Forwarding Classes on EX9200 Switches

Published: 2013-04-19

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

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

List of Figures

List of Tables

About the Documentation

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 http://www.juniper.net/techpubs/.

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 http://www.juniper.net/books.

Supported Platforms

For the features described in this document, the following platforms are supported:

- EX Series

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 the CLI User Guide.

Documentation Conventions

Table 1 on page xi 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.

Table 2 on page xi 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`

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

Convention	Description	Examples
Italic text like this	• Introduces or emphasizes important new terms. • Identifies book names. • Identifies RFC and Internet draft titles.	• A policy *term* is a named structure that defines match conditions and actions. • *Junos OS System Basics Configuration Guide* • RFC 1997, *BGP Communities Attribute*
Italic text like this	Represents variables (options for which you substitute a value) in commands or configuration statements.	Configure the machine's domain name: [edit] root@# **set system domain-name** *domain-name*
Text like this	Represents names of configuration statements, commands, files, and directories; configuration hierarchy levels; or labels on routing platform components.	• 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**.
< > (angle brackets)	Enclose optional keywords or variables.	**stub <default-metric** ***metric*****>;**
| (pipe symbol)	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.	**broadcast | multicast** **(*string1* | *string2* | *string3*)**
# (pound sign)	Indicates a comment specified on the same line as the configuration statement to which it applies.	**rsvp { # Required for dynamic MPLS only**
[ ] (square brackets)	Enclose a variable for which you can substitute one or more values.	**community name members [** ***community-ids*** **]**

Indention and braces ({ })	Identify a level in the configuration hierarchy.	[edit]
		routing-options {
		  static {
		  route default {
		  nexthop *address*;
		  retain;
		  }
		  }
		}

; (semicolon)	Identifies a leaf statement at a configuration hierarchy level.	
GUI Conventions		
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, comments, and suggestions so that we can improve the documentation. You can send your comments to techpubs-comments@juniper.net, or fill out the documentation feedback form at https://www.juniper.net/cgi-bin/docbugreport/ . If you are using e-mail, be sure to include the following information with your comments:

- Document or topic name
- URL or page number
- Software release 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 J-Care or JNASC 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 http://www.juniper.net/us/en/local/pdf/resource-guides/7100059-en.pdf.
- Product warranties—For product warranty information, visit http://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: http://www.juniper.net/customers/support/
- Search for known bugs: http://www2.juniper.net/kb/
- Find product documentation: http://www.juniper.net/techpubs/
- Find solutions and answer questions using our Knowledge Base: http://kb.juniper.net/
- Download the latest versions of software and review release notes: http://www.juniper.net/customers/csc/software/
- Search technical bulletins for relevant hardware and software notifications: https://www.juniper.net/alerts/

- Join and participate in the Juniper Networks Community Forum: http://www.juniper.net/company/communities/
- Open a case online in the CSC Case Management tool: http://www.juniper.net/cm/

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

Opening a Case with JTAC

You can open a case with JTAC on the Web or by telephone.

- Use the Case Management tool in the CSC at http://www.juniper.net/cm/.
- 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 http://www.juniper.net/support/requesting-support.html.

PART 1

Overview

CHAPTER 1

Forwarding Classes

Overview of Forwarding Classes

This topic covers the following information:

Output Queue Assignments Based on Forwarding Class

It is helpful to think of forwarding classes as output queues. In effect, the end result of classification is the identification of an output queue for a particular packet.

CoS packet classification assigns an incoming packet to an output queue based on the packet's forwarding class. Each packet is associated with one of the following default forwarding classes:

- 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 three drop probabilities: low, medium, and high.
- Best effort (BE)—Provides no service profile. For the best effort forwarding class, loss priority is typically not carried in a class-of-service (CoS) value and random early detection (RED) drop profiles are more aggressive.
- Network control (NC)—This class is typically high priority because it supports protocol control.

Devices That Support Up to Four Forwarding Classes

Some of the Juniper Networks routing platforms support up to four forwarding classes for classifying customer traffic. On these platforms, you can configure one of each type of default forwarding class. The following Juniper Networks platforms support up to four forwarding classes:

- M7i Multiservice Edge Routers with Compact Forwarding Engine Boards (CFEBs)
- M10i Multiservice Edge Routers with CFEBs

NOTE: This list does not reference any Juniper Networks device that has reached its End of Life (EOL) period and its End of Support (EOS) milestone date.

Devices That Support Up to 16 Forwarding Classes

Other Juniper Networks routing platforms support up to 16 forwarding classes, which enables you to classify packets more granularly. For example, you can configure multiple classes of EF traffic: EF, EF1, and EF2. On these platforms, the Junos OS software supports up to eight output queues; therefore, if you configure more than eight forwarding classes, you must map multiple forwarding classes to single output queues. The following Juniper Networks routing and switching platforms support up to 16 forwarding classes and up to 8 output queues:

- EX Series switches
- M7i Multiservices Edge Routers with Enhanced Compact Forwarding Engine Boards (CFEB-Es)
- M10i Multiservices Edge Routers with CFEB-Es
- M120 Multiservices Edge Routers
- M320 Multiservices Edge Routers
- MX Series 3D Universal Edge Routers
- T Series Core Routers
- PTX Packet Transport Switches

Default and Configurable Packet Loss Priority Values

By default, the loss priority is low. On most devices, you can configure high or low loss priority. On the following devices, you can configure high, low, medium-high, or medium-low loss priority:

- J Series Services Router interfaces
- M320 routers and T Series routers with Enhanced II Flexible PIC Concentrators (FPCs)

- T640 routers with Enhanced Scaling FPC4s
- PTX Series Packet Transport Switches

Configuration Statements Used to Configure and Apply Forwarding Classes

To configure CoS forwarding classes, include the **forwarding-classes** statement at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
forwarding-classes {
   class class-name queue-num queue-number priority (high | low);
   queue queue-number class-name priority (high | low);
}
forwarding-classes-interface-specific forwarding-class-map-name {
   class class-name queue-num queue-number [ restricted-queue queue-number ];
}
interfaces {
   interface-name {
      unit logical-unit-number {
         forwarding-class class-name;
         forwarding-classes-interface-specific forwarding-class-map-name;
      }
   }
}
restricted-queues {
   forwarding-class class-name queue queue-number;
}
```

Related Documentation

- Default Forwarding Classes
- Configuring Forwarding Classes on page 11
- Applying Forwarding Classes to Interfaces on page 12
- Configuring Up to 16 Forwarding Classes
- Policer Overview

Default Forwarding Classes

By default, four queues are assigned to four forwarding classes, each with a queue number, name, and abbreviation.

These default mappings apply to all routers. The four forwarding classes defined by default are shown in Table 3 on page 6.

If desired, you can rename the forwarding classes associated with the queues supported on your hardware. Assigning a new class name to an output queue does not alter the default classification or scheduling that is applicable to that queue. CoS configurations can be quite complicated, so unless it is required by your scenario, we recommend that you not alter the default class names or queue number associations.

Some routers support eight queues. Queues 4 through 7 have no default mappings to forwarding classes. To use queues 4 through 7, you must create custom forwarding class

names and map them to the queues. For more information, see the Juniper Networks J Series Services Router documentation.

Table 3: Default Forwarding Classes

Queue	Forwarding Class Name	Comments
Queue 0	**best-effort** (**be**)	The software does not apply any special CoS handling to packets with 000000 in the DiffServ field, a backward compatibility feature. These packets are usually dropped under congested network conditions.
Queue 1	**expedited-forwarding** (**ef**)	The software delivers assured bandwidth, low loss, low delay, and low delay variation (jitter) end-to-end for packets in this service class. Routers accept excess traffic in this class, but in contrast to assured forwarding, out-of-profile expedited-forwarding packets can be forwarded out of sequence or dropped.
Queue 2	**assured-forwarding** (**af**)	The software offers a high level of assurance that the packets are delivered as long as the packet flow from the customer stays within a certain service profile that you define. The software accepts excess traffic, but applies a RED drop profile to determine if the excess packets are dropped and not forwarded. Depending on router type, up to four drop probabilities (low, medium-low, medium-high, and high) are defined for this service class.
Queue 3	**network-control** (**nc**)	The software delivers packets in this service class with a low 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, delay is preferable to discard.

The following rules govern queue assignment:

- If classifiers fail to classify a packet, the packet always receives the default classification to the class associated with queue 0.
- The number of queues is dependent on the hardware plugged into the chassis. CoS configurations are inherently contingent on the number of queues on the system. Only two classes, **best-effort** and **network-control**, are referenced in the default configuration. The default configuration works on all routers.
- CoS configurations that specify more queues than the router can support are not accepted. The commit fails with a detailed message that states the total number of queues available.
- All default CoS configuration is based on queue number. The name of the forwarding class that shows up when the default configuration is displayed is the forwarding class currently associated with that queue.

This is the default configuration for the **forwarding-classes** statement:

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

```
forwarding-classes {
  queue 0 best-effort;
  queue 1 expedited-forwarding;
  queue 2 assured-forwarding;
  queue 3 network-control;
}
```

If you reassign the forwarding-class names, the **best-effort** forwarding-class name appears in the locations in the configuration previously occupied by **network-control** as follows:

```
[edit class-of-service]
forwarding-classes {
  queue 0 network-control;
  queue 1 assured-forwarding;
  queue 2 expedited-forwarding;
  queue 3 best-effort;
}
```

All the default rules of classification and scheduling that applied to Queue 3 still apply. Queue 3 is simply now renamed **best-effort**.

On Juniper Networks M320 Multiservice Edge Routers and T Series Core Routers, you can assign multiple forwarding classes to a single queue. If you do so, the first forwarding class that you assign to queue 0 acquires the default BE classification and scheduling. The first forwarding class that you assign to queue 1 acquires the default EF classification and scheduling. The first forwarding class that you assign to queue 2 acquires the default AF classification and scheduling. The first forwarding class that you assign to queue 3 acquires the default NC classification and scheduling. For more information, see Configuring Up to 16 Forwarding Classes.

- In the current default configuration:
  - Only IP precedence classifiers are associated with interfaces.
  - The only classes designated are **best-effort** and **network-control**.
  - Schedulers are not defined for the **expedited-forwarding** or **assured-forwarding** forwarding classes.
- You must explicitly classify packets to the **expedited-forwarding** or **assured-forwarding** forwarding class and define schedulers for these classes.

For more information, see Hardware Capabilities and Limitations.

PART 2

Configuration

CHAPTER 2

Configuration Tasks

Configuring Forwarding Classes

You assign each forwarding class to an internal queue number by including the **forwarding-classes** statement at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
forwarding-classes {
  class queue-num queue-number priority (high | low);
    queue queue-number class-name priority (high | low) [ policing-priority (premium |
      normal) ];
  }
```

You cannot commit a configuration that assigns the same forwarding class to two different queues.

CAUTION: We do not recommend classifying packets into a forwarding class that has no associated scheduler on the egress interface. Such a configuration can cause unnecessary packet drops because an unconfigured scheduling class might lack adequate buffer space. For example, if you configure a custom scheduler map that does not define queue 0, and the default classifier assigns incoming packets to the best-effort class (queue 0), the unconfigured egress queue for the best-effort forwarding class might not have enough space to accommodate even short packet bursts.

A default congestion and transmission control mechanism is used when an output interface is not configured for a certain forwarding class, but receives packets destined for that unconfigured forwarding class. This default mechanism uses the delay buffer and weighted round robin (WRR) credit

allocated to the designated forwarding class, with a default drop profile. Because the buffer and WRR credit allocation is minimal, packets might be lost if a larger number of packets are forwarded without configuring the forwarding class for the interface.

Applying Forwarding Classes to Interfaces

You can configure *fixed classification* on a logical interface by specifying a forwarding class to be applied to all packets received by the logical interface, regardless of the packet contents.

NOTE: On the T4000 router, BA classification and fixed classification are mutually exclusive. That is, only one of the classifications can be configured.

To apply a forwarding class configuration to the input logical interface, include the **forwarding-class** statement at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number*]** hierarchy level:

```
[edit class-of-service interfaces interface-name unit logical-unit-number]
forwarding-class class-name;
```

You can include interface wildcards for ***interface-name*** and ***logical-unit-number***.

In the following example, all packets coming into the router from the **ge-3/0/0.0** interface are assigned to the **assured-forwarding** forwarding class:

```
[edit class-of-service]
interfaces {
  ge-3/0/0 {
    unit 0 {
      forwarding-class assured-forwarding;
    }
  }
}
```

Related Documentation

- forwarding-class on page 30

Classifying Packets by Egress Interface

For Juniper Networks EX Series switches, M320 Multiservice Edge Routers, and T Series Core Routers with the Intelligent Queuing (IQ), IQ2, Enhanced IQ (IQE), or Multiservices link services intelligent queuing (LSQ) interfaces,, you can classify unicast and multicast packets based on the egress interface. For unicast traffic, you can also use a multifield filter, but only egress interface classification applies to multicast traffic as well as unicast traffic. If you configure egress classification of an interface, you cannot perform Differentiated Services code point (DSCP) rewrites on the interface. By default, the system will not perform any classification based on the egress interface.

To enable packet classification by the egress interface, you first configure a forwarding class map and one or more queue numbers for the egress interface at the **[edit class-of-service forwarding-classes-interface-specific *forwarding-class-map-name*]** hierarchy level:

```
[edit class-of-service]
forwarding-classes-interface-specific  forwarding-class-map-name {
   class class-name queue-num queue-number [ restricted-queue queue-number ];
}
```

For T Series routers that are restricted to only four queues, you can control the queue assignment with the **restricted-queue** option, or you can allow the system to automatically determine the queue in a modular fashion. For example, a map assigning packets to queue 6 would map to queue 2 on a four-queue system.

NOTE: If you configure an output forwarding class map associating a forwarding class with a queue number, this map is not supported on multiservices link services intelligent queuing (lsq-) interfaces.

Once the forwarding class map has been configured, you apply the map to the logical interface by using the **output-forwarding-class-map** statement at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number*]** hierarchy level:

```
[edit class-of-service interfaces interface-name unit logical-unit-number]
output-forwarding-class-map forwarding-class-map-name;
```

All parameters relating to the queues and forwarding class must be configured as well. For more information about configuring forwarding classes and queues, see "Configuring Forwarding Classes" on page 11.

This example shows how to configure an interface-specific forwarding-class map named **FCMAP1** that restricts queues 5 and 6 to different queues on four-queue systems and then applies **FCMAP1** to **unit 0** of interface **ge-6/0/0**:

```
[edit class-of-service]
forwarding-classes-interface-specific FCMAP1 {
   class FC1 queue-num 6 restricted-queue 3;
   class FC2 queue-num 5 restricted-queue 2;
   class FC3 queue-num 3;
   class FC4 queue-num 0;
   class FC3 queue-num 0;
   class FC4 queue-num 1;
}

[edit class-of-service]
interfaces {
   ge-6/0/0 unit 0 {
      output-forwarding-class-map FCMAP1;
   }
}
```

Note that without the **restricted-queue** option in **FCMAP1**, the example would assign **FC1** and **FC2** to queues 2 and 1, respectively, on a system restricted to four queues.

Use the **show class-of-service forwarding-class *forwarding-class-map-name*** command to display the forwarding-class map queue configuration:

```
user@host> show class-of-service forwarding-class FCMAP2

Forwarding class                          ID      Queue  Restricted queue
  FC1                                      0       6          3
  FC2                                      1       5          2
  FC3                                      2       3          3
  FC4                                      3       0          0
  FC5                                      4       0          0
  FC6                                      5       1          1
  FC7                                      6       6          2
  FC8                                      7       7          3
```

Use the **show class-of-service interface *interface-name*** command to display the forwarding-class maps (and other information) assigned to a logical interface:

```
user@host> show class-of-service interface ge-6/0/0

Physical interface: ge-6/0/0, Index: 128
Queues supported: 8, Queues in use: 8
 Scheduler map: <default>, Index: 2
 Input scheduler map: <default>, Index: 3
 Chassis scheduler map: <default-chassis>, Index: 4

Logical interface: ge-6/0/0.0, Index: 67
 Object                    Name              Type                    Index
 Scheduler-map             sch-map1          Output                   6998
 Scheduler-map             sch-map1          Input                    6998
 Classifier                dot1p             ieee8021p                4906
 forwarding-class-map      FCMAP1            Output                   1221

Logical interface: ge-6/0/0.1, Index 68
 Object                    Name              Type                    Index
 Scheduler-map             <default>         Output                      2
 Scheduler-map             <default>         Input                       3

Logical interface: ge-6/0/0.32767, Index 69
 Object                    Name              Type                    Index
 Scheduler-map             <default>         Output                      2
 Scheduler-map             <default>         Input                       3
```

Assigning Forwarding Class and DSCP Value for Routing Engine-Generated Traffic

You can set the forwarding class and differentiated service code point (DSCP) value for traffic originating in the Routing Engine. To configure forwarding class and DSCP values that apply to Routing Engine–generated traffic only, apply an output filter to the loopback (**lo.0**) interface and set the appropriate forwarding class and DSCP bit configuration for various protocols. For example, you can set the DSCP value on OSPF packets that originate in the Routing Engine to **10** and assign them to the AF (assured forwarding) forwarding class while the DSCP value on ping packets are set to **0** and use forwarding class BE (best effort).

This particular classification ability applies to packets generated by the Routing Engine only.

The following example assigns Routing Engine sourced ping packets (using ICMP) a DSCP value of **38** and a forwarding class of **af17**, OSPF packets a DSCP value of **12** and a forwarding class of **af11**, and BGP packets (using TCP) a DSCP value of **10** and a forwarding class of **af16**.

```
[edit class-of-service]
forwarding-classes {
   class af11 queue-num 7;
   class af12 queue-num 1;
   class af13 queue-num 2;
   class af14 queue-num 4;
   class af15 queue-num 5;
   class af16 queue-num 4;
   class af17 queue-num 6;
   class af18 queue-num 7;
}

[edit firewall filter family inet]
filter loopback-filter {
   term t1 {
      from {
         protocol icmp; # For pings
      }
      then {
         forwarding-class af17;
         dscp 38;
      }
   }
   term t2 {
      from {
         protocol ospf; # For OSPF
      }
      then {
         forwarding-class af11;
         dscp 12;
      }
   }
   term t3 {
      from {
         protocol tcp; # For BGP
      }
      then {
         forwarding-class af16;
         dscp 10;
      }
   }
   term t4 {
      then accept; # Do not forget!
   }
}

[edit interfaces]
```



```
lo0 {
  unit 0 {
    family inet {
      filter {
        output loopback-filter;
      }
    }
  }
}
```

NOTE: This is not a complete router configuration. You still have to assign resources to the queues, configure the routing protocols, addresses, and so on.

Overriding Fabric Priority Queuing

On EX Series switches, and on M320 and T Series routers, the default behavior is for fabric priority queuing on egress interfaces to match the scheduling priority you assign. High-priority egress traffic is automatically assigned to high-priority fabric queues. Likewise, low-priority egress traffic is automatically assigned to low-priority fabric queues.

You can override the default fabric priority queuing of egress traffic by including the **priority** statement at the **[edit class-of-service forwarding-classes queue *queue-number class-name*]** hierarchy level:

```
[edit class-of-service forwarding-classes queue queue-number class-name]
priority (high | low);
```

For information about associating a scheduler with a fabric priority, see Associating Schedulers with Fabric Priorities.

Configuring Up to 16 Forwarding Classes

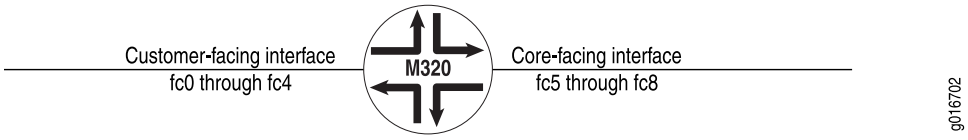
By default on all routers and switches, four output queues are mapped to four forwarding classes, as shown in the topic Default Forwarding Classes. On Juniper Networks J Series Services Routers, M120 and M320 Multiservice Edge Routers, and T Series Core Routers, you can configure more than four forwarding classes and queues. For information about configuring J Series routers, see the J Series router documentation.

NOTE: You cannot use CoS-based forwarding features if you configure more than eight forwarding classes on the device.

On M120, M320, MX Series, T Series routers, EX Series switches and PTX Series Packet Transport Switches, you can configure up to 16 forwarding classes and eight queues, with multiple forwarding classes assigned to single queues. The concept of assigning multiple forwarding classes to a queue is sometimes referred to as creating *forwarding-class aliases*. This section explains how to configure M320 and T Series routers.

Mapping multiple forwarding classes to single queues is useful. Suppose, for example, that forwarding classes are set based on multifield packet classification, and the multifield classifiers are different for core-facing interfaces and customer-facing interfaces. Suppose you need four queues for a core-facing interface and five queues for a customer-facing interface, where **fc0** through **fc4** correspond to the classifiers for the customer-facing interface, and **fc5** through **fc8** correspond to classifiers for the core-facing interface, as shown in Figure 1 on page 17.

Figure 1: Customer-Facing and Core-Facing Forwarding Classes



In this example, there are nine classifiers and, therefore, nine forwarding classes. The forwarding class-to-queue mapping is shown in Table 4 on page 17.

Table 4: Sample Forwarding Class-to-Queue Mapping

Forwarding Class Names	Queue Number
fc0 **fc5**	0
fc1 **fc6**	1
fc2 **fc7**	2
fc3 **fc8**	3
fc4	4

To configure up to 16 forwarding classes, include the **class** and **queue-num** statements at the **[edit class-of-service forwarding-classes]** hierarchy level:

```
[edit class-of-service forwarding-classes]
class class-name queue-num queue-number;
```

You can configure up 16 different forwarding-class names. The corresponding output queue number can be from 0 through 7. Therefore, you can map multiple forwarding classes to a single queue. If you map multiple forwarding classes to a queue, the multiple forwarding classes must refer to the same scheduler (at the **[edit class-of-service scheduler-maps *map-name* forwarding-class *class-name* scheduler *scheduler-name*]** hierarchy level).

When you configure up to 16 forwarding classes, you can use them as you can any other forwarding class—in classifiers, schedulers, firewall filters (multifield classifiers), policers, and rewrite rules.

When you configure up to 16 forwarding classes, the following limitations apply:

- The **class** and **queue** statements at the **[edit class-of-service forwarding-classes]** hierarchy level are mutually exclusive. In other words, you can include one or the other of the following configurations, but not both:

  [edit class-of-service forwarding-classes]
  queue *queue-number class-name*;

  [edit class-of-service forwarding-classes]
  class *class-name* queue-num *queue-number*;

- On T Series routers only, when you configure IEEE 802.1p rewrite marking on Gigabit Ethernet IQ, Gigabit Ethernet IQ2, Gigabit Ethernet Enhanced IQ (IQE), and Gigabit Ethernet Enhanced IQ2 (IQ2E) PICs, you cannot configure more than eight forwarding classes. This limitation does not apply to M Series routers. On M Series routers, you can configure up to 16 forwarding classes when you configure IEEE 802.1p rewrite marking on any of these PICs.
- For GRE and IP-IP tunnels, IP precedence and DSCP rewrite marking of the inner header do not work with more than eight forwarding classes.
- When you use CoS-based forwarding features, you cannot configure more than eight forwarding classes with a forwarding policy. However, if you try to configure CoS-based forwarding with more than eight forwarding classes configured, commit fails with a message. Therefore, you can configure CBF on a router with eight or less than eight forwarding classes only. Under this condition, the forwarding class to queue mapping can be either one-to-one or one-to-many.
- A scheduler map that maps eight different forwarding classes to eight different schedulers can only be applied to interfaces that support eight queues. If you apply this type of scheduler map to an interface that only supports four queues, then the commit will fail.
- We recommend that you configure the statements changing PICs to support eight queues and then applying an eight queue scheduler map in two separate steps. Otherwise, the commit might succeed but the PIC might not have eight queues when the scheduler map is applied, generating an error.

You can determine the ID number assigned to a forwarding class by issuing the **show class-of-service forwarding-class** command. You can determine whether the classification is fixed by issuing the **show class-of-service forwarding-table classifier mapping** command. In the command output, if the **Table Type** field appears as **Fixed**, the classification is fixed. For more information about fixed classification, see "Applying Forwarding Classes to Interfaces" on page 12.

This section discusses the following topics:

- PICs Restricted to Four Queues on page 20
- Examples: Configuring Up to 16 Forwarding Classes on page 21

Enabling Eight Queues on Interfaces

By default, Intelligent Queuing (IQ), Intelligent Queuing 2 (IQ2), Intelligent Queuing Enhanced (IQE), and Intelligent Queuing 2 Enhanced (IQ2E) PICs on M320 and T Series routers are restricted to a maximum of four egress queues per interface. To configure a maximum of eight egress queues on these interfaces, include the **max-queues-per-interface** statement at the **[edit chassis fpc *slot-number* pic *pic-number*]** hierarchy level:

```
[edit chassis fpc slot-number pic pic-number]
max-queues-per-interface (4 | 8);
```

On a TX Matrix or TX Matrix Plus router, include the **max-queues-per-interface** statement at the **[edit chassis lcc *number* fpc *slot-number* pic *pic-number*]** hierarchy level:

```
[edit chassis lcc number fpc slot-number pic pic-number]
max-queues-per-interface (4 | 8);
```

The numerical value can be **4** or **8**.

For Juniper Networks J Series routers, this statement is not supported. J Series routers always have eight queues available.

NOTE: **In addition to configuring eight queues at the [edit chassis] hierarchy level, the configuration at the [edit class-of-service] hierarchy level must support eight queues per interface.**

The maximum number of queues per IQ PIC can be **4** or **8**. If you include the **max-queues-per-interface** statement, all ports on the IQ PIC use configured mode and all interfaces on the IQ PIC have the same maximum number of queues.

To determine how many queues an interface supports, you can check the **CoS queues** output field of the **show interfaces *interface-name* extensive** command:

```
user@host> show interfaces ge-1/0/0 extensive
CoS queues: 8 supported
```

If you include the **max-queues-per-interface 4** statement, you can configure all four ports and configure up to four queues per port.

For 4-port OC3c/STM1 Type I and Type II PICs on M320 and T Series routers, when you include the **max-queues-per-interface 8** statement, you can configure up to eight queues on ports 0 and 2. After you commit the configuration, the PIC goes offline and comes back online with only ports 0 and 2 operational. No interfaces can be configured on ports 1 and 3.

For Quad T3 and Quad E3 PICs, when you include the **max-queues-per-interface 8** statement, you can configure up to eight queues on ports 0 and 2. After you commit the

configuration, the PIC goes offline and comes back online with only ports 0 and 2 operational. No interfaces can be configured on ports 1 and 3.

When you include the **max-queues-per-interface** statement and commit the configuration, all physical interfaces on the IQ PIC are deleted and readded. Also, the PIC is taken offline and then brought back online immediately. You do not need to take the PIC offline and online manually. You should change modes between four queues and eight queues only when there is no active traffic going to the IQ PIC.

Multiple Forwarding Classes and Default Forwarding Classes

For queues 0 through 3, if you assign multiple forwarding classes to a single queue, default forwarding class assignment works as follows:

- The first forwarding class that you assign to queue 0 acquires the default BE classification and scheduling.
- The first forwarding class that you assign to queue 1 acquires the default EF classification and scheduling.
- The first forwarding class that you assign to queue 2 acquires the default AF classification and scheduling.
- The first forwarding class that you assign to queue 3 acquires the default NC classification and scheduling.

Of course you can override the default classification and scheduling by configuring custom classifiers and schedulers.

If you do not explicitly map forwarding classes to queues 0 through 3, then the respective default classes are automatically assigned to those queues. When you are counting the 16 forwarding classes, you must include in the total any default forwarding classes automatically assigned to queues 0 through 3. As a result, you can map up to 13 forwarding classes to a single queue when the single queue is queue 0, 1, 2, or 3. You can map up to 12 forwarding classes to a single queue when the single queue is queue 4, 5, 6, or 7. In summary, there must be at least one forwarding class each (default or otherwise) assigned to queue 0 through 3, and you can assign the remaining 12 forwarding classes (16–4) to any queue.

For example, suppose you assign two forwarding classes to queue 0 and you assign no forwarding classes to queues 1 through 3. The software automatically assigns one default forwarding class each to queues 1 through 3. This means 11 forwarding classes (16–5) are available for you to assign to queues 4 through 7.

For more information about forwarding class defaults, see Default Forwarding Classes.

PICs Restricted to Four Queues

Some Juniper Networks T Series Core Router PICs support up to 16 forwarding classes and are restricted to 4 queues. Contact Juniper Networks customer support for a current list of T Series router PICs that are restricted to four queues. To determine how many

queues an interface supports, you can check the **CoS queues** output field of the **show interfaces *interface-name* extensive** command:

```
user@host> show interfaces ge-1/0/0 extensive
CoS queues: 8 supported
```

By default, for T Series router PICs that are restricted to four queues, the router overrides the global configuration based on the following formula:

Qr = Qd mod Rmax

Qr is the queue number assigned if the PIC is restricted to four queues.

Qd is the queue number that would have been mapped if this PIC were not restricted.

Rmax is the maximum number of restricted queues available. Currently, this is four.

For example, assume you map the forwarding class **ef** to queue 6. For a PIC restricted to four queues, the queue number for forwarding class **ef** is **Qr** = 6 mod 4 = 2.

To determine which queue is assigned to a forwarding class, use the **show class-of-service forwarding-class** command from the top level of the CLI. The output shows queue assignments for both global queue mappings and restricted queue mappings:

```
user@host> show class-of-service forwarding-class
Forwarding class            Queue     Restricted Queue  Fabric priority
  be                            0           2              low
  ef                            1           2              low
  assured-forwarding            2           2              low
  network-control               3           3              low
```

For T Series router PICs restricted to four queues, you can override the formula-derived queue assignment by including the **restricted-queues** statement at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
restricted-queues {
  forwarding-class class-name queue queue-number;
}
```

You can configure up to 16 forwarding classes. The output queue number can be from 0 through 3. Therefore, for PICs restricted to four queues, you can map multiple forwarding classes to single queues. If you map multiple forwarding classes to a queue, the multiple forwarding classes must refer to the same scheduler. This requirement applies to all PICs. The class name you configure at the **[edit class-of-service restricted-queues]** hierarchy level must be either a default forwarding class name or a forwarding class you configure at the **[edit class-of-service forwarding-classes]** hierarchy level.

Examples: Configuring Up to 16 Forwarding Classes

Configure 16 forwarding classes:

Configuring 16 Forwarding Classes

```
[edit class-of-service]
forwarding-classes {
  class fc0 queue-num 0;
  class fc1 queue-num 0;
  class fc2 queue-num 1;
```

```
    class fc3 queue-num 1;
    class fc4 queue-num 2;
    class fc5 queue-num 2;
    class fc6 queue-num 3;
    class fc7 queue-num 3;
    class fc8 queue-num 4;
    class fc9 queue-num 4;
    class fc10 queue-num 5;
    class fc11 queue-num 5;
    class fc12 queue-num 6;
    class fc13 queue-num 6;
    class fc14 queue-num 7;
    class fc15 queue-num 7;
}
```

For PICs restricted to four queues, map four forwarding classes to each queue:

Restricted Queues: Mapping Two Forwarding Classes to Each Queue

```
[edit class-of-service]
restricted-queues {
    forwarding-class fc0 queue 0;
    forwarding-class fc1 queue 0;
    forwarding-class fc2 queue 0;
    forwarding-class fc3 queue 0;
    forwarding-class fc4 queue 1;
    forwarding-class fc5 queue 1;
    forwarding-class fc6 queue 1;
    forwarding-class fc7 queue 1;
    forwarding-class fc8 queue 2;
    forwarding-class fc9 queue 2;
    forwarding-class fc10 queue 2;
    forwarding-class fc11 queue 2;
    forwarding-class fc12 queue 3;
    forwarding-class fc13 queue 3;
    forwarding-class fc14 queue 3;
    forwarding-class fc15 queue 3;
}
```

If you map multiple forwarding classes to a queue, the multiple forwarding classes must refer to the same scheduler:

Configuring a Scheduler Map Applicable to an Interface Restricted to Four Queues

```
[edit class-of-service]
scheduler-maps {
    interface-restricted {
        forwarding-class be scheduler Q0;
        forwarding-class ef scheduler Q1;
        forwarding-class ef1 scheduler Q1;
        forwarding-class ef2 scheduler Q1;
        forwarding-class af1 scheduler Q2;
        forwarding-class af scheduler Q2;
        forwarding-class nc scheduler Q3;
        forwarding-class nc1 scheduler Q3;
    }
}
[edit class-of-service]
restricted-queues {
    forwarding-class be queue 0;
    forwarding-class ef queue 1;
```

```
    forwarding-class ef1 queue 1;
    forwarding-class ef2 queue 1;
    forwarding-class af queue 2;
    forwarding-class af1 queue 2;
    forwarding-class nc queue 3;
    forwarding-class nc1 queue 3;
}
```

CHAPTER 3

Configuration Statements

[edit class-of-service] Hierarchy Level

```
class-of-service {
  classifiers {
    type classifier-name {
      forwarding-class class-name {
        loss-priority (high | low | medium-high | medium-low) code-points [ aliases bits ];
      }
      import (classifier-name | default);
    }
  }
  code-point-aliases {
    (dscp | dscp-ipv6 | exp | ieee-802.1 | ieee-802.1ad | inet-precedence) {
      alias-name bits;
    }
  }
  drop-profiles {
    profile-name {
      fill-level percentage drop-probability percentage;
      interpolate {
        drop-probability value;
        fill-level value;
      }
    }
  }
  fabric {
    scheduler-map {
      priority (high | low) scheduler scheduler-name;
    }
  }
  forwarding-class-map {
    map-name {
      class class-name queue-num queue-number <restricted-queue queue-number>;
    }
  }
  forwarding-classes {
    class class-name policing-priority (normal | premium) queue-num queue-number
      priority (high | low);
```

```
    queue queue-number class-name policing-priority (normal | premium) priority (high |
      low);
  }
  forwarding-policy {
    class class-name {
      classification-override {
        forwarding-class class-name;
      }
    }
    next-hop-map map-name {
      forwarding-class class-name {
        discard;
        lsp-next-hop [ lsp-regular-expressions ];
        next-hop [ next-hop-names ];
        non-lsp-next-hop;
      }
    }
  }
  fragmentation-maps {
    map-name {
      forwarding-class class-name {
        drop-timeout milliseconds;
        fragment-threshold bytes;
        multilink-class number;
        no-fragmentation;
      }
    }
  }
  host-outbound-traffic {
    dscp-code-point value;
    forwarding-class class-name;
    ieee-802.1 {
      default value;
      rewrite-rules;
    }
    tcp {
      raise-internet-control-priority;
    }
  }
  interfaces {
    ... the interfaces subhierarchy appears after the main [edit class-of-service] hierarchy
      ...
  }
  }
  restricted-queues {
    forwarding-class class-name queue-number;
  }
  rewrite-rules {
    (dscp | dscp-ipv6 | exp |frame-relay-de | ieee-802.1 | ieee-802.1ad | inet-precedence)
      rewrite-rule {
      forwarding-class class-name {
        loss-priority level code-point (alias | bits);
      }
      import (rewrite-rule | default);
    }
  }
```

```
    routing-instances routing-instance-name {
      classifiers {
        dscp (classifier-name | default);
        dscp-ipv6 (classifier-name | default);
        exp (classifier-name | default);
        ieee-208.1 (classifier-name | default | encapsulated | vlan-tag (inner | outer));
      }
    }
    scheduler-maps {
      map-name {
        forwarding-class class-name scheduler scheduler-name;
      }
    }
    schedulers {
      scheduler-name {
        adjust-minimum value;
        adjust-percent value;
        buffer-size (exact | percent percentage | remainder);
        drop-profile-map loss-priority (any | high | low | medium-high | medium-low)
          protocol any;
        excess-priority (high | low | medium-high | medium-low);
        excess-rate (percent percentage | proportion proportion);
        priority (high | low | medium-high | medium-low | strict-high);
        shaping-rate (bps | percent percentage | burst-size size);
        transmit-rate (bps | percent percentage | remainder) <exact | rate-limit>;
      }
    }
    traceoptions {
      file <files number> <match regular-expression> <size maximum-file-size>
        <world-readable | no-world-readable>;
      flag flag;
      no-remote-trace;
    }
    traffic-control-profiles {
      profile-name {
        adjust-minimum rate;
        delay-buffer-rate (bps | cps cps | percent percentage);
        excess-rate (percent percentage | proportion value);
        guaranteed-rate (bps | percent percentage) <burst-size bytes>;
        overhead-accounting (frame-mode | cell-mode) <bytes byte-value>;
        scheduler-map map-name;
        shaping-rate (bps | percent percentage) <burst-size bytes>;
      }
    }
    tri-color;
  }

  class-of-service {
    interfaces {
      interface-name {
        excess-bandwith-share (equal | proportional value);
        input-excess-bandwith-share (equal | proportional value);
        input-scheduler-map map-name;
        input-shaping-rate bps;
        input-traffic-control-profile profile-name;
        output-forwarding-class-map map-name;
```

```
    output-traffic-control-profile profile-name;
    scheduler-map map-name;
    scheduler-map-chassis (map-name | derived);
    shaping-rate bps;
    unit (logical-unit-number | *){
      classifiers {
        dscp (classifier-name | default) {
          family [ inet mpls ];
        }
        dscp-ipv6 (classifier-name | default) {
          family [ inet mpls ];
        }
        exp (classifier-name | default);
        ieee-208.1 (classifier-name | default)  <vlan-tag (inner | outer)>;
        ieee-208.1ad (classifier-name | default);
        inet-precedence (classifier-name | default);
      }
      forwarding-class class-name;
      input-scheduler-map map-name;
      input-shaping-rate bps;
      input-traffic-control-profile profile-name shared-instance instance-name;
      loss-priority-maps {
        (map-name | default);
      }
      loss-priority-rewrites {
        (map-name | default);
      }
      output-forwarding-class-map map-name;
      output-traffic-control-profile profile-name shared-instance instance-name;
      rewrite-rules {
        dscp (rule-name | default) <protocol mpls>;
        dscp-ipv6 (rule-name | default);
        exp (rule-name | default) <protocol [ mpls-any | mpls-inet-both |
          mpls-inet-both-non-vpn ]>;
        exp-push-push-push default;
        exp-swap-push-push default;
        ieee-802.1 (rewrite-name | default) <vlan-tag (outer | outer-and-inner)>;
        ieee-802.1ad (rewrite-name | default) <vlan-tag (outer | outer-and-inner)>;
        inet-precedence (rewrite-name | default) <protocol mpls>;
      }
      scheduler-map map-name;
      shaping-rate bps;
      translation-table (to-dscp-from-dscp | to-dscp-ipv6-from-dscp-ipv6 |
        to-exp-from-exp | to-inet-precedence-from-inet-precedence) table-name;
    }
  }
  interface-set interface-set-name {
    excess-bandwith-share (equal | proportional value);
    input-excess-bandwith-share (equal | proportional value);
    input-traffic-control-profile profile-name;
    input-traffic-control-profile-remaining profile-name;
    internal-node;
    output-traffic-control-profile profile-name;
    output-traffic-control-profile-remaining profile-name;
  }
}
```

}

Related Documentation

- Notational Conventions Used in Junos OS Configuration Hierarchies

class (Forwarding Classes)

Syntax 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.

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.

Options *class-name*—Name of forwarding class.

queue-number—Output queue number.

Range: 0 through 15. Some T Series router PICs are restricted to 0 through 3.

The remaining statement is explained separately.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Forwarding Classes on page 11
- queue (Global Queues) on page 37

forwarding-class (Interfaces)

Syntax forwarding-class *class-name*;

Hierarchy Level [edit class-of-service interfaces *interface-name* unit *logical-unit-number*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.2 for ACX Series routers.

Description Associate a forwarding class configuration or default mapping with a specific interface.

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

- Applying Forwarding Classes to Interfaces on page 12
- Configuring Fixed Classification on an ATM IMA Pseudowire
- Example: Configuring Fixed Classification on an ATM IMA Pseudowire

forwarding-class (Restricted Queues)

Syntax forwarding-class *class-name* queue *queue-number*;

Hierarchy Level [edit class-of-service restricted-queues]

Release Information Statement introduced before Junos OS Release 7.4.

Description For M320 and T Series routers only, map forwarding classes to restricted queues. You can map up to eight forwarding classes to restricted queues.

Options ***class-name***—Name of the forwarding class.

The remaining statement is explained separately.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

forwarding-classes (Class-of-Service)

Syntax

```
forwarding-classes {
    class queue-num queue-number priority (high | low);
    queue queue-number 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.
policing-priority option introduced in Junos OS Release 9.5.
Statement introduced on PTX Series Packet Transport Switches in Junos OS Release 12.1.

Description Associate the forwarding class with a queue name and number. For M320, MX Series, 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 add policing-priority options are not supported on PTX Series Packet Transport Switches.

The statements are explained separately.

See "Configuring Forwarding Classes" on page 11, "Overriding Fabric Priority Queuing" on page 16, and Example: Configuring CoS for a PBB Network. For the **policing-priority** option, see Configuring Layer 2 Policers on IQE PICs. For classification by egress interface, see Classifying Packets by Egress Interface.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

forwarding-classes-interface-specific

Syntax

```
forwarding-classes-interface-specific forwarding-class-map-name {
  class class-name queue-num queue-number [ restricted-queue queue-number ];
}
```

Hierarchy Level [edit class-of-service]

Release Information Statement introduced in Junos OS Release 9.6.

Description For the IQ, IQE, LSQ and ATM2 PICs in the T Series routers only and for EX Series switches, configure a forwarding class map for unicast and multicast traffic and a user-configured queue number for an egress interface.

Options

class-name—Name of the forwarding class.

forwarding-class-map-name—Name of the forwarding class map for traffic.

queue-number—Number of the egress queue.

Range: 0 through 3 or 7, depending on chassis and configuration

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Forwarding Classes on page 11
- Classifying Packets by Egress Interface
- output-forwarding-class-map on page 35

interfaces

Syntax

```
interfaces {
  interface-name {
    classifiers{
      dscp(classifier-name | default) {
    }
      ieee-802.1 (classifier-name | default)  vlan-tag (inner | outer | classifier-name);
      inet-precedence (rewrite-name | default);
    }
    input-scheduler-map map-name;
    input-shaping-rate rate;
    irb {
      unit logical-unit-number {
        classifiers {
          type (classifier-name | default);
        }
        rewrite-rules {
          dscp (rewrite-name | default);
          dscp-ipv6 (rewrite-name | default);
          exp (rewrite-name | default)protocol protocol-types;
          ieee-802.1 (rewrite-name | default) vlan-tag (outer | outer-and-inner);
          inet-precedence (rewrite-name | default);
        }
      }
    }
    member-link-scheduler (replicate | scale);
    rewrite-rules {
      dscp (rewrite-name | default);
      ieee-802.1 (rewrite-name | default) vlan-tag (outer);
      inet-precedence (rewrite-name | default);
      }
    }
    scheduler-map map-name;
    scheduler-map-chassis map-name;
    shaping-rate rate;
    unit logical-unit-number {
      classifiers {
        type (classifier-name | default) family (mpls | inet);
      }
      forwarding-class class-name;
      fragmentation-map map-name;
      input-shaping-rate (percent percentage | rate);
      input-traffic-control-profile profile-name shared-instance instance-name;
      output-traffic-control-profile profile-name shared-instance instance-name;
      per-session-scheduler;
      rewrite-rules {
        dscp (rewrite-name | default);
        dscp-ipv6 (rewrite-name | default);
        exp (rewrite-name | default) protocol protocol-types;
        exp-push-push-push default;
        exp-swap-push-push default;
        ieee-802.1 (rewrite-name | default) vlan-tag (outer | outer-and-inner);
        inet-precedence (rewrite-name | default);
```

```
        }
        scheduler-map map-name;
        shaping-rate rate;
        translation-table (to-dscp-from-dscp | to-dscp-ipv6-from-dscp-ipv6 | to-exp-from-exp
          | to-inet-precedence-from-inet-precedence) table-name;
      }
    }
    interface-set interface-set-name {
      excess-bandwidth-share;
      internal-node;
      output-traffic-control-profile profile-name;
      output-traffic-control-profile-remaining profile-name;
    }
}
```

Hierarchy Level [edit class-of-service]

Release Information Statement introduced before Junos OS Release 7.4.
Interface-set level added in Junos OS Release 8.5.

Description Configure interface-specific CoS properties for incoming packets.

NOTE: The dscp-ipv6 and ieee-802.1ad classifier types are not supported on ACX Series routers. For further information about support on ACX Series routers, see Understanding CoS CLI Configuration Statements on ACX Series Universal Access Routers.

Options 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.

Related Documentation

- Overview of BA Classifier Types
- Configuring Rewrite Rules
- Understanding CoS CLI Configuration Statements on ACX Series Universal Access Routers

output-forwarding-class-map

Syntax output-forwarding-class-map *forwarding-class-map-name*;

Hierarchy Level [edit class-of-service forwarding-classes-interface-specific]

Release Information Statement introduced in Junos OS Release 9.6.

Description Apply a configured forwarding class map to a logical interface.

Options *forwarding-class-map-name*—Name of a forwarding class mapping configured at the **[edit class-of-service forwarding-classes-interface-specific]** hierarchy level.

Required Privilege Level interface-control—To add this statement to the configuration.

Related Documentation

- Classifying Packets by Egress Interface
- forwarding-classes-interface-specific on page 32

priority (Fabric Priority)

Syntax priority (high | low);

Hierarchy Level [edit class-of-service forwarding-classes class *class-name* queue-num *queue-number*], [edit class-of-service forwarding-classes queue *queue-number class-name*]

Release Information Statement introduced before Junos OS Release 7.4.
[edit class-of-service forwarding-classes class *class-name* queue-num *queue-number*] hierarchy level added in Junos OS Release 8.1.

Description For M320 routers, MX Series routers, T Series routers and EX Series switches only, specify a fabric priority value.

The two hierarchy levels are mutually exclusive. To configure up to eight forwarding classes with one-to-one mapping between forwarding classes and output queues, include this statement at the **[edit class-of-service forwarding-classes queue *queue-number class-name*]** hierarchy level. To configure up to 16 forwarding classes with multiple forwarding classes mapped to single queues, include this statement at the **[edit class-of-service forwarding-classes class *class-name* queue-num *queue-number*]** hierarchy level.

Options **low**—Forwarding class's fabric queuing has low priority.

high—Forwarding class's fabric queuing has high priority.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Overriding Fabric Priority Queuing on page 16
- Configuring Up to 16 Forwarding Classes

queue (Global Queues)

Syntax queue *queue-number class-name*;

Hierarchy Level [edit class-of-service forwarding-classes]

Release Information Statement introduced before Junos OS Release 7.4.

Description Specify the output transmission queue to which to map all input from an associated forwarding class.

On M120, M320, MX Series, T Series routers and on EX Series switches, this statement enables you to configure up to eight forwarding classes with one-to-one mapping to output queues. If you want to configure up to 16 forwarding classes with multiple forwarding classes mapped to single output queues, include the **class** statement instead of the **queue** statement at the **[edit class-of-service forwarding-classes]** hierarchy level.

Options *class-name*—Name of forwarding class.

queue-number—Output queue number.

Range: For M Series routers, 0 through 3. For M120, M320, MX Series, T Series routers and EX Series switches, 0 through 7. Some T Series router PICs are restricted to 0 through 3.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Forwarding Classes on page 11
- class (Forwarding Classes) on page 29

queue (Restricted Queues)

Syntax queue *queue-number*;

Hierarchy Level [edit class-of-service restricted-queues forwarding-class *class-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description For M320, MX Series, T Series routers and EX Series switches only, map forwarding classes to restricted queues.

Options *queue-number*—Output queue number.

Range: 0 through 3.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

restricted-queues

Syntax

```
restricted-queues {
    forwarding-class class-name queue queue-number;
}
```

Hierarchy Level [edit class-of-service]

Release Information Statement introduced before Junos OS Release 7.4.

Description For M320, MX Series, T Series routers and EX Series switches only, map forwarding classes to restricted queues.

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.

unit

Syntax

```
unit logical-unit-number {
  classifiers {
    type (classifier-name | default) family (mpls | all);
  }
  forwarding-class class-name;
  fragmentation-map map-name;
  input-traffic-control-profile profile-name shared-instance instance-name;
  output-traffic-control-profile profile-name shared-instance instance-name;
  per-session-scheduler;
  rewrite-rules {
    dscp (rewrite-name | default);
    dscp-ipv6 (rewrite-name | default);
    exp (rewrite-name | default)protocol protocol-types;
    exp-push-push-push default;
    exp-swap-push-push default;
    ieee-802.1 (rewrite-name | default) vlan-tag (outer | outer-and-inner);
    inet-precedence (rewrite-name | default);
  }
  scheduler-map map-name;
  shaping-rate rate;
}
```

Hierarchy Level [edit class-of-service interfaces *interface-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Configure a logical interface on the physical device. You must configure a logical interface to be able to use the physical device.

Options ***logical-unit-number***—Number of the logical unit.

Range: 0 through 16,384

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.

Related Documentation

- Overview of BA Classifier Types
- Configuring Rewrite Rules

