

Network Configuration Example

Configuring CoS Hierarchical Port Scheduling



Published: 2014-01-10

Juniper Networks, Inc.
1194 North Mathilda Avenue
Sunnyvale, California 94089
USA
408-745-2000
www.juniper.net

Juniper Networks, Junos, Steel-Belted Radius, NetScreen, and ScreenOS are registered trademarks of Juniper Networks, Inc. in the United States and other countries. The Juniper Networks Logo, the Junos logo, and JunosE are trademarks of Juniper Networks, Inc. All other trademarks, service marks, registered trademarks, or registered service marks are the property of their respective owners.

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

Network Configuration Example Configuring CoS Hierarchical Port Scheduling
NCE0071
Copyright © 2014, Juniper Networks, Inc.
All rights reserved.

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

YEAR 2000 NOTICE

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

END USER LICENSE AGREEMENT

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

Table of Contents

Introduction	1
Benefits of Configuring CoS Hierarchical Port Scheduling	1
Understanding CoS Hierarchical Port Scheduling (ETS)	2
Hierarchical Scheduling and ETS	3
ETS Advertisement in DCBX	4
Hierarchical Scheduling Process	4
Strict-High Priority Queues and Hierarchical Scheduling	5
Default Hierarchical Scheduling	6
Understanding CoS Scheduling Behavior and Configuration Considerations	6
Example: Configuring CoS Hierarchical Port Scheduling (ETS)	10

Introduction

This document describes the hierarchical port scheduling feature supported on the Juniper Networks® QFabric™ family of products and provides a step-by-step procedure for configuring hierarchical port scheduling.

Benefits of Configuring CoS Hierarchical Port Scheduling

Hierarchical port scheduling enables you to control the amount of bandwidth allocated to different traffic classes on each port in a way that maximizes flexibility and bandwidth utilization. Hierarchical port scheduling is the Junos® operating system (Junos OS) implementation of enhanced transmission selection (ETS, described in IEEE 802.1Qaz).

ETS is part of the data center bridging (DCB) IEEE standards that provide features to support lossless transport over Ethernet networks for use in data center environments.

ETS provides the flexibility to allocate a guaranteed minimum port bandwidth to any traffic class. In addition, after meeting the minimum bandwidth guarantees, ETS dynamically shares the remaining port bandwidth among the traffic classes in accordance with your configuration. This combination of capabilities results both in bandwidth guarantees for lossless traffic and in better port bandwidth utilization, which saves you money. You can configure different hierarchical port scheduling on different interfaces to customize your bandwidth allocation and utilization.

Bandwidth guarantees ensure that lossless traffic receives the bandwidth required to prevent packet loss. Better port utilization enables you to forward the same amount of traffic on fewer links than switches that do not provide ETS hierarchical scheduling. Thus, you need fewer ports to handle network traffic, which is also cost-effective.

Related Documentation

- [Understanding CoS Hierarchical Port Scheduling \(ETS\) on page 2](#)
- [Understanding CoS Scheduling Behavior and Configuration Considerations on page 6](#)
- [Example: Configuring CoS Hierarchical Port Scheduling \(ETS\) on page 10](#)

Understanding CoS Hierarchical Port Scheduling (ETS)

Scheduling defines the class-of-service (CoS) properties of output queues. These 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 and to groups of queues. Hierarchical scheduling includes the Junos OS implementation of enhanced transmission selection (ETS, described in IEEE 802.1Qaz).

The two tiers used in hierarchical scheduling are priorities and priority groups, as shown in [Table 1 on page 2](#).

Table 1: Hierarchical Scheduling Tiers

Junos OS Configuration Construct	Equivalent ETS Construct	Description
Forwarding class	Priority	<p>Think about priorities (forwarding classes) as output queues. You map forwarding classes to queues, so each forwarding class is in essence an output queue.</p> <p>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 mapped to the forwarding class all identify the same traffic.</p>
Forwarding class set	Priority group	<p>Priority groups (forwarding class sets) are groups of priorities. Forwarding class membership in a forwarding class set defines the priority group to which each priority belongs.</p> <p>You can configure up to three unicast priority groups and one multicast forwarding class set.</p>



NOTE: If you explicitly configure one or more priority groups on an interface, any priority that is not assigned to a priority group 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.

This topic describes:

- [Hierarchical Scheduling and ETS on page 3](#)
- [ETS Advertisement in DCBX on page 4](#)
- [Hierarchical Scheduling Process on page 4](#)

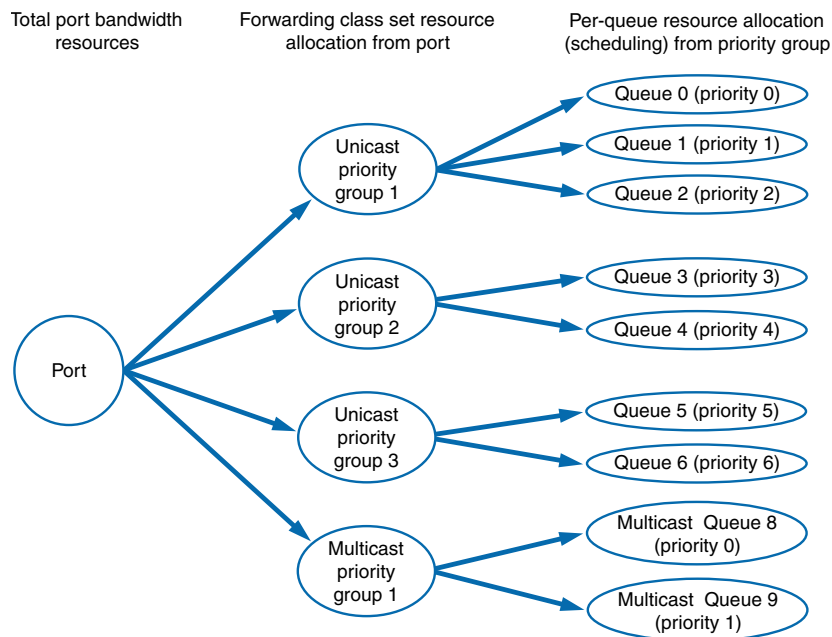
- [Strict-High Priority Queues and Hierarchical Scheduling on page 5](#)
- [Default Hierarchical Scheduling on page 6](#)

Hierarchical Scheduling and ETS

Two-tier hierarchical scheduling enables you to manage bandwidth efficiently by enabling you to define the CoS properties for each priority group and for each priority. One tier of the hierarchical scheduler allocates port bandwidth to a priority group. The other tier of the hierarchical scheduler determines the portion of the priority group bandwidth that a queue can use.

The CoS properties you configure for a priority group define the port bandwidth resources available to the queues in that priority group. The CoS properties you configure for each queue specify the portion or percentage of the total bandwidth configured for the priority group that is available to the queue. [Figure 1 on page 3](#) shows the relationship of port resource allocation to priority groups and priority group resource allocation to queues (priorities).

Figure 1: Hierarchical Scheduling Tiers



g040722

If a queue is not using its allocated bandwidth, ETS shares the unused bandwidth among the other queues in the priority group. If link bandwidth is available or if a priority group on a link is not using its allocated bandwidth, ETS shares the unused bandwidth with other priority groups on the link. In this way ETS improves link bandwidth utilization and provides each queue with the maximum bandwidth. 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.



NOTE: The available link bandwidth is the bandwidth remaining after servicing strict-high priority flows.

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.

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 2 on page 5](#):

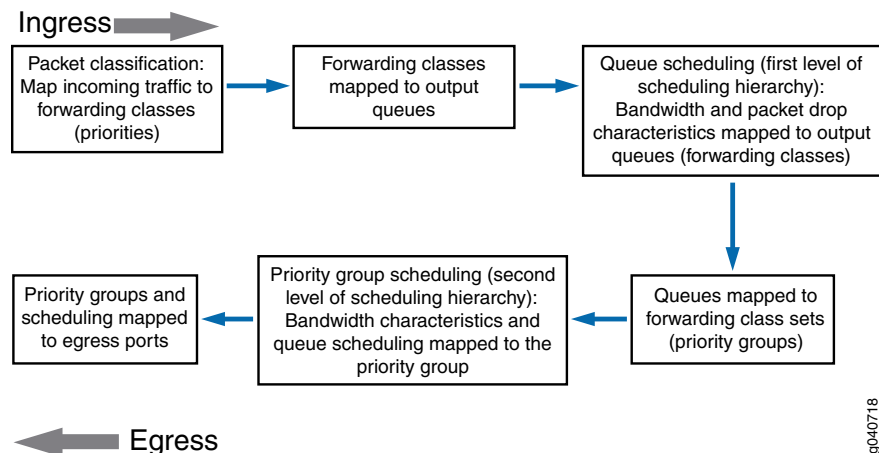
1. Packet classification:
 - Classify incoming traffic into priorities. This consists of either using the default classifiers or configuring classifiers to map IEEE 802.1p code points and loss priorities to the forwarding classes.
 - Apply the classifiers to ingress interfaces. This groups incoming traffic into forwarding classes (priorities) by mapping code points to forwarding classes and loss priorities on the specified interface.
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 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 of each queue.
 - Map resources to priorities. This consists of mapping forwarding classes to schedulers by 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 and maximum bandwidth for a priority group. Traffic control profiles also specify a scheduler map, which defines the resources (schedulers) for 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, and the schedulers specified in the scheduler map determine the amount of the priority group resources that each priority receives.

6. Apply the hierarchical scheduling to a port. This consists of attaching one or more priority groups to a port interface. For each priority group, you also attach a traffic control profile. Different priority groups on the same port can use different traffic control profiles.

Figure 2: 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.



NOTE: On a QFabric system, if a fabric (fte) interface handles strict-high priority traffic, you must define a separate fc-set (priority group) for strict-high priority traffic. Strict-high priority traffic cannot be mixed with traffic of other priorities in an fc-set. For example, you might choose to create different fc-sets for best effort, lossless, strict-high priority, and multidestination traffic.

Default Hierarchical Scheduling

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.
- 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.

Related Documentation

- [Benefits of Configuring CoS Hierarchical Port Scheduling on page 1](#)
- [Understanding CoS Scheduling Behavior and Configuration Considerations on page 6](#)
- [Example: Configuring CoS Hierarchical Port Scheduling \(ETS\) on page 10](#)

Understanding CoS Scheduling Behavior and Configuration Considerations

Many factors affect scheduling configuration and bandwidth requirements, including:

- 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.
- When you define a forwarding class that will be used on the switch (the behavior aggregate classifier has a forwarding class and you expect traffic for the 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 an interface

-
- 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, the rewrite rule rewrites only the outer VLAN tag.
 - Configuring the minimum guaranteed bandwidth (**transmit-rate**) for a queue (forwarding class) does not work unless you also configure the minimum guaranteed bandwidth (**guaranteed-rate**) for the priority group (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.) 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.

- 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.
- 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.
- There are several factors to consider if you want to configure strict-high priority queues:
 - You cannot configure a minimum guaranteed bandwidth (**transmit-rate**) for a strict-high priority queue. You cannot configure a minimum guaranteed bandwidth (**guaranteed-rate**) for a forwarding class set that includes a strict-high priority queue.
 - You must create a separate forwarding class set 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.
- In 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 may 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, you can 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.

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 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.
- 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**.
- 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.

- On an ingress port, do not configure classifiers that map the same IEEE 802.1p code point to both a multdestination traffic flow and a lossless unicast traffic flow (such as the default lossless **fcoe** or **no-loss** forwarding classes). Any code point used for multdestination traffic on a port should not be used to classify unicast traffic into a lossless forwarding class on the same port.

If a multdestination traffic flow and a lossless unicast traffic flow use the same code point on a port, the multdestination 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 multdestination traffic of the same code point is also paused. During periods of congestion, treating multdestination traffic the same as lossless unicast traffic can create ingress port congestion for the multdestination traffic and affect the multdestination traffic on all of the egress ports the multdestination traffic uses.

For example, the following configuration can cause ingress port congestion for the multdestination 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 multdestination 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 multdestination classifier that maps traffic with code point 011 to the **mcast** forwarding class is mapped to all interfaces (multdestination 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 multdestination traffic to a multdestination traffic flow, the multdestination traffic flow might experience ingress port congestion during periods of congestion.

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

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

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

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

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

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

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

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

**Related
Documentation**

- [Benefits of Configuring CoS Hierarchical Port Scheduling on page 1](#)
- [Understanding CoS Hierarchical Port Scheduling \(ETS\) on page 2](#)
- [Example: Configuring CoS Hierarchical Port Scheduling \(ETS\) on page 10](#)

Example: Configuring CoS Hierarchical Port Scheduling (ETS)

Hierarchical port scheduling defines the class-of-service (CoS) properties of output queues, which are mapped to forwarding classes (forwarding classes are mapped to IEEE 802.1p priorities, so mapping queues to forwarding classes also maps queues to priorities). Hierarchical port scheduling enables you to group priorities that require similar CoS resources 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, 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

This example describes how to configure hierarchical scheduling:

- [Requirements on page 11](#)
- [Overview on page 11](#)
- [Configuration on page 15](#)

Requirements

This example uses the following hardware and software components:

- One Juniper Networks QFX3500 Switch
- Junos OS Release 11.1 or later for the QFX Series



NOTE: This configuration example has been tested using the software release listed and is assumed to work on all later releases.

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, also called queues) to apply different CoS treatment to the 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.
 - 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 (output queue) and how much traffic you expect in each forwarding class set (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. 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. Traffic control profiles define 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 2 on page 12 shows the configuration components for this example.

Table 2: Components of the Hierarchical Port Scheduling (ETS) Configuration Topology

Property	Settings
Hardware	QFX3500 switch
Mapping of forwarding classes (priorities) to queues	<p>best-effort to queue 0</p> <p>be to queue 1</p> <p>fcoe (Fibre Channel over Ethernet) to queue 3</p> <p>no-loss to queue 4</p> <p>hpc (high-performance computing) to queue 5</p> <p>network-control to queue 7</p> <p>NOTE: 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 <i>not</i> receive lossless treatment.</p> <p>In Junos OS Release 12.3 and later, you can include the <i>no-loss</i> packet drop attribute in the explicit forwarding class configuration to configure a lossless forwarding class.</p>
Forwarding class sets (priority groups)	<p>best-effort-pg: contains forwarding classes best-effort, be, and network control</p> <p>guar-delivery-pg: contains forwarding classes fcoe and no-loss</p> <p>hpc-pg: contains forwarding class hpc</p>

Table 2: 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: <ul style="list-style-type: none"> • 000 to forwarding class best-effort and loss priority low • 001 to forwarding class be 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
Forwarding class-to-scheduler mapping	Scheduler map be-map : Forwarding class best-effort , scheduler be-sched Forwarding class be , 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%

Table 2: Components of the Hierarchical Port Scheduling (ETS) Configuration Topology *(continued)*

Property	Settings
Interfaces	<p>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:</p> <ul style="list-style-type: none"> • Classifier Name—hsclassifier1 • Forwarding class sets—best-effort-pg, guar-deliver-pg, hpc-pg • Congestion notification profile—gd-cnp

Figure 3 on page 14 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 3: Hierarchical Port Scheduling Components Block Diagram

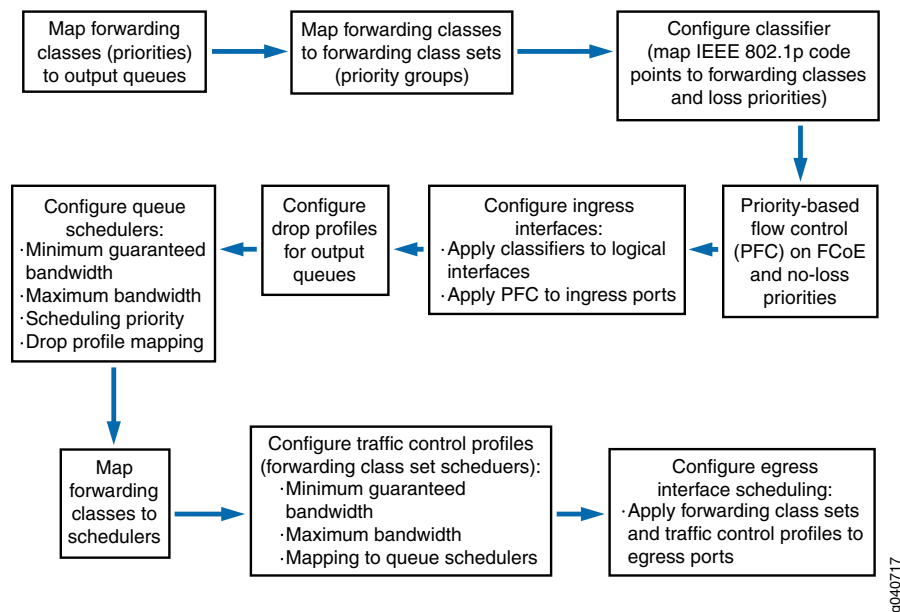
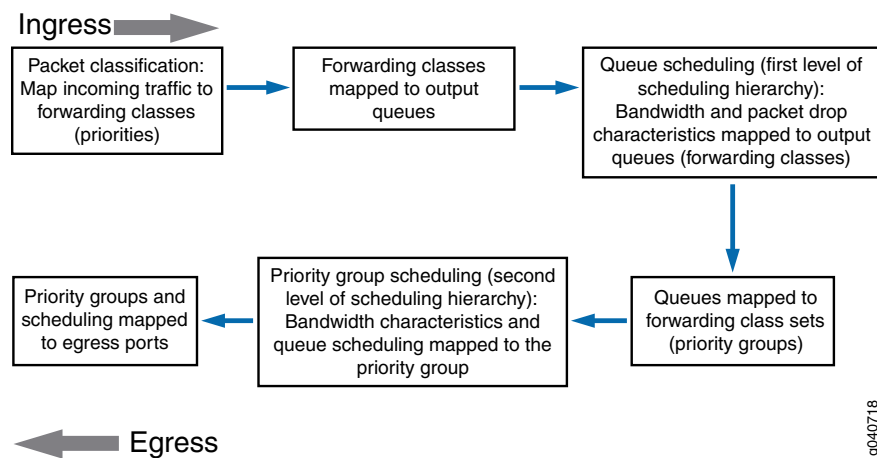


Figure 4 on page 15 shows a block diagram of the hierarchical scheduling packet flow from ingress to egress.

Figure 4: Hierarchical Port Scheduling Packet Flow Block Diagram



g040718

Configuration

CLI Quick Configuration

To quickly configure hierarchical port scheduling, 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 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
```

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
```

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
```

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
```

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
```
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
```
  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
```
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
```
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
```
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

```

Verification

To verify that the hierarchical port scheduling components have been created and are operating properly, perform these tasks:

- [Verifying That the Forwarding Classes \(Priorities\) Have Been Created on page 20](#)
- [Verifying That the Forwarding Class Sets \(Priority Groups\) Have Been Created on page 21](#)
- [Verifying That the Classifier Has Been Created on page 21](#)
- [Verifying That Priority-Based Flow Control Has Been Enabled on page 22](#)
- [Verifying That the Output Queue Schedulers Have Been Created on page 23](#)
- [Verifying That the Drop Profiles Have Been Created on page 25](#)
- [Verifying That the Priority Group Output Schedulers \(Traffic Control Profiles\) Have Been Created on page 26](#)
- [Verifying the Interface Configuration on page 27](#)

Verifying That the Forwarding Classes (Priorities) Have Been Created

Purpose Verify that the forwarding classes have been created and mapped 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 That the Forwarding Class Sets (Priority Groups) Have Been Created

Purpose Verify that the priority groups have been created 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 That the Classifier Has Been Created

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 That Priority-Based Flow Control Has Been Enabled

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 That the Output Queue Schedulers Have Been Created

Purpose Verify that the output queue schedulers have been created 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: 5000000000 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** has been 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 **3000000000 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 **5000000000 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** has been 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 **2500000000 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 **2000000000 bps**, and can consume a maximum of **100 percent** of the priority group bandwidth.
 - The scheduler map **hpc-map** has been 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 **2000000000 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 That the Drop Profiles Have Been Created

Purpose Verify that the drop profiles **dp-be-high**, **dp-be-low**, **dp-hpc**, and **dp-nc** have been created 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 That the Priority Group Output Schedulers (Traffic Control Profiles) Have Been Created

Purpose Verify that the traffic control profiles **be-tcp**, **gd-tcp**, and **hpc-tcp** have been created 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 **3500000000 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 **4500000000 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 **2000000000 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**.

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):

```

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;  
    }  
}  
}
```

**Related
Documentation**

- [Benefits of Configuring CoS Hierarchical Port Scheduling on page 1](#)
- [Understanding CoS Hierarchical Port Scheduling \(ETS\) on page 2](#)
- [Understanding CoS Scheduling Behavior and Configuration Considerations on page 6](#)