QFX3500, and QFX3600 switches, and on QFabric systems, you cannot configure a transmit rate for a strict-high priority queue. Queues (forwarding classes) with a configured transmit rate cannot be included in a forwarding class set that has a strict-high priority queue. To prevent strict-high priority queues from consuming all of the available bandwidth on these switches, we recommend that you configure a shaping rate to set a maximum amount of bandwidth for strict-high priority queues.

NOTE: 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 may 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.

Default

On QFX5100, EX4600, QFX3500, and QFX3600 switches, and on QFabric systems, if you do not configure the transmit rate, the default scheduler transmission rate and buffer size percentages for queues 0 through 11 are:

Table 133: Default Transmit Rates for QFX5100, EX4600, QFX3500, and QFX3600 Switches, and QFabric Systems

Queue Number	Default Minimum Guaranteed Bandwidth (Transmit Rate)
0 (best-effort)	5 %
1	0
2	0
3 (fcoe)	35 %
4 (no-loss)	35 %
5	0
6	0
7 (network control)	5 %
8 (mcast)	20 %
9	0
10	0
11	0

NOTE: OCX Series switches do not support lossless transport. The OCX Series default DSCP classifier does not classify traffic into the default lossless fcoe and no-loss forwarding classes. The bandwidth that the default scheduler allocates to the default fcoe and no-loss forwarding classes on other switches is allocated to the default best-effort, network-control, and mcast forwarding classes on OCX Series switches.

On QFX10000 switches, if you do not configure the transmit rate, the default scheduler transmission rate and buffer size percentages for queues 0 through 7 are:

Table 134: Default Transmit Rates for QFX10000 Switches

Queue Number	Default Minimum Guaranteed Bandwidth (Transmit Rate)
0 (best-effort)	15 %
1	0
2	0
3 (fcoe)	35 %
4 (no-loss)	35 %
5	0
6	0
7 (network control)	15 %

Configure schedulers if you want to change the minimum guaranteed bandwidth and other queue characteristics.

Options

rate—Minimum transmission rate for the queue, in bps. You can specify a value in bits-per-second either as a complete decimal number or as a decimal number followed by the abbreviation **k** (1000), m (1,000,000), or **g** (1,000,000,000).

Range: 1000 through 10,000,000,000 bps on 10-Gigabit interfaces, 1000 through 40,000,000,000 bps on 40-Gigabit interfaces.

percent *percentage*—Minimum percentage of transmission capacity allocated to the queue.

Range: 1 through 100 percent

exact—(QFX10000 switches only) Shape queues that are not strict-high priority queues to the transmit rate so that the transmit rate is the maximum bandwidth limit. Traffic that exceeds the exact transmit rate is dropped. You cannot set an excess rate on queues configured as **transmit-rate *(rate | percentage)* exact** because the purpose of setting an exact transmit rate is to set a maximum bandwidth (shaping rate) on the traffic.

> **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.

Required Privilege Level

interfaces—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Example: Configuring CoS Hierarchical Port Scheduling (ETS) | 415
Example: Configuring Queue Schedulers | 325
Example: Configuring Queue Schedulers for Port Scheduling | 361
Understanding CoS Output Queue Schedulers | 313
Understanding CoS Port Schedulers on QFX Switches | 343

tx-buffers

Syntax

```
tx-buffers (on | off);
```

Hierarchy Level

```
[edit interfaces interface-name ether-options configured-flow-control]
```

Release Information

Statement introduced in Junos OS Release 12.1 for the QFX Series.

Description

Enable or disable an interface to respond to received Ethernet PAUSE messages. If you enable the transmit buffers to respond to PAUSE messages, when the interface receives a PAUSE message from the connected peer, the interface stops transmitting frames on the entire link. When the receive buffer on the connected peer empties below a certain threshold, the peer interface sends a message to the paused interface to resume sending frames.

Ethernet PAUSE prevents buffers from overflowing and dropping packets during periods of network congestion. If the other devices in the network are also configured to support PAUSE, PAUSE supports lossless operation. Use the **tx-buffers** statement with the **rx-buffers** statement to configure asymmetric Ethernet PAUSE on an interface. (Use the **flow-control** statement to enable symmetric PAUSE and the **no-flow-control** statement to disable symmetric PAUSE on an interface. Symmetric flow control and asymmetric flow control are mutually exclusive features. If you attempt to configure both, the switch returns a commit error.)

> **NOTE:** Ethernet PAUSE temporarily stops transmitting all traffic on a link when the buffers fill to a certain threshold. To temporarily pause traffic on individual “lanes” of traffic (each lane contains the traffic associated with a particular IEEE 802.1p code point, so there can be eight lanes of traffic on a link), use priority-based flow control (PFC).
>
> Ethernet PAUSE and PFC are mutually exclusive features, so you cannot configure both of them on the same interface. If you attempt to configure both Ethernet PAUSE and PFC on an interface, the switch returns a commit error.

Default

Flow control is disabled. You must explicitly configure Ethernet PAUSE flow control on interfaces.

Options

on | off—Enable or disable an interface to respond to an Ethernet PAUSE message.

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

flow-control | **771**

rx-buffers | **836**

Configuring CoS Asymmetric Ethernet PAUSE Flow Control | **212**

Enabling and Disabling CoS Symmetric Ethernet PAUSE Flow Control | **211**

Understanding CoS Flow Control (Ethernet PAUSE and PFC) | **197**

unit

Syntax

```
unit logical-unit-number {
  classifiers {
    (dscp | dscp-ipv6 | ieee-802.1 | exp) (classifier-name | default);
  }
  forwarding-class class-name;
  rewrite-rules {
    (dscp | dscp-ipv6 | ieee-802.1 | exp) (classifier-name | default);
  }
}
```

Hierarchy Level

```
[edit class-of-service interfaces interface-name]
```

Release Information

Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Configure a logical interface on the physical device. You must configure a logical interface to use the physical device.

> **NOTE:** OCX Series switches do not support MPLS, so they do not support EXP classifiers and rewrite rules.

Options

logical-unit-number—Number of the logical unit.

Range: 0 through 16,385

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Assigning CoS Components to Interfaces | 77

CHAPTER 21

Operational Commands

IN THIS CHAPTER

show class-of-service

Syntax

```
show class-of-service
```

Release Information

Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Display the class-of-service (CoS) information.

Required Privilege Level

view

RELATED DOCUMENTATION

Monitoring CoS Code-Point Value Aliases | 82
Monitoring CoS Classifiers | 108
Monitoring CoS Forwarding Classes | 169
Monitoring Interfaces That Have CoS Components
Monitoring CoS Scheduler Maps | 341
Monitoring CoS Rewrite Rules | 137

List of Sample Output

show class-of- service on page 875

Output Fields

Table 135 on page 873 lists the output fields for the **show class-of-service** command. Output fields are listed in the approximate order in which they appear.

Table 135: show class-of-service Output Fields

Field Name	Field Description	Level of Output
Forwarding class	The forwarding class configuration: • **Forwarding class**—Name of the forwarding class. • **ID**—Forwarding class ID. • **Queue**—Queue number.	All levels

Table 135: show class-of-service Output Fields *(continued)*

Field Name	Field Description	Level of Output
Code point type	The type of code-point alias: • **dscp**—Aliases for DiffServ code point (DSCP) values. • **ieee–802.1**—Aliases for IEEE 802.1p values. • **exp**—Aliases for MPLS EXP values.	All levels
Alias	Names given to CoS values.	All levels
Bit pattern	Set of bits associated with an alias.	All levels
Classifier	Name of the classifier.	All levels
Code point	Code-point values.	All levels
Loss priority	Loss priority assigned to specific CoS values and aliases of the classifier.	All levels
Rewrite rule	Name of the rewrite rule if one has been configured.	All levels
Drop profile	Name of the drop profile.	All levels
Type	Type of drop profile. QFX Series supports only the **discrete** type of drop-profile.	All levels
Fill level	Percentage of queue buffer fullness in a drop profile at which packets begin to drop during periods of congestion.	All levels
Scheduler map	Name of the scheduler map.	All levels
Scheduler	Name of the scheduler.	All levels
Transmit rate	Transmission rate of the scheduler.	All levels
Buffer size	Delay buffer size in the queue.	All levels
Drop profiles	Drop profiles configured for the specified scheduler.	All levels
Protocol	Transport protocol corresponding to the drop profile.	All levels
Name	Name of the drop profile.	All levels
Queues supported	Number of queues that can be configured on the interface.	All levels

Table 135: show class-of-service Output Fields *(continued)*

Field Name	Field Description	Level of Output
Queues in use	Number of queues currently configured.	All levels
Physical interface	Name of the physical interface.	All levels
Scheduler map	Name of the scheduler map.	All levels
Congestion-notification	Enabled if a congestion notification profile is applied to the interface; disabled if no congestion notification profile is applied to the interface. **NOTE:** OCX Series switches do not support congestion notification profiles.	All levels
Forwarding class set	Name of the forwarding class set (priority group). **NOTE:** Only on systems that support enhanced transmission selection (ETS) hierarchical port scheduling.	
Index	Internal index of an object.	All levels

Sample Output

show class-of- service

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

```
Forwarding class                               ID       Queue
  best-effort                                   0        0
  fcoe                                          1        3
  no-loss                                       2        4
  network-control                               3        7
  mcast                                         8        8

Code point type: dscp
  Alias                 Bit pattern
  af11                  001010
  af12                  001100
  ...                   ...

Code point type: ieee-802.1
  Alias                 Bit pattern
```

```
  af11              100
  ...               ...

Classifier: dscp-default, Code point type: dscp, Index: 7
  Code point         Forwarding class                    Loss priority
  000000             best-effort                         low
  000001             best-effort                         low
  ...                ...                                 ...

Classifier: ieee8021p-default, Code point type: ieee-802.1, Index: 11
  Code point         Forwarding class                    Loss priority
  000                best-effort                         low
  001                best-effort                         low
  010                best-effort                         low
  011                fcoe                                low
  100                no-loss                             low
  101                best-effort                         low
  110                network-control                     low
  111                network-control                     low

Drop profile:<default-drop-profile>, Type: discrete, Index: 1
  Fill level
         100

Scheduler map: <default>, Index: 2

  Scheduler: <default-be>, Forwarding class: best-effort, Index: 21
    Transmit rate: 5 percent, Rate Limit: none, Buffer size: 5 percent, Buffer 
Limit: none,
    Priority: low
    Excess Priority: low
    drop-profile-map-set-type: mark
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any             1    <default-drop-profile>
      Medium high     any             1    <default-drop-profile>
      High            any             1    <default-drop-profile>

  Scheduler: <default-fcoe>, Forwarding class: fcoe, Index: 50
    Transmit rate: 35 percent, Rate Limit: none, Buffer size: 35 percent, Buffer 
Limit: none,
    Priority: low
    Excess Priority: low
    drop-profile-map-set-type: mark
```

```
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any             1    <default-drop-profile>
      Medium high     any             1    <default-drop-profile>
      High            any             1    <default-drop-profile>

  Scheduler: <default-noloss>, Forwarding class: no-loss, Index: 51
    Transmit rate: 35 percent, Rate Limit: none, Buffer size: 35 percent, Buffer
Limit: none,
    Priority: low
    Excess Priority: low
    drop-profile-map-set-type: mark
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any             1    <default-drop-profile>
      Medium high     any             1    <default-drop-profile>
      High            any             1    <default-drop-profile>

  Scheduler: <default-nc>, Forwarding class: network-control, Index: 23
    Transmit rate: 5 percent, Rate Limit: none, Buffer size: 5 percent, Buffer
Limit: none,
    Priority: low
    Excess Priority: low
    drop-profile-map-set-type: mark
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any             1    <default-drop-profile>
      Medium high     any             1    <default-drop-profile>
      High            any             1    <default-drop-profile>

  Scheduler: <default-mcast>, Forwarding class: mcast, Index: 49
    Transmit rate: 20 percent, Rate Limit: none, Buffer size: 20 percent, Buffer
Limit: none,
    Priority: low
    Excess Priority: low
    drop-profile-map-set-type: mark
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any             1    <default-drop-profile>
      Medium high     any             1    <default-drop-profile>
      High            any             1    <default-drop-profile>

Physical interface: xe-0/0/0, Index: 129
Queues supported: 12, Queues in use: 12
```

```
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled

Physical interface: xe-0/0/1, Index: 130
Queues supported: 12, Queues in use: 12
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled

...              ...               ...

Forwarding class set: lan-fcset, Type: normal-type, Forwarding class set index: 7

  Forwarding class                             Index
  best-effort                                    0
```

show class-of-service classifier

Syntax

```
show class-of-service classifier
<name name>
<type dscp | type dscp-ipv6 | type exp | type ieee-802.1 | type inet-precedence>
```

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 9.0 for EX Series switches.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

For each class-of-service (CoS) classifier, display the mapping of code point value to forwarding class and loss priority.

Options

none—Display all classifiers.

name *name*—(Optional) Display named classifier.

type dscp—(Optional) Display all classifiers of the Differentiated Services code point (DSCP) type.

type dscp-ipv6—(Optional) Display all classifiers of the DSCP for IPv6 type.

type exp—(Optional) Display all classifiers of the MPLS experimental (EXP) type.

type ieee-802.1—(Optional) Display all classifiers of the ieee-802.1 type.

type inet-precedence—(Optional) Display all classifiers of the inet-precedence type.

Required Privilege Level

view

List of Sample Output

show class-of-service classifier type ieee-802.1 on page 880
show class-of-service classifier type ieee-802.1 (QFX Series) on page 881

Output Fields

Table 136 on page 880 describes the output fields for the **show class-of-service classifier** command. Output fields are listed in the approximate order in which they appear.

Table 136: show class-of-service classifier Output Fields

Field Name	Field Description
Classifier	Name of the classifier.
Code point type	Type of the classifier: **exp** (not on EX Series switch), **dscp**, **dscp-ipv6** (not on EX Series switch), **ieee-802.1**, or **inet-precedence**.
Index	Internal index of the classifier.
Code point	Code point value used for classification
Forwarding class	Classification of a packet affecting the forwarding, scheduling, and marking policies applied as the packet transits the router.
Loss priority	Loss priority value used for classification. For most platforms, the value is **high** or **low**. For some platforms, the value is **high**, **medium-high**, **medium-low**, or **low**.

Sample Output

show class-of-service classifier type ieee-802.1

user@host> show class-of-service classifier type ieee-802.1

```
Classifier: ieee802.1-default, Code point type: ieee-802.1, Index: 3
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                  medium-high
  110              network-control                     low
  111              network-control                     high

Classifier: users-ieee802.1, Code point type: ieee-802.1
  Code point         Forwarding class                    Loss priority
  100                expedited-forwarding                low
```

show class-of-service classifier type ieee-802.1 (QFX Series)

user@switch> show class-of-service classifier type ieee-802.1

```
Classifier: ieee8021p-default, Code point type: ieee-802.1, Index: 11
  Code point          Forwarding class                    Loss priority
  000                 best-effort                         low
  001                 best-effort                         low
  010                 best-effort                         low
  011                 fcoe                                low
  100                 no-loss                             low
  101                 best-effort                         low
  110                 network-control                     low
  111                 network-control                     low

Classifier: ieee8021p-untrust, Code point type: ieee-802.1, Index: 16
  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

Classifier: ieee-mcast, Code point type: ieee-802.1, Index: 46
  Code point          Forwarding class                    Loss priority
  000                 mcast                               low
  001                 mcast                               low
  010                 mcast                               low
  011                 mcast                               low
  100                 mcast                               low
  101                 mcast                               low
  110                 mcast                               low
  111                 mcast                               low
```

show class-of-service code-point-aliases

Syntax

```
show class-of-service code-point-aliases
<dscp | dscp-ipv6 | exp | ieee-802.1 | inet-precedence>
```

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 9.0 for EX Series switches.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Display the mapping of class-of-service (CoS) code point aliases to corresponding bit patterns.

Options

none—Display code point aliases of all code point types.

dscp—(Optional) Display Differentiated Services code point (DSCP) aliases.

dscp-ipv6—(Optional) Display IPv6 DSCP aliases.

exp—(Optional) Display MPLS EXP code point aliases.

ieee-802.1—(Optional) Display IEEE-802.1 code point aliases.

inet-precedence—(Optional) Display IPv4 precedence code point aliases.

Required Privilege Level

view

List of Sample Output

show class-of-service code-point-aliases exp on page 883

Output Fields

Table 137 on page 882 describes the output fields for the **show class-of-service code-point-aliases** command. Output fields are listed in the approximate order in which they appear.

Table 137: show class-of-service code-point-aliases Output Fields

Field Name	Field Description
Code point type	Type of the code points displayed: **dscp**, **dscp-ipv6** (not on EX Series switch), **exp** (not on EX Series switch or the QFX Series), **ieee-802.1**, or **inet-precedence** (not on the QFX Series).

Table 137: show class-of-service code-point-aliases Output Fields *(continued)*

Field Name	Field Description
Alias	Alias for a bit pattern.
Bit pattern	Bit pattern for which the alias is displayed.

Sample Output

show class-of-service code-point-aliases exp

user@host> **show class-of-service code-point-aliases exp**

```
Code point type: exp
  Alias                Bit pattern
  af11                 100
  af12                 101
  be                   000
  be1                  001
  cs6                  110
  cs7                  111
  ef                   010
  ef1                  011
  nc1                  110
  nc2                  111
```

show class-of-service congestion-notification

Syntax

```
show class-of-service congestion-notification <profile-name>
```

Release Information

Command introduced in Junos OS Release 11.1 for the QFX Series.
Output for DSCP values introduced for DSCP-based PFC in Junos OS Release 17.4R1 for the QFX Series.

Description

Display whether priority-based flow control (PFC) is enabled for each IEEE 802.1p code point or for each DSCP value for DSCP-based PFC.

Options

none—Display the PFC state for all IEEE 802.1p code points or DSCP values.

profile-name—Display the PFC state for all IEEE 802.1p code points or DSCP values for the specified congestion notification profile.

Required Privilege Level

view

RELATED DOCUMENTATION

Configuring CoS PFC (Congestion Notification Profiles) | 194
Example: Configuring CoS PFC for FCoE Traffic | 490
Example: Configuring Lossless FCoE Traffic When the Converged Ethernet Network Does Not Use IEEE 802.1p Priority 3 for FCoE Traffic (FCoE Transit Switch) | 565
Example: Configuring Two or More Lossless FCoE Priorities on the Same FCoE Transit Switch Interface | 576
Example: Configuring Two or More Lossless FCoE IEEE 802.1p Priorities on Different FCoE Transit Switch Interfaces | 587
Example: Configuring Lossless IEEE 802.1p Priorities on Ethernet Interfaces for Multiple Applications (FCoE and iSCSI) | 605
Example: Configuring PFC Across Layer 3 Interfaces | 217
Understanding CoS Flow Control (Ethernet PAUSE and PFC) | 197
Understanding PFC Using DSCP at Layer 3 for Untagged Traffic | 242
Configuring DSCP-based PFC for Layer 3 Untagged Traffic | 245

List of Sample Output

Output Fields

Table 138 on page 885 describes the output fields for the **show class-of-service congestion-notification** command. Output fields are listed in the approximate order in which they appear.

Table 138: show class-of-service congestion-notification Output Fields

Field Name	Field Description
Type	Type of interfaces on which congestion notification is applied. Congestion notification is applied on input interfaces.
Index	Index of this congestion notification profile.
Name	Name of the congestion notification profile.
Cable Length	Length of the attached physical cable in meters. The default value is 100 meters.
Priority	IEEE 802.1p code points, or for DSCP-based PFC, DSCP values.
PFC	State of PFC for the corresponding code point, either **enabled** or **disabled**.
MRU	Maximum receive unit of the interface in bytes. (Incoming traffic that exceeds the MRU size of an interface is dropped.) The default values are: • 2500 bytes for priority 3 traffic • 9216 bytes for priority 4 traffic **NOTE:** If you configure flow control on a priority that is not one of the default flow control priorities, the default MRU value is 2500 bytes. For example, if you configure flow control on priority 5 and you do not configure an MRU value, the default MRU value is 2500 bytes.
Flow-Control-Queues	Output queue mapping to IEEE 802.1p code points (priorities). Explicit output queue to priority mapping overwrites the default configuration, and only explicitly mapped queues are displayed in the output. Flow control is only enabled on a queue when you enable PFC on the corresponding priority in the input stanza of the congestion notification profile.

Sample Output

show class-of-service congestion-notification

user@switch> **show class-of-service congestion-notification**

```
Name: fcoe_p3_cnp, Index: 12037
Type: Input
Cable Length: 100 m
  Priority     PFC          MRU
  000          Disabled
  001          Disabled
  010          Disabled
  011          Enabled      2500
  100          Enabled      9216
  101          Disabled
  110          Disabled
  111          Disabled
Type: Output
  Priority     Flow-Control-Queues
  000
               0
  001
               1
  010
               2
  011
               3
  100
               4
  101
               5
  110
               6
  111
               7

Name: fcoe_p3_p5_cnp, Index: 46484
Type: Input
Cable Length: 100 m
  Priority     PFC          MRU
  000          Disabled
  001          Disabled
  010          Disabled
  011          Enabled      2240
```

```
  100          Disabled
  101          Enabled      2240
  110          Disabled
  111          Disabled
Type: Output
  Priority     Flow-Control-Queues
  011
               3
  101
               5
```

Sample Output

show class-of-service congestion-notification (QFX Series with DSCP-based PFC and a specified congestion notification profile)

`user@switch>` **show class-of-service congestion-notification dscp_cnp**

```
Name: dscp_cnp, Index: 13825
Type: Input
Cable Length: 100 m
  Priority     PFC          MRU
  000000       Disabled
  000001       Disabled
  000010       Disabled
  000011       Disabled
  000100       Disabled
  000101       Disabled
  000110       Disabled
  000111       Disabled
  001000       Disabled
  001001       Disabled
  001010       Disabled
  001011       Disabled
  001100       Disabled
  001101       Disabled
  001110       Disabled
  001111       Disabled
  010000       Disabled
  010001       Disabled
  010010       Disabled
  010011       Disabled
```

```
010100        Disabled
010101        Disabled
010110        Disabled
010111        Disabled
011000        Disabled
011001        Disabled
011010        Disabled
011011        Disabled
011100        Disabled
011101        Disabled
011110        Disabled
011111        Disabled
100000        Disabled
100001        Disabled
100010        Disabled
100011        Disabled
100100        Disabled
100101        Disabled
100110        Disabled
100111        Disabled
101000        Disabled
101001        Disabled
101010        Disabled
101011        Disabled
101100        Disabled
101101        Disabled
101110        Disabled
101111        Disabled
110000        Enabled       3000
110001        Disabled
110010        Disabled
110011        Disabled
110100        Disabled
110101        Disabled
110110        Disabled
110111        Disabled
111000        Enabled       2000
111001        Disabled
111010        Disabled
111011        Disabled
111100        Enabled       4000
111101        Disabled
111110        Disabled
111111        Disabled
```

```
Type: Output
  Priority      Flow-Control-Queues
  111           1
```

show class-of-service drop-profile

Syntax

```
show class-of-service drop-profile
<profile-name profile-name>
```

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 9.0 for EX Series switches.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Display data points for each class-of-service (CoS) random early detection (RED) drop profile.

Options

none—Display all drop profiles.

profile-name *profile-name*—(Optional) Display the specified profile only.

Required Privilege Level

view

List of Sample Output

show class-of-service drop-profile on page 891
show class-of-service drop-profile (EX4200 Switch) on page 892
show class-of-service drop-profile (EX8200 Switch) on page 892

Output Fields

Table 139 on page 890 describes the output fields for the **show class-of-service drop-profile** command. Output fields are listed in the approximate order in which they appear.

Table 139: show class-of-service drop-profile Output Fields

Field Name	Field Description
Drop profile	Name of a drop profile.
Type	Type of drop profile: • **discrete** (default) • **interpolated** (EX8200 switches, QFX Series switches, QFabric systems, EX4600 switches, OCX Series switches only)

Table 139: show class-of-service drop-profile Output Fields *(continued)*

Field Name	Field Description
Index	Internal index of this drop profile.
Fill Level	Percentage fullness of a queue.
Drop probability	Drop probability at this fill level.

Sample Output

show class-of-service drop-profile

user@host> **show class-of-service drop-profile**

```
Drop profile: <default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100
Drop profile: user-drop-profile, Type: interpolated, Index: 2989
  Fill level    Drop probability
           0                   0
           1                   1
           2                   2
           4                   4
           5                   5
           6                   6
           8                   8
          10                  10
          12                  15
          14                  20
          15                  23
        ... 64 entries total
          90                  96
          92                  96
          94                  97
          95                  98
          96                  98
          98                  99
          99                  99
         100                 100
```

show class-of-service drop-profile (EX4200 Switch)

`user@switch>` **show class-of-service drop-profile**

```
Drop profile: <default-drop-profile>, Type: discrete, Index: 1
  Fill level
         100
Drop profile: dp1, Type: discrete, Index: 40496
  Fill level
          10
```

show class-of-service drop-profile (EX8200 Switch)

`user@switch>` **show class-of-service drop-profile**

```
Drop profile: <default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100
Drop profile: dp1, Type: interpolated, Index: 40496
  Fill level    Drop probability
           0                   0
           1                  80
           2                  90
           4                  90
           5                  90
           6                  90
           8                  90
          10                  90
          12                  91
          14                  91
          15                  91
          16                  91
          18                  91
          20                  91
          22                  92
          24                  92
          25                  92
          26                  92
          28                  92
          30                  92
          32                  93
          34                  93
          35                  93
          36                  93
          38                  93
```

```
          40                   93
          42                   94
          44                   94
          45                   94
          46                   94
          48                   94
          49                   94
          51                   95
          52                   95
          54                   95
          55                   95
          56                   95
          58                   95
          60                   95
          62                   96
          64                   96
          65                   96
          66                   96
          68                   96
          70                   96
          72                   97
          74                   97
          75                   97
          76                   97
          78                   97
          80                   97
          82                   98
          84                   98
          85                   98
          86                   98
          88                   98
          90                   98
          92                   99
          94                   99
          95                   99
          96                   99
          98                   99
          99                   99
         100                  100
Drop profile: dp2, Type: discrete, Index: 40499
  Fill level    Drop probability
          10                    5
          50                   50
```

show class-of-service forwarding-class

Syntax

```
show class-of-service forwarding-class
```

Release Information

Command introduced in Junos OS Release 9.0 for EX Series switches.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.
PFC priority output field introduced for DSCP-based PFC in Junos OS Release 17.4R1 for the QFX Series.

Description

Display information about forwarding classes, including the mapping of forwarding classes to queue numbers.

Required Privilege Level

view

RELATED DOCUMENTATION

Monitoring CoS Forwarding Classes

Monitoring CoS Forwarding Classes | 169
Understanding PFC Using DSCP at Layer 3 for Untagged Traffic | 242

List of Sample Output

show class-of-service forwarding-class on page 896
show class-of-service forwarding-class (EX8200 Switch) on page 896
show class-of-service forwarding-class (QFX Series) on page 896
show class-of-service forwarding-class (QFX Series with DSCP-based PFC) on page 897

Output Fields

Table 140 on page 894 describes the output fields for the **show class-of-service forwarding-class** command. Output fields are listed in the approximate order in which they appear.

Table 140: show class-of-service forwarding-class Output Fields

Field Name	Field Description
Forwarding class	Name of the forwarding class.

Table 140: show class-of-service forwarding-class Output Fields *(continued)*

Field Name	Field Description
ID	Forwarding class identifier. (QFX5110, QFX5200, and QFX5210 switches only) For DSCP-based PFC, the forwarding class ID is assigned from (and should be the same as) the configured PFC priority for the forwarding class. See "Configuring DSCP-based PFC for Layer 3 Untagged Traffic" on page 245 for details.
Queue	CoS output queue mapped to the forwarding class.
Policing priority	Not supported on EX Series switches or the QFX Series and can be ignored.
Fabric priority	(EX8200 switches only) Fabric priority for the forwarding class, either **high** or **low**. Determines the priority of packets entering the switch fabric.
No-Loss	(QFX Series only) Packet loss attribute to differentiate lossless forwarding classes from lossy forwarding classes: • Disabled—Lossless transport is not configured on the forwarding class (packet drop attribute is **drop**). • Enabled—Lossless transport is configured on the forwarding class (packet drop attribute is **no-loss**).
PFC Priority	(QFX5110, QFX5200, and QFX5210 switches only) For DSCP-based PFC, the explicitly configured PFC priority configured for the forwarding class. The DSCP value on which PFC is enabled maps to this priority, and this priority is used in PFC pause frames sent to the peer to request to pause traffic on the mapped DSCP value when the link becomes congested. The forwarding class ID is assigned from and should match this value in the output of this command. See "Configuring DSCP-based PFC for Layer 3 Untagged Traffic" on page 245 for details.

Sample Output

show class-of-service forwarding-class

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

```
Forwarding class                ID                       Queue  Policing priority
  best-effort                         0                        0            normal
  expedited-forwarding                1                        5            normal
  assured-forwarding                  2                        1            normal
  network-control                     3                        7            normal
```

Sample Output

show class-of-service forwarding-class (EX8200 Switch)

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

```
Forwarding class                        ID       Queue  Fabric priority
  best-effort                            0        0      low
  expedited-forwarding                   1        5      low
  assured-forwarding                     2        1      low
  network-control                        3        7      low
  mcast-be                               4        2      low
  mcast-ef                               5        4      low
  mcast-af                               6        6      low
```

Sample Output

show class-of-service forwarding-class (QFX Series)

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

```
Forwarding class                        ID       Queue  Policing priority  No-Loss
  best-effort                          0          0        normal         Disabled

  fcoe                                  1          3        normal         Enabled
  no-loss                               2          4        normal         Enabled
  network-control                      3          7        normal         Disabled
```

```
mcast                               8         8        normal        Disabled
```

show class-of-service forwarding-class (QFX Series with DSCP-based PFC)

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

```
Forwarding class        ID      Queue  Policing priority  No-Loss     PFC priority

  best-effort           0         0        normal         Disabled
  fcoe                  1         3        normal         Enabled
  no-loss               2         4        normal         Enabled
  fc2                   3         2        normal         Enabled          3
  network-control       5         7        normal         Disabled
  fc1                   7         1        normal         Enabled          7
  mcast                 8         8        normal         Disabled
```

On switches that do not use different forwarding classes and output queues for unicast and multidestination (multicast, broadcast, destination lookup fail) traffic, there is no **mcast** forwarding class and there is no queue 8. (Switches that use different forwarding classes and output queues for unicast and multidestination traffic support 12 forwarding classes and output queues, of which four of each are dedicated to multidestination traffic. Switches that use the same forwarding classes and output queues for unicast and multidestination traffic support eight forwarding classes and eight output queues.)

show class-of-service forwarding-class-set

Syntax

```
show class-of-service forwarding-class-set
<forwarding-class-set-name>
```

Release Information

Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Display the forwarding classes associated with each forwarding class set.

Options

none—Display all forwarding class sets.

forwarding-class-set-name—(Optional) Display the forwarding classes associated with the specified forwarding class set.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding CoS Forwarding Class Sets (Priority Groups) | 162
Defining CoS Forwarding Class Sets | 164
Example: Configuring Forwarding Class Sets | 165

Output Fields

Table 141 on page 898 describes the output fields for the **show class-of-service forwarding-class-set** command. Output fields are listed in the approximate order in which they appear.

Table 141: show class-of-service forwarding-class-set Output Fields

Field Name	Field Description
Forwarding class set	Name of the forwarding class set.
Type	Internal Junos OS type.
Forwarding class set index	Index of this forwarding class set.

Table 141: show class-of-service forwarding-class-set Output Fields *(continued)*

Field Name	Field Description
Forwarding class	Name of a forwarding class.
Index	Index of this forwarding class.

Sample Output

show class-of-service forwarding-class-set

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

```
Forwarding class set: san_fcset, Type: normal-type, Forwarding class set index:
37839
  Forwarding class                       Index
  fcoe                                     1

Forwarding class set: lan_fcset, Type: normal-type, Forwarding class set index:
37840
  Forwarding class                       Index
  best-effort                              0

Forwarding class set: multicast_fcset, Type: normal-type, Forwarding class set
index: 37841
  Forwarding class                       Index
  mcast                                    8
```

show class-of-service forwarding-table

List of Syntax

Syntax on page 900
Syntax (TX Matrix and TX Matrix Plus Router) on page 900

Syntax

```
show class-of-service forwarding-table
```

Syntax (TX Matrix and TX Matrix Plus Router)

```
show class-of-service forwarding-table
<lcc number> | <sfc number>
```

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Display the entire class-of-service (CoS) configuration as it exists in the forwarding table. Executing this command is equivalent to executing all **show class-of-service forwarding-table** commands in succession.

Options

lcc *number*—(TX Matrix and TX Matrix Plus router only) (Optional) On a TX Matrix router, display the forwarding table configuration for a specific T640 router (or line-card chassis) configured in a routing matrix. On a TX Matrix Plus router, display the forwarding table configuration for a specific router (or line-card chassis) configured in the routing matrix.

Replace *number* with the following values depending on the LCC configuration:

- 0 through 3, when T640 routers are connected to a TX Matrix router in a routing matrix.
- 0 through 3, when T1600 routers are connected to a TX Matrix Plus router in a routing matrix.
- 0 through 7, when T1600 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix.
- 0, 2, 4, or 6, when T4000 routers are connected to a TX Matrix Plus router with 3D SIBs in a routing matrix.

sfc *number*—(TX Matrix Plus routers only) (Optional) Display the forwarding table configuration for the TX Matrix Plus router. Replace ***number*** with **0**.

Required Privilege Level

view

List of Sample Output

show class-of-service forwarding-table on page 901
show class-of-service forwarding-table lcc (TX Matrix Plus Router) on page 902

Output Fields

See the output field descriptions for **show class-of-service forwarding-table** commands:

- show class-of-service forwarding-table classifier
- show class-of-service forwarding-table classifier mapping
- show class-of-service forwarding-table drop-profile
- *show class-of-service forwarding-table fabric scheduler-map*
- show class-of-service forwarding-table rewrite-rule
- show class-of-service forwarding-table rewrite-rule mapping
- show class-of-service forwarding-table scheduler-map

Sample Output

show class-of-service forwarding-table

`user@host>` **show class-of-service forwarding-table**

```
Classifier table index: 9, # entries: 8, Table type: EXP
Entry #   Code point   Forwarding-class #   PLP
   0          000        0                  0
   1          001        0                  1
   2          010        1                  0
   3          011        1                  1
   4          100        2                  0
   5          101        2                  1
   6          110        3                  0
   7          111        3                  1

                       Table Index/
Interface        Index       Q num      Table type
sp-0/0/0.1001      66          11      IPv4 precedence
sp-0/0/0.2001      67          11      IPv4 precedence
sp-0/0/0.16383     68          11      IPv4 precedence
```

```
fe-0/0/0.0        69         11      IPv4 precedence

Interface: sp-0/0/0 (Index: 129, Map index: 2, Map type: FINAL,
 Num of queues: 2):
  Entry 0 (Scheduler index: 16, Forwarding-class #: 0):
    Tx rate: 0 Kb (95%), Buffer size: 95 percent
Priority low
    PLP high: 1, PLP low: 1, PLP medium-high: 1, PLP medium-low: 1
  Entry 1 (Scheduler index: 18, Forwarding-class #: 3):
    Tx rate: 0 Kb (5%), Buffer size: 5 percent
Priority low
    PLP high: 1, PLP low: 1, PLP medium-high: 1, PLP medium-low: 1

Interface: fe-0/0/0 (Index: 137, Map index: 2, Map type: FINAL,
 Num of queues: 2):
  Entry 0 (Scheduler index: 16, Forwarding-class #: 0):
    Tx rate: 0 Kb (95%), Buffer size: 95 percent
Priority low
    PLP high: 1, PLP low: 1, PLP medium-high: 1, PLP medium-low: 1
  Entry 1 (Scheduler index: 18, Forwarding-class #: 3):
    Tx rate: 0 Kb (5%), Buffer size: 5 percent
Priority low
    PLP high: 1, PLP low: 1, PLP medium-high: 1, PLP medium-low: 1
Interface: fe-0/0/1 (Index: 138, Map index: 2, Map type: FINAL,
 Num of queues: 2):
  Entry 0 (Scheduler index: 16, Forwarding-class #: 0):
    Tx rate: 0 Kb (95%), Buffer size: 95 percent
Priority low
    PLP high: 1, PLP low: 1, PLP medium-high: 1, PLP medium-low: 1
  Entry 1 (Scheduler index: 18, Forwarding-class #: 3):
    Tx rate: 0 Kb (5%), Buffer size: 5 percent
Priority low
    PLP high: 1, PLP low: 1, PLP medium-high: 1, PLP medium-low: 1

...

RED drop profile index: 1, # entries: 1
                             Drop
Entry      Fullness(%)   Probability(%)
    0              100              100
```

show class-of-service forwarding-table lcc (TX Matrix Plus Router)

`user@host>` **show class-of-service forwarding-table lcc 0**

```
lcc0-re0:
--------------------------------------------------------------------------

Classifier table index: 9, # entries: 64, Table type: IPv6 DSCP
Entry #   Code point   Forwarding-class #   PLP
   0        000000        0       0
   1        000001        0       0
   2        000010        0       0
   3        000011        0       0
   4        000100        0       0
   5        000101        0       0
   6        000110        0       0
   7        000111        0       0
   8        001000        0       0
   9        001001        0       0
  10        001010        0       0
  11        001011        0       0
  12        001100        0       0
  13        001101        0       0
  14        001110        0       0
  15        001111        0       0
  16        010000        0       0
  17        010001        0       0
  18        010010        0       0
  19        010011        0       0
  20        010100        0       0
  21        010101        0       0
  22        010110        0       0
  23        010111        0       0
  24        011000        0       0
  25        011001        0       0
  26        011010        0       0
  27        011011        0       0
  28        011100        0       0
  29        011101        0       0
  30        011110        0       0
  31        011111        0       0
  32        100000        0       0
  33        100001        0       0
  34        100010        0       0
  35        100011        0       0
  36        100100        0       0
  37        100101        0       0
  38        100110        0       0
```

```
  39        100111        0       0
  40        101000        0       0
  41        101001        0       0
  42        101010        0       0
  43        101011        0       0
  44        101100        0       0
  45        101101        0       0
  46        101110        0       0
...
```

show class-of-service forwarding-table classifier

Syntax

```
show class-of-service forwarding-table classifier
```

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Display the mapping of code point value to queue number and loss priority for each classifier as it exists in the forwarding table.

Options

This command has no options.

Required Privilege Level

view

List of Sample Output

show class-of-service forwarding-table classifier on page 906

Output Fields

Table 142 on page 905 describes the output fields for the **show class-of-service forwarding-table classifier** command. Output fields are listed in the approximate order in which they appear.

Table 142: show class-of-service forwarding-table classifier Output Fields

Field Name	Field Description
Classifier table index	Index of the classifier table.
entries	Total number of entries.
Table type	Type of code points in the table: **DSCP**, **EXP** (not on the QFX Series), **IEEE 802.1**, **IPv4 precedence** (not on the QFX Series), or **IPv6 DSCP**.
Entry #	Entry number.
Code point	Code point value used for classification.
Forwarding-class #	Forwarding class to which the code point is assigned.

Table 142: show class-of-service forwarding-table classifier Output Fields *(continued)*

Field Name	Field Description
PLP	Packet loss priority value set by classification. For most platforms, the value can be **0** or **1**. For some platforms, the value is **0**, **1**, **2**, or **3**. The value **0** represents low PLP. The value **1** represents **high** PLP. The value **2** represents medium-low PLP. The value **3** represents medium-high PLP.

Sample Output

show class-of-service forwarding-table classifier

`user@host>` **show class-of-service forwarding-table classifier**

```
Classifier table index: 62436, # entries: 64, Table type: DSCP

Entry #   Code point   Forwarding-class #   PLP
   0        000000       0                  0
   1        000001       0                  0
   2        000010       0                  0
   3        000011       0                  0
   4        000100       0                  0
   5        000101       0                  0
   6        000110       0                  0
   7        000111       0                  0
   8        001000       0                  0
   9        001001       0                  0
  10        001010       1                  1
  11        001011       0                  0
...
  60        111100       0                  0
  61        111101       0                  0
  62        111110       0                  0
  63        111111       0                  0
```

show class-of-service forwarding-table classifier mapping

Syntax

```
show class-of-service forwarding-table classifier mapping
```

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

For each logical interface, display either the table index of the classifier for a given code point type or the queue number (if it is a fixed classification) in the forwarding table.

Options

This command has no options.

Required Privilege Level

view

List of Sample Output

show class-of-service forwarding-table classifier mapping on page 908

Output Fields

Table 143 on page 907 describes the output fields for the **show class-of-service forwarding-table classifier mapping** command. Output fields are listed in the approximate order in which they appear.

Table 143: show class-of-service forwarding-table classifier mapping Output Fields

Field Name	Field Description
Table index/ Q num	If the table type is **Fixed**, the number of the queue to which the interface is mapped. For all other types, this value is the classifier index number.
Interface	Name of the logical interface. This field can also show the physical interface (QFX Series).
Index	Logical interface index.
Table type	Type of code points in the table: **DSCP**, **EXP** (not on the QFX Series), **Fixed**, **IEEE 802.1**, **IPv4 precedence** (not on the QFX Series),or **IPv6 DSCP**. **none** if **no-default** option set.

Sample Output

show class-of-service forwarding-table classifier mapping

`user@host>` **show class-of-service forwarding-table classifier mapping**

```
                       Table index/
Interface      Index     Q num      Table type
so-5/0/0.0        10     62436      DSCP
so-0/1/0.0        11     62436      DSCP
so-0/2/0.0        12         1      Fixed
so-0/2/1.0        13     62436      DSCP
so-0/2/1.0        13     62437      IEEE 802.1
so-0/2/2.0        14     62436      DSCP
so-0/2/2.0        14     62438      IPv4 precedence
```

show class-of-service forwarding-table drop-profile

Syntax

```
show class-of-service forwarding-table drop-profile
```

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Display the data points of all random early detection (RED) drop profiles as they exist in the forwarding table.

Options

This command has no options.

Required Privilege Level

view

List of Sample Output

show class-of-service forwarding-table drop-profile on page 910

Output Fields

Table 144 on page 909 describes the output fields for the **show class-of-service forwarding-table drop-profile** command. Output fields are listed in the approximate order in which they appear.

Table 144: show class-of-service forwarding-table drop-profile Output Fields

Field Name	Field Description
RED drop profile index	Index of this drop profile.
# entries	Number of entries in a particular RED drop profile index.
Entry	Drop profile entry number.
Fullness(%)	Percentage fullness of a queue.
Drop probability(%)	Drop probability at this fill level.

Sample Output

show class-of-service forwarding-table drop-profile

`user@host>` **show class-of-service forwarding-table drop-profile**

```
RED drop profile index: 4, # entries: 1
                             Drop
Entry      Fullness(%)   Probability(%)
    0              100              100

RED drop profile index: 8742, # entries: 3
                             Drop
Entry      Fullness(%)   Probability(%)
    0               10               10
    1               20               20
    2               30               30

RED drop profile index: 24627, # entries: 64
                             Drop
Entry      Fullness(%)   Probability(%)
    0                0                0
    1                1                1
    2                2                2
    3                4                4
 ...
   61               98               99
   62               99               99
   63              100              100

RED drop profile index: 25393, # entries: 64
                             Drop
Entry      Fullness(%)   Probability(%)
    0                0                0
    1                1                1
    2                2                2
    3                4                4
 ...
   61               98               98
   62               99               99
   63              100              100
```

show class-of-service forwarding-table rewrite-rule

Syntax

```
show class-of-service forwarding-table rewrite-rule
```

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Display mapping of queue number and loss priority to code point value for each rewrite rule as it exists in the forwarding table.

Options

This command has no options.

Required Privilege Level

view

List of Sample Output

show class-of-service forwarding-table rewrite-rule on page 912

Output Fields

Table 145 on page 911 describes the output fields for the **show class-of-service forwarding-table rewrite-rule** command. Output fields are listed in the approximate order in which they appear.

Table 145: show class-of-service forwarding-table rewrite-rule Output Fields

Field Name	Field Description
Rewrite table index	Index for this rewrite rule.
# entries	Number of entries in this rewrite rule.
Table type	Type of table: **DSCP**, **EXP** (not on the QFX Series), **EXP-PUSH-3** (not on the QFX Series), **IEEE 802.1**,**IPv4 precedence** (not on the QFX Series), **IPv6 DSCP**, or **Fixed**.
Q#	Queue number to which this entry is assigned.
Low bits	Code point value for low-priority loss profile.
State	State of this code point: **enabled**, **rewritten**, or **disabled**.

Table 145: show class-of-service forwarding-table rewrite-rule Output Fields *(continued)*

Field Name	Field Description
High bits	Code point value for high-priority loss profile.

Sample Output

show class-of-service forwarding-table rewrite-rule

user@host> **show class-of-service forwarding-table rewrite-rule**

```
Rewrite table index: 3753, # entries: 4, Table type: DSCP
Q#     Low bits  State      High bits  State
 0      000111  Enabled      001010   Enabled
 2      000000  Disabled     001100   Enabled
 1      101110  Enabled      110111   Enabled
 3      110000  Enabled      111000   Enabled
```

show class-of-service forwarding-table rewrite-rule mapping

Syntax

```
show class-of-service forwarding-table rewrite-rule mapping
```

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

For each logical interface, display the table identifier of the rewrite rule map for each code point type.

Options

This command has no options.

Required Privilege Level

view

List of Sample Output

show class-of-service forwarding-table rewrite-rule mapping on page 914

Output Fields

Table 146 on page 913 describes the output fields for the **show class-of-service forwarding-table rewrite-rule mapping** command. Output fields are listed in the approximate order in which they appear.

Table 146: show class-of-service forwarding-table rewrite-rule mapping Output Fields

Field Name	Field Description
Interface	Name of the logical interface. This field can also show the physical interface (QFX Series).
Index	Logical interface index.
Table index	Rewrite table index.
Type	Type of classifier: **DSCP**, **EXP** (not on the QFX Series), **EXP-PUSH-3** (not on the QFX Series), **EXP-SWAP-PUSH-2** (not on the QFX Series),**IEEE 802.1**, **IPv4 precedence** (not on the QFX Series), **IPv6 DSCP**, or **Fixed**.

Sample Output

show class-of-service forwarding-table rewrite-rule mapping

`user@host>` **show class-of-service forwarding-table rewrite-rule mapping**

```
Interface      Index   Table index  Type
so-5/0/0.0        10       3753  DSCP
so-0/1/0.0        11       3753  DSCP
so-0/2/0.0        12       3753  DSCP
so-0/2/1.0        13       3753  DSCP
so-0/2/2.0        14       3753  DSCP
so-0/2/3.0        15       3753  DSCP
```

show class-of-service forwarding-table scheduler-map

Syntax

```
show class-of-service forwarding-table scheduler-map
```

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

For each physical interface, display the scheduler map information as it exists in the forwarding table.

Options

This command has no options.

Required Privilege Level

view

List of Sample Output

show class-of-service forwarding-table scheduler-map on page 916

Output Fields

Table 147 on page 915 describes the output fields for the **show class-of-service forwarding-table scheduler-map** command. Output fields are listed in the approximate order in which they appear.

Table 147: show class-of-service forwarding-table scheduler-map Output Fields

Field Name	Field Description
Interface	Name of the physical interface.
Index	Physical interface index.
Map index	Scheduler map index.
Num of queues	Number of queues defined in this scheduler map.
Entry	Number of this entry in the scheduler map.
Scheduler index	Scheduler policy index.
Forwarding-class #	Forwarding class number to which this entry is applied.

Table 147: show class-of-service forwarding-table scheduler-map Output Fields *(continued)*

Field Name	Field Description
Tx rate	Configured transmit rate of the scheduler (in bps). The rate is a percentage of the total interface bandwidth, or the keyword **remainder**, which indicates that the scheduler receives the remaining bandwidth of the interface.
Max buffer delay	Amount of transmit delay (in milliseconds) or buffer size of the queue. This amount is a percentage of the total interface buffer allocation or the keyword **remainder**, which indicates that the buffer is sized according to what remains after other scheduler buffer allocations.
Priority	• **high**—Queue priority is high. • **low**—Queue priority is low.
PLP high	Drop profile index for a high packet loss priority profile.
PLP low	Drop profile index for a low packet loss priority profile.
PLP medium-high	Drop profile index for a medium-high packet loss priority profile.
PLP medium-low	Drop profile index for a medium-low packet loss priority profile.
TCP PLP high	Drop profile index for a high TCP packet loss priority profile.
TCP PLP low	Drop profile index for a low TCP packet loss priority profile.
Policy is exact	If this line appears in the output, exact rate limiting is enabled. Otherwise, no rate limiting is enabled.

Sample Output

show class-of-service forwarding-table scheduler-map

`user@host>` **show class-of-service forwarding-table scheduler-map**

```
Interface: so-5/0/0 (Index: 9, Map index: 17638, Num of queues: 2):
  Entry 0 (Scheduler index: 6090, Forwarding-class #: 0):
    Tx rate: 0 Kb (30%), Max buffer delay: 39 bytes (0%)
    Priority low
    PLP high: 25393, PLP low: 24627, TCP PLP high: 25393, TCP PLP low:8742
    Policy is exact
  Entry 1 (Scheduler index: 38372, Forwarding-class #: 1):
```

```
    Traffic chunk: Max = 0 bytes, Min = 0 bytes
    Tx rate: 0 Kb (40%), Max buffer delay: 68 bytes (0%)
    Priority high
    PLP high: 25393, PLP low: 24627, TCP PLP high: 25393, TCP PLP low: 8742

Interface: at-6/1/0 (Index: 10, Map index: 17638, Num of queues: 2):
  Entry 0 (Scheduler index: 6090, Forwarding-class #: 0):
    Traffic chunk: Max = 0 bytes, Min = 0 bytes
    Tx rate: 0 Kb (30%), Max buffer delay: 39 bytes (0%)
    Priority high
    PLP high: 25393, PLP low: 24627, TCP PLP high: 25393, TCP PLP low: 8742
  Entry 1 (Scheduler index: 38372, Forwarding-class #: 1):
    Traffic chunk: Max = 0 bytes, Min = 0 bytes
    Tx rate: 0 Kb (40%), Max buffer delay: 68 bytes (0%)
    Priority low
    PLP high: 25393, PLP low: 24627, TCP PLP high: 25393, TCP PLP low: 8742
```

show class-of-service interface

Syntax

```
show class-of-service interface
<comprehensive | detail> <interface-name>
```

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 9.0 for EX Series switches.
Forwarding class map information added in Junos OS Release 9.4.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 12.1 for the PTX Series Packet Transport routers.
Command introduced in Junos OS Release 12.2 for the ACX Series Universal Metro routers.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.
Options **detail** and **comprehensive** introduced in Junos OS Release 11.4.
Command introduced in Junos OS Release 15.1R3 on MX Series routers for enhanced subscriber management.

Description

Display the logical and physical interface associations for the classifier, rewrite rules, and scheduler map objects.

> **NOTE:** On routing platforms with dual Routing Engines, running this command on the backup Routing Engine, with or without any of the available options, is not supported and produces the following error message:
>
> **error: the class-of-service subsystem is not running**

Options

none—Display CoS associations for all physical and logical interfaces.

comprehensive—(M Series, MX Series, and T Series routers) (Optional) Display comprehensive quality-of-service (QoS) information about all physical and logical interfaces.

detail—(M Series, MX Series, and T Series routers) (Optional) Display QoS and CoS information based on the interface.

If the **interface** *interface-name* is a physical interface, the output includes:

- Brief QoS information about the physical interface
- Brief QoS information about the logical interface
- CoS information about the physical interface
- Brief information about filters or policers of the logical interface
- Brief CoS information about the logical interface

If the **interface** *interface-name* is a logical interface, the output includes:

- Brief QoS information about the logical interface
- Information about filters or policers for the logical interface
- CoS information about the logical interface

interface-name—(Optional) Display class-of-service (CoS) associations for the specified interface.

none—Display CoS associations for all physical and logical interfaces.

> **NOTE:** ACX5000 routers do not support classification on logical interfaces and therefore do not show CoS associations for logical interfaces with this command.

Required Privilege Level
view

RELATED DOCUMENTATION

Verifying and Managing Junos OS Enhanced Subscriber Management

List of Sample Output

show class-of-service interface (PPPoE Subscriber Interface for Enhanced Subscriber Management) on page 956

Output Fields

Table 148 on page 920 describes the output fields for the **show class-of-service interface** command. Output fields are listed in the approximate order in which they appear.

Table 148: show class-of-service interface Output Fields

Field Name	Field Description
Physical interface	Name of a physical interface.
Index	Index of this interface or the internal index of this object. (Enhanced subscriber management for MX Series routers) Index values for dynamic CoS traffic control profiles and dynamic scheduler maps are larger for enhanced subscriber management than they are for legacy subscriber management.
Dedicated Queues	Status of dedicated queues configured on an interface. Supported only on Trio MPC/MIC interfaces on MX Series routers. (Enhanced subscriber management for MX-Series routers) This field is not displayed for enhanced subscriber management.
Maximum usable queues	Number of queues you can configure on the interface.
Maximum usable queues	Maximum number of queues you can use.
Total non-default queues created	Number of queues created in addition to the default queues. Supported only on Trio MPC/MIC interfaces on MX Series routers. (Enhanced subscriber management for MX Series routers) This field is not displayed for enhanced subscriber management.
Rewrite Input IEEE Code-point	(QFX3500 switches only) IEEE 802.1p code point (priority) rewrite value. Incoming traffic from the Fibre Channel (FC) SAN is classified into the forwarding class specified in the native FC interface (NP_Port) fixed classifier and uses the priority specified as the IEEE 802.1p rewrite value.
Shaping rate	Maximum transmission rate on the physical interface. You can configure the shaping rate on the physical interface, or on the logical interface, but not on both. Therefore, the **Shaping rate** field is displayed for either the physical interface or the logical interface.

Table 148: show class-of-service interface Output Fields *(continued)*

Field Name	Field Description
Scheduler map	Name of the output scheduler map associated with this interface. (Enhanced subscriber management for MX Series routers) The name of the dynamic scheduler map object is associated with a generated UID (for example, **SMAP-1_UID1002**) instead of with a subscriber interface.
Scheduler map forwarding class sets	(QFX Series only) Name of the output fabric scheduler map associated with a QFabric system Interconnect device interface.
Input shaping rate	For Gigabit Ethernet IQ2 PICs, maximum transmission rate on the input interface.
Input scheduler map	For Gigabit Ethernet IQ2 PICs, name of the input scheduler map associated with this interface.
Chassis scheduler map	Name of the scheduler map associated with the packet forwarding component queues.
Rewrite	Name and type of the rewrite rules associated with this interface.
Traffic-control-profile	Name of the associated traffic control profile. (Enhanced subscriber management for MX Series routers) The name of the dynamic traffic control profile object is associated with a generated UID (for example, **TC_PROF_100_199_SERIES_UID1006**) instead of with a subscriber interface.
Classifier	Name and type of classifiers associated with this interface.
Forwarding-class-map	Name of the forwarding map associated with this interface.
Congestion-notification	(QFX Series and EX4600 switches only) Congestion notification state, **enabled** or **disabled**.
Logical interface	Name of a logical interface.
Object	Category of an object: **Classifier**, **Fragmentation-map** (for LSQ interfaces only), **Scheduler-map**, **Rewrite**, **Translation Table** (for IQE PICs only), or **traffic-class-map** (for T4000 routers with Type 5 FPCs).
Name	Name of an object.
Type	Type of an object: **dscp**, **dscp-ipv6**, **exp**, **ieee-802.1**, **ip**, **inet-precedence**, or **ieee-802.1ad** (for traffic class map on T4000 routers with Type 5 FPCs)..
Link-level type	Encapsulation on the physical interface.

Table 148: show class-of-service interface Output Fields *(continued)*

Field Name	Field Description
MTU	MTU size on the physical interface.
Speed	Speed at which the interface is running.
Loopback	Whether loopback is enabled and the type of loopback.
Source filtering	Whether source filtering is enabled or disabled.
Flow control	Whether flow control is enabled or disabled.
Auto-negotiation	(Gigabit Ethernet interfaces) Whether autonegotiation is enabled or disabled.
Remote-fault	(Gigabit Ethernet interfaces) Remote fault status. • **Online**—Autonegotiation is manually configured as online. • **Offline**—Autonegotiation is manually configured as offline.
Device flags	The **Device flags** field provides information about the physical device and displays one or more of the following values: • **Down**—Device has been administratively disabled. • **Hear-Own-Xmit**—Device receives its own transmissions. • **Link-Layer-Down**—The link-layer protocol has failed to connect with the remote endpoint. • **Loopback**—Device is in physical loopback. • **Loop-Detected**—The link layer has received frames that it sent, thereby detecting a physical loopback. • **No-Carrier**—On media that support carrier recognition, no carrier is currently detected. • **No-Multicast**—Device does not support multicast traffic. • **Present**—Device is physically present and recognized. • **Promiscuous**—Device is in promiscuous mode and recognizes frames addressed to all physical addresses on the media. • **Quench**—Transmission on the device is quenched because the output buffer is overflowing. • **Recv-All-Multicasts**—Device is in multicast promiscuous mode and therefore provides no multicast filtering. • **Running**—Device is active and enabled.

Table 148: show class-of-service interface Output Fields *(continued)*

Field Name	Field Description

Interface flags	The **Interface flags** field provides information about the physical interface and displays one or more of the following values:
	
	• **Admin-Test**—Interface is in test mode and some sanity checking, such as loop detection, is disabled.
	• **Disabled**—Interface is administratively disabled.
	• **Down**—A hardware failure has occurred.
	• **Hardware-Down**—Interface is nonfunctional or incorrectly connected.
	• **Link-Layer-Down**—Interface keepalives have indicated that the link is incomplete.
	• **No-Multicast**—Interface does not support multicast traffic.
	• **No-receive No-transmit**—Passive monitor mode is configured on the interface.
	• **Point-To-Point**—Interface is point-to-point.
	• **Pop all MPLS labels from packets of depth**—MPLS labels are removed as packets arrive on an interface that has the **pop-all-labels** statement configured. The depth value can be one of the following:
	  ◦ **1**—Takes effect for incoming packets with one label only.
	  ◦ **2**—Takes effect for incoming packets with two labels only.
	  ◦ **[1 2]**—Takes effect for incoming packets with either one or two labels.
	• **Promiscuous**—Interface is in promiscuous mode and recognizes frames addressed to all physical addresses.
	• **Recv-All-Multicasts**—Interface is in multicast promiscuous mode and provides no multicast filtering.
	• **SNMP-Traps**—SNMP trap notifications are enabled.
	• **Up**—Interface is enabled and operational.

Table 148: show class-of-service interface Output Fields *(continued)*

Field Name	Field Description
Flags	The **Logical interface flags** field provides information about the logical interface and displays one or more of the following values: • **ACFC Encapsulation**—Address control field Compression (ACFC) encapsulation is enabled (negotiated successfully with a peer). • **Device-down**—Device has been administratively disabled. • **Disabled**—Interface is administratively disabled. • **Down**—A hardware failure has occurred. • **Clear-DF-Bit**—GRE tunnel or IPsec tunnel is configured to clear the Don't Fragment (DF) bit. • **Hardware-Down**—Interface protocol initialization failed to complete successfully. • **PFC**—Protocol field compression is enabled for the PPP session. • **Point-To-Point**—Interface is point-to-point. • **SNMP-Traps**—SNMP trap notifications are enabled. • **Up**—Interface is enabled and operational.
Encapsulation	Encapsulation on the logical interface.
Admin	Administrative state of the interface (**Up** or **Down**)
Link	Status of physical link (**Up** or **Down**).
Proto	Protocol configured on the interface.
Input Filter	Names of any firewall filters to be evaluated when packets are received on the interface, including any filters attached through activation of dynamic service.
Output Filter	Names of any firewall filters to be evaluated when packets are transmitted on the interface, including any filters attached through activation of dynamic service.

Table 148: show class-of-service interface Output Fields *(continued)*

Field Name	Field Description
Link flags	Provides information about the physical link and displays one or more of the following values: • **ACFC**—Address control field compression is configured. The Point-to-Point Protocol (PPP) session negotiates the ACFC option. • **Give-Up**—Link protocol does not continue connection attempts after repeated failures. • **Loose-LCP**—PPP does not use the Link Control Protocol (LCP) to indicate whether the link protocol is operational. • **Loose-LMI**—Frame Relay does not use the Local Management Interface (LMI) to indicate whether the link protocol is operational. • **Loose-NCP**—PPP does not use the Network Control Protocol (NCP) to indicate whether the device is operational. • **Keepalives**—Link protocol keepalives are enabled. • **No-Keepalives**—Link protocol keepalives are disabled. • **PFC**—Protocol field compression is configured. The PPP session negotiates the PFC option.
Hold-times	Current interface hold-time up and hold-time down, in milliseconds.
CoS queues	Number of CoS queues configured.
Last flapped	Date, time, and how long ago the interface went from down to up. The format is **Last flapped:** ***year-month-day hour:minute:second:timezone*** **(*hour:minute:second* ago)**. For example, **Last flapped: 2002-04-26 10:52:40 PDT (04:33:20 ago).**
Statistics last cleared	Number and rate of bytes and packets received and transmitted on the physical interface. • **Input bytes**—Number of bytes received on the interface. • **Output bytes**—Number of bytes transmitted on the interface. • **Input packets**—Number of packets received on the interface. • **Output packets**—Number of packets transmitted on the interface.
Exclude Overhead Bytes	Exclude the counting of overhead bytes from aggregate queue statistics. • **Disabled**—Default configuration. Includes the counting of overhead bytes in aggregate queue statistics. • **Enabled**—Excludes the counting of overhead bytes from aggregate queue statistics for just the physical interface. • **Enabled for hierarchy**—Excludes the counting of overhead bytes from aggregate queue statistics for the physical interface as well as all child interfaces, including logical interfaces and interface sets.
IPv6 transit statistics	Number of IPv6 transit bytes and packets received and transmitted on the logical interface if IPv6 statistics tracking is enabled.

Table 148: show class-of-service interface Output Fields *(continued)*

Field Name	Field Description
Input errors	Input errors on the interface. The labels are explained in the following list: • **Errors**—Sum of the incoming frame aborts and FCS errors. • **Drops**—Number of packets dropped by the input queue of the I/O Manager ASIC. If the interface is saturated, this number increments once for every packet that is dropped by the ASIC's RED mechanism. • **Framing errors**—Number of packets received with an invalid frame checksum (FCS). • **Runts**—Number of frames received that are smaller than the runt threshold. • **Giants**—Number of frames received that are larger than the giant threshold. • **Bucket Drops**—Drops resulting from the traffic load exceeding the interface transmit or receive leaky bucket configuration. • **Policed discards**—Number of frames that the incoming packet match code discarded because they were not recognized or not of interest. Usually, this field reports protocols that Junos OS does not handle. • **L3 incompletes**—Number of incoming packets discarded because they failed Layer 3 (usually IPv4) sanity checks of the header. For example, a frame with less than 20 bytes of available IP header is discarded. Layer 3 incomplete errors can be ignored by configuring the **ignore-l3-incompletes** statement. • **L2 channel errors**—Number of times the software did not find a valid logical interface for an incoming frame. • **L2 mismatch timeouts**—Number of malformed or short packets that caused the incoming packet handler to discard the frame as unreadable. • **HS link CRC errors**—Number of errors on the high-speed links between the ASICs responsible for handling the router interfaces. • **HS link FIFO overflows**—Number of FIFO overflows on the high-speed links between the ASICs responsible for handling the router interfaces.

Table 148: show class-of-service interface Output Fields *(continued)*

Field Name	Field Description
Output errors	Output errors on the interface. The labels are explained in the following list: • **Carrier transitions**—Number of times the interface has gone from **down** to **up**. This number does not normally increment quickly, increasing only when the cable is unplugged, the far-end system is powered down and up, or another problem occurs. If the number of carrier transitions increments quickly (perhaps once every 10 seconds), the cable, the far-end system, or the PIC is malfunctioning. • **Errors**—Sum of the outgoing frame aborts and FCS errors. • **Drops**—Number of packets dropped by the output queue of the I/O Manager ASIC. If the interface is saturated, this number increments once for every packet that is dropped by the ASIC's RED mechanism. NOTE: Due to accounting space limitations on certain Type 3 FPCs (which are supported in M320 and T640 routers), the **Drops** field does not always use the correct value for queue 6 or queue 7 for interfaces on 10-port 1-Gigabit Ethernet PICs. • **Aged packets**—Number of packets that remained in shared packet SDRAM so long that the system automatically purged them. The value in this field should never increment. If it does, it is most likely a software bug or possibly malfunctioning hardware. • **HS link FIFO underflows**—Number of FIFO underflows on the high-speed links between the ASICs responsible for handling the router interfaces. • **MTU errors**—Number of packets whose size exceeds the MTU of the interface.
Egress queues	Total number of egress Maximum usable queues on the specified interface.
Queue counters	CoS queue number and its associated user-configured forwarding class name. • **Queued packets**—Number of queued packets. • **Transmitted packets**—Number of transmitted packets. • **Dropped packets**—Number of packets dropped by the ASIC's RED mechanism. NOTE: Due to accounting space limitations on certain Type 3 FPCs (which are supported in M320 and T640 routers), the **Dropped packets** field does not always display the correct value for queue 6 or queue 7 for interfaces on 10-port 1-Gigabit Ethernet PICs.
SONET alarms **SONET defects**	(SONET) SONET media-specific alarms and defects that prevent the interface from passing packets. When a defect persists for a certain period, it is promoted to an alarm. Based on the router configuration, an alarm can ring the red or yellow alarm bell on the router or light the red or yellow alarm LED on the craft interface. See these fields for possible alarms and defects: **SONET PHY**, **SONET section**, **SONET line**, and **SONET path**.

Table 148: show class-of-service interface Output Fields *(continued)*

Field Name	Field Description
SONET PHY	Counts of specific SONET errors with detailed information. • **Seconds**—Number of seconds the defect has been active. • **Count**—Number of times that the defect has gone from inactive to active. • **State**—State of the error. A state other than **OK** indicates a problem. The **SONET PHY** field has the following subfields: • **PLL Lock**—Phase-locked loop • **PHY Light**—Loss of optical signal
SONET section	Counts of specific SONET errors with detailed information. • **Seconds**—Number of seconds the defect has been active. • **Count**—Number of times that the defect has gone from inactive to active. • **State**—State of the error. A state other than **OK** indicates a problem. The **SONET section** field has the following subfields: • **BIP-B1**—Bit interleaved parity for SONET section overhead • **SEF**—Severely errored framing • **LOS**—Loss of signal • **LOF**—Loss of frame • **ES-S**—Errored seconds (section) • **SES-S**—Severely errored seconds (section) • **SEFS-S**—Severely errored framing seconds (section)

Table 148: show class-of-service interface Output Fields *(continued)*

Field Name	Field Description
SONET line	Active alarms and defects, plus counts of specific SONET errors with detailed information. • **Seconds**—Number of seconds the defect has been active. • **Count**—Number of times that the defect has gone from inactive to active. • **State**—State of the error. A state other than **OK** indicates a problem. The **SONET line** field has the following subfields: • **BIP-B2**—Bit interleaved parity for SONET line overhead • **REI-L**—Remote error indication (near-end line) • **RDI-L**—Remote defect indication (near-end line) • **AIS-L**—Alarm indication signal (near-end line) • **BERR-SF**—Bit error rate fault (signal failure) • **BERR-SD**—Bit error rate defect (signal degradation) • **ES-L**—Errored seconds (near-end line) • **SES-L**—Severely errored seconds (near-end line) • **UAS-L**—Unavailable seconds (near-end line) • **ES-LFE**—Errored seconds (far-end line) • **SES-LFE**—Severely errored seconds (far-end line) • **UAS-LFE**—Unavailable seconds (far-end line)

Table 148: show class-of-service interface Output Fields *(continued)*

Field Name	Field Description
SONET path	Active alarms and defects, plus counts of specific SONET errors with detailed information. • **Seconds**—Number of seconds the defect has been active. • **Count**—Number of times that the defect has gone from inactive to active. • **State**—State of the error. A state other than **OK** indicates a problem. The **SONET path** field has the following subfields: • **BIP-B3**—Bit interleaved parity for SONET section overhead • **REI-P**—Remote error indication • **LOP-P**—Loss of pointer (path) • **AIS-P**—Path alarm indication signal • **RDI-P**—Path remote defect indication • **UNEQ-P**—Path unequipped • **PLM-P**—Path payload (signal) label mismatch • **ES-P**—Errored seconds (near-end STS path) • **SES-P**—Severely errored seconds (near-end STS path) • **UAS-P**—Unavailable seconds (near-end STS path) • **ES-PFE**—Errored seconds (far-end STS path) • **SES-PFE**—Severely errored seconds (far-end STS path) • **UAS-PFE**—Unavailable seconds (far-end STS path)
Received SONET overhead **Transmitted SONET overhead**	Values of the received and transmitted SONET overhead: • **C2**—Signal label. Allocated to identify the construction and content of the STS-level SPE and for PDI-P. • **F1**—Section user channel byte. This byte is set aside for the purposes of users. • **K1** and **K2**—These bytes are allocated for APS signaling for the protection of the multiplex section. • **J0**—Section trace. This byte is defined for STS-1 number 1 of an STS-*N* signal. Used to transmit a 1-byte fixed-length string or a 16-byte message so that a receiving terminal in a section can verify its continued connection to the intended transmitter. • **S1**—Synchronization status. The S1 byte is located in the first STS-1 number of an STS-*N* signal. • **Z3** and **Z4**—Allocated for future use.

Table 148: show class-of-service interface Output Fields *(continued)*

Field Name	Field Description
Received path trace **Transmitted path trace**	SONET/SDH interfaces allow path trace bytes to be sent inband across the SONET/SDH link. Juniper Networks and other router manufacturers use these bytes to help diagnose misconfigurations and network errors by setting the transmitted path trace message so that it contains the system hostname and name of the physical interface. The received path trace value is the message received from the router at the other end of the fiber. The transmitted path trace value is the message that this router transmits.
HDLC configuration	Information about the HDLC configuration. • **Policing bucket**—Configured state of the receiving policer. • **Shaping bucket**—Configured state of the transmitting shaper. • **Giant threshold**—Giant threshold programmed into the hardware. • **Runt threshold**—Runt threshold programmed into the hardware.
Packet Forwarding Engine configuration	Information about the configuration of the Packet Forwarding Engine: • **Destination slot**—FPC slot number. • **PLP byte**—Packet Level Protocol byte.
CoS information	Information about the CoS queue for the physical interface. • **CoS transmit queue**—Queue number and its associated user-configured forwarding class name. • **Bandwidth %**—Percentage of bandwidth allocated to the queue. • **Bandwidth bps**—Bandwidth allocated to the queue (in bps). • **Buffer %**—Percentage of buffer space allocated to the queue. • **Buffer usec**—Amount of buffer space allocated to the queue, in microseconds. This value is nonzero only if the buffer size is configured in terms of time. • **Priority**—Queue priority: **low** or **high**. • **Limit**—Displayed if rate limiting is configured for the queue. Possible values are **none** and **exact**. If **exact** is configured, the queue transmits only up to the configured bandwidth, even if excess bandwidth is available. If **none** is configured, the queue transmits beyond the configured bandwidth if bandwidth is available.
Forwarding classes	Total number of forwarding classes supported on the specified interface.
Egress queues	Total number of egress Maximum usable queues on the specified interface.
Queue	Queue number.
Forwarding classes	Forwarding class name.

Table 148: show class-of-service interface Output Fields *(continued)*

Field Name	Field Description
Queued Packets	Number of packets queued to this queue.
Queued Bytes	Number of bytes queued to this queue. The byte counts vary by PIC type.
Transmitted Packets	Number of packets transmitted by this queue. When fragmentation occurs on the egress interface, the first set of packet counters shows the postfragmentation values. The second set of packet counters (displayed under the **Packet Forwarding Engine Chassis Queues** field) shows the prefragmentation values.
Transmitted Bytes	Number of bytes transmitted by this queue. The byte counts vary by PIC type.
Tail-dropped packets	Number of packets dropped because of tail drop.

RED-dropped packets	Number of packets dropped because of random early detection (RED).
	• (M Series and T Series routers only) On M320 and M120 routers and the T Series routers, the total number of dropped packets is displayed. On all other M Series routers, the output classifies dropped packets into the following categories:
	  • **Low, non-TCP**—Number of low-loss priority non-TCP packets dropped because of RED.
	  • **Low, TCP**—Number of low-loss priority TCP packets dropped because of RED.
	  • **High, non-TCP**—Number of high-loss priority non-TCP packets dropped because of RED.
	  • **High, TCP**—Number of high-loss priority TCP packets dropped because of RED.
	• (MX Series routers with enhanced DPCs, and T Series routers with enhanced FPCs only) The output classifies dropped packets into the following categories:
	  • **Low**—Number of low-loss priority packets dropped because of RED.
	  • **Medium-low**—Number of medium-low loss priority packets dropped because of RED.
	  • **Medium-high**—Number of medium-high loss priority packets dropped because of RED.
	  • **High**—Number of high-loss priority packets dropped because of RED.
	NOTE: Due to accounting space limitations on certain Type 3 FPCs (which are supported in M320 and T640 routers), this field does not always display the correct value for queue 6 or queue 7 for interfaces on 10-port 1-Gigabit Ethernet PICs.

Table 148: show class-of-service interface Output Fields *(continued)*

<table>
<tr><th>Field Name</th><th>Field Description</th></tr>
<tr><td>RED-dropped bytes</td><td>Number of bytes dropped because of RED. The byte counts vary by PIC type.
(M Series and T Series routers only) On M320 and M120 routers and the T Series routers, only the total number of dropped bytes is displayed. On all other M Series routers, the output classifies dropped bytes into the following categories:
Low, non-TCP—Number of low-loss priority non-TCP bytes dropped because of RED.
Low, TCP—Number of low-loss priority TCP bytes dropped because of RED.
High, non-TCP—Number of high-loss priority non-TCP bytes dropped because of RED.
High, TCP—Number of high-loss priority TCP bytes dropped because of RED.
NOTE: Due to accounting space limitations on certain Type 3 FPCs (which are supported in M320 and T640 routers), this field does not always display the correct value for queue 6 or queue 7 for interfaces on 10-port 1-Gigabit Ethernet PICs.</td></tr>
<tr><td>Transmit rate</td><td>Configured transmit rate of the scheduler. The rate is a percentage of the total interface bandwidth.</td></tr>
<tr><td>Rate Limit</td><td>Rate limiting configuration of the queue. Possible values are :
None—No rate limit.
exact—Queue transmits at the configured rate.</td></tr>
<tr><td>Buffer size</td><td>Delay buffer size in the queue.</td></tr>
<tr><td>Priority</td><td>Scheduling priority configured as low or high.</td></tr>
<tr><td>Excess Priority</td><td>Priority of the excess bandwidth traffic on a scheduler: low, medium-low, medium-high, high, or none.</td></tr>
<tr><td>Drop profiles</td><td>Display the assignment of drop profiles.
Loss priority—Packet loss priority for drop profile assignment.
Protocol—Transport protocol for drop profile assignment.
Index—Index of the indicated object. Objects that have indexes in this output include schedulers and drop profiles.
Name—Name of the drop profile.
Type—Type of the drop profile: discrete or interpolated.
Fill Level—Percentage fullness of a queue.
Drop probability—Drop probability at this fill level.</td></tr>
<tr><td>Excess Priority</td><td>Priority of the excess bandwidth traffic on a scheduler.</td></tr>
</table>

Table 148: show class-of-service interface Output Fields *(continued)*

Field Name	Field Description
Drop profiles	Display the assignment of drop profiles. • **Loss priority**—Packet loss priority for drop profile assignment. • **Protocol**—Transport protocol for drop profile assignment. • **Index**—Index of the indicated object. Objects that have indexes in this output include schedulers and drop profiles. • **Name**—Name of the drop profile. • **Type**—Type of the drop profile: **discrete** or **interpolated**. • **Fill Level**—Percentage fullness of a queue. • **Drop probability**—Drop probability at this fill level.

Adjustment information	Display the assignment of shaping-rate adjustments on a scheduler node or queue.
	• **Adjusting application**—Application that is performing the shaping-rate adjustment.
	  ◦ The adjusting application can appear as **ancp LS-0**, which is the Junos OS Access Node Control Profile process (**ancpd**) that performs shaping-rate adjustments on schedule nodes.
	  ◦ The adjusting application can appear as **DHCP**, which adjusts the shaping-rate and overhead-accounting class-of-service attributes based on DSL Forum VSA conveyed in DHCP option 82, suboption 9 (Vendor Specific Information). The shaping rate is based on the actual-data-rate-downstream attribute. The overhead accounting value is based on the access-loop-encapsulation attribute and specifies whether the access loop uses Ethernet (frame mode) or ATM (cell mode).
	  ◦ The adjusting application can also appear as **pppoe**, which adjusts the shaping-rate and overhead-accounting class-of-service attributes on dynamic subscriber interfaces in a broadband access network based on access line parameters in Point-to-Point Protocol over Ethernet (PPPoE) Tags [TR-101]. This feature is supported on MPC/MIC interfaces on MX Series routers. The shaping rate is based on the actual-data-rate-downstream attribute. The overhead accounting value is based on the access-loop-encapsulation attribute and specifies whether the access loop uses Ethernet (frame mode) or ATM (cell mode).
	• **Adjustment type**—Type of adjustment: **absolute** or **delta**.
	• **Configured shaping rate**—Shaping rate configured for the scheduler node or queue.
	• **Adjustment value**—Value of adjusted shaping rate.
	• **Adjustment target**—Level of shaping-rate adjustment performed: **node** or **queue**.
	• **Adjustment overhead-accounting mode**—Configured shaping mode: **frame** or **cell**.
	• **Adjustment overhead bytes**—Number of bytes that the ANCP agent adds to or subtracts from the actual downstream frame overhead before reporting the adjusted values to CoS.
	• **Adjustment target**—Level of shaping-rate adjustment performed: **node** or **queue**.
	• **Adjustment multicast index**—

Sample Output

show class-of-service interface (Physical)

user@host> **show class-of-service interface so-0/2/3**

```
Physical interface: so-0/2/3, Index: 135
Maximum usable queues: 8, Queues in use: 4
Total non–default queues created: 4
  Scheduler map: <default>, Index: 2032638653

  Logical interface: fe-0/0/1.0, Index: 68, Dedicated Queues: no
    Shaping rate: 32000
    Object                   Name                   Type            Index
    Scheduler-map            <default>                               27
    Rewrite                  exp-default            exp              21
    Classifier               exp-default            exp              5
    Classifier               ipprec-compatibility   ip               8
    Forwarding–class–map     exp-default            exp              5
```

show class-of-service interface (Logical)

user@host> **show class-of-service interface so-0/2/3.0**

```
Logical interface: so-0/2/3.0, Index: 68, Dedicated Queues: no
    Shaping rate: 32000
    Object                   Name                   Type              Index
    Scheduler-map            <default>                                 27
    Rewrite                  exp-default            exp                21
    Classifier               exp-default            exp                5
    Classifier               ipprec-compatibility   ip                 8
    Forwarding–class–map     exp-default            exp                5
```

show class-of-service interface (Gigabit Ethernet)

user@host> **show class-of-service interface ge-6/2/0**

```
Physical interface: ge-6/2/0, Index: 175
Maximum usable queues: 4, Queues in use: 4
  Scheduler map: <default>, Index: 2
  Input scheduler map: <default>, Index: 3
  Chassis scheduler map: <default-chassis>, Index: 4
```

show class-of-service interface (ANCP)

user@host> **show class-of-service interface pp0.1073741842**

```
  Logical interface: pp0.1073741842, Index: 341
Object                  Name                   Type                    Index
Traffic-control-profile TCP-CVLAN              Output                  12408
Classifier              dscp-ipv6-compatibility dscp-ipv6                  9
Classifier              ipprec-compatibility   ip                         13

    Adjusting application: ancp LS-0
      Adjustment type: absolute
      Configured shaping rate: 4000000
      Adjustment value: 11228000
      Adjustment overhead-accounting mode: Frame Mode
      Adjustment overhead bytes: 50
      Adjustment target: node
```

show class-of-service interface (PPPoE Interface)

user@host> **show class-of-service interface pp0.1**

```
Logical interface: pp0.1, Index: 85
    Object                  Name                   Type           Index
    Traffic-control-profile tcp-pppoe.o.pp0.1      Output         2726446535
    Classifier              ipprec-compatibility   ip             13

    Adjusting application: PPPoE
      Adjustment type: absolute
      Adjustment value: 5000000
      Adjustment overhead-accounting mode: cell
      Adjustment target: node
```

show class-of-service interface (DHCP Interface)

user@host> **show class-of-service interface demux0.1**

```
Logical interface: pp0.1, Index: 85
    Object                  Name                   Type           Index
    Traffic-control-profile tcp-dhcp.o.demux0.1    Output         2726446535
    Classifier              ipprec-compatibility   ip             13

    Adjusting application: DHCP
      Adjustment type: absolute
```

```
      Adjustment value: 5000000
      Adjustment overhead-accounting mode: cell
      Adjustment target: node
```

show class-of-service interface (T4000 Routers with Type 5 FPCs)

user@host> **show class-of-service interface xe-4/0/0**

```
Physical interface: xe-4/0/0, Index: 153
    Maximum usable queues: 8, Queues in use: 4
      Shaping rate: 5000000000 bps
      Scheduler map: <default>, Index: 2
      Congestion-notification: Disabled

      Logical interface: xe-4/0/0.0, Index: 77
        Object                  Name                    Type
Index
        Classifier              ipprec-compatibility   ip
 13
```

show class-of-service interface detail

user@host> **show class-of-service interface ge-0/3/0 detail**

```

Physical interface: ge-0/3/0, Enabled, Physical link is Up
  Link-level type: Ethernet, MTU: 1518, Speed: 1000mbps, Loopback: Disabled, Source
 filtering: Disabled, Flow control: Enabled, Auto-negotiation: Enabled, Remote
fault: Online
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000

  Physical interface: ge-0/3/0, Index: 138
  Maximum usable queues: 4, Queues in use: 5
  Shaping rate: 50000 bps
  Scheduler map: interface-schedular-map, Index: 58414
  Input shaping rate: 10000 bps
  Input scheduler map: schedular-map, Index: 15103
  Chassis scheduler map: <default-chassis>, Index: 4
  Congestion-notification: Disabled

  Logical interface ge-0/3/0.0
    Flags: SNMP-Traps 0x4000 VLAN-Tag [ 0x8100.1 ]  Encapsulation: ENET2
```

```
    inet
    mpls
Interface       Admin Link Proto Input Filter         Output Filter
ge-0/3/0.0      up    up   inet
                           mpls
Interface       Admin Link Proto Input Policer         Output Policer
ge-0/3/0.0      up    up   inet
                           mpls

  Logical interface: ge-0/3/0.0, Index: 68
    Object                  Name                   Type                  Index
    Rewrite                 exp-default            exp (mpls-any)          33
    Classifier              exp-default            exp                     10
    Classifier              ipprec-compatibility   ip                      13


  Logical interface ge-0/3/0.1
    Flags: SNMP-Traps 0x4000 VLAN-Tag [ 0x8100.2 ]  Encapsulation: ENET2
    inet
Interface       Admin Link Proto Input Filter         Output Filter
ge-0/3/0.1      up    up   inet
Interface       Admin Link Proto Input Policer         Output Policer
ge-0/3/0.1      up    up   inet

  Logical interface: ge-0/3/0.1, Index: 69
    Object                  Name                   Type               Index
    Classifier              ipprec-compatibility   ip                  13
```

show class-of-service interface comprehensive

user@host> **show class-of-service interface ge-0/3/0 comprehensive**

```
Physical interface: ge-0/3/0, Enabled, Physical link is Up
  Interface index: 138, SNMP ifIndex: 601, Generation: 141
  Link-level type: Ethernet, MTU: 1518, Speed: 1000mbps, BPDU Error: None,
MAC-REWRITE Error: None, Loopback: Disabled, Source filtering: Disabled, Flow
control: Enabled,
  Auto-negotiation: Enabled, Remote fault: Online
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
  CoS queues     : 4 supported, 4 maximum usable queues
  Schedulers     : 256
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 00:14:f6:f4:b4:5d, Hardware address: 00:14:f6:f4:b4:5d
```

```
  Last flapped   : 2010-09-07 06:35:22 PDT (15:14:42 ago)
  Statistics last cleared: Never  Exclude Overhead Bytes: Disabled
  Traffic statistics:
   Input  bytes  :                    0                   0 bps
   Output bytes  :                    0                   0 bps
   Input  packets:                    0                   0 pps
   Output packets:                    0                   0 pps
   IPv6 total statistics:
    Input  bytes  :                   0
    Output bytes  :                   0
    Input  packets:                   0
    Output packets:                   0
  Ingress traffic statistics at Packet Forwarding Engine:
   Input  bytes  :                    0                   0 bps
   Input  packets:                    0                   0 pps
   Drop   bytes  :                    0                   0 bps
   Drop   packets:                    0                   0 pps
  Label-switched interface (LSI) traffic statistics:
   Input  bytes  :                    0                   0 bps
   Input  packets:                    0                   0 pps
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Policed discards: 0, L3
incompletes: 0, L2 channel errors: 0, L2 mismatch timeouts: 0, FIFO errors: 0,
Resource errors: 0
  Output errors:
    Carrier transitions: 5, Errors: 0, Drops: 0, Collisions: 0, Aged packets: 0,
FIFO errors: 0, HS link CRC errors: 0, MTU errors: 0, Resource errors: 0
  Ingress queues: 4 supported, 5 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 af3                            0                    0                    0
    1 af2                            0                    0                    0
    2 ef2                            0                    0                    0
    3 ef1                            0                    0                    0
  Egress queues: 4 supported, 5 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets
    0 af3                            0                    0                    0
    1 af2                            0                    0                    0
    2 ef2                            0                    0                    0
    3 ef1                            0                    0                    0
  Active alarms  : None
  Active defects : None
  MAC statistics:                      Receive         Transmit
    Total octets                             0                0
    Total packets                            0                0
```

```
    Unicast packets                         0                 0
    Broadcast packets                       0                 0
    Multicast packets                       0                 0
    CRC/Align errors                        0                 0
    FIFO errors                             0                 0
    MAC control frames                      0                 0
    MAC pause frames                        0                 0
    Oversized frames                        0
    Jabber frames                           0
    Fragment frames                         0
    VLAN tagged frames                      0
    Code violations                         0
  Filter statistics:
    Input packet count                      0
    Input packet rejects                    0
    Input DA rejects                        0
    Input SA rejects                        0
    Output packet count                                       0
    Output packet pad count                                   0
    Output packet error count                                 0
    CAM destination filters: 0, CAM source filters: 0
  Autonegotiation information:
    Negotiation status: Complete
    Link partner:
        Link mode: Full-duplex, Flow control: Symmetric/Asymmetric, Remote fault:
 OK
    Local resolution:
      Flow control: Symmetric, Remote fault: Link OK
  Packet Forwarding Engine configuration:
    Destination slot: 0
  CoS information:
    Direction : Output
    CoS transmit queue               Bandwidth               Buffer Priority
Limit
                              %             bps     %            usec
    2 ef2                    39           19500     0             120     high
none
    Direction : Input
    CoS transmit queue               Bandwidth               Buffer Priority
Limit
                              %             bps     %            usec
    0 af3                    30            3000    45               0      low
none
```

```
Physical interface: ge-0/3/0, Enabled, Physical link is Up
  Interface index: 138, SNMP ifIndex: 601
Forwarding classes: 16 supported, 5 in use
Ingress queues: 4 supported, 5 in use
Queue: 0, Forwarding classes: af3
  Queued:
    Packets              :                    0                      0 pps
    Bytes                :                    0                      0 bps
  Transmitted:
    Packets              :                    0                       0 pps
    Bytes                :                    0                      0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                      0 pps
    RED-dropped bytes    :                    0                      0 bps
Queue: 1, Forwarding classes: af2
  Queued:
    Packets              :                    0                      0 pps
    Bytes                :                    0                      0 bps
  Transmitted:
    Packets              :                    0                       0 pps
    Bytes                :                    0                      0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                      0 pps
    RED-dropped bytes    :                    0                      0 bps
Queue: 2, Forwarding classes: ef2
  Queued:
    Packets              :                    0                      0 pps
    Bytes                :                    0                      0 bps
  Transmitted:
    Packets              :                    0                       0 pps
    Bytes                :                    0                      0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                      0 pps
    RED-dropped bytes    :                    0                      0 bps
Queue: 3, Forwarding classes: ef1
  Queued:
    Packets              :                    0                      0 pps
    Bytes                :                    0                      0 bps
  Transmitted:
    Packets              :                    0                       0 pps
    Bytes                :                    0                      0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                      0 pps
    RED-dropped bytes    :                    0                      0 bps
```

```
Forwarding classes: 16 supported, 5 in use
Egress queues: 4 supported, 5 in use
Queue: 0, Forwarding classes: af3
  Queued:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
  Transmitted:
    Packets              :                     0                      0 pps
    Bytes                :                     0                     0 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                     0                     0 pps
    RL-dropped bytes     :                     0                     0 bps
    RED-dropped packets  :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
Queue: 1, Forwarding classes: af2
  Queued:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
  Transmitted:
    Packets              :                     0                      0 pps
    Bytes                :                     0                     0 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                     0                     0 pps
    RL-dropped bytes     :                     0                     0 bps
    RED-dropped packets  :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
Queue: 2, Forwarding classes: ef2
  Queued:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
  Transmitted:
    Packets              :                     0                      0 pps
    Bytes                :                     0                     0 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                     0                     0 pps
    RL-dropped bytes     :                     0                     0 bps
    RED-dropped packets  :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
Queue: 3, Forwarding classes: ef1
  Queued:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
  Transmitted:
    Packets              :                     0                      0 pps
```

```
    Bytes                :                    0                     0 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                    0                     0 pps
    RL-dropped bytes     :                    0                     0 bps
    RED-dropped packets  :                    0                     0 pps
    RED-dropped bytes    :                    0                     0 bps

Packet Forwarding Engine Chassis Queues:
Queues: 4 supported, 5 in use
Queue: 0, Forwarding classes: af3
  Queued:
    Packets              :                    0                     0 pps
    Bytes                :                    0                     0 bps
  Transmitted:
    Packets              :                    0                      0 pps
    Bytes                :                    0                     0 bps
    Tail-dropped packets :                    0                     0 pps
    RED-dropped packets  : Not Available
    RED-dropped bytes    : Not Available
Queue: 1, Forwarding classes: af2
  Queued:
    Packets              :                    0                     0 pps
    Bytes                :                    0                     0 bps
  Transmitted:
    Packets              :                    0                      0 pps
    Bytes                :                    0                     0 bps
    Tail-dropped packets :                    0                     0 pps
    RED-dropped packets  : Not Available
    RED-dropped bytes    : Not Available
Queue: 2, Forwarding classes: ef2
  Queued:
    Packets              :                    0                     0 pps
    Bytes                :                    0                     0 bps
  Transmitted:
    Packets              :                    0                      0 pps
    Bytes                :                    0                     0 bps
    Tail-dropped packets :                    0                     0 pps
    RED-dropped packets  : Not Available
    RED-dropped bytes    : Not Available
Queue: 3, Forwarding classes: ef1
  Queued:
    Packets              :               108546                     0 pps
    Bytes                :             12754752                   376 bps
  Transmitted:
```

```
    Packets              :                108546                        0 pps
    Bytes                :              12754752                      376 bps
    Tail-dropped packets :                     0                        0 pps
    RED-dropped packets  : Not Available
    RED-dropped bytes    : Not Available

Physical interface: ge-0/3/0, Index: 138
Maximum usable queues: 4, Queues in use: 5
  Shaping rate: 50000 bps

Scheduler map: interface-schedular-map, Index: 58414

  Scheduler: ef2, Forwarding class: ef2, Index: 39155
    Transmit rate: 39 percent, Rate Limit: none, Buffer size: 120 us, Buffer Limit:
 none, Priority: high
    Excess Priority: unspecified
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any             1    < default-drop-profile>
      Medium low      any             1    < default-drop-profile>
      Medium high     any             1    < default-drop-profile>
      High            any             1    < default-drop-profile>
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100
  Input shaping rate: 10000 bps
  Input scheduler map: schedular-map

Scheduler map: schedular-map, Index: 15103

  Scheduler: af3, Forwarding class: af3, Index: 35058
    Transmit rate: 30 percent, Rate Limit: none, Buffer size: 45 percent, Buffer
Limit: none, Priority: low
    Excess Priority: unspecified
    Drop profiles:
```

```
      Loss priority   Protocol    Index    Name
      Low             any         40582    green
      Medium low      any             1    < default-drop-profile>
      Medium high     any             1    < default-drop-profile>
      High            any         18928    yellow
Drop profile: green, Type: discrete, Index: 40582
  Fill level    Drop probability
          50                   0
         100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100
Drop profile: yellow, Type: discrete, Index: 18928
  Fill level    Drop probability
          50                   0
         100                 100
  Chassis scheduler map: < default-drop-profile>
Scheduler map: < default-drop-profile>, Index: 4

  Scheduler: < default-drop-profile>, Forwarding class: af3, Index: 25
    Transmit rate: 25 percent, Rate Limit: none, Buffer size: 25 percent, Buffer
Limit: none, Priority: low
    Excess Priority: low
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any             1    < default-drop-profile>
      Medium low      any             1    < default-drop-profile>
      Medium high     any             1    < default-drop-profile>
      High            any             1    < default-drop-profile>
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100
```

```
  Scheduler: < default-drop-profile>, Forwarding class: af2, Index: 25
    Transmit rate: 25 percent, Rate Limit: none, Buffer size: 25 percent, Buffer
Limit: none, Priority: low
    Excess Priority: low
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any             1    < default-drop-profile>
      Medium low      any             1    < default-drop-profile>
      Medium high     any             1    < default-drop-profile>
      High            any             1    < default-drop-profile>
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100

  Scheduler: < default-drop-profile>, Forwarding class: ef2, Index: 25
    Transmit rate: 25 percent, Rate Limit: none, Buffer size: 25 percent, Buffer
Limit: none, Priority: low
    Excess Priority: low
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any             1    < default-drop-profile>
      Medium low      any             1    < default-drop-profile>
      Medium high     any             1    < default-drop-profile>
      High            any             1    < default-drop-profile>
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
```

```
  Fill level    Drop probability
         100                 100

  Scheduler: < default-drop-profile>, Forwarding class: ef1, Index: 25
    Transmit rate: 25 percent, Rate Limit: none, Buffer size: 25 percent, Buffer
Limit: none, Priority: low
    Excess Priority: low
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any             1    < default-drop-profile>
      Medium low      any             1    < default-drop-profile>
      Medium high     any             1    < default-drop-profile>
      High            any             1    < default-drop-profile>
Drop profile: , Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100
  Congestion-notification: Disabled
Forwarding class                       ID      Queue  Restricted queue  Fabric
priority  Policing priority
  af3                                   0       0          0             low
           normal
  af2                                   1       1          1             low
           normal
  ef2                                   2       2          2             high
           normal
  ef1                                   3       3          3             high
           normal
  af1                                   4       4          0             low
           normal

  Logical interface ge-0/3/0.0 (Index 68) (SNMP ifIndex 152) (Generation 159)
    Flags: SNMP-Traps 0x4000 VLAN-Tag [ 0x8100.1 ]  Encapsulation: ENET2
    Traffic statistics:
     Input  bytes  :                    0
     Output bytes  :                    0
```

```
      Input  packets:                   0
      Output packets:                   0
    Local statistics:
      Input  bytes  :                   0
      Output bytes  :                   0
      Input  packets:                   0
      Output packets:                   0
    Transit statistics:
      Input  bytes  :                   0                    0 bps
      Output bytes  :                   0                    0 bps
      Input  packets:                   0                    0 pps
      Output packets:                   0                    0 pps
    Protocol inet, MTU: 1500, Generation: 172, Route table: 0
      Flags: Sendbcast-pkt-to-re
      Input Filters: filter-in-ge-0/3/0.0-i,
      Policer: Input: p1-ge-0/3/0.0-inet-i
    Protocol mpls, MTU: 1488, Maximum labels: 3, Generation: 173, Route table: 0
      Flags: Is-Primary
      Output Filters: exp-filter,,,,,

Logical interface ge-1/2/0.0 (Index 347) (SNMP ifIndex 638) (Generation 156)

Forwarding class ID  Queue  Restricted queue  Fabric priority  Policing priority
  SPU priority
best-effort      0   0      0                 low              normal
   low

Aggregate Forwarding-class statistics per forwarding-class
  Aggregate Forwarding-class statistics:
   Forwarding-class statistics:

Forwarding-class best-effort statistics:
    Input unicast bytes:      0
    Output unicast bytes:     0
    Input unicast packets:    0
    Output unicast packets:   0

    Input multicast bytes:    0
    Output multicast bytes:   0
    Input multicast packets:  0
    Output multicast packets: 0

Forwarding-class expedited-forwarding statistics:
    Input unicast bytes:      0
```

```
    Output unicast bytes:     0
    Input unicast packets:    0
    Output unicast packets:   0

    Input multicast bytes:    0
    Output multicast bytes:   0
    Input multicast packets:  0
    Output multicast packets: 0

IPv4 protocol forwarding-class statistics:
  Forwarding-class statistics:
  Forwarding-class best-effort statistics:

    Input unicast bytes:      0
    Output unicast bytes:     0
    Input unicast packets:    0
    Output unicast packets:   0

    Input multicast bytes:    0
    Output multicast bytes:   0
    Input multicast packets:  0
    Output multicast packets: 0

Forwarding-class expedited-forwarding statistics:
    Input unicast bytes:      0
    Output unicast bytes:     0
    Input unicast packets:    0
    Output unicast packets:   0

    Input multicast bytes:    0
    Output multicast bytes:   0
    Input multicast packets:  0
    Output multicast packets: 0

    IPv6  protocol forwarding-class statistics:
    Forwarding-class statistics:
     Forwarding-class best-effort statistics:

    Input unicast bytes:      0
    Output unicast bytes:     0
    Input unicast packets:    0
    Output unicast packets:   0

    Input multicast bytes:    0
```

```
    Output multicast bytes:   0
    Input multicast packets:  0
    Output multicast packets: 0

Forwarding-class expedited-forwarding statistics:
    Input unicast bytes:      0
    Output unicast bytes:     0
    Input unicast packets:    0
    Output unicast packets:   0

    Input multicast bytes:    0
    Output multicast bytes:   0
    Input multicast packets:  0
    Output multicast packets: 0

Logical interface ge-0/3/0.0 (Index 68) (SNMP ifIndex 152)
    Flags: SNMP-Traps 0x4000 VLAN-Tag [ 0x8100.1 ]  Encapsulation: ENET2
    Input packets : 0
    Output packets: 0

Interface       Admin Link Proto Input Filter         Output Filter
ge-0/3/0.0      up    up   inet  filter-in-ge-0/3/0.0-i
                           mpls                       exp-filter
Interface       Admin Link Proto Input Policer         Output Policer
ge-0/3/0.0      up    up
                           inet  p1-ge-0/3/0.0-inet-i
                           mpls

Filter: filter-in-ge-0/3/0.0-i
Counters:
Name                                                Bytes              Packets
count-filter-in-ge-0/3/0.0-i                            0                    0

Filter: exp-filter
Counters:
Name                                                Bytes              Packets
count-exp-seven-match                                   0                    0
count-exp-zero-match                                    0                    0
Policers:
Name                                              Packets
p1-ge-0/3/0.0-inet-i                                    0


  Logical interface: ge-0/3/0.0, Index: 68
```

```
    Object                  Name                    Type                    Index
    Rewrite                 exp-default             exp (mpls-any)             33

Rewrite rule: exp-default, Code point type: exp, Index: 33
  Forwarding class                    Loss priority       Code point
  af3                                 low                 000
  af3                                 high                001
  af2                                 low                 010
  af2                                 high                011
  ef2                                 low                 100
  ef2                                 high                101
  ef1                                 low                 110
  ef1                                 high                111
    Object                  Name                    Type                    Index
    Classifier              exp-default             exp                        10

Classifier: exp-default, Code point type: exp, Index: 10
  Code point         Forwarding class                    Loss priority
  000                af3                                 low
  001                af3                                 high
  010                af2                                 low
  011                af2                                 high
  100                ef2                                 low
  101                ef2                                 high
  110                ef1                                 low
  111                ef1                                 high
    Object                  Name                    Type                    Index
    Classifier              ipprec-compatibility    ip                         13

Classifier: ipprec-compatibility, Code point type: inet-precedence, Index: 13
  Code point         Forwarding class                    Loss priority
  000                af3                                 low
  001                af3                                 high
  010                af3                                 low
  011                af3                                 high
  100                af3                                 low
  101                af3                                 high
  110                ef1                                 low
  111                ef1                                 high
Forwarding class                       ID      Queue  Restricted queue  Fabric
priority  Policing priority
  af3                                   0       0          0             low
           normal
  af2                                   1       1          1             low
```

```
          normal
ef2                               2       2          2             high
          normal
ef1                               3       3          3             high
          normal
af1                               4       4          0             low
          normal

Logical interface ge-0/3/0.1 (Index 69) (SNMP ifIndex 154) (Generation 160)
  Flags: SNMP-Traps 0x4000 VLAN-Tag [ 0x8100.2 ]  Encapsulation: ENET2
  Traffic statistics:
   Input  bytes  :                    0
   Output bytes  :                    0
   Input  packets:                    0
   Output packets:                    0
  Local statistics:
   Input  bytes  :                    0
   Output bytes  :                    0
   Input  packets:                    0
   Output packets:                    0
  Transit statistics:
   Input  bytes  :                    0                    0 bps
   Output bytes  :                    0                    0 bps
   Input  packets:                    0                    0 pps
   Output packets:                    0                    0 pps
  Protocol inet, MTU: 1500, Generation: 174, Route table: 0
    Flags: Sendbcast-pkt-to-re

Logical interface ge-0/3/0.1 (Index 69) (SNMP ifIndex 154)
  Flags: SNMP-Traps 0x4000 VLAN-Tag [ 0x8100.2 ]  Encapsulation: ENET2
  Input packets : 0
  Output packets: 0


Interface       Admin Link Proto Input Filter         Output Filter
ge-0/3/0.1      up    up   mpls
Interface       Admin Link Proto Input Policer         Output Policer
ge-0/3/0.1      up    up
                           mpls

Logical interface: ge-0/3/0.1, Index: 69
  Object                 Name                   Type                   Index
  Classifier             ipprec-compatibility   ip                        13
```

```
Classifier: ipprec-compatibility, Code point type: inet-precedence, Index: 13
  Code point         Forwarding class                    Loss priority
  000                af3                                 low
  001                af3                                 high
  010                af3                                 low
  011                af3                                 high
  100                af3                                 low
  101                af3                                 high
  110                ef1                                 low
  111                ef1                                 high
Forwarding class                       ID      Queue  Restricted queue  Fabric
priority  Policing priority
  af3                                   0       0          0             low
           normal
  af2                                   1       1          1             low
           normal
  ef2                                   2       2          2             high
           normal
  ef1                                   3       3          3             high
           normal
  af1                                   4       4          0             low
           normal
```

show class-of-service interface (ACX Series Routers)

`user@host-g11# show class-of-service interface`

```
Physical interface: at-0/0/0, Index: 130
Maximum usable queues: 4, Queues in use: 4
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled

  Logical interface: at-0/0/0.0, Index: 69

  Logical interface: at-0/0/0.32767, Index: 70

Physical interface: at-0/0/1, Index: 133
Maximum usable queues: 4, Queues in use: 4
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled

  Logical interface: at-0/0/1.0, Index: 71
```

```
  Logical interface: at-0/0/1.32767, Index: 72

Physical interface: ge-0/1/0, Index: 146
Maximum usable queues: 8, Queues in use: 5
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled
Object                  Name                   Type                    Index
Rewrite                 dscp-default           dscp                       31
Classifier              d1                     dscp                    11331
Classifier              ci                     ieee8021p                 583

  Logical interface: ge-0/1/0.0, Index: 73
Object                  Name                   Type                    Index
Rewrite                 custom-exp             exp (mpls-any)          46413

  Logical interface: ge-0/1/0.1, Index: 74

  Logical interface: ge-0/1/0.32767, Index: 75

Physical interface: ge-0/1/1, Index: 147
Maximum usable queues: 8, Queues in use: 5
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled
Object                  Name                   Type                    Index
Classifier              ipprec-compatibility   ip                         13

  Logical interface: ge-0/1/1.0, Index: 76

Physical interface: ge-0/1/2, Index: 148
Maximum usable queues: 8, Queues in use: 5
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled
Object                  Name                   Type                    Index
Rewrite                 ri                     ieee8021p (outer)       35392
Classifier              ci                     ieee8021p                 583

Physical interface: ge-0/1/3, Index: 149
Maximum usable queues: 8, Queues in use: 5
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled
Object                  Name                   Type                    Index
Classifier              ipprec-compatibility   ip                         13
```

```
  Logical interface: ge-0/1/3.0, Index: 77
Object                 Name                   Type                    Index
Rewrite                custom-exp2            exp (mpls-any)          53581

Physical interface: ge-0/1/4, Index: 150
Maximum usable queues: 8, Queues in use: 5
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled
Object                 Name                   Type                    Index
Classifier             ipprec-compatibility   ip                         13

Physical interface: ge-0/1/5, Index: 151
Maximum usable queues: 8, Queues in use: 5
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled
Object                 Name                   Type                    Index
Classifier             ipprec-compatibility   ip                         13

Physical interface: ge-0/1/6, Index: 152
Maximum usable queues: 8, Queues in use: 5
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled
Object                 Name                   Type                    Index
Classifier             ipprec-compatibility   ip                         13

Physical interface: ge-0/1/7, Index: 153
Maximum usable queues: 8, Queues in use: 5
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled
Object                 Name                   Type                    Index
Classifier             d1                     dscp                    11331

Physical interface: ge-0/2/0, Index: 154
Maximum usable queues: 8, Queues in use: 5
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled
Object                 Name                   Type                    Index
Classifier             ipprec-compatibility   ip                         13

Physical interface: ge-0/2/1, Index: 155
Maximum usable queues: 8, Queues in use: 5
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled
Object                 Name                   Type                    Index
```

```
Classifier              ipprec-compatibility   ip                              13

  Logical interface: ge-0/2/1.0, Index: 78

  Logical interface: ge-0/2/1.32767, Index: 79

Physical interface: xe-0/3/0, Index: 156
Maximum usable queues: 8, Queues in use: 5
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled
Object                  Name                   Type                         Index
Classifier              ipprec-compatibility   ip                              13

  Logical interface: xe-0/3/0.0, Index: 80

Physical interface: xe-0/3/1, Index: 157
Maximum usable queues: 8, Queues in use: 5
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled
Object                  Name                   Type                         Index
Classifier              ipprec-compatibility   ip                              13

  Logical interface: xe-0/3/1.0, Index: 81

[edit]
user@host-g11#
```

show class-of-service interface (PPPoE Subscriber Interface for Enhanced Subscriber Management)

user@host> **show class-of-service interface pp0.3221225474**

```
  Logical interface: pp0.3221225475, Index: 3221225475
Object                  Name                   Type                         Index
Traffic-control-profile TC_PROF_100_199_SERIES_UID1006 Output          4294967312
Scheduler-map           SMAP-1_UID1002         Output                  4294967327
Rewrite-Output          ieee-rewrite           ieee8021p                    60432
Rewrite-Output          rule1                  ip                           50463

    Adjusting application: PPPoE IA tags
      Adjustment type: absolute
      Configured shaping rate: 11000000
      Adjustment value: 5000000
      Adjustment target: node
```

```
Adjusting application: ucac
  Adjustment type: delta
  Configured shaping rate: 5000000
  Adjustment value: 100000
  Adjustment target: node
```

show class-of-service multi-destination

Syntax

```
show class-of-service multi-destination
```

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.
Command introduced in Junos OS Release 17.1 for the EX4300 switch.

Description

For each class-of-service (CoS) multidestination classifier, display the classifier type.

Options

none—Display all multidestination classifiers.

Required Privilege Level

view

RELATED DOCUMENTATION

Defining CoS BA Classifiers (DSCP, DSCP IPv6, IEEE 802.1p) | **94**
Example: Configuring Multidestination (Multicast, Broadcast, DLF) Classifiers | **104**
Understanding CoS Classifiers | **84**
Understanding CoS Classifiers
Understanding Applying CoS Classifiers and Rewrite Rules to Interfaces | **116**
Understanding Applying CoS Classifiers and Rewrite Rules to Interfaces

Output Fields

Table 149 on page 958 describes the output fields for the **show class-of-service multi-destination** command. Output fields are listed in the approximate order in which they appear.

Table 149: show class-of-service multi-destination Output Fields

Field Name	Field Description
Family ethernet	Family to which the classifier belongs.
Classifier Name	Name of the classifier.

Table 149: show class-of-service multi-destination Output Fields *(continued)*

Field Name	Field Description
Classifier Type	Type of the classifier: **dscp** or **ieee-802.1**.
Classifier Index	Internal index of the classifier.

Sample Output

show class-of-service multi-destination

user@switch> show class-of-service multi-destination

```
Family ethernet:
Classifier Name                Classifier Type               Classifier Index
  ba-mcast-classifier            ieee-802.1                    62376
```

show class-of-service rewrite-rule

Syntax

```
show class-of-service rewrite-rule
<name name>
<type type>
```

Release Information

Command introduced before Junos OS Release 7.4.

Description

Display the mapping of forwarding classes and loss priority to code point values.

Options

none—Display all rewrite rules.

name *name*—(Optional) Display the specified rewrite rule.

type *type*—(Optional) Display the rewrite rule of the specified type. The rewrite rule type can be one of the following:

- **dscp**—For IPv4 traffic.
- **dscp-ipv6**—For IPv6 traffic.
- **exp**—For MPLS traffic.
- **frame-relay-de**—(SRX Series only) For Frame Relay traffic.
- **ieee-802.1**—For Layer 2 traffic.
- **inet-precedence**—For IPv4 traffic.

Required Privilege Level

view

RELATED DOCUMENTATION

Rewrite Rules Overview

List of Sample Output

show class-of-service rewrite-rule type dscp on page 961

Output Fields

Table 150 on page 961 describes the output fields for the **show class-of-service rewrite-rule** command. Output fields are listed in the approximate order in which they appear.

Table 150: show class-of-service rewrite-rule Output Fields

Field Name	Field Description
Rewrite rule	Name of the rewrite rule.
Code point type	Type of rewrite rule: **dscp**, **dscp-ipv6**, **exp**, **frame-relay-de**, or **inet-precedence**.
Forwarding class	Classification of a packet affecting the forwarding, scheduling, and marking policies applied as the packet transits the router or switch.
Index	Internal index for this particular rewrite rule.
Loss priority	Loss priority for rewriting.
Code point	Code point value to rewrite.

Sample Output

show class-of-service rewrite-rule type dscp

`user@host>` **show class-of-service rewrite-rule type dscp**

```
Rewrite rule: dscp-default, Code point type: dscp
  Forwarding class                    Loss priority       Code point
  gold                                high                000000
  silver                              low                 110000
  silver                              high                111000
  bronze                              low                 001010
  bronze                              high                001100
  lead                                high                101110

Rewrite rule: abc-dscp-rewrite, Code point type: dscp, Index: 3245
Forwarding class                    Loss priority       Code point
  gold                                low                 000111
  gold                                high                001010
  silver                              low                 110000
  silver                              high                111000
```

```
bronze          high      001100
lead            low       101110
lead            high      110111
```

show class-of-service scheduler-map

Syntax

```
show class-of-service scheduler-map
<name>
```

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 15.1R3 on MX Series routers for enhanced subscriber management.

Description

Display the mapping of schedulers to forwarding classes and a summary of scheduler parameters for each entry.

Options

none—Display all scheduler maps.

name—(Optional) Display a summary of scheduler parameters for each forwarding class to which the named scheduler is assigned.

Required Privilege Level

view

RELATED DOCUMENTATION

Verifying and Managing Junos OS Enhanced Subscriber Management

List of Sample Output

show class-of-service scheduler-map on page 965
show class-of-service scheduler-map (QFX Series) on page 966

Output Fields

Table 151 on page 964 describes the output fields for the **show class-of-service scheduler-map** command. Output fields are listed in the approximate order in which they appear.

Table 151: show class-of-service scheduler-map Output Fields

Field Name	Field Description
Scheduler map	Name of the scheduler map. (Enhanced subscriber management for MX Series routers) The name of the dynamic scheduler map object is associated with a generated UID (for example, **SMAP-1_UID1002**) instead of with a subscriber interface.
Index	Index of the indicated object. Objects having indexes in this output include scheduler maps, schedulers, and drop profiles. (Enhanced subscriber management for MX Series routers) Index values for dynamic CoS traffic control profiles are larger for enhanced subscriber management than they are for legacy subscriber management.
Scheduler	Name of the scheduler.
Forwarding class	Classification of a packet affecting the forwarding, scheduling, and marking policies applied as the packet transits the router.
Transmit rate	Configured transmit rate of the scheduler (in bps). The rate is a percentage of the total interface bandwidth, or the keyword **remainder**, which indicates that the scheduler receives the remaining bandwidth of the interface.
Rate Limit	Rate limiting configuration of the queue. Possible values are **none**, meaning no rate limiting, and **exact**, meaning the queue only transmits at the configured rate.
Maximum buffer delay	Amount of transmit delay (in milliseconds) or the buffer size of the queue. The buffer size is shown as a percentage of the total interface buffer allocation, or by the keyword **remainder** to indicate that the buffer is sized according to what remains after other scheduler buffer allocations.
Priority	Scheduling priority: **low** or **high**.
Excess priority	Priority of excess bandwidth: **low**, **medium-low**, **medium-high**, **high**, or **none**.
Explicit Congestion Notification	(QFX Series, OCX Series, and EX4600 switches only) Explicit congestion notification (ECN) state: • Disable—ECN is disabled on the specified scheduler • Enable—ECN is enabled on the specified scheduler ECN is disabled by default.
Adjust minimum	Minimum shaping rate for an adjusted queue, in bps.

Table 151: show class-of-service scheduler-map Output Fields *(continued)*

Field Name	Field Description
Adjust percent	Bandwidth adjustment applied to a queue, in percent.
Drop profiles	Table displaying the assignment of drop profiles by name and index to a given loss priority and protocol pair.
Loss priority	Packet loss priority for drop profile assignment.
Protocol	Transport protocol for drop profile assignment.
Name	Name of the drop profile.

Sample Output

show class-of-service scheduler-map

`user@host>` **show class-of-service scheduler-map**

```
Scheduler map: dd-scheduler-map, Index: 84

  Scheduler: aa-scheduler, Index: 8721, Forwarding class: aa-forwarding-class
   Transmit rate: 30 percent, Rate Limit: none, Maximum buffer delay: 39 ms,
   Priority: high
   Drop profiles:
    Loss priority   Protocol    Index    Name
     Low            non-TCP     8724     aa-drop-profile
     Low            TCP         9874     bb-drop-profile
     High           non-TCP     8833     cc-drop-profile
     High           TCP         8484     dd-drop-profile

  Scheduler: bb-scheduler, Forwarding class: aa-forwarding-class
   Transmit rate: 40 percent, Rate limit: none, Maximum buffer delay: 68 ms,
   Priority: high
   Drop profiles:
    Loss priority   Protocol    Index    Name
     Low            non-TCP     8724     aa-drop-profile
     Low            TCP         9874     bb-drop-profile
     High           non-TCP     8833     cc-drop-profile
     High           TCP         8484     dd-drop-profile
```

show class-of-service scheduler-map (QFX Series)

```
user@switch# show class-of-service scheduler-map
```

```
Scheduler map: be-map, Index: 12240

  Scheduler:be-sched, Forwarding class: best-effort, Index: 115
    Transmit rate: 30 percent, Rate Limit: none, Buffer size: remainder,
    Buffer Limit: none, Priority: low
    Excess Priority: unspecified, Explicit Congestion Notification: disable

    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any          3312    lan-dp
      Medium-high     any          2714    be-dp1
      High            any          3178    be-dp2
```

show class-of-service shared-buffer

Syntax

```
show class-of-service shared-buffer
<egress | ingress>
```

Release Information

Command introduced in Junos OS Release 12.3 for the QFX Series.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description

Display the shared buffer allocation and partitioning configuration.

> **NOTE:** Due to QFX5200 cross-point architecture, all buffer usage counters are maintained separately. When usage counters are displayed with the command show class-of-service shared-buffer on QFX5200, various pipe counters are displayed separately.

Options

none—Display ingress and egress shared buffer settings.

egress—(Optional) Display the egress shared buffer settings.

ingress—(Optional) Display the ingress shared buffer settings.

Required Privilege Level

view

RELATED DOCUMENTATION

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Best-Effort Unicast Traffic | 659

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Multicast Traffic | 675

Example: Recommended Configuration of the Shared Buffer Pool for Networks with Mostly Lossless Traffic | 683

Configuring Global Ingress and Egress Shared Buffers | 657

Understanding CoS Buffer Configuration | 635

List of Sample Output

show class-of-service shared-buffer on page 969

Output Fields

Table 152 on page 968 describes the output fields for the **show class-of-service shared-buffer** command. Output fields are listed in the approximate order in which they appear.

Table 152: show class-of-service shared-buffer Output Fields

Field Name	Field Description
Ingress	Ingress shared buffer configuration.
Total Buffer	Total buffer space available to the ports in KB. This is the combined dedicated buffer pool and shared buffer pool.
Dedicated Buffer	Buffer space allocated to the dedicated buffer pool in KB.
Shared Buffer	Buffer space allocated to the shared buffer pool in KB.
Lossless	Buffer space allocated to the lossless traffic buffer pool in KB.
Lossless Headroom	Buffer space allocated to the lossless headroom traffic buffer pool to support priority-based flow control (PFC) and Ethernet PAUSE in KB. (Ingress ports only.) **NOTE:** OCX Series switches do not support PFC.
Lossy	Buffer space allocated to the lossy (best-effort) traffic buffer pool in KB.
Lossless Headroom Utilization	Utilization of the ingress lossless headroom buffer pool. (These fields can help you to determine how much headroom buffer space you need to reserve to support PFC and Ethernet PAUSE for lossless flows.) **NOTE:** OCX Series switches do not support PFC.
Node Device	Index number that identifies the switch. On a QFX3500 switch, this field always has a value of zero (0).
Total	Size of the lossless headroom ingress buffer pool in KB.
Used	Amount in KB of lossless headroom ingress buffer used.
Free	Amount in KB of lossless headroom ingress buffer free (unused).
Egress	Egress shared buffer configuration.

Table 152: show class-of-service shared-buffer Output Fields *(continued)*

Field Name	Field Description
Multicast	Buffer space allocated to the multicast traffic buffer pool in KB. (Egress ports only.)

Sample Output

show class-of-service shared-buffer

user@switch> **show class-of-service shared-buffer**

```
Ingress:
  Total Buffer     :  9360.00 KB
  Dedicated Buffer :  2158.00 KB
  Shared Buffer    :  7202.00 KB
    Lossless          :  648.18 KB
    Lossless Headroom :  3240.90 KB
    Lossy             :  3312.92 KB

  Lossless Headroom Utilization:
  Node Device         Total          Used                 Free
  0                   3240.90 KB     0.00 KB              3240.90 KB

Egress:
  Total Buffer     :  9360.00 KB
  Dedicated Buffer :  2704.00 KB
  Shared Buffer    :  6656.00 KB
    Lossless          :  3328.00 KB
    Multicast         :  1264.64 KB
    Lossy             :  2063.36 KB
```

show class-of-service traffic-control-profile

Syntax

```
show class-of-service traffic-control-profile
<profile-name>
```

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 12.2 for ACX Series Routers.
Command introduced in Junos OS Release 15.1R3 on MX Series routers for enhanced subscriber management.

Description

For Gigabit Ethernet IQ PICs, Channelized IQ PICs, EQ DPCs, and MPC/MIC interfaces only, display traffic shaping and scheduling profiles.

(ACX Series routers) For ATM IMA pseudowire interfaces, display traffic shaping and scheduling profiles.

Options

none—Display all profiles.

profile-name—(Optional) Display information about a single profile.

Required Privilege Level

view

RELATED DOCUMENTATION

Verifying and Managing Junos OS Enhanced Subscriber Management

List of Sample Output

Output Fields

Table 153 on page 971 describes the output fields for the **show class-of-service traffic-control-profile** command. Output fields are listed in the approximate order in which they appear.

Table 153: show class-of-service traffic-control-profile Output Fields

Field Name	Field Description
Traffic control profile	Name of the traffic control profile. (Enhanced subscriber management for MX Series routers) The name of the dynamic traffic control profile object is associated with a generated UID (for example, **TC_PROF_100_199_SERIES_UID1000**) instead of with a subscriber interface.
Index	Index number of the traffic control profile. (Enhanced subscriber management for MX Series routers) Index values for dynamic CoS traffic control profiles are larger for enhanced subscriber management than they are for legacy subscriber management.
ATM Service	(MX Series routers with ATM Multi-Rate CE MIC) Configured category of ATM service. Possible values: • cbr—Constant bit rate. • rtvbr—Real time variable bit rate. • nrtvbr—Non real time variable bit rate. • ubr—Unspecified bit rate.
Maximum Burst Size	Configured maximum burst size, in cells.
Peak rate	Configured peak rate, in cps.
Sustained rate	Configured sustained rate, in cps.
Shaping rate	Configured shaping rate, in bps. **NOTE:** (MX Series routers with ATM Multi-Rate CE MIC) Configured peak rate, in cps.
Shaping rate burst	Configured burst size for the shaping rate, in bytes. **NOTE:** (MX Series routers with ATM Multi-Rate CE MIC) Configured maximum burst rate, in cells.
Shaping rate priority high	Configured shaping rate for high-priority traffic, in bps.
Shaping rate priority medium	Configured shaping rate for medium-priority traffic, in bps.
Shaping rate priority low	Configured shaping rate for low-priority traffic, in bps.

Table 153: show class-of-service traffic-control-profile Output Fields *(continued)*

Field Name	Field Description
Shaping rate excess high	Configured shaping rate for high-priority excess traffic, in bps.
Shaping rate excess low	Configured shaping rate for low-priority excess traffic, in bps.
Scheduler map	Name of the associated scheduler map. (Enhanced subscriber management for MX Series routers) The name of the dynamic scheduler map object is associated with a generated UID (for example, **SMAP-1_UID1002**) instead of with a subscriber interface.
Delay Buffer rate	Configured delay buffer rate, in bps.
Excess rate	Configured excess rate, in percent or proportion.
Excess rate high	Configured excess rate for high priority traffic, in percent or proportion.
Excess rate low	Configured excess rate for low priority traffic, in percent or proportion.
Guaranteed rate	Configured guaranteed rate, in bps or cps. NOTE: (MX Series routers with ATM Multi-Rate CE MIC) This value depends on the ATM service category chosen. Possible values: • **cbr**—Guaranteed rate is equal to the configured peak rate in cps. • **rtvbr**—Guaranteed rate is equal to the configured sustained rate in cps. • **nrtvbr**—Guaranteed rate is equal to the configured sustained rate in cps.
Guaranteed rate burst	Configured burst size for the guaranteed rate, in bytes.
adjust-minimum	Configured minimum shaping rate for an adjusted queue, in bps.
overhead accounting mode	Configured shaping mode: **Frame Mode** or **Cell Mode**.
Overhead bytes	Configured byte adjustment value.
Adjust parent	Configured shaping-rate adjustment for parent scheduler nodes. If enabled, this field appears. **flow-aware** indicates that the parent scheduler node is adjusted only once per multicast channel.

Sample Output

show class-of-service traffic-control-profile

user@host> **show class-of-service traffic-control-profile**

```
Traffic control profile: Profile1, Index: 57625
  Scheduler map: m1
  Delay Buffer rate: 500000
  Guaranteed rate: 1000000

Traffic control profile: Profile2, Index: 57624
  Scheduler map: m2
  Delay Buffer rate: 600000
  Guaranteed rate: 2000000

Traffic control profile: Profile3, Index: 57627
  Scheduler map: m3
  Delay Buffer rate: 800000
  Guaranteed rate: 3000000
 .Excess rate high: proportion 4

Traffic control profile: Profile4, Index: 57626
  Scheduler map: m4
  Delay Buffer rate: 750000
  Guaranteed rate: 4000000
..adjust-minimum 20000000

 Traffic control profile: foo, Index: 57626
   Shaping rate: 100000000
   Scheduler map: <default>
   Overhead accounting mode: Frame Mode
   Frame mode overhead accounting bytes: -12
   Adjust parent: flow-aware
```

show class-of-service traffic-control-profile (MX Series routers with Clear Channel Multi-Rate CE MIC)

user@host> **show class-of-service traffic-control-profile**

```
Traffic control profile: at-vbr1, Index: 11395
  ATM Service: RTVBR
  Scheduler map: m3
  overhead accounting mode: Frame Mode
  Shaping rate: 1000 cps
```

```
  Shaping rate burst: 500 cells
  Delay Buffer rate: 2000 cps
  Guaranteed rate: 1000 cps

Traffic control profile: foo, Index: 38286
  ATM Service: UBR
  Scheduler map: m3
  overhead accounting mode: Frame Mode
```

show class-of-service traffic-control-profile (ACX Series routers with ATM IMA pseudowire interfaces)

`user@host>` **show class-of-service traffic-control-profile**

```
Traffic control profile: foo, Index: 38286
  ATM Service: RTVBR
  Shaping rate: 2000 cps
  Shaping rate burst: 200 cells
  Scheduler map: <default>
  Delay Buffer rate: 1000 cps
  Guaranteed rate: 1700 cps
```

show class-of-service traffic-control-profile (Enhanced Subscriber Management)

`user@host>` **show class-of-service traffic-control-profile**

```
Traffic control profile: TC_PROF_100_199_SERIES_UID1000, Index: 4294967313
  Shaping rate: 11000000
  Shaping rate burst: 1 bytes
  Scheduler map: SMAP-1_UID1002
  Delay Buffer rate: 5000000
  Overhead accounting mode: Cell Mode
  Frame mode overhead accounting bytes: -4
  Cell mode overhead accounting bytes: 20
```

show dcbx

Syntax

```
show dcbx
```

Release Information

Command introduced in Junos OS Release 11.3 for the QFX Series.

Description

List DCBX status (enabled or disabled) and the interfaces on which DCBX is enabled.

Required Privilege Level

view

RELATED DOCUMENTATION

show dcbx neighbors | 977
Configuring DCBX Autonegotiation | 465

Output Fields

Table 154 on page 975 lists the output fields for the **show dcbx** command. Output fields are listed in the approximate order in which they appear.

Table 154: show dcbx output fields

Field Name	Field Description
DCBX	Status of DCBX on the switch or for the specified interface: • Enabled—DCBX is enabled on the switch or on the specified interface • Disabled—DCBX is disabled on the switch or on the specified interface
Interface	Name of the interface

Sample Output

show dcbx

`user@switch>` **show dcbx**

```
DCBX                   : Enabled
Interface      DCBX
xe-0/0/9.0     enabled
xe-0/0/32.0    enabled
xe-0/0/36.0    enabled
```

show dcbx neighbors

Syntax

```
show dcbx neighbors
<interface interface-name>
<terse>
```

Release Information

Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 11.3 for EX Series switches.

Description

Display information about Data Center Bridging Capability Exchange protocol (DCBX) neighbor interfaces.

Options

none—Display information about all DCBX neighbor interfaces.

interface-name—(Optional) Display information for the specified interface.

terse—Display the specified level of output.

Required Privilege Level

view

RELATED DOCUMENTATION

Configuring DCBX Autonegotiation | 465
Example: Configuring DCBX Application Protocol TLV Exchange | 476
Example: Configuring an FCoE Transit Switch
Example: Configuring DCBX to Support an iSCSI Application
Understanding DCB Features and Requirements | 450
Understanding Data Center Bridging Capability Exchange Protocol for EX Series Switches
dcbx | 740

List of Sample Output

show dcbx neighbors interface (QFX Series, DCBX Version 1.01 Mode) on page 994
show dcbx neighbors interface (QFX Series, IEEE DCBX Mode) on page 997
show dcbx neighbors terse (QFX Series) on page 1000
show dcbx neighbors (EX4500 Switch: FCoE Interfaces on Both Local and Peer with PFC Configured Compatibly) on page 1000

show dcbx neighbors (EX4500 Switch: DCBX Interfaces on Local and Peer Are Configured Compatibly with iSCSI Application) on page 1002
show dcbx neighbors (EX4500 Switch: Includes ETS) on page 1003

Output Fields

Table 155 on page 978 lists the output fields for the **show dcbx neighbors** command. Output fields are listed in the approximate order in which they appear.

Table 155: show dcbx neighbors Output Fields

Field Name	Field Description
Interface	Name of the interface.
Parent Interface	Name of the link aggregation group (LAG) interface to which the DCBX interface belongs.
Active-application-map	Name of the application map applied to the interface.
Protocol-Mode	(QFX Series) DCBX protocol mode the interface uses: • IEEE DCBX Version—The interface uses IEEE DCBX mode. • DCBX Version 1.01—The interface uses DCBX version 1.01. **NOTE:** On interfaces that use the IEEE DCBX mode, the **show dcbx neighbors interface *interface-name*** operational command does not include application, PFC, or ETS operational state in the output.
Protocol-State	(DCBX Version 1.01 only) DCBX protocol state synchronization status: • **in-sync**—The local interface received an acknowledge message from the peer to indicate that the peer received a state change message sent by the local interface. • **ack-pending**—The local interface has not yet received an acknowledge message from the peer to indicate that the peer received a state change message sent by the local interface.

Table 155: show dcbx neighbors Output Fields *(continued)*

Field Name			Field Description
Local-Advertisement			(DCBX Version 1.01 only) Status of advertisements that the local interface sends to the peer.
	Operational version		Version of the DCBX standard used.
	sequence-number		Number of state change messages sent to the peer. If the interface **Protocol-State** value is **in-sync**, this number should match the **acknowledge-id** number in the **Peer-Advertisement** section. If the interface **Protocol-State** value is **ack-pending**, this number does not match the **acknowledge-id** number in the **Peer-Advertisement** section.
	acknowledge-id		Number of acknowledge messages received from the peer. If the **Protocol-State** value is **in-sync**, this number should match the **sequence-number** value in the **Peer-Advertisement** section. If the **Protocol-State** value is **ack-pending**, this number does not match the **sequence-number** value in the **Peer-Advertisement** section.

Table 155: show dcbx neighbors Output Fields *(continued)*

Field Name		Field Description
Peer-Advertisement		(DCBX Version 1.01 only) Status of advertisements that the peer sends to the local interface.
	Operational version	Version of the DCBX standard used.
	sequence-number	Number of state change messages the peer sent to the local interface. If this number matches the **acknowledge-id** number in the **Local-Advertisement** field, this indicates that the local interface has acknowledged all of the peer's state change messages and is synchronized. If this number does not match the **acknowledge-id** number in the **Local-Advertisement** field, this indicates that the peer has not yet received an acknowledgment for a state change message from the local interface.
	acknowledge-id	Number of acknowledge messages the peer has received from the local interface. If this number matches the **sequence-number** value in the **Local-Advertisement** field, this indicates that the peer has acknowledged all of the local interface's state change messages and is in synchronization. If this number does not match the **sequence-number** value in the **Local-Advertisement** field, this indicates that the peer has not yet sent an acknowledgment for a state change message from the local interface.

Table 155: show dcbx neighbors Output Fields *(continued)*

Field Name			Field Description
Feature: PFC			Priority-based flow control (PFC) feature DCBX state information.
	Protocol-State		(DCBX Version 1.01 only) DCBX protocol state synchronization status: • **ack-pending**—The local interface has not yet received an acknowledge message from the peer to indicate that the peer received a PFC state change message sent by the local interface. • **in-sync**—The local interface received an acknowledge message from the peer to indicate that the peer received a PFC state change message sent by the local interface. • **not-applicable**—PFC autonegotiation is disabled.
	Operational State		(DCBX Version 1.01 only) Operational state of the feature: **enabled** or **disabled**.
	Local-Advertisement		Status of advertisements that the local interface sends to the peer.
		Enable	(DCBX Version 1.01 only) State that the local interface advertises to the peer: • **Yes**—The feature is enabled. • **No**—The feature is disabled.
		Willing	Willingness of the local interface to learn the PFC configuration from the peer using DCBX: • **Yes**—The local interface is willing to learn the PFC configuration from the peer. • **No**—The local interface is not willing to learn the PFC configuration from the peer.

Table 155: show dcbx neighbors Output Fields *(continued)*

Field Name			Field Description
		Mac auth Bypass Capability	(IEEE DCBX only) (QFX Series) Media access controller (MAC) authentication bypass provides access to devices based on MAC address authentication. This is not supported, so the only value seen in the local advertisement field is **no**.
		Error	(DCBX Version 1.01 only) Configuration compatibility error status: • **No**—No error detected. Local and peer configuration are compatible. • **Yes**—Error detected. Local and peer configuration are not compatible.
		Operational State	PFC operational state on the interface: • Enabled—PFC is enabled on the interface • Disabled—PFC is disabled on the interface
		Maximum Traffic Classes capable to support PFC	Largest number of traffic classes the local interface supports for PFC: • **6** (EX Series switches) • **6** (QFX Series)
		Code Point	PFC code point, which is specified in the 3-bit class-of-service field in the VLAN header.
		Admin Mode	PFC administrative state for each code point on the local interface: • **Enabled**—PFC is enabled for the code point. • **Disabled**—PFC is disabled for the code point.
		Operational Mode	(QFX Series) PFC operational mode for each code point: • Enable—PFC is enabled on the code point. • Disable—PFC is disabled on the code point.
	Peer-Advertisement		Status of advertisements that the peer sends to the local interface.

Table 155: show dcbx neighbors Output Fields *(continued)*

Field Name			Field Description
		Enable	(DCBX Version 1.01 only) State that the peer advertises to the local interface: • **Yes**—The feature is enabled. • **No**—The feature is disabled.
		Willing	Willingness of the peer to learn the PFC configuration from the local interface using DCBX: • **Yes**—The peer is willing to learn the PFC configuration from the local interface. • **No**—The peer is not willing to learn the PFC configuration from the local interface.
		Error	(DCBX Version 1.01 only) Configuration compatibility error status: • **No**—No error detected. Local and peer configuration are compatible. • **Yes**—Error detected. Local and peer configuration are not compatible.
		Operational State	PFC operational state on the interface: • Enabled—PFC is enabled on the interface • Disabled—PFC is disabled on the interface
		Mac auth Bypass Capability	(IEEE DCBX only) (QFX Series) Media access controller (MAC) authentication bypass provides access to devices based on MAC address authentication. Although the QFX Series does not support this feature, the connected peer might support it. This field reports the peer state: • **Yes**—The connected peer supports MAC authentication bypass. • **No**—The connected peer does not support MAC authentication bypass.

Table 155: show dcbx neighbors Output Fields *(continued)*

Field Name			Field Description
		Maximum Traffic Classes capable to support PFC	Largest number of traffic classes the peer supports for PFC: • **6** (EX Series switches) • **8** (QFX Series)
		Code Point	PFC code point, which is specified in the 3-bit class-of-service field in the VLAN header.
		Admin Mode	PFC administrative state for each code point on the peer: • **Enabled**—PFC is enabled for the code point. • **Disabled**—PFC is disabled for the code point.

Table 155: show dcbx neighbors Output Fields *(continued)*

Field Name			Field Description
Feature: Application			State information for the DCBX application.
	Protocol-State		(DCBX Version 1.01 only) DCBX protocol state synchronization status: • **in-sync**—The local interface received an acknowledge message from the peer to indicate that the peer received an FCoE state change message sent by the local interface. • **ack-pending**—The local interface has not yet received an acknowledge message from the peer to indicate that the peer received an FCoE state change message sent by the local interface. • **not-applicable**—The local interface is set to **no-auto-negotiation** (autonegotiation is disabled). If the interface is associated with an FCoE forwarding class, the interface advertises FCoE capability even if the connected peer does not advertise FCoE capability.
	Local-Advertisement		Status of advertisements that the local interface sends to the peer. If the local interface is set to **no-auto-negotiation** (autonegotiation is disabled), the local advertisement portion of the output is not shown.
		Enable	(DCBX Version 1.01 only) State that the local interface advertises to the peer: • **Yes**—The feature is enabled. • **No**—The feature is disabled.
		Willing	

Table 155: show dcbx neighbors Output Fields ***(continued)***

Field Name			Field Description
			(DCBX Version 1.01 only) Willingness of the local interface to learn the FCoE interface state from the peer using DCBX: • **Yes**—The local interface is willing to learn the FCoE interface state from the peer. • **No**—The local interface is not willing to learn the FCoE interface state from the peer.
		Error	(DCBX Version 1.01 only) Configuration compatibility error status: • **No**—No error detected. The local and peer configuration are compatible. • **Yes**—Error detected. The local and peer configuration are not compatible.
		Appl-Name	Name of the application:
		Ethernet-Type	(DCBX Version 1.01 only) Ethernet type (EtherType) of the application. For example, **0x8906** indicates the EtherType for the FCoE application. Either the EtherType (for Layer 2 applications) or the Socket Number (for Layer 4 applications) of the application is displayed in the output.
		Socket-Number	Destination port socket number of the application, if applicable. Either the EtherType (for Layer 2 applications) or the Socket Number (for Layer 4 applications) of the application is displayed in the output.
		Priority-Field or **Priority-Map**	Priority assigned to the application. For EX Series switches, the priority of the FCoE application is determined by the PFC congestion notification profile that has been configured and associated with the FCoE interface. For other applications, the priority is based on the application map.

Table 155: show dcbx neighbors Output Fields *(continued)*

Field Name			Field Description
		Status	(DCBX Version 1.01 only) Local status when autonegotiation is enabled: • **Enabled**—The application feature is enabled on both the local interface and the peer interface. (The local configuration and the peer configuration match.) • **Disabled**—The local configuration and the peer configuration do not match. NOTE: If there is a configuration mismatch in one application between the switch and the peer, all the other applications including FCoE are disabled.
	Peer-Advertisement		Status of advertisements that the peer sends to the local interface.
		Enable	(DCBX Version 1.01 only) State that the peer advertises to the local interface: • **Yes**—The feature is enabled. • **No**—The feature is disabled.
		Willing	(DCBX Version 1.01 only) Willingness of the peer to learn the FCoE interface state from the local interface using DCBX: • **Yes**—The peer is willing to learn the FCoE interface state from the local interface. • **No**—The peer is not willing to learn the FCoE interface state from the local interface.
		Error	(DCBX Version 1.01 only) Configuration compatibility error status: • **No**—No error detected. Local and peer configuration are compatible. • **Yes**—Error detected. Local and peer configuration are not compatible.

Table 155: show dcbx neighbors Output Fields *(continued)*

Field Name			Field Description
		Appl-Name	Name of the application: • **FCoE**—Fibre Channel over Ethernet
		Ethernet-Type	Ethernet type (EtherType) of the application. For example, **0x8906** indicates the EtherType for the FCoE application. Either the EtherType (for Layer 2 applications) or the Socket-Number (for Layer 4 applications) of the application is displayed in the output.
		Socket-Number	Destination port socket number of the application, if applicable. Either the EtherType (for Layer 2 applications) or the Socket Number (for Layer 4 applications) of the application is displayed in the output.
		Priority-Field or **Priority-Map**	Priority assigned to the application.
		Status	(DCBX Version 1.01 only) Peer interface status: • **Enabled**—The application feature is enabled on both the local interface and the peer interface. (The local configuration and the peer configuration match.) • **Disabled**—The local configuration and the peer configuration do not match.

Table 155: show dcbx neighbors Output Fields *(continued)*

Field Name			Field Description
Feature: ETS			Enhanced Transmission Selection (ETS) DCBX state information.
	Protocol-State		(DCBX Version 1.01 only) ETS protocol state synchronization status: • **in-sync**—The local interface received an acknowledge message from the peer to indicate that the peer received an ETS state change message sent by the local interface. • **ack-pending**—The local interface has not yet received an acknowledge message from the peer to indicate that the peer received an ETS state change message sent by the local interface.
	Operational State		(DCBX Version 1.01 only) Operational state of the feature, **enabled** or **disabled**.
	Local-Advertisement		Status of advertisements that the local interface sends to the peer.
		Enable	(DCBX Version 1.01 only) State that the local interface advertises to the peer: • **Yes**—The feature is enabled. • **No**—The feature is disabled.
		TLV Type	

Table 155: show dcbx neighbors Output Fields *(continued)*

Field Name			Field Description
			(IEEE DCBX only) Type of ETS TLV: • Configuration—Advertises the Configuration TLV, which communicates the local ETS configuration to the peer but does not ask the peer to use the configuration. • Recommendation—Advertises the Recommendation TLV, which communicates the local ETS configuration to the peer, and if the peer is "willing," configures the peer interface to match the local ETS configuration. • Recommendation-or-Configuration—Advertises both TLVs.
		Willing	Willingness of the local interface to learn the ETS state from the peer using DCBX (EX Series switches always advertise **No** for this field): • **Yes**—Local interface is willing to learn the ETS state from the peer. • **No**—Local interface is not willing to learn the ETS state from the peer.
		Credit Based Shaper	(IEEE DCBX only) Alternative method of flow control to buffer-to-buffer credit. The QFX Series does not support a credit-based shaper, so the value of this field is always **No**.
		Error	(DCBX Version 1.01 only) Configuration error status: • **No**—No error. This should always be the switch ETS error state. • **Yes**—Error detected.
		Maximum Traffic Classes capable to support PFC	(DCBX Version 1.01 only) Largest number of traffic classes the local interface supports for PFC.

Table 155: show dcbx neighbors Output Fields *(continued)*

Field Name			Field Description
		Maximum Traffic Classes supported	(IEEE DCBX only) Largest number of traffic classes the local interface supports for ETS. (EX Series switches support only one traffic class for ETS. However, a different value might be shown for this field.)
		Code Point	PFC code point, which is specified in the 3-bit class-of-service field in the VLAN header.
		Priority-Group	Class-of-service (CoS) priority group (forwarding class set) identification number.
		Percentage B/W	Configured minimum percentage of link bandwidth allocated to the priority group. Only explicitly configured values appear in this output column. If the link bandwidth is the default percentage, it is not shown. (EX Series switches allocate 100% of link bandwidth to the default priority group, group 7.)
		Transmission Selection Algorithm	(IEEE DCBX only) The transmission selection algorithm used by the interface. The QFX Series supports ETS but does not support using the credit-based shaper algorithm, so the only value shown in this field is **ETS**.
	Peer-Advertisement		Status of advertisements that the peer sends to the local interface.
		Enable	(DCBX Version 1.01 only) State that the peer advertises to the local interface: • **Yes**—The feature is enabled. • **No**—The feature is disabled.
		TLV Type	

Table 155: show dcbx neighbors Output Fields *(continued)*

Field Name			Field Description
			(IEEE DCBX only) Type of ETS TLV: • Configuration—Advertises the Configuration TLV, which communicates the local ETS configuration to the peer but does not ask the peer to use the configuration. • Recommendation—Advertises the Recommendation TLV, which communicates the local ETS configuration to the peer, and if the peer is “willing,” configures the peer interface to match the local ETS configuration. • Configuration/Recommendation—Advertises both TLVs.
		Willing	Willingness of the peer to learn the ETS state from the local interface using DCBX: • **Yes**—Peer is willing to learn the ETS state from the local interface. • **No**—Peer is not willing to learn the ETS state from the local interface.
		Credit Based Shaper	(IEEE DCBX only) Alternative method of flow control to buffer-to-buffer credit. The QFX Series does not support a credit-based shaper, so the value of this field is always **No**.
		Error	(DCBX Version 1.01 only) Configuration error status of the peer: • **No**—No error in peer ETS TLV. • **Yes**—Error in peer ETS TLV.
		Maximum Traffic Classes capable to support PFC	(DCBX Version 1.01 only) Largest number of traffic classes the local interface supports for PFC.

Table 155: show dcbx neighbors Output Fields *(continued)*

Field Name			Field Description
		Maximum Traffic Classes supported	(IEEE DCBX only) Largest number of traffic classes the local interface supports for ETS. (EX Series switches support only one traffic class for ETS. However, a different value might be shown for this field.)
		Code Point	PFC code point, which is specified in the 3-bit class-of-service field in the VLAN header.
		Priority-Group	CoS priority group (forwarding class set) identification number.
		Percentage B/W	Configured minimum percentage of link bandwidth allocated to the priority group. (EX Series switches allocate 100% of link bandwidth to the default priority group, group 7.)
		Transmission Selection Algorithm	(IEEE DCBX only) Transmission selection algorithm used by the interface. The QFX Series supports ETS but does not support using the credit-based shaper algorithm, so the only value shown in this field is **ETS**.
PFC			(QFX Series, **terse** option only) DCBX TLV advertisement state for PFC: • Disabled—PFC configuration matches the configuration on the connected peer and PFC is disabled • Enabled—PFC configuration matches the configuration on the connected peer and PFC is enabled • Not Advt—Interface does not advertise PFC to the connected peer

Table 155: show dcbx neighbors Output Fields *(continued)*

Field Name	Field Description
ETS	(**terse** option only) Local DCBX TLV advertisement state for ETS: • Advt—Interface advertises ETS TLVs • Disabled—ETS is disabled on the interface (interface does not advertise ETS)
ETS Rec	(**terse** option only) DCBX TLV peer advertisement state for ETS (state received from the connected DCBX peer): • Advt—Peer interface advertises ETS TLVs • Not Advt—Peer interface does not advertise ETS **NOTE:** When the DCBX mode is DCBX version 1.01, no peer information is displayed.
Version	(**terse** option only) The DCBX version used on the interface and whether the DCBX version was autonegotiated or explicitly configured: • **IEEE**—The interface uses IEEE DCBX. • **1.01**—The interface uses DCBX version 1.01. When the DCBX version used is the result of autonegotiation, the term **(Auto)** appears next to the version. For example, **IEEE (Auto)** indicates that the interface autonegotiated with the connected peer to use IEEE DCBX. Autonegotiation is enabled by default.

Sample Output

show dcbx neighbors interface (QFX Series, DCBX Version 1.01 Mode)

`user@switch>` **show dcbx neighbors interface xe-0/0/0**

```
Interface : xe-0/0/0.0 - Parent Interface: ae0.0
    Active-application-map: app-map-1
    Protocol-State: in-sync
```

```
Protocol-Mode: DCBX Version 1.01

Local-Advertisement:
    Operational version: 1
    sequence-number: 130, acknowledge-id: 102

Peer-Advertisement:
    Operational version: 1
    sequence-number: 102, acknowledge-id: 130

Feature: PFC, Protocol-State: in-sync

  Operational State: Enabled

  Local-Advertisement:
    Enable: Yes, Willing: No, Error: No
    Maximum Traffic Classes capable to support PFC: 8

    Code Point          Admin Mode   Operational Mode
      000                Disabled    Disable
      001                Disabled    Disable
      010                Disabled    Disable
      011                Enabled     Enable
      100                Enabled     Enable
      101                Disabled    Disable
      110                Disabled    Disable
      111                Disabled    Disable

  Peer-Advertisement:
    Enable: Yes, Willing: No, Error: No
    Maximum Traffic Classes capable to support PFC: 8

    Code Point          Admin Mode
      000                Disabled
      001                Disabled
      010                Disabled
      011                Enabled
      100                Enabled
      101                Disabled
      110                Disabled
      111                Disabled

Feature: Application, Protocol-State: in-sync
```

```
    Local-Advertisement:
      Enable: Yes, Willing: No, Error: No

      Appl-Name    Ethernet-Type    Socket-Number    Priority-Map    Status
        FCoE         0x8906                            00001110        Enabled
        iSCSI                           3260           10000000        Enabled

    Peer-Advertisement:
      Enable: Yes, Willing: Yes, Error: No

      Appl-Name    Ethernet-Type    Socket-Number    Priority-Map    Status
        FCoE         0x8906             N/A            00001110        Enabled

  Feature: ETS, Protocol-State: in-sync

    Operational State: Enabled

    Local-Advertisement:
      Enable: Yes, Willing: No, Error: No
      Maximum Traffic Classes capable to support PFC: 8

      Code Point            Priority-Group
        000                   0
        001                   7
        010                   7
        011                   7
        100                   0
        101                   1
        110                   1
        111                   7

      Priority-Group        Percentage B/W
        0                     40%
        1                     5%

    Peer-Advertisement:
      Enable: Yes, Willing: No, Error: No
      Maximum Traffic Classes capable to support PFC: 8

      Code Point            Priority-Group
        000                   0
        001                   7
        010                   7
        011                   7
```

```
    100                 0
    101                 1
    110                 1
    111                 7

  Priority-Group       Percentage B/W
    0                   40%
    1                   5%
```

show dcbx neighbors interface (QFX Series, IEEE DCBX Mode)

`user@switch>` **show dcbx neighbors interface xe-0/0/0**

```
Interface : xe-0/0/0.0 - Parent Interface: ae0.0
    Active-application-map: app-map-1
    Protocol-Mode: IEEE-DCBX Version

    Feature: PFC

      Local-Advertisement:
        Willing: No
        Mac auth Bypass Capability: No
        Operational State: Enabled

        Maximum Traffic Classes capable to support PFC: 8

        Code Point           Admin Mode
          000                 Disabled
          001                 Disabled
          010                 Disabled
          011                 Enabled
          100                 Enabled
          101                 Disabled
          110                 Disabled
          111                 Disabled

      Peer-Advertisement:
        Willing: No
        Mac auth Bypass Capability: No
        Operational State: Enabled

        Maximum Traffic Classes capable to support PFC: 8
```

```
    Code Point          Admin Mode
      000                 Disabled
      001                 Disabled
      010                 Disabled
      011                 Enabled
      100                 Enabled
      101                 Disabled
      110                 Disabled
      111                 Disabled

Feature: Application

  Local-Advertisement:

    Appl-Name    Ethernet-Type    Socket-Number    Priority-field
      FCoE         0x8906                            00001110
      iSCSI                           3260           10000000

  Peer-Advertisement:
    Appl-Name    Ethernet-Type    Socket-Number    Priority-field
      FCoE         0x8906             N/A            00001110

Feature: ETS

  Local-Advertisement:
    TLV Type: Configuration/Recommendation
    Willing: No
    Credit Based Shaper: No
    Maximum Traffic Classes supported: 3

    Code Point           Priority-Group
      000                  0
      001                  7
      010                  7
      011                  7
      100                  0
      101                  1
      110                  1
      111                  7

    Priority-Group       Percentage B/W
      0                    40%
      1                    5%
```

```
    Priority-Group   Transmission Selection Algorithm
      0              Enhanced Transmission Selection
      1              Enhanced Transmission Selection

  Peer-Advertisement:
    TLV Type: Configuration
    Willing: No
    Credit Based Shaper: No

    Code Point           Priority-Group
      000                  0
      001                  7
      010                  7
      011                  7
      100                  0
      101                  1
      110                  1
      111                  7

    Priority-Group       Percentage B/W
      0                    40%
      1                    5%

    Priority-Group   Transmission Selection Algorithm
      0              Enhanced Transmission Selection
      1              Enhanced Transmission Selection

  Peer-Advertisement:

    TLV Type: Recommendation

    Code Point           Priority-Group
      000                  0
      001                  7
      010                  7
      011                  7
      100                  0
      101                  1
      110                  1
      111                  7

    Priority-Group       Percentage B/W
      0                    40%
```

```
         1                    5%

       Priority-Group   Transmission Selection Algorithm
         0              Enhanced Transmission Selection
         1              Enhanced Transmission Selection
```

show dcbx neighbors terse (QFX Series)

`user@switch>` **show dcbx neighbors terse**

```
Interface   Parent      PFC        ETS        ETS        Version
            Interface                                    Rec
xe-0/0/8.0  -           Enabled    Advt       Advt       IEEE (Auto)
xe-0/0/9.0  -           Disabled   Disabled              1.01
xe-0/0/11.0 ae0.0       Enabled    Advt       Advt       IEEE (Auto)
xe-0/0/12.0 ae0.0       Enabled    Advt       Advt       IEEE (Auto)
xe-0/0/32.0 -           Enabled    Advt       Not Advt   IEEE
xe-0/0/36.0 -           Not Advt   Advt       Advt       IEEE
```

show dcbx neighbors (EX4500 Switch: FCoE Interfaces on Both Local and Peer with PFC Configured Compatibly)

`user@switch>` **show dcbx neighbors interface xe-0/0/14**

```
    Interface : xe-0/0/14.0 - Parent Interface: ae0.0
     Protocol-State: in-sync

     Local-Advertisement:
         Operational version: 0
         sequence-number: 6, acknowledge-id: 6

     Peer-Advertisement:
         Operational version: 0
         sequence-number: 6, acknowledge-id: 6

     Feature: PFC, Protocol-State: in-sync

       Operational State: Enabled
```

```
    Local-Advertisement:
      Enable: Yes, Willing: No, Error: No
      Maximum Traffic Classes capable to support PFC: 6

       Code Point            Admin Mode
        000                 Disabled
        001                 Disabled
        010                 Disabled
        011                 Enabled
        100                 Disabled
        101                 Disabled
        110                 Disabled
        111                 Disabled



    Peer-Advertisement:
      Enable: Yes, Willing: No, Error: No
      Maximum Traffic Classes capable to support PFC: 6

      Code Point             Admin Mode
        000                 Disabled
        001                 Disabled
        010                 Disabled
        011                 Enabled
        100                 Disabled
        101                 Disabled
        110                 Disabled
        111                 Disabled



  Feature: Application, Protocol-State: in-sync

    Local-Advertisement:
      Enable: Yes, Willing: No, Error: No   <<< Error bit will not be set as
there is no miss configuration between local and peer.


      Appl-Name    Ethernet-Type    Socket-Number    Priority-Map    Status
      FCoE          0x8906                            00001000        Enabled


    Peer-Advertisement:
```

```
        Enable: Yes, Willing: No, Error: No

                 Appl-Name     Ethernet-Type     Socket-Number     Priority-Map
Status
                 FCoE           0x8906                             00001000
Enabled
```

show dcbx neighbors (EX4500 Switch: DCBX Interfaces on Local and Peer Are Configured Compatibly with iSCSI Application)

`user@switch>` **show dcbx neighbors interface xe-0/0/14**

```
  Interface : xe-0/0/14.0 - Parent Interface: ae0.0

    Protocol-State: in-sync

    Active-application-map: iscsi-map

     Local-Advertisement:
        Operational version: 0
        sequence-number: 9, acknowledge-id: 12


    Peer-Advertisement:
        Operational version: 0
        sequence-number: 12, acknowledge-id: 9


    Feature: PFC, Protocol-State: in-sync
      Operational State: Enabled
       Local-Advertisement:
        Enable: Yes, Willing: No, Error: No
        Maximum Traffic Classes capable to support PFC: 6


     Code Point              Admin Mode
          000                 Disabled
          001                 Disabled
          010                 Disabled
          011                 Enabled
          100                 Disabled
```

```
         101                 Disabled
         110                 Disabled
         111                 Disabled

     Peer-Advertisement:
       Enable: Yes, Willing: No, Error: No
       Maximum Traffic Classes capable to support PFC: 6

    Code Point               Admin Mode
         000                 Disabled
         001                 Disabled
         010                 Disabled
         011                 Enabled
         100                 Disabled
         101                 Disabled
         110                 Disabled
         111                 Disabled

    Feature: Application, Protocol-State: in-sync
     Local-Advertisement:
       Enable: Yes, Willing: No, Error: No

            Appl-Name    Ethernet-Type    Socket-Number   Priority-Map    Status

             FCoE        0x8906                          00001000       Enabled

             iscsi                       3260            00100000       Enabled



     Peer-Advertisement:
       Enable: Yes, Willing: No, Error: No

            Appl-Name    Ethernet-Type    Socket-Number   Priority-Map    Status

             FCoE        0x8906                          00001000       Enabled

             iscsi                       3260            00100000       Enabled
```

show dcbx neighbors (EX4500 Switch: Includes ETS)

`user@switch>` **show dcbx neighbors interface xe-0/0/3**

```
Interface : xe-0/0/3.0
 Protocol-State: in-sync
 Active-application-map: map_iscsi

 Local-Advertisement:
     Operational version: 0
     sequence-number: 1, acknowledge-id: 5

 Peer-Advertisement:
     Operational version: 0
     sequence-number: 5, acknowledge-id: 1

 Feature: PFC, Protocol-State: in-sync

   Operational State: Enabled

   Local-Advertisement:
     Enable: Yes, Willing: No, Error: No
     Maximum Traffic Classes capable to support PFC: 6

     Code Point          Admin Mode
       000                 Enabled
       001                 Enabled
       010                 Disabled
       011                 Disabled
       100                 Disabled
       101                 Disabled
       110                 Disabled
       111                 Disabled

   Peer-Advertisement:
     Enable: Yes, Willing: Yes, Error: No
     Maximum Traffic Classes capable to support PFC: 8

     Code Point          Admin Mode
       000                 Enabled
       001                 Disabled
       010                 Disabled
       011                 Disabled
       100                 Enabled
       101                 Disabled
       110                 Disabled
       111                 Disabled
```

```
Feature: Application, Protocol-State: in-sync

  Local-Advertisement:
    Enable: Yes, Willing: No, Error: No

    Appl-Name    Ethernet-Type    Socket-Number    Priority-Map    Status
      FCoE         0x8906                            00000001        Enabled
      iscsi                         3260             00000010        Enabled

  Peer-Advertisement:
    Enable: Yes, Willing: Yes, Error: No

    Appl-Name    Ethernet-Type    Socket-Number    Priority-Map    Status
      FCoE         0x8906                            00001000        Enabled
      iscsi                         3260             00010000        Enabled

Feature: ETS, Protocol-State: in-sync

  Operational State: Enabled

  Local-Advertisement:
    Enable: Yes, Willing: No, Error: No
    Maximum Traffic Classes supported : 3

    Code Point           Priority-Group
      000                  7
      001                  7
      010                  7
      011                  7
      100                  7
      101                  7
      110                  7
      111                  7

    Priority-Group       Percentage B/W
      7                    100%

  Peer-Advertisement:
    Enable: Yes, Willing: Yes, Error: No
    Maximum Traffic Classes supported : 8

    Code Point           Priority-Group
      000                  0
```

```
      001                   1
      010                   0
      011                   0
      100                   2
      101                   0
      110                   0
      111                   0

    Priority-Group        Percentage B/W
      0                     30%
      1                     40%
      2                     30%
```

show interfaces priority-group

Syntax

```
show interfaces priority-group
<interface-name>
<buffer-occupancy>
```

Release Information

Command introduced before Junos OS Release 7.4.
buffer-occupancy statement introduced in Junos OS Release 19.1R1 for QFX5000 Series switches.

Description

Display class-of-service (CoS) priority group information for physical interfaces.

Options

interface-name—(Optional) Show detailed CoS priority group statistics for the specified interface.

buffer-occupancy—Displays the peak interface ingress priority-group buffer occupancy while **buffer-monitor-enable** is enabled at the **[edit chassis fpc *slot-number* traffic-manager]** hierarchy level.

Additional Information

For related CoS operational mode commands, see the CLI Explorer.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding CoS Priority Group Scheduling | 373
Defining CoS Traffic Control Profiles (Priority Group Scheduling) | 384

List of Sample Output

show interfaces priority-group buffer-occupancy on page 1008

Output Fields

Table 156 on page 1008 lists the output fields for the **show interfaces priority-group** command. Output fields are listed in the approximate order in which they appear.

Table 156: show interfaces queue Output Fields

Field Name	Field Description
Physical interface	Name of the physical interface.
Enabled	State of the interface. Possible values are described in the “Enabled Field” section under *Common Output Fields Description*.
Interface index	Physical interface's index number, which reflects its initialization sequence.
SNMP ifIndex	SNMP index number for the interface.
PG	Priority group number.
PG-Utilization bytes	
Peak	(QFX5000 Series switches only) Diplays the peak buffer occupancy for the priority group while **buffer-monitor-enable** is enabled at the **[edit chassis fpc *slot-number* traffic-manager]** hierarchy level.

Sample Output

show interfaces priority-group buffer-occupancy

`user@host>` **show interfaces priority-group xe-0/0/6 buffer-occupancy**

```
Physical interface: xe-0/0/6, Enabled, Physical link is Down
  Interface index: 650, SNMP ifIndex: 580
            PG: 0
            PG-Utilization bytes  :
            Peak                 : 88192
            PG: 1
            PG-Utilization bytes  :
            Peak                 : 87984
            PG: 2
            PG-Utilization bytes  :
            Peak                 : 87984
            PG: 3
            PG-Utilization bytes  :
            Peak                 : 88192
            PG: 4
            PG-Utilization bytes  :
```

```
Peak               : 88608
PG: 5
PG-Utilization bytes  :
Peak               : 87776
PG: 6
PG-Utilization bytes  :
Peak               : 0
PG: 7
PG-Utilization bytes  :
Peak               : 0
```

show interfaces queue

Syntax

```
show interfaces queue
<aggregate | remaining-traffic>
<both-ingress-egress>
<egress>
<forwarding-class forwarding-class>
<ingress>
< interface-name>
<l2-statistics>
<buffer-occupancy>
```

Release Information

Command introduced before Junos OS Release 7.4.

both-ingress-egress, **egress**, and **ingress** options introduced in Junos OS Release 7.6.

Command introduced in Junos OS Release 9.0 for EX Series.

Command introduced in Junos OS Release 11.1 for the QFX Series.

l2-statistics option introduced in Junos OS Release 12.1.

Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

buffer-occupancy statement introduced in Junos OS Release 19.1R1 for QFX5000 Series switches.

Description

Display class-of-service (CoS) queue information for physical interfaces.

Options

none—Show detailed CoS queue statistics for all physical interfaces.

aggregate—(Optional) Display the aggregated queuing statistics of all logical interfaces that have traffic-control profiles configured. (Not on the QFX Series.)

both-ingress-egress—(Optional) On Gigabit Ethernet Intelligent Queuing 2 (IQ2) PICs, display both ingress and egress queue statistics. (Not on the QFX Series.)

egress—(Optional) Display egress queue statistics.

forwarding-class *forwarding-class*—(Optional) Forwarding class name for this queue. Shows detailed CoS statistics for the queue associated with the specified forwarding class.

ingress—(Optional) On Gigabit Ethernet IQ2 PICs, display ingress queue statistics. (Not on the QFX Series.)

interface-name—(Optional) Show detailed CoS queue statistics for the specified interface.

l2-statistics—(Optional) Display Layer 2 statistics for MLPPP, FRF.15, and FRF.16 bundles

buffer-occupancy—Displays the peak buffer occupancy for each queue while **buffer-monitor-enable** is enabled at the **[edit chassis fpc *slot-number* traffic-manager]** hierarchy level.

remaining-traffic—(Optional) Display the remaining-traffic queue statistics of all logical interfaces that have traffic-control profiles configured.

Overhead for Layer 2 Statistics

Transmitted packets and transmitted byte counts are displayed for the Layer 2 level with the addition of encapsulation overheads applied for fragmentation, as shown in Table 157 on page 1011. Others counters, such as packets and bytes queued (input) and drop counters, are displayed at the Layer 3 level. In the case of link fragmentation and interleaving (LFI) for which fragmentation is not applied, corresponding Layer 2 overheads are added, as shown in Table 157 on page 1011.

Table 157: Layer 2 Overhead and Transmitted Packets or Byte Counts

Protocol	Fragmentation		LFI
	First fragmentation	Second to *n* fragmentations	
	Bytes	Bytes	
MLPPP (Long)	13	12	8
MLPPP (short)	11	10	8
MLFR (FRF15)	12	10	8
MFR (FRF16)	10	8	-
MCMLPPP(Long)	13	12	-
MCMLPPP(Short)	11	10	-

Layer 2 Statistics—Fragmentation Overhead Calculation

```
MLPPP/MC-MLPPP Overhead details:
   ===============================
   Fragment 1:

    Outer PPP header                         : 4 bytes
```

```
 Long or short sequence MLPPP header      : 4 bytes or 2 bytes
 Inner PPP header                         : 1 byte
 HDLC flag and FCS bytes                  : 4 bytes

Fragments 2 .. n :

 Outer PPP header                         : 4 bytes
 Long or short sequence MLPPP header      : 4 bytes or 2 bytes
 HDLC flag and FCS bytes                  : 4 bytes


MLFR (FRF15) Overhead details:
=============================
Fragment 1:
    Framerelay header     :  2 bytes
    Control,NLPID         :  2 bytes
    Fragmentaion header   :  2 bytes
    Inner proto           :  2 bytes
    HDLC flag and  FCS    :  4 bytes

Fragments 2 ...n :

    Framerelay header     :  2 bytes
    Control,NLPID         :  2 bytes
    Fragmentaion header   :  2 bytes
    HDLC flag and  FCS    :  4 bytes


MFR (FRF16) Overhead details:
==============================
Fragment 1:
   Fragmentaion header   :  2 bytes
   Framerelay header     :  2 bytes
   Inner proto           :  2 bytes
   HDLC flag and  FCS    :  4 bytes

Fragments 2 ...n :
   Fragmentaion header   :  2 bytes
   Framerelay header     :  2 bytes
   HDLC flag and  FCS    :  4 bytes
```

Overhead with LFI

```
MLPPP(Long & short sequence):
   ============================
       Outer PPP header      :  4 bytes
       HDLC flag and  FCS    :  4 bytes

  MLFR (FRF15):
  =============
       Framerelay header     :  2 bytes
       Control,NLPID         :  2 bytes
       HDLC flag and  FCS    :  4 bytes
```

The following examples show overhead for different cases:

- A 1000-byte packet is sent to a mlppp bundle without any fragmentation. At the Layer 2 level, bytes transmitted is 1013 in 1 packet. This overhead is for MLPPP long sequence encap.
- A 1000-byte packet is sent to a mlppp bundle with a fragment threshold of 250byte. At the Layer 2 level, bytes transmitted is 1061 bytes in 5 packets.
- A 1000-byte LFI packet is sent to an mlppp bundle. At the Layer 2 level, bytes transmitted is 1008 in 1 packet.

remaining-traffic—(Optional) Display the queuing statistics of all logical interfaces that do not have traffic-control profiles configured. (Not on the QFX Series.)

Additional Information

For rate-limited interfaces hosted on Modular Interface Cards (MICs), Modular Port Concentrators (MPCs), or Enhanced Queuing DPCs, rate-limit packet-drop operations occur *before* packets are queued for transmission scheduling. For such interfaces, the statistics for queued traffic do not include the packets that have already been dropped due to rate limiting, and consequently the displayed statistics for queued traffic are the same as the displayed statistics for transmitted traffic.

> **NOTE:** For rate-limited interfaces hosted on other types of hardware, rate-limit packet-drop operations occur *after* packets are queued for transmission scheduling. For these other interface types, the statistics for queued traffic include the packets that are later dropped due to rate limiting, and consequently the displayed statistics for queued traffic equals the sum of the statistics for transmitted and rate-limited traffic.

On M Series routers (except for the M320 and M120 routers), this command is valid only for a PIC installed on an enhanced Flexible PIC Concentrator (FPC).

Queue statistics for aggregated interfaces are supported on the M Series and T Series routers only. Statistics for an aggregated interface are the summation of the queue statistics of the child links of that aggregated interface. You can view the statistics for a child interface by using the **show interfaces statistics** command for that child interface.

When you configure tricolor marking on a 10-port 1-Gigabit Ethernet PIC, for queues 6 and 7 only, the output does not display the number of queued bytes and packets, or the number of bytes and packets dropped because of RED. If you do not configure tricolor marking on the interface, these statistics are available for all queues.

For the 4-port Channelized OC12 IQE PIC and 1-port Channelized OC48 IQE PIC, the **Packet Forwarding Engine Chassis Queues** field represents traffic bound for a particular physical interface on the PIC. For all other PICs, the **Packet Forwarding Engine Chassis Queues** field represents the total traffic bound for the PIC.

For Gigabit Ethernet IQ2 PICs, the **show interfaces queue** command output does not display the number of tail-dropped packets. This limitation does not apply to Packet Forwarding Engine chassis queues.

When fragmentation occurs on the egress interface, the first set of packet counters shows the postfragmentation values. The second set of packet counters (under the **Packet Forwarding Engine Chassis Queues** field) shows the prefragmentation values.

The behavior of the **egress** queues for the **Routing Engine-Generated Traffic** is not same as the configured queue for MLPPP and MFR configurations.

For related CoS operational mode commands, see the CLI Explorer.

Required Privilege Level

view

Release History Table

Release	Description
18.3R1	Starting with Junos OS 18.3R1, the **Tail-dropped packets** counter is supported on PTX Series Packet Transport Routers.
16.1	Starting with Junos OS Release 16.1, **Last-packet enqueued** output field is introduced.

RELATED DOCUMENTATION

Monitoring Interface Status and Traffic

Monitoring Interfaces That Have CoS Components

Defining CoS Schedulers and Scheduler Maps (CLI Procedure)

Configuring CoS Traffic Classification for Ingress Queuing on Oversubscribed Ports on EX8200 Line Cards (CLI Procedure)

ATM Interfaces User Guide for Routing Devices

Layer 2 Bridging, Address Learning, and Forwarding User Guide

List of Sample Output

Output Fields

Table 158 on page 1015 lists the output fields for the **show interfaces queue** command. Output fields are listed in the approximate order in which they appear.

Table 158: show interfaces queue Output Fields

Field Name	Field Description
Physical interface	Name of the physical interface.
Enabled	State of the interface. Possible values are described in the “Enabled Field” section under *Common Output Fields Description*.
Interface index	Physical interface's index number, which reflects its initialization sequence.
SNMP ifIndex	SNMP index number for the interface.
Forwarding classes supported	Total number of forwarding classes supported on the specified interface.

Table 158: show interfaces queue Output Fields *(continued)*

Field Name	Field Description
Forwarding classes in use	Total number of forwarding classes in use on the specified interface.
Ingress queues supported	On Gigabit Ethernet IQ2 PICs only, total number of ingress queues supported on the specified interface.
Ingress queues in use	On Gigabit Ethernet IQ2 PICs only, total number of ingress queues in use on the specified interface.
Output queues supported	Total number of output queues supported on the specified interface.
Output queues in use	Total number of output queues in use on the specified interface.
Egress queues supported	Total number of egress queues supported on the specified interface.
Egress queues in use	Total number of egress queues in use on the specified interface.
Queue counters (Ingress)	CoS queue number and its associated user-configured forwarding class name. Displayed on IQ2 interfaces. • **Queued packets**—Number of queued packets. NOTE: This field is not supported on QFX5100, QFX5110, QFX5200, and QFX5210 switches due to hardware limitations. • **Transmitted packets**—Number of transmitted packets. • **Dropped packets**—Number of packets dropped by the ASIC's RED mechanism.
Burst size	(Logical interfaces on IQ PICs only) Maximum number of bytes up to which the logical interface can burst. The burst size is based on the shaping rate applied to the interface.
The following output fields are applicable to both interface component and Packet Forwarding component in the **show interfaces queue** command:	
Queue	Queue number.
Forwarding classes	Forwarding class name.

Table 158: show interfaces queue Output Fields *(continued)*

Field Name	Field Description
Queued Packets	Number of packets queued to this queue. **NOTE:** For Gigabit Ethernet IQ2 interfaces, the Queued Packets count is calculated by the Junos OS interpreting one frame buffer as one packet. If the queued packets are very large or very small, the calculation might not be completely accurate for transit traffic. The count is completely accurate for traffic terminated on the router. For rate-limited interfaces hosted on MICs or MPCs only, this statistic does not include traffic dropped due to rate limiting. For more information, see "Additional Information" on page 1013. **NOTE:** This field is not supported on QFX5100, QFX5110, QFX5200, and QFX5210 switches due to hardware limitations. This field is not supported on EX Series switches due to hardware limitations.
Queued Bytes	Number of bytes queued to this queue. The byte counts vary by interface hardware. For more information, see Table 159 on page 1020. For rate-limited interfaces hosted on MICs or MPCs only, this statistic does not include traffic dropped due to rate limiting. For more information, see "Additional Information" on page 1013. **NOTE:** This field is not supported on QFX5100, QFX5110, QFX5200, and QFX5210 switches due to hardware limitations. This field is not supported on EX Series switches due to hardware limitations.
Transmitted Packets	Number of packets transmitted by this queue. When fragmentation occurs on the egress interface, the first set of packet counters shows the postfragmentation values. The second set of packet counters (displayed under the **Packet Forwarding Engine Chassis Queues** field) shows the prefragmentation values. **NOTE:** For Layer 2 statistics, see "Overhead for Layer 2 Statistics" on page 1011
Transmitted Bytes	Number of bytes transmitted by this queue. The byte counts vary by interface hardware. For more information, see Table 159 on page 1020. **NOTE:** On MX Series routers, this number can be inaccurate when you issue the command for a physical interface repeatedly and in quick succession, because the statistics for the child nodes are collected infrequently. Wait ten seconds between successive iterations to avoid this situation. **NOTE:** For Layer 2 statistics, see "Overhead for Layer 2 Statistics" on page 1011

Table 158: show interfaces queue Output Fields *(continued)*

Field Name	Field Description
Tail-dropped packets	Number of packets dropped because of tail drop. NOTE: Starting with Junos OS 18.3R1, the **Tail-dropped packets** counter is supported on PTX Series Packet Transport Routers.
RL-dropped packets	Number of packets dropped due to rate limiting. For rate-limited interfaces hosted on MICs, MPCs, and Enhanced Queuing DPCs only, this statistic is not included in the queued traffic statistics. For more information, see “Additional Information” on page 1013. NOTE: The **RL-dropped packets** counter is not supported on the PTX Series Packet Transport Routers, and is omitted from the output.
RL-dropped bytes	Number of bytes dropped due to rate limiting. For rate-limited interfaces hosted on MICs, MPCs, and Enhanced Queuing DPCs only, this statistic is not included in the queued traffic statistics. For more information, see “Additional Information” on page 1013.

RED-dropped packets	Number of packets dropped because of random early detection (RED).
	
	• (M Series and T Series routers only) On M320 and M120 routers and the T Series routers, the total number of dropped packets is displayed. On all other M Series routers, the output classifies dropped packets into the following categories:
	  ◦ **Low, non-TCP**—Number of low-loss priority non-TCP packets dropped because of RED.
	  ◦ **Low, TCP**—Number of low-loss priority TCP packets dropped because of RED.
	  ◦ **High, non-TCP**—Number of high-loss priority non-TCP packets dropped because of RED.
	  ◦ **High, TCP**—Number of high-loss priority TCP packets dropped because of RED.
	• (MX Series routers with enhanced DPCs, and T Series routers with enhanced FPCs only) The output classifies dropped packets into the following categories:
	  ◦ **Low**—Number of low-loss priority packets dropped because of RED.
	  ◦ **Medium-low**—Number of medium-low loss priority packets dropped because of RED.
	  ◦ **Medium-high**—Number of medium-high loss priority packets dropped because of RED.
	  ◦ **High**—Number of high-loss priority packets dropped because of RED.
	
	NOTE: Due to accounting space limitations on certain Type 3 FPCs (which are supported in M320 and T640 routers), this field does not always display the correct value for queue 6 or queue 7 for interfaces on 10-port 1-Gigabit Ethernet PICs.

Table 158: show interfaces queue Output Fields *(continued)*

Field Name	Field Description

RED-dropped bytes	Number of bytes dropped because of RED. The byte counts vary by interface hardware. For more information, see Table 159 on page 1020.
	• (M Series and T Series routers only) On M320 and M120 routers and the T Series routers, only the total number of dropped bytes is displayed. On all other M Series routers, the output classifies dropped bytes into the following categories:
	  • **Low, non-TCP**—Number of low-loss priority non-TCP bytes dropped because of RED.
	  • **Low, TCP**—Number of low-loss priority TCP bytes dropped because of RED.
	  • **High, non-TCP**—Number of high-loss priority non-TCP bytes dropped because of RED.
	  • **High, TCP**—Number of high-loss priority TCP bytes dropped because of RED.
	NOTE: Due to accounting space limitations on certain Type 3 FPCs (which are supported in M320 and T640 routers), this field does not always display the correct value for queue 6 or queue 7 for interfaces on 10-port 1-Gigabit Ethernet PICs.

Queue-depth bytes	Displays queue-depth average, current, peak, and maximum values for RTP queues. Because queue-depth values cannot be aggregated, displays the values for RTP queues regardless of whether **aggregate**, **remaining-traffic**, or neither option is selected.
Peak	(QFX5000 Series switches only) Diplays the peak buffer occupancy for the queue while **buffer-monitor-enable** is enabled at the **[edit chassis fpc *slot-number* traffic-manager]** hierarchy level.
Last-packet enqueued	Starting with Junos OS Release 16.1, **Last-packet enqueued** output field is introduced. If **packet-timestamp** is enabled for an FPC, shows the day, date, time, and year in the format *day-of-the-week month day-date hh:mm:ss yyyy* when a packet was enqueued in the CoS queue. When the timestamp is aggregated across all active Packet Forwarding Engines, the latest timestamp for each CoS queue is reported.

Byte counts vary by interface hardware. Table 159 on page 1020 shows how the byte counts on the outbound interfaces vary depending on the interface hardware. Table 159 on page 1020 is based on the assumption that outbound interfaces are sending IP traffic with 478 bytes per packet.

Table 159: Byte Count by Interface Hardware

Interface Hardware	Output Level	Byte Count Includes	Comments
Gigabit Ethernet IQ and IQE PICs	Interface	Queued: 490 bytes per packet, representing 478 bytes of Layer 3 packet + 12 bytes Transmitted: 490 bytes per packet, representing 478 bytes of Layer 3 packet + 12 bytes RED dropped: 496 bytes per packet representing 478 bytes of Layer 3 packet + 18 bytes	The 12 additional bytes include 6 bytes for the destination MAC address + 4 bytes for the VLAN + 2 bytes for the Ethernet type. For RED dropped, 6 bytes are added for the source MAC address.
	Packet forwarding component	Queued: 478 bytes per packet, representing 478 bytes of Layer 3 packet Transmitted: 478 bytes per packet, representing 478 bytes of Layer 3 packet	–

Table 159: Byte Count by Interface Hardware *(continued)*

Interface Hardware	Output Level	Byte Count Includes	Comments
Non-IQ PIC	Interface	T Series, TX Series, T1600, and MX Series routers: • Queued: 478 bytes of Layer 3 packet. • Transmitted: 478 bytes of Layer 3 packet. T4000 routers with Type 5 FPCs : • Queued: 478 bytes of Layer 3 packet + the full Layer 2 overhead including 4 bytes CRC + the full Layer 1 overhead 8 bytes preamble + 12 bytes Inter frame Gap. • Transmitted: 478 bytes of Layer 3 packet + the full Layer 2 overhead including 4 bytes CRC + the full Layer 1 overhead 8 bytes preamble + 12 bytes Interframe Gap. M Series routers: • Queued: 478 bytes of Layer 3 packet. • Transmitted: 478 bytes of Layer 3 packet + the full Layer 2 overhead. PTX Series Packet Transport Routers: • Queued: The sum of the transmitted bytes and the RED dropped bytes. • Transmitted: Full Layer 2 overhead (including all L2 encapsulation and CRC) + 12 inter-packet gap + 8 for the preamble. • RED dropped: Full Layer 2 overhead (including all L2 encapsulation and CRC) + 12 inter-packet gap + 8 for the preamble (does not include the VLAN header or MPLS pushed bytes).	The Layer 2 overhead is 14 bytes for non-VLAN traffic and 18 bytes for VLAN traffic.

Table 159: Byte Count by Interface Hardware *(continued)*

Interface Hardware	Output Level	Byte Count Includes	Comments
IQ and IQE PICs with a SONET/SDH interface	Interface	Queued: 482 bytes per packet, representing 478 bytes of Layer 3 packet + 4 bytes Transmitted: 482 bytes per packet, representing 478 bytes of Layer 3 packet + 4 bytes RED dropped: 482 bytes per packet, representing 478 bytes of Layer 3 packet + 4 bytes	The additional 4 bytes are for the Layer 2 Point-to-Point Protocol (PPP) header.
	Packet forwarding component	Queued: 478 bytes per packet, representing 478 bytes of Layer 3 packet Transmitted: 486 bytes per packet, representing 478 bytes of Layer 3 packet + 8 bytes	For transmitted packets, the additional 8 bytes includes 4 bytes for the PPP header and 4 bytes for a cookie.
Non-IQ PIC with a SONET/SDH interface	Interface	T Series, TX Series, T1600, and MX Series routers: • Queued: 478 bytes of Layer 3 packet. • Transmitted: 478 bytes of Layer 3 packet. M Series routers: • Queued: 478 bytes of Layer 3 packet. • Transmitted: 483 bytes per packet, representing 478 bytes of Layer 3 packet + 5 bytes • RED dropped: 478 bytes per packet, representing 478 bytes of Layer 3 packet	For transmitted packets, the additional 5 bytes includes 4 bytes for the PPP header and 1 byte for the packet loss priority (PLP).
Interfaces configured with Frame Relay Encapsulation	Interface	The default Frame Relay overhead is 7 bytes. If you configure the Frame Check Sequence (FCS) to 4 bytes, then the overhead increases to 10 bytes.	

Table 159: Byte Count by Interface Hardware *(continued)*

Interface Hardware	Output Level	Byte Count Includes	Comments
1-port 10-Gigabit Ethernet IQ2 and IQ2–E PICs	Interface	Queued: 478 bytes of Layer 3 packet + the full Layer 2 overhead including CRC. Transmitted: 478 bytes of Layer 3 packet + the full Layer 2 overhead including CRC.	The Layer 2 overhead is 18 bytes for non-VLAN traffic and 22 bytes for VLAN traffic.
4-port 1G IQ2 and IQ2-E PICs 8-port 1G IQ2 and IQ2-E PICs	Packet forwarding component	Queued: 478 bytes of Layer 3 packet. Transmitted: 478 bytes of Layer 3 packet.	–

Sample Output

show interfaces queue (Rate-Limited Interface on a Gigabit Ethernet MIC in an MPC)

The following example shows queue information for the rate-limited interface ge-4/2/0 on a Gigabit Ethernet MIC in an MPC. For rate-limited queues for interfaces hosted on MICs or MPCs, rate-limit packet drops occur prior to packet output queuing. In the command output, the nonzero statistics displayed in the **RL-dropped packets** and **RL-dropped bytes** fields quantify the traffic dropped to rate-limit queue 0 output to 10 percent of 1 gigabyte (100 megabits) per second. Because the RL-dropped traffic is not included in the **Queued** statistics, the statistics displayed for queued traffic are the same as the statistics for transmitted traffic.

user@host> **show interfaces queue ge-4/2/0**

```
Physical interface: ge-4/2/0, Enabled, Physical link is Up
  Interface index: 203, SNMP ifIndex: 1054
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets              :             131300649               141751 pps
    Bytes                :           11287964840             99793248 bps
  Transmitted:
```

```
    Packets              :           131300649              141751 pps
    Bytes                :         11287964840            99793248 bps
    Tail-dropped packets :                   0                   0 pps
    RL-dropped packets   :           205050862              602295 pps
    RL-dropped bytes     :         13595326612           327648832 bps
    RED-dropped packets  :                   0                   0 pps
     Low                 :                   0                   0 pps
     Medium-low          :                   0                   0 pps
     Medium-high         :                   0                   0 pps
     High                :                   0                   0 pps
    RED-dropped bytes    :                   0                   0 bps
     Low                 :                   0                   0 bps
     Medium-low          :                   0                   0 bps
     Medium-high         :                   0                   0 bps
     High                :                   0                   0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              :                   0                   0 pps
    Bytes                :                   0                   0 bps
```

show interfaces queue (Aggregated Ethernet on a T320 Router)

The following example shows that the aggregated Ethernet interface, **ae1**, has traffic on queues **af1** and **af12**:

`user@host>` **show interfaces queue ae1**

```
Physical interface: ae1, Enabled, Physical link is Up
Interface index: 158, SNMP ifIndex: 33 Forwarding classes: 8 supported, 8 in use
Output queues: 8 supported, 8 in use
Queue: 0, Forwarding classes: be
  Queued:
    Packets              :                   5                   0 pps
    Bytes                :                 242                   0 bps
  Transmitted:
    Packets              :                   5                   0 pps
    Bytes                :                 242                   0 bps
    Tail-dropped packets :                   0                   0 pps
    RED-dropped packets  :                   0                   0 pps
    RED-dropped bytes    :                   0                   0 bps
Queue: 1, Forwarding classes: af1
  Queued:
    Packets              :            42603765              595484 pps
```

```
    Bytes                :          5453281920            609776496 bps
  Transmitted:
    Packets              :            42603765               595484 pps
    Bytes                :          5453281920            609776496 bps
    Tail-dropped packets :                   0                    0 pps
    RED-dropped packets  :                   0                    0 pps
    RED-dropped bytes    :                   0                    0 bps
Queue: 2, Forwarding classes: ef1
  Queued:
    Packets              :                   0                    0 pps
    Bytes                :                   0                    0 bps
  Transmitted:
    Packets              :                   0                    0 pps
    Bytes                :                   0                    0 bps
    Tail-dropped packets :                   0                    0 pps
    RED-dropped packets  :                   0                    0 pps
    RED-dropped bytes    :                   0                    0 bps
Queue: 3, Forwarding classes: nc
  Queued:
    Packets              :                  45                    0 pps
    Bytes                :                3930                    0 bps
  Transmitted:
    Packets              :                  45                    0 pps
    Bytes                :                3930                    0 bps
    Tail-dropped packets :                   0                    0 pps
    RED-dropped packets  :                   0                    0 pps
    RED-dropped bytes    :                   0                    0 bps
Queue: 4, Forwarding classes: af11
  Queued:
    Packets              :                   0                    0 pps
    Bytes                :                   0                    0 bps
  Transmitted:
    Packets              :                   0                    0 pps
    Bytes                :                   0                    0 bps
    Tail-dropped packets :                   0                    0 pps
    RED-dropped packets  :                   0                    0 pps
    RED-dropped bytes    :                   0                    0 bps
Queue: 5, Forwarding classes: ef11
  Queued:
    Packets              :                   0                    0 pps
    Bytes                :                   0                    0 bps
  Transmitted:
    Packets              :                   0                    0 pps
    Bytes                :                   0                    0 bps
```

```
    Tail-dropped packets :                    0                      0 pps
    RED-dropped packets  :                    0                      0 pps
    RED-dropped bytes    :                    0                      0 bps
Queue: 6, Forwarding classes: af12
  Queued:
    Packets              :             31296413                 437436 pps
    Bytes                :           4005940864              447935200 bps
  Transmitted:
    Packets              :             31296413                 437436 pps
    Bytes                :           4005940864              447935200 bps
    Tail-dropped packets :                    0                      0 pps
    RED-dropped packets  :                    0                      0 pps
    RED-dropped bytes    :                    0                      0 bps
Queue: 7, Forwarding classes: nc2
  Queued:
    Packets              :                    0                      0 pps
    Bytes                :                    0                      0 bps
  Transmitted:
    Packets              :                    0                      0 pps
    Bytes                :                    0                      0 bps
    Tail-dropped packets :                    0                      0 pps
    RED-dropped packets  :                    0                      0 pps
    RED-dropped bytes    :                    0                      0 bps
```

show interfaces queue (Gigabit Ethernet on a T640 Router)

`user@host>` **show interfaces queue**

```
Physical interface: ge-7/0/1, Enabled, Physical link is Up
   Interface index: 150, SNMP ifIndex: 42
 Forwarding classes: 8 supported, 8 in use
 Output queues: 8 supported, 8 in use
 Queue: 0, Forwarding classes: be
   Queued:
     Packets              :                   13                     0 pps
     Bytes                :                  622                     0 bps
   Transmitted:
     Packets              :                   13                     0 pps
     Bytes                :                  622                     0 bps
     Tail-dropped packets :                    0                     0 pps
     RED-dropped packets  :                    0                     0 pps
     RED-dropped bytes    :                    0                     0 bps
 Queue: 1, Forwarding classes: af1
```

```
  Queued:
    Packets              :          1725947945               372178 pps
    Bytes                :        220921336960            381110432 bps
  Transmitted:
    Packets              :          1725947945               372178 pps
    Bytes                :        220921336960            381110432 bps
    Tail-dropped packets :                   0                    0 pps
    RED-dropped packets  :                   0                    0 pps
    RED-dropped bytes    :                   0                    0 bps
Queue: 2, Forwarding classes: ef1
  Queued:
    Packets              :                   0                    0 pps
    Bytes                :                   0                    0 bps
  Transmitted:
    Packets              :                   0                    0 pps
    Bytes                :                   0                    0 bps
    Tail-dropped packets :                   0                    0 pps
    RED-dropped packets  :                   0                    0 pps
    RED-dropped bytes    :                   0                    0 bps
Queue: 3, Forwarding classes: nc
  Queued:
    Packets              :                 571                    0 pps
    Bytes                :               49318                  336 bps
  Transmitted:
    Packets              :                 571                    0 pps
    Bytes                :               49318                  336 bps
    Tail-dropped packets :                   0                    0 pps
    RED-dropped packets  :                   0                    0 pps
    RED-dropped bytes    :                   0                    0 bps
```

show interfaces queue aggregate (Gigabit Ethernet Enhanced DPC)

`user@host>` **show interfaces queue ge-2/2/9 aggregate**

```
Physical interface: ge-2/2/9, Enabled, Physical link is Up
  Interface index: 238, SNMP ifIndex: 71
Forwarding classes: 16 supported, 4 in use
Ingress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets              :           148450735               947295 pps
    Bytes                :          8016344944            409228848 bps
  Transmitted:
```

```
    Packets              :              76397439                487512 pps
    Bytes                :            4125461868             210602376 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :              72053285                459783 pps
     Low                 :              72053285                459783 pps
     Medium-low          :                     0                     0 pps
     Medium-high         :                     0                     0 pps
     High                :                     0                     0 pps
    RED-dropped bytes    :            3890877444             198626472 bps
     Low                 :            3890877444             198626472 bps
     Medium-low          :                     0                     0 bps
     Medium-high         :                     0                     0 bps
     High                :                     0                     0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
  Transmitted:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                     0                     0 pps
     Low                 :                     0                     0 pps
     Medium-low          :                     0                     0 pps
     Medium-high         :                     0                     0 pps
     High                :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
     Low                 :                     0                     0 bps
     Medium-low          :                     0                     0 bps
     Medium-high         :                     0                     0 bps
     High                :                     0                     0 bps
Queue: 2, Forwarding classes: assured-forwarding
  Queued:
    Packets              :             410278257                473940 pps
    Bytes                :           22156199518             204742296 bps
  Transmitted:
    Packets              :               4850003                  4033 pps
    Bytes                :             261900162               1742256 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :             405425693                469907 pps
     Low                 :             405425693                469907 pps
     Medium-low          :                     0                     0 pps
     Medium-high         :                     0                     0 pps
     High                :                     0                     0 pps
```

```
    RED-dropped bytes     :           21892988124             203000040 bps
     Low                  :           21892988124             203000040 bps
     Medium-low           :                     0                     0 bps
     Medium-high          :                     0                     0 bps
     High                 :                     0                     0 bps
Queue: 3, Forwarding classes: network-control
  Queued:
    Packets               :                     0                     0 pps
    Bytes                 :                     0                     0 bps
  Transmitted:
    Packets               :                     0                     0 pps
    Bytes                 :                     0                     0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets   :                     0                     0 pps
     Low                  :                     0                     0 pps
     Medium-low           :                     0                     0 pps
     Medium-high          :                     0                     0 pps
     High                 :                     0                     0 pps
    RED-dropped bytes     :                     0                     0 bps
     Low                  :                     0                     0 bps
     Medium-low           :                     0                     0 bps
     Medium-high          :                     0                     0 bps
     High                 :                     0                     0 bps
Forwarding classes: 16 supported, 4 in use
Egress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets               :              76605230                485376 pps
    Bytes                 :            5209211400             264044560 bps
  Transmitted:
    Packets               :              76444631                484336 pps
    Bytes                 :            5198235612             263478800 bps
    Tail-dropped packets : Not Available
    RED-dropped packets   :                160475                  1040 pps
     Low                  :                160475                  1040 pps
     Medium-low           :                     0                     0 pps
     Medium-high          :                     0                     0 pps
     High                 :                     0                     0 pps
    RED-dropped bytes     :              10912300                565760 bps
     Low                  :              10912300                565760 bps
     Medium-low           :                     0                     0 bps
     Medium-high          :                     0                     0 bps
     High                 :                     0                     0 bps
Queue: 1, Forwarding classes: expedited-forwarding
```

```
  Queued:
    Packets              :                    0                    0 pps
    Bytes                :                    0                    0 bps
  Transmitted:
    Packets              :                    0                    0 pps
    Bytes                :                    0                    0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                    0 pps
     Low                 :                    0                    0 pps
     Medium-low          :                    0                    0 pps
     Medium-high         :                    0                    0 pps
     High                :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
     Low                 :                    0                    0 bps
     Medium-low          :                    0                    0 bps
     Medium-high         :                    0                    0 bps
     High                :                    0                    0 bps
Queue: 2, Forwarding classes: assured-forwarding
  Queued:
    Packets              :              4836136                 3912 pps
    Bytes                :            333402032              2139056 bps
  Transmitted:
    Packets              :              3600866                 1459 pps
    Bytes                :            244858888               793696 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :              1225034                 2450 pps
     Low                 :              1225034                 2450 pps
     Medium-low          :                    0                    0 pps
     Medium-high         :                    0                    0 pps
     High                :                    0                    0 pps
    RED-dropped bytes    :             83302312              1333072 bps
     Low                 :             83302312              1333072 bps
     Medium-low          :                    0                    0 bps
     Medium-high         :                    0                    0 bps
     High                :                    0                    0 bps
Queue: 3, Forwarding classes: network-control
  Queued:
    Packets              :                    0                    0 pps
    Bytes                :                    0                    0 bps
  Transmitted:
    Packets              :                    0                    0 pps
    Bytes                :                    0                    0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                    0 pps
```

```
     Low                 :                 0                     0 pps
     Medium-low          :                 0                     0 pps
     Medium-high         :                 0                     0 pps
     High                :                 0                     0 pps
    RED-dropped bytes    :                 0                     0 bps
     Low                 :                 0                     0 bps
     Medium-low          :                 0                     0 bps
     Medium-high         :                 0                     0 bps
     High                :                 0                     0 bps

Packet Forwarding Engine Chassis Queues:
Queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets              :          77059796                486384 pps
    Bytes                :        3544750624             178989576 bps
  Transmitted:
    Packets              :          77059797                486381 pps
    Bytes                :        3544750670             178988248 bps
    Tail-dropped packets :                 0                     0 pps
    RED-dropped packets  :                 0                     0 pps
     Low                 :                 0                     0 pps
     Medium-low          :                 0                     0 pps
     Medium-high         :                 0                     0 pps
     High                :                 0                     0 pps
    RED-dropped bytes    :                 0                     0 bps
     Low                 :                 0                     0 bps
     Medium-low          :                 0                     0 bps
     Medium-high         :                 0                     0 bps
     High                :                 0                     0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              :                 0                     0 pps
    Bytes                :                 0                     0 bps
  Transmitted:
    Packets              :                 0                     0 pps
    Bytes                :                 0                     0 bps
    Tail-dropped packets :                 0                     0 pps
    RED-dropped packets  :                 0                     0 pps
     Low                 :                 0                     0 pps
     Medium-low          :                 0                     0 pps
     Medium-high         :                 0                     0 pps
     High                :                 0                     0 pps
    RED-dropped bytes    :                 0                     0 bps
```

```
      Low                 :                    0                   0 bps
      Medium-low          :                    0                   0 bps
      Medium-high         :                    0                   0 bps
      High                :                    0                   0 bps
 Queue: 2, Forwarding classes: assured-forwarding
   Queued:
     Packets              :              4846580                3934 pps
     Bytes                :            222942680             1447768 bps
   Transmitted:
     Packets              :              4846580                3934 pps
     Bytes                :            222942680             1447768 bps
     Tail-dropped packets :                    0                   0 pps
     RED-dropped packets  :                    0                   0 pps
      Low                 :                    0                   0 pps
      Medium-low          :                    0                   0 pps
      Medium-high         :                    0                   0 pps
      High                :                    0                   0 pps
     RED-dropped bytes    :                    0                   0 bps
      Low                 :                    0                   0 bps
      Medium-low          :                    0                   0 bps
      Medium-high         :                    0                   0 bps
      High                :                    0                   0 bps
 Queue: 3, Forwarding classes: network-control
   Queued:
     Packets              :                    0                   0 pps
     Bytes                :                    0                   0 bps
   Transmitted:
     Packets              :                    0                   0 pps
     Bytes                :                    0                   0 bps
     Tail-dropped packets :                    0                   0 pps
     RED-dropped packets  :                    0                   0 pps
      Low                 :                    0                   0 pps
      Medium-low          :                    0                   0 pps
      Medium-high         :                    0                   0 pps
      High                :                    0                   0 pps
     RED-dropped bytes    :                    0                   0 bps
      Low                 :                    0                   0 bps
      Medium-low          :                    0                   0 bps
      Medium-high         :                    0                   0 bps
      High                :                    0                   0 bps
```

show interfaces queue (Gigabit Ethernet IQ2 PIC)

`user@host>` **show interfaces queue ge-7/1/3**

```
Physical interface: ge-7/1/3, Enabled, Physical link is Up
  Interface index: 170, SNMP ifIndex: 70 Forwarding classes: 16 supported, 4 in
use Ingress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets              :             418390039                    10 pps
    Bytes                :           38910269752                  7440 bps
  Transmitted:
    Packets              :             418390039                    10 pps
    Bytes                :           38910269752                  7440 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
  Transmitted:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
Queue: 2, Forwarding classes: assured-forwarding
  Queued:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
  Transmitted:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
Queue: 3, Forwarding classes: network-control
  Queued:
    Packets              :                  7055                     1 pps
    Bytes                :                451552                   512 bps
  Transmitted:
    Packets              :                  7055                     1 pps
    Bytes                :                451552                   512 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
Forwarding classes: 16 supported, 4 in use Egress queues: 4 supported, 4 in use
```

```
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets              :                  1031                     0 pps
    Bytes                :                143292                     0 bps
  Transmitted:
    Packets              :                  1031                     0 pps
    Bytes                :                143292                     0 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                     0                     0 pps
    RL-dropped bytes     :                     0                     0 bps
    RED-dropped packets  :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
  Transmitted:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                     0                     0 pps
    RL-dropped bytes     :                     0                     0 bps
    RED-dropped packets  :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
Queue: 2, Forwarding classes: assured-forwarding
  Queued:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
  Transmitted:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                     0                     0 pps
    RL-dropped bytes     :                     0                     0 bps
    RED-dropped packets  :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
Queue: 3, Forwarding classes: network-control
  Queued:
    Packets              :                 77009                    11 pps
    Bytes                :               6894286                  7888 bps
  Transmitted:
    Packets              :                 77009                    11 pps
    Bytes                :               6894286                  7888 bps
    Tail-dropped packets : Not Available
```

```
    RL-dropped packets   :                    0                     0 pps
    RL-dropped bytes     :                    0                     0 bps
    RED-dropped packets  :                    0                     0 pps
    RED-dropped bytes    :                    0                     0 bps

Packet Forwarding Engine Chassis Queues:
Queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets              :                 1031                     0 pps
    Bytes                :               147328                     0 bps
  Transmitted:
    Packets              :                 1031                     0 pps
    Bytes                :               147328                     0 bps
    Tail-dropped packets :                    0                     0 pps
    RED-dropped packets  :                    0                     0 pps
     Low, non-TCP        :                    0                     0 pps
     Low, TCP            :                    0                     0 pps
     High, non-TCP       :                    0                     0 pps
     High, TCP           :                    0                     0 pps
    RED-dropped bytes    :                    0                     0 bps
     Low, non-TCP        :                    0                     0 bps
     Low, TCP            :                    0                     0 bps
     High, non-TCP       :                    0                     0 bps
     High, TCP           :                    0                     0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              :                    0                     0 pps
    Bytes                :                    0                     0 bps
  Transmitted:
    Packets              :                    0                     0 pps
    Bytes                :                    0                     0 bps
    Tail-dropped packets :                    0                     0 pps
    RED-dropped packets  :                    0                     0 pps
     Low, non-TCP        :                    0                     0 pps
     Low, TCP            :                    0                     0 pps
     High, non-TCP       :                    0                     0 pps
     High, TCP           :                    0                     0 pps
    RED-dropped bytes    :                    0                     0 bps
     Low, non-TCP        :                    0                     0 bps
     Low, TCP            :                    0                     0 bps
     High, non-TCP       :                    0                     0 bps
     High, TCP           :                    0                     0 bps
Queue: 2, Forwarding classes: assured-forwarding
```

```
  Queued:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
  Transmitted:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
    Tail-dropped packets :                     0                     0 pps
    RED-dropped packets  :                     0                     0 pps
     Low, non-TCP        :                     0                     0 pps
     Low, TCP            :                     0                     0 pps
     High, non-TCP       :                     0                     0 pps
     High, TCP           :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
     Low, non-TCP        :                     0                     0 bps
     Low, TCP            :                     0                     0 bps
     High, non-TCP       :                     0                     0 bps
     High, TCP           :                     0                     0 bps
Queue: 3, Forwarding classes: network-control
  Queued:
    Packets              :                 94386                    12 pps
    Bytes                :              13756799                  9568 bps
  Transmitted:
    Packets              :                 94386                    12 pps
    Bytes                :              13756799                  9568 bps
    Tail-dropped packets :                     0                     0 pps
    RED-dropped packets  :                     0                     0 pps
     Low, non-TCP        :                     0                     0 pps
     Low, TCP            :                     0                     0 pps
     High, non-TCP       :                     0                     0 pps
     High, TCP           :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
     Low, non-TCP        :                     0                     0 bps
     Low, TCP            :                     0                     0 bps
     High, non-TCP       :                     0                     0 bps
     High, TCP           :                     0                     0 bps
```

show interfaces queue both-ingress-egress (Gigabit Ethernet IQ2 PIC)

`user@host>` **show interfaces queue ge-6/2/0 both-ingress-egress**

```
Physical interface: ge-6/2/0, Enabled, Physical link is Up
  Interface index: 175, SNMP ifIndex: 121
Forwarding classes: 8 supported, 4 in use
```

```
Ingress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets              : Not Available
    Bytes                :                    0                    0 bps
  Transmitted:
    Packets              :                  254                    0 pps
    Bytes                :                16274                    0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              : Not Available
    Bytes                :                    0                    0 bps
  Transmitted:
    Packets              :                    0                    0 pps
    Bytes                :                    0                    0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
Queue: 2, Forwarding classes: assured-forwarding
  Queued:
    Packets              : Not Available
    Bytes                :                    0                    0 bps
  Transmitted:
    Packets              :                    0                    0 pps
    Bytes                :                    0                    0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
Queue: 3, Forwarding classes: network-control
  Queued:
    Packets              : Not Available
    Bytes                :                    0                    0 bps
  Transmitted:
    Packets              :                    0                    0 pps
    Bytes                :                    0                    0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
Forwarding classes: 8 supported, 4 in use
Egress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
```

```
  Queued:
    Packets              : Not Available
    Bytes                :                    0                     0 bps
  Transmitted:
    Packets              :                    3                     0 pps
    Bytes                :                  126                     0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                     0 pps
    RED-dropped bytes    :                    0                     0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              : Not Available
    Bytes                :                    0                     0 bps
  Transmitted:
    Packets              :                    0                     0 pps
    Bytes                :                    0                     0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                     0 pps
    RED-dropped bytes    :                    0                     0 bps
Queue: 2, Forwarding classes: assured-forwarding
  Queued:
    Packets              : Not Available
    Bytes                :                    0                     0 bps
  Transmitted:
    Packets              :                    0                     0 pps
    Bytes                :                    0                     0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                     0 pps
    RED-dropped bytes    :                    0                     0 bps
Queue: 3, Forwarding classes: network-control
  Queued:
    Packets              : Not Available
    Bytes                :                    0                     0 bps
  Transmitted:
    Packets              :                    0                     0 pps
    Bytes                :                    0                     0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                     0 pps
    RED-dropped bytes    :                    0                     0 bps
Packet Forwarding Engine Chassis Queues:
Queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets              :             80564692                     0 pps
```

```
    Bytes                 :             3383717100                   0 bps
  Transmitted:
    Packets               :               80564692                   0 pps
    Bytes                 :             3383717100                   0 bps
    Tail-dropped packets :                      0                   0 pps
    RED-dropped packets  :                      0                   0 pps
    RED-dropped bytes    :                      0                   0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets               :               80564685                   0 pps
    Bytes                 :             3383716770                   0 bps
  Transmitted:
    Packets               :               80564685                   0 pps
    Bytes                 :             3383716770                   0 bps
    Tail-dropped packets :                      0                   0 pps
    RED-dropped packets  :                      0                   0 pps
    RED-dropped bytes    :                      0                   0 bps
Queue: 2, Forwarding classes: assured-forwarding
  Queued:
    Packets               :                      0                   0 pps
    Bytes                 :                      0                   0 bps
  Transmitted:
    Packets               :                      0                   0 pps
    Bytes                 :                      0                   0 bps
    Tail-dropped packets :                      0                   0 pps
    RED-dropped packets  :                      0                   0 pps
    RED-dropped bytes    :                      0                   0 bps
Queue: 3, Forwarding classes: network-control
  Queued:
    Packets               :                   9397                   0 pps
    Bytes                 :                3809052                 232 bps
  Transmitted:
    Packets               :                   9397                   0 pps
    Bytes                 :                3809052                 232 bps
    Tail-dropped packets :                      0                   0 pps
    RED-dropped packets  :                      0                   0 pps
    RED-dropped bytes    :                      0                   0 bps
```

show interfaces queue ingress (Gigabit Ethernet IQ2 PIC)

user@host> **show interfaces queue ge-6/2/0 ingress**

```
Physical interface: ge-6/2/0, Enabled, Physical link is Up
  Interface index: 175, SNMP ifIndex: 121
Forwarding classes: 8 supported, 4 in use
Ingress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets              : Not Available
    Bytes                :                    0                      0 bps
  Transmitted:
    Packets              :                  288                      0 pps
    Bytes                :                18450                      0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                      0 pps
    RED-dropped bytes    :                    0                      0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              : Not Available
    Bytes                :                    0                      0 bps
  Transmitted:
    Packets              :                    0                      0 pps
    Bytes                :                    0                      0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                      0 pps
    RED-dropped bytes    :                    0                      0 bps
Queue: 2, Forwarding classes: assured-forwarding
  Queued:
    Packets              : Not Available
    Bytes                :                    0                      0 bps
  Transmitted:
    Packets              :                    0                      0 pps
    Bytes                :                    0                      0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                      0 pps
    RED-dropped bytes    :                    0                      0 bps
Queue: 3, Forwarding classes: network-control
  Queued:
    Packets              : Not Available
    Bytes                :                    0                      0 bps
  Transmitted:
    Packets              :                    0                      0 pps
    Bytes                :                    0                      0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                      0 pps
    RED-dropped bytes    :                    0                      0 bps
```

show interfaces queue egress (Gigabit Ethernet IQ2 PIC)

user@host> **show interfaces queue ge-6/2/0 egress**

```
Physical interface: ge-6/2/0, Enabled, Physical link is Up
  Interface index: 175, SNMP ifIndex: 121
Forwarding classes: 8 supported, 4 in use
Egress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets              : Not Available
    Bytes                :                     0                     0 bps
  Transmitted:
    Packets              :                     3                     0 pps
    Bytes                :                   126                     0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              : Not Available
    Bytes                :                     0                     0 bps
  Transmitted:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
Queue: 2, Forwarding classes: assured-forwarding
  Queued:
    Packets              : Not Available
    Bytes                :                     0                     0 bps
  Transmitted:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
Queue: 3, Forwarding classes: network-control
  Queued:
    Packets              : Not Available
    Bytes                :                     0                     0 bps
  Transmitted:
    Packets              :                     0                     0 pps
```

```
    Bytes                 :                    0                     0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                     0 pps
    RED-dropped bytes    :                    0                     0 bps
Packet Forwarding Engine Chassis Queues:
Queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets               :             80564692                     0 pps
    Bytes                 :           3383717100                     0 bps
  Transmitted:
    Packets               :             80564692                     0 pps
    Bytes                 :           3383717100                     0 bps
    Tail-dropped packets :                    0                     0 pps
    RED-dropped packets  :                    0                     0 pps
    RED-dropped bytes    :                    0                     0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets               :             80564685                     0 pps
    Bytes                 :           3383716770                     0 bps
  Transmitted:
    Packets               :             80564685                     0 pps
    Bytes                 :           3383716770                     0 bps
    Tail-dropped packets :                    0                     0 pps
    RED-dropped packets  :                    0                     0 pps
    RED-dropped bytes    :                    0                     0 bps
Queue: 2, Forwarding classes: assured-forwarding
  Queued:
    Packets               :                    0                     0 pps
    Bytes                 :                    0                     0 bps
  Transmitted:
    Packets               :                    0                     0 pps
    Bytes                 :                    0                     0 bps
    Tail-dropped packets :                    0                     0 pps
    RED-dropped packets  :                    0                     0 pps
    RED-dropped bytes    :                    0                     0 bps
Queue: 3, Forwarding classes: network-control
  Queued:
    Packets               :                 9538                     0 pps
    Bytes                 :              3819840                     0 bps
  Transmitted:
    Packets               :                 9538                     0 pps
    Bytes                 :              3819840                     0 bps
    Tail-dropped packets :                    0                     0 pps
```

```
    RED-dropped packets  :                    0                     0 pps
    RED-dropped bytes    :                    0                     0 bps
```

show interfaces queue remaining-traffic (Gigabit Ethernet Enhanced DPC)

user@host> **show interfaces queue ge-2/2/9 remaining-traffic**

```
Physical interface: ge-2/2/9, Enabled, Physical link is Up
  Interface index: 238, SNMP ifIndex: 71
Forwarding classes: 16 supported, 4 in use
Ingress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets              :            110208969                472875 pps
    Bytes                :           5951284434             204282000 bps
  Transmitted:
    Packets              :            110208969                472875 pps
    Bytes                :           5951284434             204282000 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                     0 pps
     Low                 :                    0                     0 pps
     Medium-low          :                    0                     0 pps
     Medium-high         :                    0                     0 pps
     High                :                    0                     0 pps
    RED-dropped bytes    :                    0                     0 bps
     Low                 :                    0                     0 bps
     Medium-low          :                    0                     0 bps
     Medium-high         :                    0                     0 bps
     High                :                    0                     0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              :                    0                     0 pps
    Bytes                :                    0                     0 bps
  Transmitted:
    Packets              :                    0                     0 pps
    Bytes                :                    0                     0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                     0 pps
     Low                 :                    0                     0 pps
     Medium-low          :                    0                     0 pps
     Medium-high         :                    0                     0 pps
     High                :                    0                     0 pps
    RED-dropped bytes    :                    0                     0 bps
```

```
     Low                 :                     0                     0 bps
     Medium-low          :                     0                     0 bps
     Medium-high         :                     0                     0 bps
     High                :                     0                     0 bps
Queue: 2, Forwarding classes: assured-forwarding
  Queued:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
  Transmitted:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                     0                     0 pps
     Low                 :                     0                     0 pps
     Medium-low          :                     0                     0 pps
     Medium-high         :                     0                     0 pps
     High                :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
     Low                 :                     0                     0 bps
     Medium-low          :                     0                     0 bps
     Medium-high         :                     0                     0 bps
     High                :                     0                     0 bps
Queue: 3, Forwarding classes: network-control
  Queued:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
  Transmitted:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                     0                     0 pps
     Low                 :                     0                     0 pps
     Medium-low          :                     0                     0 pps
     Medium-high         :                     0                     0 pps
     High                :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
     Low                 :                     0                     0 bps
     Medium-low          :                     0                     0 bps
     Medium-high         :                     0                     0 bps
     High                :                     0                     0 bps
Forwarding classes: 16 supported, 4 in use
Egress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
```

```
    Packets              :           109355853            471736 pps
    Bytes                :          7436199152         256627968 bps
  Transmitted:
    Packets              :           109355852            471736 pps
    Bytes                :          7436198640         256627968 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                   0                 0 pps
     Low                 :                   0                 0 pps
     Medium-low          :                   0                 0 pps
     Medium-high         :                   0                 0 pps
     High                :                   0                 0 pps
    RED-dropped bytes    :                   0                 0 bps
     Low                 :                   0                 0 bps
     Medium-low          :                   0                 0 bps
     Medium-high         :                   0                 0 bps
     High                :                   0                 0 bps
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              :                   0                 0 pps
    Bytes                :                   0                 0 bps
  Transmitted:
    Packets              :                   0                 0 pps
    Bytes                :                   0                 0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                   0                 0 pps
     Low                 :                   0                 0 pps
     Medium-low          :                   0                 0 pps
     Medium-high         :                   0                 0 pps
     High                :                   0                 0 pps
    RED-dropped bytes    :                   0                 0 bps
     Low                 :                   0                 0 bps
     Medium-low          :                   0                 0 bps
     Medium-high         :                   0                 0 bps
     High                :                   0                 0 bps
Queue: 2, Forwarding classes: assured-forwarding
  Queued:
    Packets              :                   0                 0 pps
    Bytes                :                   0                 0 bps
  Transmitted:
    Packets              :                   0                 0 pps
    Bytes                :                   0                 0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                   0                 0 pps
     Low                 :                   0                 0 pps
```

```
      Medium-low          :                    0                      0 pps
      Medium-high         :                    0                      0 pps
      High                :                    0                      0 pps
     RED-dropped bytes    :                    0                      0 bps
      Low                 :                    0                      0 bps
      Medium-low          :                    0                      0 bps
      Medium-high         :                    0                      0 bps
      High                :                    0                      0 bps
 Queue: 3, Forwarding classes: network-control
   Queued:
     Packets              :                    0                      0 pps
     Bytes                :                    0                      0 bps
   Transmitted:
     Packets              :                    0                      0 pps
     Bytes                :                    0                      0 bps
     Tail-dropped packets : Not Available
     RED-dropped packets  :                    0                      0 pps
      Low                 :                    0                      0 pps
      Medium-low          :                    0                      0 pps
      Medium-high         :                    0                      0 pps
      High                :                    0                      0 pps
     RED-dropped bytes    :                    0                      0 bps
      Low                 :                    0                      0 bps
      Medium-low          :                    0                      0 bps
      Medium-high         :                    0                      0 bps
      High                :                    0                      0 bps
```

show interfaces queue (Channelized OC12 IQE Type 3 PIC in SONET Mode)

user@host> **show interfaces queue t3-1/1/0:7**

```
Physical interface: t3-1/1/0:7, Enabled, Physical link is Up
  Interface index: 192, SNMP ifIndex: 1948
  Description: full T3 interface connect to 6ce13 t3-3/1/0:7 for FR testing - Lam

Forwarding classes: 16 supported, 9 in use
Egress queues: 8 supported, 8 in use
Queue: 0, Forwarding classes: DEFAULT
  Queued:
    Packets              :               214886                 13449 pps
    Bytes                :              9884756               5164536 bps
  Transmitted:
    Packets              :               214886                 13449 pps
```

```
    Bytes                :              9884756             5164536 bps
    Tail-dropped packets :                    0                   0 pps
    RED-dropped packets  :                    0                   0 pps
     Low                 :                    0                   0 pps
     Medium-low          :                    0                   0 pps
     Medium-high         :                    0                   0 pps
     High                :                    0                   0 pps
    RED-dropped bytes    :                    0                   0 bps
     Low                 :                    0                   0 bps
     Medium-low          :                    0                   0 bps
     Medium-high         :                    0                   0 bps
     High                :                    0                   0 bps
Queue: 1, Forwarding classes: REALTIME
  Queued:
    Packets              :                    0                   0 pps
    Bytes                :                    0                   0 bps
  Transmitted:
    Packets              :                    0                   0 pps
    Bytes                :                    0                   0 bps
    Tail-dropped packets :                    0                   0 pps
    RED-dropped packets  :                    0                   0 pps
     Low                 :                    0                   0 pps
     Medium-low          :                    0                   0 pps
     Medium-high         :                    0                   0 pps
     High                :                    0                   0 pps
    RED-dropped bytes    :                    0                   0 bps
     Low                 :                    0                   0 bps
     Medium-low          :                    0                   0 bps
     Medium-high         :                    0                   0 bps
     High                :                    0                   0 bps
Queue: 2, Forwarding classes: PRIVATE
  Queued:
    Packets              :                    0                   0 pps
    Bytes                :                    0                   0 bps
  Transmitted:
    Packets              :                    0                   0 pps
    Bytes                :                    0                   0 bps
    Tail-dropped packets :                    0                   0 pps
    RED-dropped packets  :                    0                   0 pps
     Low                 :                    0                   0 pps
     Medium-low          :                    0                   0 pps
     Medium-high         :                    0                   0 pps
     High                :                    0                   0 pps
    RED-dropped bytes    :                    0                   0 bps
```

```
      Low                   :                     0                     0 bps
      Medium-low            :                     0                     0 bps
      Medium-high           :                     0                     0 bps
      High                  :                     0                     0 bps
Queue: 3, Forwarding classes: CONTROL
  Queued:
    Packets                 :                    60                     0 pps
    Bytes                   :                  4560                     0 bps
  Transmitted:
    Packets                 :                    60                     0 pps
    Bytes                   :                  4560                     0 bps
    Tail-dropped packets    :                     0                     0 pps
    RED-dropped packets     :                     0                     0 pps
     Low                    :                     0                     0 pps
     Medium-low             :                     0                     0 pps
     Medium-high            :                     0                     0 pps
     High                   :                     0                     0 pps
    RED-dropped bytes       :                     0                     0 bps
     Low                    :                     0                     0 bps
     Medium-low             :                     0                     0 bps
     Medium-high            :                     0                     0 bps
     High                   :                     0                     0 bps
Queue: 4, Forwarding classes: CLASS_B_OUTPUT
  Queued:
    Packets                 :                     0                     0 pps
    Bytes                   :                     0                     0 bps
  Transmitted:
    Packets                 :                     0                     0 pps
    Bytes                   :                     0                     0 bps
    Tail-dropped packets    :                     0                     0 pps
    RED-dropped packets     :                     0                     0 pps
     Low                    :                     0                     0 pps
     Medium-low             :                     0                     0 pps
     Medium-high            :                     0                     0 pps
     High                   :                     0                     0 pps
    RED-dropped bytes       :                     0                     0 bps
     Low                    :                     0                     0 bps
     Medium-low             :                     0                     0 bps
     Medium-high            :                     0                     0 bps
     High                   :                     0                     0 bps
Queue: 5, Forwarding classes: CLASS_C_OUTPUT
  Queued:
    Packets                 :                     0                     0 pps
    Bytes                   :                     0                     0 bps
```

```
  Transmitted:
    Packets              :                    0                   0 pps
    Bytes                :                    0                   0 bps
    Tail-dropped packets :                    0                   0 pps
    RED-dropped packets  :                    0                   0 pps
     Low                 :                    0                   0 pps
     Medium-low          :                    0                   0 pps
     Medium-high         :                    0                   0 pps
     High                :                    0                   0 pps
    RED-dropped bytes    :                    0                   0 bps
     Low                 :                    0                   0 bps
     Medium-low          :                    0                   0 bps
     Medium-high         :                    0                   0 bps
     High                :                    0                   0 bps
Queue: 6, Forwarding classes: CLASS_V_OUTPUT
  Queued:
    Packets              :                    0                   0 pps
    Bytes                :                    0                   0 bps
  Transmitted:
    Packets              :                    0                   0 pps
    Bytes                :                    0                   0 bps
    Tail-dropped packets :                    0                   0 pps
    RED-dropped packets  :                    0                   0 pps
     Low                 :                    0                   0 pps
     Medium-low          :                    0                   0 pps
     Medium-high         :                    0                   0 pps
     High                :                    0                   0 pps
    RED-dropped bytes    :                    0                   0 bps
     Low                 :                    0                   0 bps
     Medium-low          :                    0                   0 bps
     Medium-high         :                    0                   0 bps
     High                :                    0                   0 bps
Queue: 7, Forwarding classes: CLASS_S_OUTPUT, GETS
  Queued:
    Packets              :                    0                   0 pps
    Bytes                :                    0                   0 bps
  Transmitted:
    Packets              :                    0                   0 pps
    Bytes                :                    0                   0 bps
    Tail-dropped packets :                    0                   0 pps
    RED-dropped packets  :                    0                   0 pps
     Low                 :                    0                   0 pps
     Medium-low          :                    0                   0 pps
     Medium-high         :                    0                   0 pps
```

```
     High                :                 0                 0 pps
    RED-dropped bytes    :                 0                 0 bps
     Low                 :                 0                 0 bps
     Medium-low          :                 0                 0 bps
     Medium-high         :                 0                 0 bps
     High                :                 0                 0 bps

Packet Forwarding Engine Chassis Queues:
Queues: 8 supported, 8 in use
Queue: 0, Forwarding classes: DEFAULT
  Queued:
    Packets              :            371365             23620 pps
    Bytes                :          15597330           7936368 bps
  Transmitted:
    Packets              :            371365             23620 pps
    Bytes                :          15597330           7936368 bps
    Tail-dropped packets :                 0                 0 pps
    RED-dropped packets  :                 0                 0 pps
     Low                 :                 0                 0 pps
     Medium-low          :                 0                 0 pps
     Medium-high         :                 0                 0 pps
     High                :                 0                 0 pps
    RED-dropped bytes    :                 0                 0 bps
     Low                 :                 0                 0 bps
     Medium-low          :                 0                 0 bps
     Medium-high         :                 0                 0 bps
     High                :                 0                 0 bps
Queue: 1, Forwarding classes: REALTIME
  Queued:
    Packets              :                 0                 0 pps
    Bytes                :                 0                 0 bps
  Transmitted:
    Packets              :                 0                 0 pps
    Bytes                :                 0                 0 bps
    Tail-dropped packets :                 0                 0 pps
    RED-dropped packets  :                 0                 0 pps
     Low                 :                 0                 0 pps
     Medium-low          :                 0                 0 pps
     Medium-high         :                 0                 0 pps
     High                :                 0                 0 pps
    RED-dropped bytes    :                 0                 0 bps
     Low                 :                 0                 0 bps
     Medium-low          :                 0                 0 bps
     Medium-high         :                 0                 0 bps
```

```
     High                 :                     0                     0 bps
Queue: 2, Forwarding classes: PRIVATE
  Queued:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
  Transmitted:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
    Tail-dropped packets :                     0                     0 pps
    RED-dropped packets  :                     0                     0 pps
     Low                 :                     0                     0 pps
     Medium-low          :                     0                     0 pps
     Medium-high         :                     0                     0 pps
     High                :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
     Low                 :                     0                     0 bps
     Medium-low          :                     0                     0 bps
     Medium-high         :                     0                     0 bps
     High                :                     0                     0 bps
Queue: 3, Forwarding classes: CONTROL
  Queued:
    Packets              :                 32843                     0 pps
    Bytes                :               2641754                    56 bps
  Transmitted:
    Packets              :                 32843                     0 pps
    Bytes                :               2641754                    56 bps
    Tail-dropped packets :                     0                     0 pps
    RED-dropped packets  :                     0                     0 pps
     Low                 :                     0                     0 pps
     Medium-low          :                     0                     0 pps
     Medium-high         :                     0                     0 pps
     High                :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
     Low                 :                     0                     0 bps
     Medium-low          :                     0                     0 bps
     Medium-high         :                     0                     0 bps
     High                :                     0                     0 bps
Queue: 4, Forwarding classes: CLASS_B_OUTPUT
  Queued:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
  Transmitted:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
```

```
    Tail-dropped packets :                     0                     0 pps
    RED-dropped packets  :                     0                     0 pps
     Low                 :                     0                     0 pps
     Medium-low          :                     0                     0 pps
     Medium-high         :                     0                     0 pps
     High                :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
     Low                 :                     0                     0 bps
     Medium-low          :                     0                     0 bps
     Medium-high         :                     0                     0 bps
     High                :                     0                     0 bps
Queue: 5, Forwarding classes: CLASS_C_OUTPUT
  Queued:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
  Transmitted:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
    Tail-dropped packets :                     0                     0 pps
    RED-dropped packets  :                     0                     0 pps
     Low                 :                     0                     0 pps
     Medium-low          :                     0                     0 pps
     Medium-high         :                     0                     0 pps
     High                :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
     Low                 :                     0                     0 bps
     Medium-low          :                     0                     0 bps
     Medium-high         :                     0                     0 bps
     High                :                     0                     0 bps
Queue: 6, Forwarding classes: CLASS_V_OUTPUT
  Queued:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
  Transmitted:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
    Tail-dropped packets :                     0                     0 pps
    RED-dropped packets  :                     0                     0 pps
     Low                 :                     0                     0 pps
     Medium-low          :                     0                     0 pps
     Medium-high         :                     0                     0 pps
     High                :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
     Low                 :                     0                     0 bps
```

```
     Medium-low          :                    0                    0 bps
     Medium-high         :                    0                    0 bps
     High                :                    0                    0 bps
 Queue: 7, Forwarding classes: CLASS_S_OUTPUT, GETS
   Queued:
    Packets              :                    0                    0 pps
    Bytes                :                    0                    0 bps
   Transmitted:
    Packets              :                    0                    0 pps
    Bytes                :                    0                    0 bps
    Tail-dropped packets :                    0                    0 pps
    RED-dropped packets  :                    0                    0 pps
     Low                 :                    0                    0 pps
     Medium-low          :                    0                    0 pps
     Medium-high         :                    0                    0 pps
     High                :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
     Low                 :                    0                    0 bps
     Medium-low          :                    0                    0 bps
     Medium-high         :                    0                    0 bps
     High                :                    0                    0 bps
```

show interfaces queue (QFX Series)

`user@switch>` **show interfaces queue xe-0/0/15**

```
  Physical interface: xe-0/0/15, Enabled, Physical link is Up
    Interface index: 49165, SNMP ifIndex: 539
  Forwarding classes: 12 supported, 8 in use
  Egress queues: 12 supported, 8 in use
  Queue: 0, Forwarding classes: best-effort
    Queued:
      Packets              :                    0                    0 pps
      Bytes                :                    0                    0 bps
    Transmitted:
      Packets              :                    0                    0 pps
      Bytes                :                    0                    0 bps
      Tail-dropped packets : Not Available
      Total-dropped packets:                    0                    0 pps
      Total-dropped bytes  :                    0                    0 bps
  Queue: 3, Forwarding classes: fcoe
    Queued:
      Packets              :                    0                    0 pps
```

```
      Bytes                :                      0                      0 bps
    Transmitted:
      Packets              :                      0                      0 pps
      Bytes                :                      0                      0 bps
      Tail-dropped packets : Not Available
      Total-dropped packets:                      0                      0 pps
      Total-dropped bytes  :                      0                      0 bps
  Queue: 4, Forwarding classes: no-loss
    Queued:
      Packets              :                      0                      0 pps
      Bytes                :                      0                      0 bps
    Transmitted:
      Packets              :                      0                      0 pps
      Bytes                :                      0                      0 bps
      Tail-dropped packets : Not Available
      Total-dropped packets:                      0                      0 pps
      Total-dropped bytes  :                      0                      0 bps
  Queue: 7, Forwarding classes: network-control
    Queued:
      Packets              :                      0                      0 pps
      Bytes                :                      0                      0 bps
    Transmitted:
      Packets              :                      0                      0 pps
      Bytes                :                      0                      0 bps
      Tail-dropped packets : Not Available
      Total-dropped packets:                      0                      0 pps
      Total-dropped bytes  :                      0                      0 bps
  Queue: 8, Forwarding classes: mcast
    Queued:
      Packets              :                      0                      0 pps
      Bytes                :                      0                      0 bps
    Transmitted:
      Packets              :                      0                      0 pps
      Bytes                :                      0                      0 bps
      Tail-dropped packets : Not Available
      Total-dropped packets:                      0                      0 pps
      Total-dropped bytes  :                      0                      0 bps
```

show interfaces queue l2-statistics (lsq interface)

`user@switch>` **show interfaces queue lsq-2/2/0.2 l2-statistics**

```
Logical interface lsq-2/2/0.2 (Index 69) (SNMP ifIndex 1598)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: be
  Queued:
    Packets              :                     1                    0 pps
    Bytes                :                  1001                    0 bps
  Transmitted:
    Packets              :                     5                     0 pps
    Bytes                :                  1062                    0 bps
    Tail-dropped packets :                     0                    0 pps
    RED-dropped packets  :                     0                    0 pps
    RED-dropped bytes    :                     0                    0 bps
Queue: 1, Forwarding classes: ef
  Queued:
    Packets              :                     1                    0 pps
    Bytes                :                  1500                    0 bps
  Transmitted:
    Packets              :                     6                     0 pps
    Bytes                :                  1573                    0 bps
    Tail-dropped packets :                     0                    0 pps
    RED-dropped packets  :                     0                    0 pps
    RED-dropped bytes    :                     0                    0 bps
Queue: 2, Forwarding classes: af
  Queued:
    Packets              :                     1                    0 pps
    Bytes                :                   512                    0 bps
  Transmitted:
    Packets              :                     3                     0 pps
    Bytes                :                   549                    0 bps
    Tail-dropped packets :                     0                    0 pps
    RED-dropped packets  :                     0                    0 pps
    RED-dropped bytes    :                     0                    0 bps
Queue: 3, Forwarding classes: nc
  Queued:
    Packets              :                     0                    0 pps
    Bytes                :                     0                    0 bps
  Transmitted:
    Packets              :                     0                     0 pps
    Bytes                :                     0                    0 bps
    Tail-dropped packets :                     0                    0 pps
    RED-dropped packets  :                     0                    0 pps
```

```
    RED-dropped bytes    :                     0                     0 bps
=========
```

show interfaces queue lsq (lsq-ifd)

user@switch> **show interfaces queue lsq-1/0/0**

```
Logical interface lsq-1/0/0 (Index 348) (SNMP ifIndex 660)
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Burst size: 0
Queue: 0, Forwarding classes: be
  Queued:
    Packets              :                 55576                  1206 pps
    Bytes                :              29622008               5145472 bps
  Transmitted:
    Packets              :                 55576                  1206 pps
    Bytes                :              29622008               5145472 bps
    Tail-dropped packets :                     0                     0 pps
    RL-dropped packets   :                     0                     0 pps
    RL-dropped bytes     :                     0                     0 bps
    RED-dropped packets  :                     0                     0 pps
     Low                 :                     0                     0 pps
     Medium-low          :                     0                     0 pps
     Medium-high         :                     0                     0 pps
     High                :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
     Low                 :                     0                     0 bps
     Medium-low          :                     0                     0 bps
     Medium-high         :                     0                     0 bps
     High                :                     0                     0 bps
Queue: 1, Forwarding classes: ef
  Queued:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
  Transmitted:
    Packets              :                     0                     0 pps
    Bytes                :                     0                     0 bps
    Tail-dropped packets :                     0                     0 pps
    RL-dropped packets   :                     0                     0 pps
    RL-dropped bytes     :                     0                     0 bps
    RED-dropped packets  :                     0                     0 pps
     Low                 :                     0                     0 pps
```

```
      Medium-low          :                  0                    0 pps
      Medium-high         :                  0                    0 pps
      High                :                  0                    0 pps
     RED-dropped bytes    :                  0                    0 bps
      Low                 :                  0                    0 bps
      Medium-low          :                  0                    0 bps
      Medium-high         :                  0                    0 bps
      High                :                  0                    0 bps
 Queue: 2, Forwarding classes: af
   Queued:
     Packets              :                  0                    0 pps
     Bytes                :                  0                    0 bps
   Transmitted:
     Packets              :                  0                    0 pps
     Bytes                :                  0                    0 bps
     Tail-dropped packets :                  0                    0 pps
     RL-dropped packets   :                  0                    0 pps
     RL-dropped bytes     :                  0                    0 bps
     RED-dropped packets  :                  0                    0 pps
      Low                 :                  0                    0 pps
      Medium-low          :                  0                    0 pps
      Medium-high         :                  0                    0 pps
      High                :                  0                    0 pps
     RED-dropped bytes    :                  0                    0 bps
      Low                 :                  0                    0 bps
      Medium-low          :                  0                    0 bps
      Medium-high         :                  0                    0 bps
      High                :                  0                    0 bps
 Queue: 3, Forwarding classes: nc
   Queued:
     Packets              :              22231                  482 pps
     Bytes                :           11849123              2057600 bps
   Transmitted:
     Packets              :              22231                  482 pps
     Bytes                :           11849123              2057600 bps
     Tail-dropped packets :                  0                    0 pps
     RL-dropped packets   :                  0                    0 pps
     RL-dropped bytes     :                  0                    0 bps
     RED-dropped packets  :                  0                    0 pps
      Low                 :                  0                    0 pps
      Medium-low          :                  0                    0 pps
      Medium-high         :                  0                    0 pps
      High                :                  0                    0 pps
     RED-dropped bytes    :                  0                    0 bps
```

```
      Low                 :                 0                 0 bps
      Medium-low          :                 0                 0 bps
      Medium-high         :                 0                 0 bps
      High                :                 0                 0 bps
```

show interfaces queue (Aggregated Ethernet on a MX series Router)

user@host> **show interfaces queue ae0 remaining-traffic**

```

Physical interface: ae0    , Enabled, Physical link is Up
  Interface index: 128, SNMP ifIndex: 543
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    Packets             :                16                 0 pps
    Bytes               :              1896                 0 bps
  Transmitted:
    Packets             :                16                 0 pps
    Bytes               :              1896                 0 bps
    Tail-dropped packets :                0                 0 pps
    RL-dropped packets  :                 0                 0 pps
    RL-dropped bytes    :                 0                 0 bps
    RED-dropped packets  :                0                 0 pps
     Low                :                 0                 0 pps
     Medium-low         :                 0                 0 pps
     Medium-high        :                 0                 0 pps
     High               :                 0                 0 pps
    RED-dropped bytes   :                 0                 0 bps
     Low                :                 0                 0 bps
     Medium-low         :                 0                 0 bps
     Medium-high        :                 0                 0 bps
     High               :                 0                 0 bps
  Queue-depth bytes     :
    Average             :                 0
    Current             :                 0
    Peak                :                 0
    Maximum             :         119013376
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets             :                 0                 0 pps
    Bytes               :                 0                 0 bps
  Transmitted:
```

```
    Packets             :                  0                     0 pps
    Bytes               :                  0                     0 bps
    Tail-dropped packets :                 0                     0 pps
    RL-dropped packets  :                  0                     0 pps
    RL-dropped bytes    :                  0                     0 bps
    RED-dropped packets :                  0                     0 pps
     Low                :                  0                     0 pps
     Medium-low         :                  0                     0 pps
     Medium-high        :                  0                     0 pps
     High               :                  0                     0 pps
    RED-dropped bytes   :                  0                     0 bps
     Low                :                  0                     0 bps
     Medium-low         :                  0                     0 bps
     Medium-high        :                  0                     0 bps
     High               :                  0                     0 bps
  Queue-depth bytes     :
    Average             :                  0
    Current             :                  0
    Peak                :                  0
    Maximum             :              32768
Queue: 2, Forwarding classes: assured-forwarding
  Queued:
    Packets             :                  0                     0 pps
    Bytes               :                  0                     0 bps
  Transmitted:
    Packets             :                  0                     0 pps
    Bytes               :                  0                     0 bps
    Tail-dropped packets :                 0                     0 pps
    RL-dropped packets  :                  0                     0 pps
    RL-dropped bytes    :                  0                     0 bps
    RED-dropped packets :                  0                     0 pps
     Low                :                  0                     0 pps
     Medium-low         :                  0                     0 pps
     Medium-high        :                  0                     0 pps
     High               :                  0                     0 pps
    RED-dropped bytes   :                  0                     0 bps
     Low                :                  0                     0 bps
     Medium-low         :                  0                     0 bps
     Medium-high        :                  0                     0 bps
     High               :                  0                     0 bps
  Queue-depth bytes     :
    Average             :                  0
    Current             :                  0
    Peak                :                  0
```

```
    Maximum              :                  32768
Queue: 3, Forwarding classes: network-control
  Queued:
    Packets              :                      0                     0 pps
    Bytes                :                      0                     0 bps
  Transmitted:
    Packets              :                      0                     0 pps
    Bytes                :                      0                     0 bps
    Tail-dropped packets :                      0                     0 pps
    RL-dropped packets   :                      0                     0 pps
    RL-dropped bytes     :                      0                     0 bps
    RED-dropped packets  :                      0                     0 pps
     Low                 :                      0                     0 pps
     Medium-low          :                      0                     0 pps
     Medium-high         :                      0                     0 pps
     High                :                      0                     0 pps
    RED-dropped bytes    :                      0                     0 bps
     Low                 :                      0                     0 bps
     Medium-low          :                      0                     0 bps
     Medium-high         :                      0                     0 bps
     High                :                      0                     0 bps
  Queue-depth bytes      :
    Average              :                      0
    Current              :                      0
    Peak                 :                      0
    Maximum              :                6258688
```

show interfaces queue ge-0/0/0 (EX2200 Switch)

`user@switch>` **show interfaces queue ge–0/0/0**

```
Physical interface: ge-0/0/0, Enabled, Physical link is Down
  Interface index: 130, SNMP ifIndex: 501
Forwarding classes: 16 supported, 4 in use
Egress queues: 8 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
  Transmitted:
    Packets              :                      0
    Bytes                :                      0
    Tail-dropped packets :                      0
Queue: 1, Forwarding classes: assured-forwarding
  Queued:
  Transmitted:
```

```
    Packets              :                     0
    Bytes                :                     0
    Tail-dropped packets :                     0
Queue: 5, Forwarding classes: expedited-forwarding
  Queued:
  Transmitted:
    Packets              :                     0
    Bytes                :                     0
    Tail-dropped packets :                     0
Queue: 7, Forwarding classes: network-control
  Queued:
  Transmitted:
    Packets              :                     0
    Bytes                :                     0
    Tail-dropped packets :                     0
```

show interfaces queue xe-6/0/39 (Line Card with Oversubscribed Ports in an EX8200 Switch)

`user@switch>` **show interfaces queue xe-6/0/39**

```

Physical interface: xe-6/0/39, Enabled, Physical link is Up
  Interface index: 291, SNMP ifIndex: 1641
Forwarding classes: 16 supported, 7 in use
Ingress queues: 1 supported, 1 in use
  Transmitted:
    Packets              :          337069086018
    Bytes                :        43144843010304
    Tail-dropped packets :            8003867575
PFE chassis queues: 1 supported, 1 in use
  Transmitted:
    Packets              :                     0
    Bytes                :                     0
    Tail-dropped packets :                     0
Forwarding classes: 16 supported, 7 in use
Egress queues: 8 supported, 7 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
  Transmitted:
    Packets              :          334481399932
    Bytes                :        44151544791024
    Tail-dropped packets :                     0
  Queue: 1, Forwarding classes: assured-forwarding
  Queued:
```

```
  Transmitted:
    Packets              :                    0
    Bytes                :                    0
    Tail-dropped packets :                    0
  Queue: 2, Forwarding classes: mcast-be
  Queued:
  Transmitted:
    Packets              :            274948977
    Bytes                :          36293264964
    Tail-dropped packets :                    0
Queue: 4, Forwarding classes: mcast-ef
  Queued:
  Transmitted:
    Packets              :                    0
    Bytes                :                    0
    Tail-dropped packets :                    0
 Queue: 5, Forwarding classes: expedited-forwarding
  Queued:
  Transmitted:
    Packets              :                    0
    Bytes                :                    0
    Tail-dropped packets :                    0
  Queue: 6, Forwarding classes: mcast-af
  Queued:
  Transmitted:
    Packets              :                    0
    Bytes                :                    0
    Tail-dropped packets :                    0
  Queue: 7, Forwarding classes: network-control
  Queued:
  Transmitted:
    Packets              :                46714
    Bytes                :              6901326
    Tail-dropped packets :                    0

Packet Forwarding Engine Chassis Queues:
Queues: 8 supported, 7 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
  Transmitted:
    Packets              :         739338141426
    Bytes                :       94635282101928
    Tail-dropped packets :                    0
    RED-dropped packets  :           5606426444
```

```
     Low                :          5606426444
     High               :                   0
    RED-dropped bytes   :        683262846464
     Low                :        683262846464
     High               :                   0
Queue: 1, Forwarding classes: assured-forwarding
  Queued:
  Transmitted:
    Packets             :                   0
    Bytes               :                   0
    Tail-dropped packets :                  0
    RED-dropped packets  :                  0
     Low                :                   0
     High               :                   0
    RED-dropped bytes   :                   0
     Low                :                   0
     High               :                   0
Queue: 2, Forwarding classes: mcast-be
  Queued:
  Transmitted:
    Packets             :                   0
    Bytes               :                   0
    Tail-dropped packets :                  0
    RED-dropped packets  :                  0
     Low                :                   0
     High               :                   0
    RED-dropped bytes   :                   0
     Low                :                   0
     High               :                   0
Queue: 4, Forwarding classes: mcast-ef
  Queued:
  Transmitted:
    Packets             :                   0
    Bytes               :                   0
    Tail-dropped packets :                  0
    RED-dropped packets  :                  0
     Low                :                   0
     High               :                   0
    RED-dropped bytes   :                   0
     Low                :                   0
     High               :                   0
Queue: 5, Forwarding classes: expedited-forwarding
  Queued:
  Transmitted:
```

```
    Packets              :                    0
    Bytes                :                    0
    Tail-dropped packets :                    0
    RED-dropped packets  :                    0
     Low                 :                    0
     High                :                    0
    RED-dropped bytes    :                    0
     Low                 :                    0
     High                :                    0
Queue: 6, Forwarding classes: mcast-af
  Queued:
  Transmitted:
    Packets              :                    0
    Bytes                :                    0
    Tail-dropped packets :                    0
    RED-dropped packets  :                    0
     Low                 :                    0
     High                :                    0
    RED-dropped bytes    :                    0
     Low                 :                    0
     High                :                    0
Queue: 7, Forwarding classes: network-control
  Queued:
  Transmitted:
    Packets              :                97990
    Bytes                :             14987506
    Tail-dropped packets :                    0
    RED-dropped packets  :                    0
     Low                 :                    0
     High                :                    0
    RED-dropped bytes    :                    0
     Low                 :                    0
     High                :                    0
```

show interfaces queue xe-0/0/2 buffer-occupancy (QFX5000 Series switch)

user@switch> **show interfaces queue xe-0/0/2 buffer-occupancy**

```

Physical interface: xe-0/0/2, Enabled, Physical link is Up
  Interface index: 689, SNMP ifIndex: 514
Forwarding classes: 12 supported, 5 in use
Egress queues: 12 supported, 5 in use
            Queue: 0, Forwarding classes: fc0
```

```
Queue-depth bytes  :
Peak               : 1786720
Queue: 3, Forwarding classes: fcoe
Queue-depth bytes  :
Peak               : 0
Queue: 4, Forwarding classes: no-loss
Queue-depth bytes  :
Peak               : 0
Queue: 7, Forwarding classes: network-control
Queue-depth bytes  :
Peak               : 416
Queue: 8, Forwarding classes: mcast
Queue-depth bytes  :
Peak               : 0
```

show interfaces voq

Syntax

```
show interfaces voq interface-name
<forwarding-class forwarding-class-name>
<non-zero>
```

Syntax (Junos OS Evolved)

```
show interfaces voq interface-name
<forwarding-class forwarding-class-name>
<non-zero>
<source-fpc source-fpc-number>
```

Release Information

Command introduced in Junos OS Release 14.1 for the PTX Series Routers
Command introduced in Junos OS Release 15.1X53-D20 for QFX10000 switches.

Description

Display the random early detection (RED) drop statistics from all ingress Packet Forwarding Engines associated with the specified physical egress interface. In the VOQ architecture, egress output queues (shallow buffers) buffer data in virtual queues on ingress Packet Forwarding Engines. In cases of congestion, you can use this command to identify which ingress Packet Forwarding Engine is the source of RED-dropped packets contributing to congestion.

> **NOTE:** On the PTX Series routers and QFX10000 switches, these statistics include tail-dropped packets.

Options

interface *interface-name*—Display the ingress VOQ RED drop statistics for the specified egress interface.

forwarding-class *forwarding-class-name*—Display VOQ RED drop statistics for a specified forwarding class.

non-zero—Display only non-zero VOQ RED drop statistics counters.

source-fpc *source-fpc-number*—Display VOQ RED drop statistics for the specified source FPC.

Additional Information

- On PTX Series routers, you can display VOQ statistics for only the WAN physical interface.

- VOQ statistics for aggregated physical interfaces are not supported. Statistics for an aggregated interface are the summation of the queue statistics of the child links of that aggregated interface. You can use the **show interfaces queue** command to identify the child link which is experiencing congestion and then view the VOQ statistics on the respective child link using the **show interfaces voq** command.

For information on virtual output queuing on PTX routers, see *Understanding Virtual Output Queues on PTX Series Packet Transport Routers*. For information on virtual output queueing on QFX10000 switches, see "Understanding CoS Virtual Output Queues (VOQs) on QFX10000 Switches" on page 377.

Required Privilege Level
view

RELATED DOCUMENTATION

Understanding Virtual Output Queues on PTX Series Packet Transport Routers

Understanding CoS Virtual Output Queues (VOQs) on QFX10000 Switches | **377**

List of Sample Output

Output Fields

Table 160 on page 1067 lists the output fields for the show interfaces queue command. Output fields are listed in the approximate order in which they appear.

Table 160: show interfaces voq Output Fields

Field Name	Field Description
Physical interface	Name of the physical interface.
Enabled	State of the interface. Possible values are described in the "Enabled Field" section under *Common Output Fields Description*.
Interface index	Physical interface's index number, which reflects its initialization sequence.
SNMP ifIndex	SNMP index number for the interface.

Table 160: show interfaces voq Output Fields *(continued)*

Field Name	Field Description
Queue	Egress queue number.
Forwarding classes	Forwarding class name.
FPC number	Number of the Flexible PIC Concentrator (FPC) located on ingress.
PFE	Number of the Packet Forwarding Engine providing virtual output queues on the ingress.
RED-dropped packets	Number of packets per second (pps) dropped because of random early detection (RED). **NOTE:** On the PTX Series routers, these statistics include tail-dropped packets.
RED-dropped bytes	Number of bytes per second dropped because of RED. The byte counts vary by interface hardware. **NOTE:** On the PTX Series routers, these statistics include tail-dropped packets.

Sample Output

show interfaces voq (For a Specific Physical Interface) (PTX Series Routers)

The following example shows ingress RED-dropped statistics for the egress Ethernet interface configured on port 0 of Physical Interface Card (PIC) 0, located on the FPC in slot 7.

The sample output below shows that the cause of the congestion is ingress Packet Forwarding Engine PFE 0, which resides on FPC number 4, as denoted by the count of RED-dropped packets and RED-dropped bytes for egress queue 0, forwarding classes best-effort and egress queue 3, forwarding class network control.

user@host> **show interfaces voq et-7/0/0**

```
Physical interface: et-7/0/0, Enabled, Physical link is Up
  Interface index: 155, SNMP ifIndex: 699

Queue: 0, Forwarding classes: best-effort
```

```
FPC number: 1
    PFE: 0
      RED-dropped packets  :              0                 0 pps
      RED-dropped bytes    :              0                 0 bps
    PFE: 1
      RED-dropped packets  :              0                 0 pps
      RED-dropped bytes    :              0                 0 bps
    PFE: 2
      RED-dropped packets  :              0                 0 pps
      RED-dropped bytes    :              0                 0 bps
    PFE: 3
      RED-dropped packets  :              0                 0 pps
      RED-dropped bytes    :              0                 0 bps

  FPC number: 4
    PFE: 0
      RED-dropped packets  :       19969426           2323178 pps
      RED-dropped bytes    :     2196636860        2044397464 bps
    PFE: 1
      RED-dropped packets  :              0                 0 pps
      RED-dropped bytes    :              0                 0 bps
    PFE: 2
      RED-dropped packets  :              0                 0 pps
      RED-dropped bytes    :              0                 0 bps
    PFE: 3
      RED-dropped packets  :              0                 0 pps
      RED-dropped bytes    :              0                 0 bps

  FPC number: 6
    PFE: 0
      RED-dropped packets  :       19969424           2321205 pps
      RED-dropped bytes    :     2196636640        2042660808 bps
    PFE: 1
      RED-dropped packets  :              0                 0 pps
      RED-dropped bytes    :              0                 0 bps
    PFE: 2
      RED-dropped packets  :              0                 0 pps
      RED-dropped bytes    :              0                 0 bps
    PFE: 3
      RED-dropped packets  :              0                 0 pps
      RED-dropped bytes    :              0                 0 bps
    PFE: 4
      RED-dropped packets  :              0                 0 pps
      RED-dropped bytes    :              0                 0 bps
```

```
    PFE: 5
      RED-dropped packets  :                     0                      0 pps
      RED-dropped bytes    :                     0                      0 bps
    PFE: 6
      RED-dropped packets  :                     0                      0 pps
      RED-dropped bytes    :                     0                      0 bps
    PFE: 7
      RED-dropped packets  :                     0                      0 pps
      RED-dropped bytes    :                     0                      0 bps

  FPC number: 7
    PFE: 0
      RED-dropped packets  :                     0                      0 pps
      RED-dropped bytes    :                     0                      0 bps
    PFE: 1
      RED-dropped packets  :                     0                      0 pps
      RED-dropped bytes    :                     0                      0 bps
    PFE: 2
      RED-dropped packets  :                     0                      0 pps
      RED-dropped bytes    :                     0                      0 bps
    PFE: 3
      RED-dropped packets  :                     0                      0 pps
      RED-dropped bytes    :                     0                      0 bps

Queue: 1, Forwarding classes: expedited-forwarding

  FPC number: 1
    PFE: 0
      RED-dropped packets  :                     0                      0 pps
      RED-dropped bytes    :                     0                      0 bps
    PFE: 1
      RED-dropped packets  :                     0                      0 pps
      RED-dropped bytes    :                     0                      0 bps
    PFE: 2
      RED-dropped packets  :                     0                      0 pps
      RED-dropped bytes    :                     0                      0 bps
    PFE: 3
      RED-dropped packets  :                     0                      0 pps
      RED-dropped bytes    :                     0                      0 bps

  FPC number: 4
    PFE: 0
      RED-dropped packets  :                     0                      0 pps
      RED-dropped bytes    :                     0                      0 bps
```

```
  PFE: 1
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 2
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 3
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps

FPC number: 6
  PFE: 0
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 1
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 2
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 3
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 4
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 5
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 6
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 7
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps

FPC number: 7
  PFE: 0
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 1
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
  PFE: 2
```

```
      RED-dropped packets   :                     0                     0 pps
      RED-dropped bytes     :                     0                     0 bps
    PFE: 3
      RED-dropped packets   :                     0                     0 pps
      RED-dropped bytes     :                     0                     0 bps

Queue: 2, Forwarding classes: assured-forwarding

  FPC number: 1
    PFE: 0
      RED-dropped packets   :                     0                     0 pps
      RED-dropped bytes     :                     0                     0 bps
    PFE: 1
      RED-dropped packets   :                     0                     0 pps
      RED-dropped bytes     :                     0                     0 bps
    PFE: 2
      RED-dropped packets   :                     0                     0 pps
      RED-dropped bytes     :                     0                     0 bps
    PFE: 3
      RED-dropped packets   :                     0                     0 pps
      RED-dropped bytes     :                     0                     0 bps

  FPC number: 4
    PFE: 0
      RED-dropped packets   :                     0                     0 pps
      RED-dropped bytes     :                     0                     0 bps
    PFE: 1
      RED-dropped packets   :                     0                     0 pps
      RED-dropped bytes     :                     0                     0 bps
    PFE: 2
      RED-dropped packets   :                     0                     0 pps
      RED-dropped bytes     :                     0                     0 bps
    PFE: 3
      RED-dropped packets   :                     0                     0 pps
      RED-dropped bytes     :                     0                     0 bps

  FPC number: 6
    PFE: 0
      RED-dropped packets   :                     0                     0 pps
      RED-dropped bytes     :                     0                     0 bps
    PFE: 1
      RED-dropped packets   :                     0                     0 pps
      RED-dropped bytes     :                     0                     0 bps
    PFE: 2
```

```
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 3
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 4
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 5
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 6
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 7
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps

  FPC number: 7
    PFE: 0
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 1
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 2
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 3
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps

Queue: 3, Forwarding classes: network-control

  FPC number: 1
    PFE: 0
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 1
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
    PFE: 2
      RED-dropped packets  :                    0                    0 pps
      RED-dropped bytes    :                    0                    0 bps
```

```
  PFE: 3
    RED-dropped packets   :                   0                     0 pps
    RED-dropped bytes     :                   0                     0 bps

FPC number: 4
  PFE: 0
    RED-dropped packets   :            16338670               1900314 pps
    RED-dropped bytes     :          1797253700            1672276976 bps
  PFE: 1
    RED-dropped packets   :                   0                     0 pps
    RED-dropped bytes     :                   0                     0 bps
  PFE: 2
    RED-dropped packets   :                   0                     0 pps
    RED-dropped bytes     :                   0                     0 bps
  PFE: 3
    RED-dropped packets   :                   0                     0 pps
    RED-dropped bytes     :                   0                     0 bps

FPC number: 6
  PFE: 0
    RED-dropped packets   :            16338698               1899163 pps
    RED-dropped bytes     :          1797256780            1671263512 bps
  PFE: 1
    RED-dropped packets   :                   0                     0 pps
    RED-dropped bytes     :                   0                     0 bps
  PFE: 2
    RED-dropped packets   :                   0                     0 pps
    RED-dropped bytes     :                   0                     0 bps
  PFE: 3
    RED-dropped packets   :                   0                     0 pps
    RED-dropped bytes     :                   0                     0 bps
  PFE: 4
    RED-dropped packets   :                   0                     0 pps
    RED-dropped bytes     :                   0                     0 bps
  PFE: 5
    RED-dropped packets   :                   0                     0 pps
    RED-dropped bytes     :                   0                     0 bps
  PFE: 6
    RED-dropped packets   :                   0                     0 pps
    RED-dropped bytes     :                   0                     0 bps
  PFE: 7
    RED-dropped packets   :                   0                     0 pps
    RED-dropped bytes     :                   0                     0 bps
```

```
  FPC number: 7
    PFE: 0
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 1
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 2
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 3
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
```

show interfaces voq (For a Specific Physical Interface) (QFX10000 Switches)

The sample output below shows congestion on ingress PFE 1 on FPC number 0, and on ingress PFE 2 on FPC number 1, as denoted by the count of RED-dropped packets and RED-dropped bytes for best-effort egress queue 0.

`user@host>` **show interfaces voq et-1/0/0**

```
Physical interface: et-1/0/0, Enabled, Physical link is Up
  Interface index: 659, SNMP ifIndex: 539

Queue: 0, Forwarding classes: best-effort

  FPC number: 0
    PFE: 0
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 1
      RED-dropped packets  :             411063248              16891870 pps
      RED-dropped bytes    :           52616095744           17297275600 bps
    PFE: 2
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps

  FPC number: 1
    PFE: 0
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 1
      RED-dropped packets  :                     0                     0 pps
```

```
      RED-dropped bytes    :                0                  0 bps
    PFE: 2
      RED-dropped packets  :        411063012           16891870 pps
      RED-dropped bytes    :      52616065536        17297275376 bps

Queue: 3, Forwarding classes: fcoe

  FPC number: 0
    PFE: 0
      RED-dropped packets  :                0                  0 pps
      RED-dropped bytes    :                0                  0 bps
    PFE: 1
      RED-dropped packets  :                0                  0 pps
      RED-dropped bytes    :                0                  0 bps
    PFE: 2
      RED-dropped packets  :                0                  0 pps
      RED-dropped bytes    :                0                  0 bps

  FPC number: 1
    PFE: 0
      RED-dropped packets  :                0                  0 pps
      RED-dropped bytes    :                0                  0 bps
    PFE: 1
      RED-dropped packets  :                0                  0 pps
      RED-dropped bytes    :                0                  0 bps
    PFE: 2
      RED-dropped packets  :                0                  0 pps
      RED-dropped bytes    :                0                  0 bps

Queue: 4, Forwarding classes: no-loss

  FPC number: 0
    PFE: 0
      RED-dropped packets  :                0                  0 pps
      RED-dropped bytes    :                0                  0 bps
    PFE: 1
      RED-dropped packets  :                0                  0 pps
      RED-dropped bytes    :                0                  0 bps
    PFE: 2
      RED-dropped packets  :                0                  0 pps
      RED-dropped bytes    :                0                  0 bps

  FPC number: 1
    PFE: 0
```

```
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 1
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 2
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps

Queue: 7, Forwarding classes: network-control

  FPC number: 0
    PFE: 0
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 1
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 2
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps

  FPC number: 1
    PFE: 0
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 1
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 2
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
```

show interfaces voq et-7/0/0 (For a Specific Forwarding Class)

`user@host>` **show interfaces voq et-7/0/0 forwarding-class best-effort**

```
Physical interface: et-7/0/0, Enabled, Physical link is Up
  Interface index: 155, SNMP ifIndex: 699

Queue: 0, Forwarding classes: best-effort

  FPC number: 1
    PFE: 0
```

```
    RED-dropped packets  :               0               0 pps
    RED-dropped bytes    :               0               0 bps
  PFE: 1
    RED-dropped packets  :               0               0 pps
    RED-dropped bytes    :               0               0 bps
  PFE: 2
    RED-dropped packets  :               0               0 pps
    RED-dropped bytes    :               0               0 bps
  PFE: 3
    RED-dropped packets  :               0               0 pps
    RED-dropped bytes    :               0               0 bps

FPC number: 4
  PFE: 0
    RED-dropped packets  :        66604786         2321519 pps
    RED-dropped bytes    :      7326526460      2042936776 bps
  PFE: 1
    RED-dropped packets  :               0               0 pps
    RED-dropped bytes    :               0               0 bps
  PFE: 2
    RED-dropped packets  :               0               0 pps
    RED-dropped bytes    :               0               0 bps
  PFE: 3
    RED-dropped packets  :               0               0 pps
    RED-dropped bytes    :               0               0 bps

FPC number: 6
  PFE: 0
    RED-dropped packets  :        66604794          371200 pps
    RED-dropped bytes    :      7326527340       326656000 bps
  PFE: 1
    RED-dropped packets  :               0               0 pps
    RED-dropped bytes    :               0               0 bps
  PFE: 2
    RED-dropped packets  :               0               0 pps
    RED-dropped bytes    :               0               0 bps
  PFE: 3
    RED-dropped packets  :               0               0 pps
    RED-dropped bytes    :               0               0 bps
  PFE: 4
    RED-dropped packets  :               0               0 pps
    RED-dropped bytes    :               0               0 bps
  PFE: 5
    RED-dropped packets  :               0               0 pps
```

```
    RED-dropped bytes    :                     0                     0 bps
  PFE: 6
    RED-dropped packets  :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
  PFE: 7
    RED-dropped packets  :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps

FPC number: 7
  PFE: 0
    RED-dropped packets  :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
  PFE: 1
    RED-dropped packets  :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
  PFE: 2
    RED-dropped packets  :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
  PFE: 3
    RED-dropped packets  :                     0                     0 pps
    RED-dropped bytes    :                     0                     0 bps
```

show interfaces voq et-5/0/12 (For a Specific Source FPC)

`user@host>` **show interfaces voq et-5/0/12 source-fpc 0**

```
Physical interface: et-5/0/12, Enabled, Physical link is Up
    Interface index: 166, SNMP ifIndex: 1104

  Queue: 0, Forwarding classes: best-effort

    FPC number: 0
      PFE: 0
        RED-dropped packets  :                     0                     0 pps
        RED-dropped bytes    :                     0                     0 bps
      PFE: 1
        RED-dropped packets  :                     0                     0 pps
        RED-dropped bytes    :                     0                     0 bps
      PFE: 2
        RED-dropped packets  :                     0                     0 pps
        RED-dropped bytes    :                     0                     0 bps
      PFE: 3
        RED-dropped packets  :                     0                     0 pps
        RED-dropped bytes    :                     0                     0 bps
```

```
Queue: 1, Forwarding classes: expedited-forwarding

  FPC number: 0
    PFE: 0
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 1
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 2
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 3
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps

Queue: 2, Forwarding classes: assured-forwarding

  FPC number: 0
    PFE: 0
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 1
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 2
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 3
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps

Queue: 3, Forwarding classes: network-control

  FPC number: 0
    PFE: 0
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 1
      RED-dropped packets  :                     0                     0 pps
      RED-dropped bytes    :                     0                     0 bps
    PFE: 2
      RED-dropped packets  :                     0                     0 pps
```

```
        RED-dropped bytes    :                    0                    0 bps
      PFE: 3
        RED-dropped packets  :                    0                    0 pps
        RED-dropped bytes    :                    0                    0 bps
```

show interfaces voq et-5/0/12 (For a Specific Forwarding Class and Source FPC)

`user@host>` **show interfaces voq et-5/0/12 forwarding-class best-effort source-fpc 5**

```
Physical interface: et-5/0/12, Enabled, Physical link is Up
    Interface index: 166, SNMP ifIndex: 1104

  Queue: 0, Forwarding classes: best-effort

    FPC number: 5
      PFE: 0
        RED-dropped packets  :                    0                    0 pps
        RED-dropped bytes    :                    0                    0 bps
      PFE: 1
        RED-dropped packets  :                    0                    0 pps
        RED-dropped bytes    :                    0                    0 bps
      PFE: 2
        RED-dropped packets  :                    0                    0 pps
        RED-dropped bytes    :                    0                    0 bps
      PFE: 3
        RED-dropped packets  :                    0                    0 pps
        RED-dropped bytes    :                    0                    0 bps
      PFE: 4
        RED-dropped packets  :                    0                    0 pps
        RED-dropped bytes    :                    0                    0 bps
      PFE: 5
        RED-dropped packets  :                    0                    0 pps
        RED-dropped bytes    :                    0                    0 bps
      PFE: 6
        RED-dropped packets  :                    0                    0 pps
        RED-dropped bytes    :                    0                    0 bps
      PFE: 7
        RED-dropped packets  :                    0                    0 pps
        RED-dropped bytes    :                    0                    0 bps
```

show interfaces voq et-7/0/0 (Non-Zero)

`user@host>` **show interfaces voq et-7/0/0 non-zero**

```
Physical interface: et-7/0/0, Enabled, Physical link is Up
  Interface index: 155, SNMP ifIndex: 699

Queue: 0, Forwarding classes: best-effort

  FPC number: 4
    PFE: 0
      RED-dropped packets  :           95862238              2301586 pps
      RED-dropped bytes    :        10544846180           2025396264 bps

  FPC number: 6
    PFE: 0
      RED-dropped packets  :           95866639              2322569 pps
      RED-dropped bytes    :        10545330290           2043860728 bps

Queue: 3, Forwarding classes: network-control

  FPC number: 4
    PFE: 0
      RED-dropped packets  :           78433066              1899727 pps
      RED-dropped bytes    :         8627637260           1671760384 bps

  FPC number: 6
    PFE: 0
      RED-dropped packets  :           78436704              1900628 pps
      RED-dropped bytes    :         8628037440           1672553432 bps
```

show interfaces voq et-7/0/0 (For a Specific Forwarding Class and Non-Zero)

`user@host` **show interfaces voq et-7/0/0 forwarding-class best-effort non-zero**

```
Physical interface: et-7/0/0, Enabled, Physical link is Up
  Interface index: 155, SNMP ifIndex: 699

Queue: 0, Forwarding classes: best-effort

  FPC number: 4
    PFE: 0
      RED-dropped packets  :          119540012              2322319 pps
      RED-dropped bytes    :        13149401320           2043640784 bps

  FPC number: 6
```

```
    PFE: 0
      RED-dropped packets  :             119540049               2322988 pps
      RED-dropped bytes    :           13149405390            2044229744 bps
```