53-1003037-02
9 December, 2013

Multi-Service IronWare

QoS and Traffic Management Configuration Guide

Supporting Multi-Service IronWare R05.6.00a

BROCADE

Copyright © 2013 Brocade Communications Systems, Inc. All Rights Reserved.

ADX, AnyIO, Brocade, Brocade Assurance, the B-wing symbol, DCX, Fabric OS, ICX, MLX, MyBrocade, OpenScript, VCS, VDX, and Vyatta are registered trademarks, and HyperEdge, The Effortless Network, and The On-Demand Data Center are trademarks of Brocade Communications Systems, Inc., in the United States and/or in other countries. Other brands, products, or service names mentioned may be trademarks of their respective owners.

Notice: This document is for informational purposes only and does not set forth any warranty, expressed or implied, concerning any equipment, equipment feature, or service offered or to be offered by Brocade. Brocade reserves the right to make changes to this document at any time, without notice, and assumes no responsibility for its use. This informational document describes features that may not be currently available. Contact a Brocade sales office for information on feature and product availability. Export of technical data contained in this document may require an export license from the United States government.

The authors and Brocade Communications Systems, Inc. shall have no liability or responsibility to any person or entity with respect to any loss, cost, liability, or damages arising from the information contained in this book or the computer programs that accompany it.

The product described by this document may contain “open source” software covered by the GNU General Public License or other open source license agreements. To find out which open source software is included in Brocade products, view the licensing terms applicable to the open source software, and obtain a copy of the programming source code, please visit *http://www.brocade.com/support/oscd*.

Brocade Communications Systems, Incorporated

Corporate and Latin American Headquarters
Brocade Communications Systems, Inc.
130 Holger Way
San Jose, CA 95134
Tel: 1-408-333-8000
Fax: 1-408-333-8101
E-mail: info@brocade.com

Asia-Pacific Headquarters
Brocade Communications Systems China HK, Ltd.
No. 1 Guanghua Road
Chao Yang District
Units 2718 and 2818
Beijing 100020, China
Tel: +8610 6588 8888
Fax: +8610 6588 9999
E-mail: china-info@brocade.com

European Headquarters
Brocade Communications Switzerland Sàrl
Centre Swissair
Tour B - 4ème étage
29, Route de l'Aéroport
Case Postale 105
CH-1215 Genève 15
Switzerland
Tel: +41 22 799 5640
Fax: +41 22 799 5641
E-mail: emea-info@brocade.com

Asia-Pacific Headquarters
Brocade Communications Systems Co., Ltd. (Shenzhen WFOE)
Citic Plaza
No. 233 Tian He Road North
Unit 1308 – 13th Floor
Guangzhou, China
Tel: +8620 3891 2000
Fax: +8620 3891 2111
E-mail: china-info@brocade.com

Document History

Title	Publication number	Summary of changes	Date
Multi-Service IronWare QoS and Traffic Management Configuration Guide	*53-1003037-01*	Release 05.5.00c document updated with enhancements in Release 05.6.00	30 September, 2013
Multi-Service IronWare QoS and Traffic Management Configuration Guide	*53-1003037-02*	Release 05.6.00 document updated with enhancements in Release 05.6.00a	9 December, 2013

Contents

15

71

Chapter 4 Configuring Quality of Service for the Brocade NetIron XMR and Brocade MLX series

171

Chapter 5 Hierarchical Quality of Service (HQoS)

About This Document

Audience

This document is designed for system administrators with a working knowledge of Layer 2 and Layer 3 switching and routing.

If you are using a Brocade device, you should be familiar with the following protocols if applicable to your network – IP, RIP, OSPF, BGP, ISIS, IGMP, PIM, MPLS, and VRRP.

Supported hardware and software

The following hardware platforms are supported by this release of this guide:

TABLE 1 Supported devices

Brocade NetIron XMR Series	Brocade MLX Series	NetIron CES 2000 and NetIron CER 2000 Series
Brocade NetIron XMR 4000	Brocade MLX-4	Brocade NetIron CES 2024C
Brocade NetIron XMR 8000	Brocade MLX-8	Brocade NetIron CES 2024F
Brocade NetIron XMR 16000	Brocade MLX-16	Brocade NetIron CES 2048C
Brocade NetIron XMR 32000	Brocade MLX-32	Brocade NetIron CES 2048CX
	Brocade MLXe-4	Brocade NetIron CES 2048F
	Brocade MLXe-8	Brocade NetIron CES 2048FX
	Brocade MLXe-16	Brocade NetIron CER 2024C
	Brocade MLXe-32	Brocade NetIron CER-RT 2024C
		Brocade NetIron CER 2024F
		Brocade NetIron CER-RT 2024F
		Brocade NetIron CER 2048C
		Brocade NetIron CER-RT 2048C
		Brocade NetIron CER 2048CX
		Brocade NetIron CER-RT 2048CX
		Brocade NetIron CER 2048F
		Brocade NetIron CER-RT 2048F
		Brocade NetIron CER 2048FX
		Brocade NetIron CER-RT 2048FX

Supported software

For the complete list of supported features and the summary of enhancements and configuration notes for this release, refer to the *Multi-Service IronWare R05.6.00 Release Notes*.

Document conventions

This section describes text formatting conventions and important notice formats used in this document.

Text formatting

The narrative-text formatting conventions that are used are as follows:

bold text	Identifies command names
	Identifies the names of user-manipulated GUI elements
	Identifies keywords
	Identifies text to enter at the GUI or CLI
italic text	Provides emphasis
	Identifies variables
	Identifies document titles
`code` text	Identifies CLI output

For readability, command names in the narrative portions of this guide are presented in bold: for example, **show version**.

Notes, cautions, and danger notices

The following notices and statements are used in this manual. They are listed below in order of increasing severity of potential hazards.

NOTE
A note provides a tip, guidance or advice, emphasizes important information, or provides a reference to related information.

CAUTION
A Caution statement alerts you to situations that can be potentially hazardous to you or cause damage to hardware, firmware, software, or data.

DANGER
A Danger statement indicates conditions or situations that can be potentially lethal or extremely hazardous to you. Safety labels are also attached directly to products to warn of these conditions or situations.

Notice to the reader

This document may contain references to the trademarks of the following corporations. These trademarks are the properties of their respective companies and corporations.

These references are made for informational purposes only.

Corporation	Referenced Trademarks and Products
Microsoft Corporation	Internet Explorer
Mozilla Corporation	Mozilla Firefox
Sun Microsystems	Java Runtime Environment

Related publications

For the latest edition of these documents, which contain the most up-to-date information, see Documentation at http://www.brocade.com/ethernetproducts

- *Multi-Service IronWare Administration Guide*
- *Multi-Service IronWare Security Configuration Guide*
- *Multi-Service IronWare Switching Configuration Guide*
- *Multi-Service IronWare Routing Configuration Guide*
- *Multi-Service IronWare Traffic Management Configuration Guide*
- *Multi-Service IronWare Multicast Configuration Guide*
- *Multi-Service IronWare Multiprotocol Label Switch (MPLS) Configuration Guide*
- *Multi-Service IronWare Software Defined Networking (SDN) Guide*
- *Multi-Service IronWare Family YANG Guide*
- *Brocade MLX Series and NetIron XMR Series Diagnostic Reference*
- *Unified IP MIB Reference*
- *Multi-Service IronWare Software Upgrade Guide*
- *Brocade MLXe Series Installation Guide*
- *Brocade MLX Series and Brocade NetIron XMR Installation Guide*
- *Brocade NetIron CES 2000 Series and Brocade NetIron CER 2000 Series Hardware Installation Guide*

Getting technical help or reporting errors

To contact Technical Support, go to http://www.brocade.com/services-support/index.page for the latest e-mail and telephone contact information.

Chapter 1

Configuring Traffic Policing for the Brocade NetIron CES and Brocade NetIron CER

Traffic policing on Brocade NetIron CES and Brocade NetIron CER devices

Brocade NetIron CES and Brocade NetIron CER devices provide line-rate traffic policing in hardware on inbound and outbound ports.

You can configure a device to use one of the following traffic policing modes:

- **Port-based** – Limits the rate on an individual physical port to a specified number. Only one inbound and one outbound port-based traffic policing policy can be applied to a port. (Refer to "Configuring port-based traffic policing for inbound and outbound ports" on page 5.) These policies can be applied to inbound and outbound traffic.
- **Port-and-ACL-based** – Limits the rate of IP traffic on an individual physical port that matches the permit conditions in IP Access Control Lists (ACLs). Layer-2 ACL-based traffic policing is supported. You can use standard or extended IP ACLs. Standard IP ACLs match traffic based on source IP address information. Extended ACLs match traffic based on source and destination IP address and IP protocol information. Extended ACLs for TCP and UDP also match on source and destination TCP or UDP addresses, and protocol information. These policies can be applied to inbound and outbound traffic. Brocade NetIron CES and Brocade NetIron CER devices support up to 3967 (1984 egress and 1983 ingress) policies for a port per packet processor (PPCR).

Multi-Service IronWare software lets you apply traffic policing parameters directly to a port, or create a policy map to define a set of traffic policing parameters and apply that policy map to one or more ports.

For information about how to apply traffic parameters directly to a port, refer to "Applying traffic policing parameters directly to a port" on page 1.

For information about how to apply traffic policing parameters using a policy map, refer to "Applying traffic policing parameters using a policy map" on page 2.

Applying traffic policing parameters directly to a port

When you apply a traffic policing parameters directly to a port, two parameters are specified: average rate and maximum burst. These parameters configure credits and credit totals.

Average rate

The *average rate* is the maximum number of bits a port can receive during a one-second interval. The rate of the traffic will not exceed the average rate as specified by the traffic policing policy.

The *average rate* represents a percentage of line rate (bandwidth) for an interface, expressed in bits per second (bps). It cannot be larger than the line rate for the port. For the Brocade NetIron CER and Brocade NetIron CES devices, the *average rate* can be entered in as any value from 0 up to the line rate of the port.

Maximum burst

Maximum burst allows a higher-than-average rate to traffic that meets the rate limiting criteria. Traffic is allowed to pass through the port for a short period of time. The unused bandwidth can be accumulated up to a maximum equal to the *maximum burst* value.

Maximum burst size is adjusted according the configured average line rate. If the user configured maximum burst size value exceeds the maximum burst size allowed, the maximum burst size will be automatically adjusted to the values indicated in Table 1.

TABLE 1 Maximum Burst Size

Average rate (bps)	Maximum burst size (Bits)
1 Mbps	524,280
1 - 10 Mbps	4,194,240
10 - 100 Mbps	33,553,920
100 Mbps - 1 Gbps	268,431,360
1 Gbps - 10 Gbps	1,410,064,384

Credits and credit total

Each rate limiting policy is assigned a class. The *class* uses the average rate and maximum burst in the rate limit policy to calculate credits and credit totals.

Credit size is measured in bits. A credit is a forwarding allowance for a traffic policed port, and is the smallest number of bits that can be allowed during a rate limiting interval. Minimum credit size can be 1 bit.

During a rate limiting interval, a port can send or receive only as many bits as the port has credits for. For example, if an inbound rate limiting policy results in a port receiving two credits per rate limiting interval, the port can send or receive a maximum of 2 bits of data during that interval.

In each interval, the number of bits equal to the credit size is added to the running total of the class. The running total of a class represents the number of bits that can pass without being subject to rate limiting.

The second parameter is the maximum *credit total*. The maximum credit total is based on the maximum burst value and is measured in bits.

The running total can never exceed the maximum credit total. When a packet arrives at the port, a class is assigned to the packet based on the traffic policing policies. If the running total of the class is less than the size of the packet, the packet is dropped. Otherwise, the size of the packet is subtracted from the running total and the packet is forwarded. If there is no traffic that matches traffic policing criteria, then the running total can grow up to the maximum credit total.

Applying traffic policing parameters using a policy map

The policy map configuration ties a policy name to a set of traffic policing policies. The policy name is then applied to ports that you want to rate limit using the defined policy. This allows you to set a policy in a single location the affects multiple ports and to make changes to that policy. Refer to "Configuring a policy map" on page 4.

In the policy map configuration, the traffic policing policy determines the rate of inbound or outbound traffic (in bits per second or bps) that is allowed per port. This traffic is initially traffic policed by a Committed Information Rate (CIR) bucket. Traffic that is not accommodated in the CIR bucket is then subject to the Excess Information Rate (EIR) bucket.

The CIR bucket

The CIR rate limiting bucket is defined by two parameters: the CIR rate, and the Committed Burst Size (CBS) rate. The CIR rate is the maximum number of bits a port is allowed to receive or send during a one-second interval. The rate of the traffic that matches the traffic policing policy can not exceed the CIR rate. The CIR rate represents a portion of the line rate (bandwidth) for an interface expressed in bits per second (bps) and cannot be larger than the line rate of the port. CIR-defined traffic that does not use the available CIR rate accumulates credits that be used later in circumstances where it temporarily exceeds the CIR rate.

When traffic exceeds the bandwidth that has been reserved for it by the CIR rate defined in its policy, it becomes subject to the CBS rate. The CBS rate is higher than the CIR rate to traffic that exceeds the CIR rate. The bandwidth in the CBS rate accumulates during periods when traffic that has been defined by a policy does not use the full CIR rate available. Traffic is allowed to pass through the port for a short period of time at the CBS rate.

When inbound or outbound traffic exceeds the bandwidth available for the defined CIR and CBS rates, it is either dropped, or made subject to the conditions set in the EIR bucket.

The EIR bucket

The EIR bucket provides an option for traffic that has exceeded the conditions set by policy for the CIR bucket. In the EIR bucket, two parameters define traffic that is available: the Excess Information Rate (EIR) and the Excess Burst Size (EBS) rate. The EIR and EBS operate exactly like the CIR and CBS except that they only act upon traffic that has been passed to the EIR bucket because it could not be accommodated by the CIR bucket. Like the CIR, the EIR provides an initial bandwidth allocation to accommodate inbound and outbound traffic. If the bandwidth provided by the EIR is insufficient to accommodate the excess traffic, the defined EBS rate provides for burst traffic. Like the CBS, the bandwidth available for burst traffic from the EBS is subject to the amount of bandwidth that is accumulated during periods of time when traffic that has been allocated by the EIR policy is not used.

In addition to providing additional bandwidth for traffic that exceeds that available for the CIR bucket, traffic rate limited by the EIR bucket can have its excess priority and excess dscp values changed. Using this option, priority parameters are set following the EBS value that change the priority of traffic that is being rate limited using the EIR bucket.

Configuration considerations

- Only one type of traffic policing policy can be applied on a physical port. For example, you cannot apply port-and-ACL-based and port-based traffic policing policies on the same port.
- For policy-map based rate-limiting, the committed burst in a traffic policing policy cannot be less than 1250 bytes and cannot be more than the port's line rate in bytes.
- For non policy-map based rate-limiting, the maximum burst in a traffic policing policy cannot be less than 10000 bits and cannot be more than the port's line rate in bits.
- Control packets are not subject to traffic policing.
- Source MAC address with Virtual Leased Line (VLL) endpoints are not subject to traffic policing.
- Up to four different sets of excess parameters are supported. Delete or unbind other policy maps from other interfaces. You can also change the excess parameters in the policy map to match one of the existing profiles to share the Remark Profile Tables.
- BUM rate-limit may not work properly with a lower configured-rate and bursty traffic.

Limitations

In the Brocade NetIron CES and Brocade NetIron CER, UDP rate-limiting is applicable only in the following scenarios:

- When sending 1% of 1G traffic with packet size of 64 bytes to the device for configured Burst-max value (up to 8000)
- When sending 10% of 1G traffic with packet size of 64 bytes to the device for configured Burst-max value (up to 1500)
- When sending 100% of 1G traffic with packet size of 64 bytes to the device for configured Burst-max value (up to 500).

Configuring traffic policing

The following sections show examples of how to configure each type of traffic policing.

Configuring a policy map

To configure a policy map, enter commands such as these.

```
Brocade(config)#policy-map map1
Brocade(config-policymap map1)#cir 1000000 cbs 500000 eir 1000000 ebs 500000
```

This command configures traffic policing policy-map map1 to limit CIR rate to 1000000, the CBS rate to 500000, the EIR rate to 1000000, and the EBS to 500000. This command only creates a policy, it must be applied to one or more ports to be operational.

Syntax: [**no**] **policy-map** *map-name*

Syntax: [**no**] **cir** *cir-rate* **cbs** *cbs-rate* **eir** *eir-rate* **ebs** *ebs-rate* [**excess-dp** *dp-val* **excess-dscp** *dscp-num* **excess-priority** *priority-num* **excess-pcp** *pcp-num* **excess-exp** *exp-num*]

The *map-name* variable is the name you use to reference the policy map in traffic policing command. It can be a character string up to 64 characters long.

The **cir** parameter defines the value of the CIR as the rate defined in the *cir-rate* variable. Acceptable values are: 0 - 100,000,000,000 bps.

The **cbs** parameter defines the value of the CBS as the rate defined in the *cbs-rate* variable. Acceptable values are: 1250 - 12,500,000,000 bytes in increments of 1 byte.

The **eir** parameter defines the value of the EIR as the rate defined in the *eir-rate* variable. Acceptable values are: 0 - 100,000,000,000 bps.

The **ebs** parameter defines the value of the EBS as the rate defined in the *ebs-rate* variable. Acceptable values are: 1250 - 12,500,000,000 bytes in increments of 1 byte.

The following parameters are optional. If configured, they will specify the remarking of the Quality of Service parameters, if the rate is over the limits that are specified by CIR or CBS, but less than the limit of EIR and EBS.

The **excess-dp** parameter specifies the drop precedence for traffic whose bandwidth requirements exceed what is available in the CIR bucket and is sent to the EIR bucket. Acceptable values for the *dp-value* are 0 - 3. Packets with a value of 0 are least likely to be dropped and packets with a value of 3 are most likely to be dropped.

For Drop Precedence, the Brocade MLX Series and Brocade NetIron XMR has 4 levels, while Brocade NetIron CER and Brocade NetIron CES have 3 levels. The Brocade NetIron CER and Brocade NetIron CES internally converts the 4 levels as follows:

0 -> 0, 1 -> 1, 2 -> 1, 3 -> 2

The **excess-dscp** parameter specifies that traffic whose bandwidth requirements exceeds what is available in the CIR bucket and is sent to the EIR bucket. These packets will have their DSCP priority set to the value set in the *dscp-num* variable. Acceptable values for the *dscp-num* are 0 - 63. When this parameter is used together with the **excess-dp** parameter, the value set for bits 2:1 (zero-based) in the **excess-dscp** parameter must be equal to the value set for excess-dp. The **excess-dp** parameter is compared with **excess-dscp** in bit [2:1] first, then it is converted.

The **excess-priority** parameter specifies that traffic whose bandwidth requirements exceeds what is available in the CIR bucket and is sent to the EIR bucket. These packets will have their priority queue or TC set to the value set in the *priority num* variable. Acceptable values for the *priority-num* are 0 - 7.

The **excess-pcp** parameter specifies that traffic whose bandwidth requirements exceeds what is available in the CIR bucket and is sent to the EIR bucket. These packets will have their pcp or UP (user priority) set to the value set in the *pcp num* variable. Acceptable values for the *priority-num* are 0 - 7.

The **excess-exp** parameter specifies that traffic whose bandwidth requirements exceeds what is available in the CIR bucket and is sent to the EIR bucket. These packets will their EXP set to the value set in the *exp num* variable. Acceptable values for the *exp num* are 0-7.

Configuring a policy-map to remark profile tables

The Brocade NetIron CER and Brocade NetIron CES each use have four remark profile tables per packet processor. The profile determines the CoS remapping tables sets to use, and each profile contains the 5 excess parameters (i.e., excess-dp, excess-dscp, excess-priority, excess-pcp, excess-exp), which remark the Quality of Service parameters if the rate is over the limits that are specified in the CIR, CBS, EIR and EBS parameters.

The profile is selected per port configuration. For the Ingress policier, the profile is selected per the source port; and for the Egress policier, the profile is selected per target port. You can configure as many policy-maps as you want. However, when you apply or bind the policy-map to an interface, only four different profiles are available, where each of the four profiles may be applied to multiple ports.

Note that a policy-map includes both rates and sizes and the 5 excess parameters. A remark profile table has the 5 excess parameters only, so multiple policy-maps may share a single remark profile table, if the 5 excess parameters match.

Configuring port-based traffic policing for inbound and outbound ports

Port-based traffic policing limits the rate on an individual inbound or outbound physical port to a specified rate.

To configure port-based traffic policing policy for outbound ports, enter commands such as the following at the interface level.

```
Brocade(config)# interface ethernet 1/1
Brocade(config-if-1/1)# rate-limit out 500000000 750000000
```

These commands configure a traffic policing policy for outbound traffic on port 1/1. The policy limits the average rate of all outbound traffic to 500 Mbps with a maximum burst size of 750 bits.

Configuring port-based traffic policing using a policy map

To configure port based traffic policing policy through a policy map, enter commands such as the following.

```
Brocade(config)# interface ethernet 1/1
Brocade(config-if-1/1)# rate-limit input policy-map map1
```

These commands configure a traffic policing policy for inbound traffic on port 1/1. The policy references policy map1 for rate limiting policy parameters.

The complete syntax for configuring a port-based traffic policing policy is:

Syntax: [**no**] **rate-limit** [**in** | **out**] [*average-rate maximum-burst* | **policy-map** *map-name*]

The **input** parameter applies the policy to traffic on inbound ports.

The **output** parameter applies the policy to traffic on outbound ports.

Only one inbound and one outbound port-based traffic policing policy can be applied to a port.

The *average-rate* parameter specifies the maximum rate allowed on a port during a one-second interval. For the Brocade NetIron CER and Brocade NetIron CES devices, the Average Rate can be entered in as any value from 0 up to the line rate of the port. Refer to "Average rate" on page 1 for more details. This command is only used when configuring traffic policing directly to a port as described in "Applying traffic policing parameters directly to a port" on page 1.

The *maximum-burst* parameter specifies the extra bits above the average rate that traffic can have. Refer to "Maximum burst" on page 2 for more details. This command is only used when configuring traffic policing directly to a port as described in "Applying traffic policing parameters directly to a port" on page 1.

The **policy-map** parameter specifies the policy map named in the *policy-map* variable to be used to provide parameters for rate limiting the port and VLAN specified. This command is only used when configuring traffic policing to a port using a policy map as described in "Applying traffic policing parameters using a policy map" on page 2.

NOTE
Excess-dp is not supported on egress.

Configuring a port-and-ACL-based traffic policing policy

You can use standard or extended IP ACLs for port-and-ACL-based traffic policing:

- Standard IP ACLs match traffic based on source IP address information.
- Extended ACLs match traffic based on source and destination IP addresses and IP protocol information. Extended ACLs for TCP and UDP protocols must also match on source and destination IP addresses and TCP or UDP protocol information.
- You can apply an ACL ID to a port-and-ACL-based traffic policing policy before you define the ACL. The traffic policing policy does not take effect until the ACL is defined.
- It is not necessary to remove an ACL from a port-and-ACL-based rate limiting policy before deleting the ACL.
- Layer-2 ACL rate limiting is supported.

Port-and-ACL-based traffic policing is supported for traffic on inbound and outbound ports. To configure port-and-ACL-based traffic policing policies, enter commands such as the following.

```
Brocade(config)#access-list 50 permit host 1.1.1.2
Brocade(config)#access-list 50 deny host 1.1.1.3
Brocade(config)#access-list 60 permit host 2.2.2.3
Brocade(config-if-1/1)# rate-limit input access-group 50 500000000 20480
Brocade(config-if-1/1)# rate-limit input access-group 60 100000000 24194240
```

These commands first configure access-list groups that contain the ACLs that will be used in the traffic policing policy. Use the **permit** condition for traffic that will be policed. Traffic that matches the **deny** condition is not subject to traffic policing.

Next, the commands configure two traffic policing policies on port 1/1. The policies limit the average rate of all inbound IP traffic that matches the permit rules of ACLs 50 and 60. The first policy limits the rate of all permitted IP traffic from host 1.1.1.2 to an average rate of 500 Mbps with a maximum burst size of 20480 Mbits. Traffic from host 1.1.1.3 is not subject to rate limiting since it is denied by ACL 50; it is merely forwarded on the port.

The second policy limits the rate of all IP traffic from host 2.2.2.3 to an average of 100 Mbps with a maximum burst size of 4194240 Mbits.

Traffic that does not match ACLs 50 and 60 is not subject to traffic policing.

Syntax: [**no**] **rate-limit** [**input** | **output**] **access-grou**p *group-number* [*average-rate maximum-burst* | **policy-map** *map-name*]

The **input** parameter applies the policy to traffic on inbound ports.

The **output** parameter applies the policy to traffic on outbound ports.

The **access-group**, *group-number* parameter specifies the group number to which the ACLs used in the policy belong.

NOTE
An ACL must exist in the configuration before it can take effect in a traffic policing policy.

The *average-rate* variable specifies the maximum rate allowed on a port during a one-second interval. The software automatically adjusts the number you enter to the nearest multiple of 8,144 bps. Refer to "Average rate" on page 1 for more details. This command is only used when configuring rate limiting directly to a port as described in "Applying traffic policing parameters directly to a port" on page 1.

The *maximum-burst* variable specifies the extra Mbits above the average rate that traffic can have. Refer to "Maximum burst" on page 2 for more details. This command is only used when configuring traffic policing directly to a port as described in "Applying traffic policing parameters directly to a port" on page 1.

The **policy-map** parameter specifies the policy map named in the *policy-map* variable to be used to provide parameters for traffic policing the VLAN specified. This command is only used when configuring traffic policing to a port using a policy map as described in "Applying traffic policing parameters using a policy map" on page 2.

Using ACLs for filtering in addition to rate limiting

When you use the ACL-based mode, the permit and deny conditions in an ACL you use in a rate limiting policy work as follows:

- **Permit** – The traffic is rate limited according to the other parameters in the rate limiting policy.
- **Deny** – The traffic is forwarded instead of dropped, by default.

You can configure the device to drop traffic that is denied by the ACL instead of forwarding the traffic, on an individual port basis.

NOTE
Once you configure an ACL-based rate limiting policy on a port, you cannot configure a regular (traffic filtering) ACL on the same port. To filter traffic, you must enable the strict ACL option.

To configure the device to drop traffic that is denied by a rate limiting ACL, enter the following command at the configuration level for the port.

```
Brocade(config-if-1/1)# rate-limit strict-acl
```

Syntax: [no] **rate-limit strict-acl**

Displaying rate limiting policies

Use one of the following commands to view the rate limiting policies that have been configured:

- **show rate limit counters** – Displays accounting information for rate limit usage. Only ACL based counters are displayed.
- **show rate limit** – Displays rate limiting policies implemented per interface.
- **show policy map** – Displays rate limiting policies implemented in the configured policy maps.

You can configure a device to exclude the 20-byte per-packet Ethernet overhead from traffic policing byte accounting using the **vlan-counter exclude-overhead** command.

Displaying accounting information for rate limit usage

To display accounting information for rate limit usage, enter the following command.

```
Brocade# show rate-limit counters
```

Syntax: show rate-limit counters *slot/port*

The *slot/port* option allows you to get accounting information for a specified interface only.

Output such as the following is displayed.

```
Brocade# show rate-limit counters
interface e 1/1
 rate-limit input 959904 2000000
  Fwd:        10000                   Drop:  1000 bytes
  Re-mark:    0                       Total: 11000 bytes
 rate-limit output 2986368 2000000
  Fwd:        20000                   Drop:  2340 bytes
  Re-mark:    0                       Total: 22340 bytes

This display shows the following information.
```

TABLE 2 Rate limit counters parameters

This field...	Displays...
rate-limit input	Defines rate limit policy for inbound traffic on the defined interface.
rate-limit output	Defines rate limit policy for outbound traffic on the defined interface.
Fwd	Traffic (in bytes) that has been forwarded as a result of this rate limit policy since the device was started or the counter was reset.
Drop	Traffic (in bytes) that has been dropped as a result of the defined rate limit policy since the device was started up or the counter has been reset.

TABLE 2 Rate limit counters parameters (Continued)

This field...	Displays...
Re-mark	The number of packets for which priority has been remarked as a result of exceeding the bandwidth available in the CIR bucket for this rate limit policy.
Total	Total traffic (in bytes) that has been carried on this interface for the defined rate limit policy since the device was started or the counter was reset.

NOTE
Port based rate limit counters are not supported for Brocade NetIron CER and Brocade NetIron CES.

Resetting the rate limit counters

You can reset all of the rate limit counters using the following command.

```
Brocade# clear rate-limit counters
```

Syntax: clear rate-limit counters *slot/port*

The *slot/port* variable specifies a port for which you want to clear the rate limit counters. If you do not specify a port, all rate limit counters on the device are reset.

Displaying rate limit policies for a specific interface

To display information about rate limit policies that are configured for a specific interface, enter the following command at the interface level.

```
Brocade(config-if-e10000-1/1)#show rate-limit
```

Syntax: show rate-limit

Output such as the following is displayed.

```
interface e 1/1
 rate-limit input 959904 2000000
 rate-limit output 2986368 2000000
```

This display shows the following information.

TABLE 3 Rate limit interface parameters

This field...	Displays...
rate-limit input	The average-rate and maximum burst rate configured for inbound traffic on the specified interface.
rate-limit output	The average-rate and maximum burst rate configured for outbound traffic on the specified interface.

Displaying rate limit policies configured in policy maps

To display information about rate limit policy maps, enter the following command.

```
Brocade(config-policymap)#show policy-map pmap1
```

Syntax: show policy-map *name*

The *name* variable limits the display to the map specified. If this variable is not used, configuration information is displayed for all policy maps configured on the device.

Output such as the following is displayed.

```
policy-map pmap1
  cir 106656       bps cbs 24000        bytes
  eir 53328        bps ebs 20000        bytes
  excess-priority 2 excess-dscp 43

policy-map pmap2
  cir 106656       bps cbs 24000        bytes
  eir 53328        bps ebs 30000        bytes
  excess-priority 1 excess-dscp 30
```

This display shows the following information.

TABLE 4 Rate limit policy map parameters

This field...	Displays...
policy-map	The name of the policy map for which the configuration is being displayed
cir	The value of the CIR configured for this policy map. Possible values are: 1 - 100000000000 bps.
cbs	Value of the CBS configured for this policy map. Possible values are: 1250 - 12500000000 bytes.
eir	Value of the EIR configured for this policy map. Possible values are: 1 - 100000000000 bps.
ebs	Value of the EBS configured for this policy map. Possible values are: 1250 - 12500000000 bytes.

Rate limiting BUM packets

To prevent the CPU from being flooded by the broadcast, unknown-unicast, and multicast (BUM) packets or bytes, you can restrict the number of BUM packets received on a port. If a high rate of BUM traffic is received by the device on a given port, you can configure the per-port ingress rate limit for the BUM traffic. When you configure a BUM rate limit, the device accepts the maximum configured number of packets and drops additional BUM packets. A port can have BUM rate limits independent of its interface module. The BUM packets entering the device are rate limited before being replicated.

When the received BUM traffic exceeds the pre-defined rate limit, you can close or shut down the physical port using the **shutdown** option of the **rate-limit** command. When the port is configured to be shut down and the BUM traffic experiences packet drops due to the BUM rate limit, the port is shut down automatically. The port shut down occurs within 2.5 seconds after the BUM traffic exceeds the defined limit. The port can be enabled again using the **no** form of the **rate-limit** command.

Limitations of the BUM rate limit

The configuration of the BUM rate limit has the following limitations:

- The BUM rate limit applies only to ingress ports.
- The per-port rate limit cannot be configured individually per traffic type of the broadcast, unknown unicast, or multicast traffic.
- The port shutdown cannot be configured per traffic type.
- The BUM rate limit does not drop packets based on the priorities.

- When the port reaches the configured rate limit, the device will check if the **shutdown** option is enabled for the port.
- The device counts the rate of BUM packets received on the port, for every port configured for shutdown.
- A single drop counter moves over each port to check for the **shutdown** option in a round robin fashion.
- If the drop counter finds the BUM packets dropped on a port, the port will be shut down until the port is explicitly enabled.
- Global BUM rate limiting is not supported on non-default ESI configured interfaces.
- The rate-limit is configured to count for a minimum of 10 milliseconds (ms) for a 1 GbE port and 1 ms for a 10 GbE port.
- The granularity of the rate limit is 51200 bits for 1 Gigabit per second (Gbps) port and 512000 bits for 10 GbE port.
- Due to hardware limitations, 10G ports with BUM rate-limit can shutdown with a rate lower than the actual configured shutdown threshold rate. When the minimum rate of 512Kbit/sec is configured, 10K bit/sec will be rate limited. With a large shutdown threshold rate 2G bit/sec is configured, 1.5G bit/sec will be rate limited. This limitation does not affect 1G ports.

Configuring per-port rate limiting for BUM traffic

For example, to configure the rate limit on BUM traffic packets to a million bits per second, on port 1/1, enter the following command.

```
Brocade(config-if-e1000-1/1)# rate-limit input broadcast unknown-unicast
multicast 1000000 shutdown
```

Syntax: [**no**] **rate-limit input** [**broadcast** | **unknown-unicast** | **multicast**] [*average-rate* [**shutdown**]]

The **input** parameter applies the rate limiting policy to traffic on inbound ports.

The **broadcast**, **unknown-unicast**, and **multicast** parameters define a rate limit for ingress broadcast, unknown-unicast, and multicast packets on the port. Any combination of these parameters can be used to define the rate limit.

The *average-rate* variable specifies the maximum number of bits a port is allowed to receive during a one-second interval and is the aggregate sum of the broadcast, unknown-unicast, and multicast packets rate limit, if the rate limit is configured for all three packets. The software automatically adjusts the number you enter to the nearest multiple of 8,144 bits per second (bps).

The **shutdown** option specifies that the port is to be shut down if the amount of BUM traffic exceeds the pre-defined limit.

When the user tries to add or modify an existing BUM rate limiting policy, the following error message is displayed.

```
Error: There is already a rate limit policy applied on the port.
```

When the BUM traffic exceeds the defined rate limit, port 1/1 is shut down and the reason for the shutdown is displayed in the output of the **show interface** command.

```
Brocade# show interface ethernet 1/1
GigabitEthernet1/1 is down (rate-limit BUM), line protocol is down
STP Root Guard is disabled, STP BPDU Guard is disabled
```

The following syslog message is displayed with the port shutdown information in the output to the **show log** command.

```
Brocade# show log
Nov  4 23:07:52:I:BUM rate-limit is shutting down port 0 on PPCR 0
Nov  4 23:07:52:I:System: Interface ethernet 1/1, state down - shut down by
rate-limiting broadcast, unknown unicast & multicast
```

To enable the shutdown port, delete the previous rate limit by entering the **clear rate-limit bum interface slot/port** command.

```
Brocade(config-if-e1000-1/1)# clear rate-limit bum interface 1/1
```

NOTE
If the user binds different types of rate limiting, such as access group, ACL-based, and BUM rate limits to an interface, the lowest rate limit is configured on the interface.

Displaying BUM rate limit information

You can use show commands to display the following information about the BUM rate limits:

- Accounting information for the BUM rate limit
- BUM rate limit policies per interface

Displaying accounting information for the BUM rate limit

To display the accounting information for the BUM rate limit, enter the following command.

```
Brocade# show rate-limit counters bum-drop
interface 1/1 to 1/24       Drop: 560656640 bytes
interface 1/25 to 1/48      Drop: 212962201728 bytes
interface 2/1 to 2/2        Drop: 148174664000 bytes
```

Syntax: show rate-limit counters bum-drop

Table 5 describes the output parameters of the **show rate-limit counters bum-drop** command.

TABLE 5 Output parameters of the **show rate-limit counters bum-drop** command

Field	Description
interface	Shows the interface for which the accounting information is displayed.
Drop	Shows the BUM traffic (in bytes) that has been dropped as a result of the defined rate limit policy.

Displaying BUM rate limit policies per interface

To display the BUM rate limit policies that are configured per interface, enter the following command.

```
Brocade# show rate-limit
interface e 1/2
rate-limit input broadcast unknown-unicast multicast 972800
interface e 1/12
rate-limit input broadcast multicast 102400 shutdown
```

Syntax: show rate-limit

Table 6 describes the output parameters of the **show rate-limit** command.

TABLE 6 Output parameters of the **show rate-limit** command

Field	Description
interface	Shows the interface for which the BUM rate limit policy information is displayed.
rate-limit input broadcast unknown-unicast multicast	Shows the average rate configured for the inbound broadcast, unknown-unicast, and multicast traffic on the interface.
rate-limit input broadcast multicast	Shows the average rate and the port shutdown option configured for inbound broadcast and multicast traffic on the interface.

Clearing accounting information for the BUM rate limit

To clear the accounting information for the BUM rate limit, enter the following command.

```
Brocade# clear rate-limit counters bum-drop
```

Syntax: clear rate-limit counters bum-drop

Chapter 2

Configuring Traffic Policing for the Brocade NetIron XMR and Brocade MLX series

Traffic policing on the Brocade device

The Brocade device provides line-rate traffic policing in hardware on inbound ports and outbound ports.

You can configure a Brocade device to use one of the following modes of traffic policing policies:

- **Port-based** – Limits the rate on an individual physical port to a specified rate. Only one inbound and one outbound port-based traffic policing policy can be applied to a port. (Refer to "Configuring port-based traffic policing for inbound and outbound ports" on page 20.) These policies can be applied to inbound and outbound traffic.

  NOTE
  The MLX series does not support BUM rate limiting on a per port level.

- **Port-and-priority-based** – Limits the rate on an individual hardware forwarding queue on an individual physical port. Only one port-and-priority-based traffic policing policy can be specified per priority queue for a port. (Refer to "Configuring a port and priority-based traffic policing policy for inbound and outbound ports" on page 21.) These policies can be applied to inbound and outbound traffic.
- **VLAN-based** – Untagged packets as well as tagged packets can be rate-limited. Only one rate can be specified for each VLAN. (Refer to "Configuring a VLAN-based traffic policing policy" on page 21.) Up to 990 VLAN-based policies can be configured for a port under normal conditions or 3960 policies if priority-based traffic policing is disabled as described in "Configuring for no priority-based traffic policing" on page 25. These policies can be applied to inbound and outbound traffic.
- **VLAN group based** – Limits the traffic for a group of VLANs. Members of a VLAN group share the specified bandwidth defined in the traffic policing policy that has been applied to that group. (Refer to "Configuring a VLAN group-based traffic policing policy" on page 22.) Up to 990 VLAN Group-based policies can be configured for a port under normal conditions or 3960 policies if priority-based traffic policing is disabled as described in "Configuring for no priority-based traffic policing" on page 25. These policies can only be applied to inbound traffic.

  NOTE
  If a VLAN based policing is configured on a port for a particular VLAN, the policing will be applicable to all ports on that Network Processor that belong to that VLAN.

- **Port-and-ACL-based** – Limits the rate of IP traffic on an individual physical port that matches the permit conditions in IP Access Control Lists (ACLs). Layer 2 ACL-based traffic policing is supported. You can use standard or extended IP ACLs. Standard IP ACLs match traffic based on source IP address information. Extended ACLs match traffic based on source and destination IP address and IP protocol information. Extended ACLs for TCP and UDP also match

on source and destination TCP or UDP addresses. and protocol information. These policies can be applied to inbound and outbound traffic. Up to 990 Port-and-ACL-based policies can be configured for a port under normal conditions or 3960 policies if priority-based traffic policing is disabled as described in "Configuring for no priority-based traffic policing" on page 25.

- **Rate Limiting for Copied-CPU-bound Traffic** – You can limit the rate of Copied-CPU-bound packets from applications such as sFlow, ACL logging, RPF logging, and source MAC address learning (with known destination address). Copied-CPU-bound packets are handled and queued separately from packets destined to the CPU such as protocol packets and using this feature they can be assigned to one of eight priority queues which has a rate limit assigned to it. The queue and rate are assigned by port and apply to all of the ports that are supported by the same packet processor. Table 7 describes the ports that are associated a packet processor.

Multi-Service IronWare supports applying traffic policing parameters directly to a port or creating a policy map to define a set of traffic policing parameters and then applying that policy map to one or more ports. In addition, the traffic policing parameters available from each of these options are different. The parameters used when applying traffic policing parameters directly to a port reflect the Multi-Service IronWare features that were available before this release. These parameters and the information required to use them are described in "Applying traffic policing parameters directly to a port" on page 16.

The parameters used when applying traffic policing through use of a policy map reflect the traffic policing features that have been added with this release. These parameters and the information required to use them are described in "Applying traffic policing parameters using a policy map" on page 17.

Applying traffic policing parameters directly to a port

When applying a traffic policing policy directly to a port, there are specific parameters that are applied to implement the policy that are different than those used when using a policy map. The Brocade NetIron XMR supports this mode in addition to policy maps. Using this method, a traffic policing policy specifies two parameters: average rate and maximum burst. These parameters are used to configure credits and credit totals.

Average rate

The *average rate* is the maximum number of bits a port is allowed to receive during a one-second interval. The rate of the traffic that matches the traffic policing policy will not exceed the average rate.

The *average rate* represents a percentage of an interface's line rate (bandwidth), expressed in bits per second (bps). It cannot be smaller than 8,144 bits per second (bps) and it cannot be larger than the port's line rate.

For Brocade MLX series and Brocade NetIron XMR devices, the *average rate* must be entered in multiples of 8,144 bps. If you enter a number that is not a multiple of 8,144, the software adjusts the rate down to the lowest multiple of the number so that the calculation of credits does not result in a remainder of a partial Credit. For example, if you enter 10,000 bps, the value will be adjusted to 8,144 bps. The adjusted rate is sometimes called the *adjusted average rate*.

For Brocade NetIron CER and Brocade NetIron CES devices, the *average rate* can be entered in as any value from 0 up to the line rate of the port. Multiples of 8,144 do not need to be used.

Maximum burst

Maximum burst provides a higher than average rate to traffic that meet the rate limiting criteria. Traffic will be allowed to pass through the port for a short period of time. The unused bandwidth can be accumulated up to a maximum of “maximum burst” value expressed in bits.

Credits and credit total

Each rate limiting policy is assigned a class. A *class* uses the average rate and maximum burst in the rate limit policy to calculate credits and credit totals.

Credit size is measured in bytes. A credit is a forwarding allowance for a traffic policed port, and is the smallest number of bytes that can be allowed during a rate limiting interval. Minimum credit size can be 1 byte.

During a rate limiting interval, a port can send or receive only as many bytes as the port has Credits for. For example, if an inbound rate limiting policy results in a port receiving two credits per rate limiting interval, the port can send or receive a maximum of 2 bytes of data during that interval.

In each interval, the number of bytes equal to the credit size is added to the running total of the class. The running total of a class represents the number of bytes that can be allowed to pass through without being subject to rate limiting.

The second parameter is the maximum *credit total*, which is also measured in bytes. The maximum credit total is based on the maximum burst value and is also measured in bytes.

The running total can never exceed the maximum credit total. When packets arrive at the port, a class is assigned to the packet based on the traffic policing policies. If the running total of the class is less than the size of the packet, then the packet is dropped. Otherwise, the size of the packet is subtracted from the running total and the packet is forwarded. If there is no traffic that matches traffic policing criteria, then the running total can grow up to the maximum credit total.

Applying traffic policing parameters using a policy map

When using the traffic policing policies available from previous versions, the policy parameters are provided explicitly for each port during port configuration. In this version, the policies must be defined using a policy map. The policy map configuration ties a policy name to a set of traffic policing policies. The policy name is then applied to the port or ports that you want to rate limit using the defined policy. This allows you to set a policy in a single location the affects multiple ports and to make changes to that policy. Configuration of a policy map is described in “Configuring a policy map” on page 19.

Within the policy map configuration, the parameters used to define traffic policing have been changed. When configuring traffic policing within a policy map, these new parameters apply. With this release, traffic policing policy determines the rate of inbound or outbound traffic (in bits per second or bps) that is allowed per port. This traffic is initially traffic policed by a Committed Information Rate (CIR) bucket. Traffic that is not accommodated in the CIR bucket is then subject to the Excess Information Rate (EIR) bucket.

The CIR bucket

The CIR rate limiting bucket is defined by two separate parameters: the CIR rate, and the Committed Burst Size (CBS) rate. The CIR rate is the maximum number of bits a port is allowed to receive or send during a one-second interval. The rate of the traffic that matches the traffic policing policy can not exceed the CIR rate. The CIR rate represents a portion of an interface's line rate (bandwidth), expressed in bits per second (bps) and it cannot be larger than the port's line rate. CIR-defined traffic that does not use the CIR rate available to it accumulates credits that it can use later in circumstances where it temporarily exceeds the CIR rate.

When traffic exceeds the bandwidth that has been reserved for it by the CIR rate defined in its policy, it becomes subject to the CBS rate. The CBS rate provides a rate higher than the CIR rate to traffic that exceeded its CIR rate. The bandwidth in the CBS rate, as expressed in bytes, is accumulated during periods of time when traffic that has been defined by a policy does not use the full CIR rate available to it. Traffic is allowed to pass through the port for a short period of time at the CBS rate.

When inbound or outbound traffic exceeds the bandwidth available for the defined CIR and CBS rates, it is either dropped, or made subject to the conditions set in it EIR bucket.

The EIR bucket

The EIR bucket provides an option for traffic that has exceeded the conditions set by policy for the CIR bucket. In the EIR bucket, there are two parameters that define the traffic that is available: the Excess Information Rate (EIR) and the Excess Burst Size (EBS) rate. The EIR and EBS operate exactly like the CIR and CBS except that they only act upon traffic that has been passed to the EIR bucket because it could not be accommodated by the CIR bucket. Like the CIR, the EIR provides an initial bandwidth allocation to accommodate inbound and outbound traffic. If the bandwidth provided by the EIR is insufficient to accommodate the excess traffic, the defined EBS rate provides for burst traffic. Like the CBS, the bandwidth available for burst traffic from the EBS is subject to the amount of bandwidth that is accumulated during periods of time when traffic that has been allocated by the EIR policy is not used.

In addition, to providing additional bandwidth for traffic that exceeds that available for the CIR bucket, traffic rate limited by the EIR bucket can have its excess priority and excess dscp values changed. Using this option, priority parameters are set following the EBS value that change the priority of traffic that is being rate limited using the EIR bucket.

Configuration considerations

- Only one type of traffic policing policy can be applied on a physical port. For example, you cannot apply port-and-ACL-based and port-based traffic policing policies on the same port.
- When a VLAN-based traffic policing policy is applied to a port, all the ports controlled by the same packet processor are rate limited for that VLAN. You cannot apply a VLAN-based traffic policing policy on another port of the same packet processor for the same VLAN ID.
- The Multi-Service IronWare software supports VLAN-based traffic policing that can limit tagged and untagged packets that match the VLAN ID specified in the policy. Untagged packets are not subject to traffic policing.
- The maximum burst in a traffic policing policy cannot be less than the average rate and cannot be more than the port's line rate.
- Control packets are not subject to traffic policing.
- Source MAC address with Virtual Leased Line (VLL) endpoints are not subject to traffic policing.

Configuring traffic policing on Brocade devices

The following sections show examples of how to configure each traffic policing policy type.

Configuring a policy map

To configure a policy map, enter a command such as the following.

```
Brocade(config)#policy-map map5
Brocade(config-policymap map5)#cir 1000000 cbs 2000000 eir 1000000 ebs 2000000
excess-dp 2 excess-dscp 37
```

The command configures the traffic policing policy map map1 to limit CIR rate to 1000000 the CBS rate to 2000000, the EIR rate to 1000000 and the EBS to 2000000. In addition, traffic that exceeds the bandwidth available in the CIR bucket will have its packets drop precedence set to 2 and its DSCP set to 37. This command only creates a policy, it must be applied to one or more ports to be operational.

Syntax: [no] **policy-map** *map-name* **cir** *cir-rate* **cbs** *cbs-rate* [**eir** *eir-rate* **ebs** *ebs-rate* **excess-priority** *priority-num* [**excess-dscp** *dscp-num*]] | [**eir** *eir-rate* **ebs** *ebs-rate* **excess-dp** *dp-val* [**excess-dscp** *dscp-num*]]

The *map-name* variable is the name you will use to reference the policy map in traffic policing command. It can be a character string up to 64 characters long.

The **cir** parameter defines the value of the Committed Information Rate (CIR) as the rate defined in the *cir-rate* variable. Acceptable values are: 0 - 10000000000 bits per second (bps) in increments of 8,144 bps.

The **cbs** parameter defines the value of the Committed Burst Size (CBS) as the rate defined in the *cbs-rate* variable. Acceptable values are: 1250 - 1250000000 bytes in increments of 1 byte.

The **eir** parameter defines the value of the Excess Information Rate (EIR) as the rate defined in the *eir-rate* variable. Acceptable values are: 0 - 10000000000 bits per second (bps) in increments of 8,144 bps.

The **ebs** parameter defines the value of the Excess Burst Size (EBS) as the rate defined in the *ebs-rate* variable. Acceptable values are: 1250 - 1250000000 bytes in increments of 1 byte.

The **excess-priority** parameter specifies that traffic whose bandwidth requirements exceeds what is available in the CIR bucket and is sent to the EIR bucket will have its packets priority queue set to the value set in the *priority-num* variable. Acceptable values for the *priority-num* are 0-7.

The **excess-dp** parameter specifies the WRED drop precedence for traffic whose bandwidth requirements exceed what is available in the CIR bucket and is sent to the EIR bucket. Acceptable values for the *dp-value* are 0-3. Packets with a value of 0 are least likely to be dropped and packets with a value of 3 are most likely to be dropped.

The **excess-dp** parameter is compared with the **excess-dscp** parameter in bit [2:1] first, then it is converted.

The **excess-dscp** parameter specifies that traffic whose bandwidth requirements exceeds what is available in the CIR bucket and is sent to the EIR bucket will have its packets DSCP priority set to the value set in the *dscp-num* variable. Acceptable values for the *dscp-num* are 0-63. When this parameter is used together with the **excess-dp** parameter, the value set for bits 2:1 (zero-based) in the **excess-dscp** parameter must be equal to the value set for **excess-dp**.

Configuring port-based traffic policing for inbound and outbound ports

Port-based traffic policing limits the rate on an individual inbound or outbound physical port to a specified rate.

To configure port-based traffic policing policy for outbound ports, enter commands such as the following at the interface level.

```
Brocade(config)# interface ethernet 1/1
Brocade(config-if-1/1)# rate-limit out 500000000 250000000
```

The commands configure a traffic policing policy for outbound traffic on port 1/1. The policy limits the average rate of all outbound traffic to 500000000 bits per second (bps) with a maximum burst size of 250000000 bits per second (bps).

To configure port based traffic policing policy through a policy map, enter a command such as the following.

```
Brocade(config)# interface ethernet 1/1
Brocade(config-if-1/1)# rate-limit input policy-map map1
```

The commands configure a traffic policing policy for inbound traffic on port 1/1. The policy references the policy map map1 for rate limiting policy parameters.

The complete syntax for configuring a port-based traffic policing policy is:

Syntax: [no] rate-limit [in | out] [*average-rate maximum-burst* **| policy-map** *map-name***]**

The **input** parameter applies the policy to traffic on inbound ports.

The **output** parameter applies the policy to traffic on outbound ports.

Only one inbound and one outbound port-based traffic policing policy can be applied to a port.

The *average-rate* parameter specifies the maximum rate allowed on a port during a one-second interval. The software automatically adjusts the number you enter to the lower multiple of 8,144 bits per second (bps). Refer to the section "Average rate" on page 16 for more details. This command is only used when configuring traffic policing directly to a port as described in "Applying traffic policing parameters directly to a port" on page 16.

The *maximum-burst* parameter specifies the extra bits above the average-rate that traffic can have. Refer to the section "Maximum burst" on page 17 for more details. This command is only used when configuring traffic policing directly to a port as described in "Applying traffic policing parameters directly to a port" on page 16.

The **policy-map** parameter specifies the policy map named in the *policy-map* variable to be used to provide parameters for rate limiting the port and VLAN specified. This command is only used when configuring traffic policing to a port using a policy map as described in "Applying traffic policing parameters using a policy map" on page 17.

Policy change remarks

When a rate-limit policy is changed an automated remark is generated. The following examples show the remarks displayed for adding or removing a rate-limit policy.

```
Brocade(config-if-e1000-1/12)#no rate-limit input access-group 102 policy-map
abc2
Delete Existing Remark Profile to table 2 with used count 0

Brocade(config-if-e1000-1/12)#rate-limit input access-group 102 policy-map abc2
Add New Remark Profile to table 2 with used count 1
```

Configuring a port and priority-based traffic policing policy for inbound and outbound ports

To configure port based traffic policing policy directly, enter a command such as the following.

```
Brocade(config)# interface ethernet 1/1
Brocade(config-if-1/1)# rate-limit input priority q1 500000000 33553920
```

The commands configure a traffic policing policy for inbound traffic on port 1/1. The policy limits the average rate of all inbound traffic to 500000000 bits per second (bps) with a maximum burst size of 33553920 bits per second (bps) for packets with their priority queue set to 1.

Syntax: **[no] rate-limit [input | output] priority** *queue-num* [*average-rate maximum-burst* | **policy-map** *map-name*]

The **input** parameter applies the policy to traffic on inbound ports.

The **output** parameter applies the policy to traffic on outbound ports.

Only one port-based traffic policing policy can be applied to a port.

The **priority** parameter specifies the internal queue in the *queue-num* variable which is rate limited by this command.

The priority queues for rate limiting internal queue mapping are shown in the following example.

```
Brocade(config-if-e10000-1/1)#rate-limit input priority
  q0                    priority queue 0 (internal priority 0 and 1)
  q1                    priority queue 1 (internal priority 2 and 3)
  q2                    priority queue 2 (internal priority 4 and 5)
  q3                    priority queue 3 (internal priority 6 and 7)
Brocade(config-if-e10000-1/1)#rate-limit input priority
```

The *average-rate* parameter specifies the maximum rate allowed on a port during a one-second interval. The software automatically adjusts the number you enter to the nearest multiple of 8,144 bits per second (bps). Refer to the section "Average rate" on page 16 for more details. This command is only used when configuring rate limiting directly to a port as described in "Applying traffic policing parameters directly to a port" on page 16.

The *maximum-burst* parameter specifies the extra bits above the average-rate that traffic can have. Refer to the section "Maximum burst" on page 17 for more details. This command is only used when configuring rate limiting directly to a port as described in "Applying traffic policing parameters directly to a port" on page 16.

The **policy-map** parameter specifies the policy map named in the *policy-map* variable to be used to provide parameters for rate limiting the port. This command is only used when configuring rate limiting to a port using a policy map as described in "Applying traffic policing parameters using a policy map" on page 17.

Configuring a VLAN-based traffic policing policy

To configure a port-and-VLAN based traffic policing policy, enter commands such as the following.

```
Brocade(config)# interface ethernet 1/1
Brocade(config-if-1/1)# rate-limit input vlan 10 500000000 33553920

Brocade(config)# interface ethernet 1/2
Brocade(config-if-1/2)# rate-limit output vlan 20 policy-map map1
```

These commands configure two traffic policing policies that limit the average rate of all inbound traffic on port 1/1 with VLAN tag 10 and all outbound traffic on port 1/2 VLAN tag 20. The first policy limits packets with VLAN tag 10 to an average rate of 500000000 bits per second (bps) with a maximum burst size of 33553920 bits on port 1/1. The second policy limits packets with VLAN tag 20 to values defined in policy map map1. Tagged packets belonging to VLANs other than 10 and 20 and untagged packets are not subject to traffic policing on these ports.

Syntax: [no] **rate-limit** [**input** | **output**] [**priority** *queue-num*] **vlan-id** *vlan-num* [*average-rate maximum-burst* | **policy-map** *map-name*]

The **input** parameter applies the policy to traffic on inbound ports.

The **output** parameter applies the policy to traffic on outbound ports.

The **priority** parameter specifies an 802.1p value in the *queue-num* variable that is used to identify packets that will be rate limited by this command. This parameter is optional.

The vlan-id *vlan-number* parameter species the VLAN ID to which the policy applies. You can specify up to 990 priority or 3960 non-priority VLAN-based traffic policing policies on a port.

The *average-rate* parameter specifies the maximum rate allowed on a port during a one-second interval. The software automatically adjusts the number you enter to the nearest multiple of 8,144 bits per second (bps). Refer to the section "Average rate" on page 16 for more details. This command is only used when configuring traffic policing directly to a port as described in "Applying traffic policing parameters directly to a port" on page 16.

The *maximum-burst* parameter specifies the extra bits above the average-rate that traffic can have. Refer to the section "Maximum burst" on page 17 for more details. This command is only used when configuring traffic policing directly to a port as described in "Applying traffic policing parameters directly to a port" on page 16.

The **policy-map** parameter specifies the policy map named in the *policy-map* variable to be used to provide parameters for traffic policing the port and VLAN specified. This command is only used when configuring traffic policing to a port using a policy map as described in "Applying traffic policing parameters using a policy map" on page 17.

Configuring a VLAN group-based traffic policing policy

A traffic policing policy can be applied to a VLAN group. VLANs that are members of a VLAN group share the specified bandwidth defined in the traffic policing policy applied to that group.

To configure a traffic policing policy for a VLAN group, perform the following tasks.

1. Define the VLANs that you want to place in a traffic policing VLAN group.
2. Define a rate limiting VLAN group. This VLAN group is specific to the traffic policing feature. Enter commands such as the following.

```
Brocade(config)# rl-vlan-group 10
Brocade(config-vlan-rate-group)# vlan 3 5 to 7 10
```

The commands assign VLANs 3, 5,6, 7, and 10 to traffic policing VLAN group 10.

Syntax: [no] **rl-vlan-group** *vlan-group-number*

Syntax: [no] **vlan** *vlan-number* [**to vlan-number**]

The **rl-vlan-group** command takes you to the VLAN group traffic policing level. Enter the ID of the VLAN group that you want to create or update by entering a value for *vlan-group-number*.

Use the vlan command to assign or remove VLANs to the rate limiting VLAN group. You can enter the individual VLAN IDs or a range of VLAN IDs.

3. Create a policy for the VLAN group and apply it to the interface you want. Enter commands such as the following.

```
Brocade(config)# interface ethernet 1/1
Brocade(config-if-1/1)# rate-limit input group 10 500000000 33553920
```

These commands configure a traffic policing policy that limits the average rate of all inbound traffic on port 1/1 from vlan group VlanGroupA. This policy limits packets from VlanGroupA to an average rate of 500000000 bits per second (bps) with a maximum burst size of 33553920 bits on port 1/1. VLAN Group based traffic policing is only available for inbound ports.

Syntax: [no] **rate-limit input group** *vlan-group-id* [**priority** *queue-num*] [*average-rate maximum-burst* | **policy-map** *map-name*]

The **input** parameter applies the policy to traffic on inbound ports.

The **priority** parameter specifies an 802.1p value in the *queue-num* variable that is used to identify packets that will be traffic policed by this command. This parameter is optional.

The *vlan-group-id* parameter species the VLAN Group ID to which the policy applies.

The *average-rate* parameter specifies the maximum rate allowed on a port during a one-second interval. The software automatically adjusts the number you enter to the nearest multiple of 8,144 bits per second (bps). Refer to the section "Average rate" on page 16 for more details. This command is only used when configuring rate limiting directly to a port as described in "Applying traffic policing parameters directly to a port" on page 16.

The *maximum-burst* parameter specifies the extra bits above the average-rate that traffic can have. Refer to the section "Maximum burst" on page 17 for more details. This command is only used when configuring rate limiting directly to a port as described in "Applying traffic policing parameters directly to a port" on page 16.

The **policy-map** parameter specifies the policy map named in the *policy-map* variable to be used to provide parameters for rate limiting the VLAN specified. This command is only used when configuring traffic policing to a port using a policy map as described in "Applying traffic policing parameters using a policy map" on page 17.

Configuring a port and ACL-based rate limiting

You can use standard or extended IP ACLs for port-and-ACL-based rate limiting:

- Standard IP ACLs match traffic based on source IP address information.
- Extended ACLs match traffic based on source and destination IP addresses and IP protocol information. Extended ACLs for TCP and UDP protocol must also match on source and destination IP addresses and TCP or UDP protocol information.
- You can bind multiple rate limiting policies to a single port. However, once a matching ACL clause is found for a packet, the device does not evaluate subsequent clauses in that rate limiting ACL and subsequent rate limiting ACLs.
- You can apply an ACL ID to a port-and-ACL-based traffic policing policy even before you define the ACL. The traffic policing policy does not take effect until the ACL is defined.
- It is not necessary to remove an ACL from a port-and-ACL-based rate limiting policy before deleting the ACL.
- Layer 2 ACL rate limiting is supported.

Port-and-ACL-based traffic policing is supported for traffic on inbound and outbound ports. To configure port-and-ACL-based traffic policing policies, enter commands such as the following.

```
Brocade(config)#access-list 50 permit host 1.1.1.2
Brocade(config)#access-list 50 deny host 1.1.1.3
Brocade(config)#access-list 60 permit host 2.2.2.3
Brocade(config-if-1/1)# rate-limit input access-group 50 priority q1 500000000
33553920
Brocade(config-if-1/1)# rate-limit input access-group 60 100000000 268431230
```

These commands first configure access-list groups that contain the ACLs that will be used in the traffic policing policy. Use the **permit** condition for traffic that will be traffic policed. Traffic that match the **deny** condition are not subject to traffic policing.

Next, the commands configure two traffic policing policies on port 1/1. The policies limit the average rate of all inbound IP traffic that match the permit rules of ACLs 50 and 60. The first policy limits the rate of all permitted IP traffic with a priority queue value of q1 from host 1.1.1.2 to an average rate of 500000000 bits per second (bps) with a maximum burst size of 33553920 bits. Rate of all traffic from host 1.1.1.3 is not subject to rate limiting since it is denied by ACL 50; it is merely forwarded on the port.

The second policy limits the rate of all IP traffic from host 2.2.2.3 to an average rate of 100000000 bits per second (bps) with a maximum burst size of 268431230 bits.

All IP traffic that does not match ACLs 50 and 60 are not subject to traffic policing.

Syntax: **[no] rate-limit [input | output] [vrf** *vrf-name*] **access-group** *group-number* **[priority** *queue-num*] **[** *average-rate maximum-burst* **| policy-map** *map-name*]

The **input** parameter applies the policy to traffic on inbound ports.

The **output** parameter applies the policy to traffic on outbound ports.

The **VRF** parameter specifies that the access-group will only apply to traffic within the VRF whose name is specified in the *vrf-name* variable. This feature is only supported on inbound traffic with Layer-3 ACLs.

The **access-group**, *group-number* parameter specifies the group number to which the ACLs used in the policy belong.

NOTE
An ACL must exist in the configuration before it can take effect in a traffic policing policy.

The **priority** parameter specifies a priority queue value in the *queue-num* variable that is used to identify packets that will be traffic policed by this command. The possible values for this parameter are: q0, q1, q2, or q3. Multiple queues can be specified. This parameter is optional.

The *average-rate* parameter specifies the maximum rate allowed on a port during a one-second interval. The software automatically adjusts the number you enter to the nearest multiple of 8,144 bits per second (bps). Refer to the section "Average rate" on page 16 for more details. This command is only used when configuring rate limiting directly to a port as described in "Applying traffic policing parameters directly to a port" on page 16.

The *maximum-burst* parameter specifies the extra bits above the average-rate that traffic can have. Refer to the section "Maximum burst" on page 17 for more details. This command is only used when configuring traffic policing directly to a port as described in "Applying traffic policing parameters directly to a port" on page 16.

The **policy-map** parameter specifies the policy map named in the *policy-map* variable to be used to provide parameters for traffic policing the VLAN specified. This command is only used when configuring traffic policing to a port using a policy map as described in "Applying traffic policing parameters using a policy map" on page 17.

Using ACLs for filtering in addition to rate limiting

When you use the ACL-based mode, the permit and deny conditions in an ACL you use in a rate limiting policy work as follows:

- **Permit** – The traffic is rate limited according to the other parameters in the rate limiting policy.
- **Deny** – The traffic is forwarded instead of dropped, by default.

You can configure the device to drop traffic that is denied by the ACL instead of forwarding the traffic, on an individual port basis.

NOTE
Once you configure an ACL-based rate limiting policy on a port, you cannot configure a regular (traffic filtering) ACL on the same port. To filter traffic, you must enable the strict ACL option.

To configure the device to drop traffic that is denied by a rate limiting ACL, enter the following command at the configuration level for the port.

```
Brocade(config-if-1/1)# rate-limit strict-acl
```

Syntax: [no] rate-limit strict-acl

Configuring for no priority-based traffic policing

By default, up to 990 different traffic policing policies can be applied to a single 10 GB Ethernet port. This combined with the 4 priorities utilizes 3960 rate limiting classes. You can configure a system-wide policy so that up to 3960 individual traffic policing policies can be applied to a single 10 GB Ethernet port.

To configure a Brocade device to not allow priority-based traffic policing, enter commands such as the following at the interface level.

```
Brocade(config)# qos-policy
Brocade(qos-policy)# no rate-limit internal-priority-based
```

Syntax: [no] rate-limit internal-priority-based

If this command is implemented, the number of different rate limiting policies that can be applied to a single port is increased from 990 to 3960.

Configuring rate limiting for Copied-CPU-bound traffic

A new feature was added that allows you to limit the rate of Copied-CPU-bound packets from applications such as sFlow, ACL logging, RPF logging, and source MAC address learning (with known destination address). This feature can be configured as described in this section

The following command assigns a rate limit of 200,000,000 bits per second (bps) and a priority queue of 0 to copied-CPU-bound incoming traffic on PPCR 1 though its assignment on port 3/2.

```
Brocade(config)# rl-cpu-copy 0 200000000 ethernet 3/2
```

Syntax: rl-cpu-copy *priority-number limit-rate* **ethernet** *slot/port* **[to ethernet** *slot/port***]**

The *priority-number* variable specifies the CPU-bound traffic priority queue to apply the rate limiting. This can be a value from 0 to 7.

The *limit-rate* variable specifies the limiting rate for the specified CPU-bound traffic priority queue. Acceptable values are from 1 to 300000000 bits per second (bps). The default rate for all is 300,000,000 bps.

The *slot/port* variable specifies the port that you want to apply copied-CPU-bound rate limiting to. You can apply the command to a range of ports using the **to ethernet** *slot/port* option. When you assign a port, the command applies to all ports that are associated with the same packet processor (PPCR). Table 7 describes the ports that are associated a packet processor.

TABLE 7 Ports per packet processor

Interface module	Ports Per Packet Processor (PPCR)	
	PPCR1	PPCR2
4 X 10 Gbps	1 - 2	3 - 4
20 X 1 Gbps	1 - 20	

You can display the **rl-cpu-copy** configuration displayed previously using the following command.

```
Brocade# show rl-cpu-copy
Rate shaping configuration on CPU Copy priority queues
priority 0 200000000 ethernet 3/1 to 3/20
```

Notice that although the command was only executed for port 3/2, it applies to all the ports attached to the same PPCR. In this case ports 3/1 to 3/20.

Syntax: show rl-cpu-copy

Displaying rate limiting policies

Use one of the following commands to view the rate limiting policies that have been configured:

- **show rate limit counters** – Displays accounting information for rate limit usage.
- **show rate limit group** – Displays the VLANs that are in the specified group.
- **show rate limit** – Displays rate limiting policies implemented per interface.
- **show policy map** – Displays rate limiting policies implemented in the configured policy maps.

You can configure a Brocade device to exclude the 20-byte per-packet Ethernet overhead from Traffic Policing byte accounting. This can be done by configuring the **vlan-counter exclude-overhead** command.

Displaying accounting information for rate limit usage

To display accounting information for rate limit usage, enter the following command.

```
Brocade# show rate-limit counters
```

Syntax: show rate-limit counters [interface slot/port]

The **interface slot/port** option allows you to get accounting information for a specified interface only.

Output such as the following will display.

```
Brocade# show rate-limit counters
interface e 2/1
 rate-limit input access-group 400 999993616 1000000000
  Fwd:        0                      Drop:  0 bytes
  Re-mark:    0                      Total: 0 bytes
```

This display shows the following information.

TABLE 8 Rate limit counters parameters

This field...	Displays...
Interface	The interface that rate limit accounting information is being displayed for.
rate-limit input	A rate limit configuration that defines rate limit policy for inbound traffic on the defined interface.
Fwd	The traffic in bytes that has been forwarded from this interface as a result of this rate limit policy since the device was started up or the counter has been reset.
Drop	The traffic in bytes that has been dropped from this interface as a result of the defined rate limit policy since the device was started up or the counter has been reset.
Re-mark	The number of packets whose priority have been remarked as a result of exceeding the bandwidth available in the CIR bucket for this rate limit policy.
Total	The total traffic in bytes that has been carried on this interface for the defined rate limit policy since the device was started up or the counter has been reset.

Monitoring rate limit usage by SNMP

Accounting information for rate limit usage can also be monitored by SNMP. The agAclAccntTable supports the following objects, which map to the rate limit usage counters.

agAclAccntRaclDropCnt: drop counters

agAclAccntRaclFwdCnt: fwd counters

agAclAccntRaclRemarkCnt: re-mark counters

agAclAccntRaclTotalCnt: total counters

For detailed information, please refer to the “Filtering Traffic” chapter of the *Unified IP MIB Reference.*

Among other applications, this accounting feature allows per-port VLAN statistics in the inbound or outbound direction to be extracted by means of SNMP. This can be achieved by adding ACL filters for the monitored VLAN on the appropriate port. This accounting feature works for all modules of the Brocade NetIron XMR and Brocade MLX series platforms. For modules that do not support extended VLAN statistics, this feature provides a means of extracting per-port VLAN statistics.

Resetting the rate limit counters

You can reset all of the rate limit counters using the following command.

```
Brocade# clear rate-limit counters
```

Syntax: clear rate-limit counters [interface]

The **interface** variable specifies an interface that you want to clear the rate limit counters for. If you do not specify an interface, all rate limit counters on the device will be reset.

Displaying information about rate limit VLAN groups

To display information about rate limit VLAN groups, enter the following command.

```
Brocade# show rate-limit group
```

Syntax: show rate-limit group

Output such as the following will display

```
rl-vlan-group 1
 vlan 10 to 15
```

This display shows the following information.

TABLE 9 Rate limit VLAN group parameters

This field...	Displays...
rl-vlan-group	The VLAN group whose contents are displayed.
vlan	VLANs contained in the VLAN group specified.

Displaying rate limit policies per interface

To display information about rate limit policies that are configured per interface, enter the following command.

```
Brocade# show rate-limit
```

Syntax: show rate-limit

Output such as the following will display.

```
Brocade(config-if-e10000-1/1)#show rate-limit

interface e 1/1
 rate-limit input 959904 2000000
 rate-limit output 2986368 2000000
```

This display shows the following information.

TABLE 10 Rate limit interface parameters

This field...	Displays...
rate-limit input	The average-rate and maximum burst rate configured for inbound traffic on the specified interface.
rate-limit output	The average-rate and maximum burst rate configured for outbound traffic on the specified interface.

Displaying rate limit policies configured in policy maps

To display information about rate limit policy maps, enter the following command.

```
Brocade# show policy-map
```

Syntax: show policy-map [*map-name*]

The *map-name* variable limits the display of policy map configuration information to the map specified. If this variable is not used, configuration information will be displayed for all policy maps configured on the device.

Output such as the following will display.

```
Brocade(config-policymap pmap1)#show policy-map

policy-map pmap1
  cir 106656       bps cbs 24000        bytes
  eir 53328        bps ebs 20000        bytes
  excess-priority 2 excess-dscp 43

policy-map pmap2
  cir 106656       bps cbs 24000        bytes
  eir 53328        bps ebs 30000        bytes
  excess-priority 1 excess-dscp 30
```

This display shows the following information.

TABLE 11 Rate limit policy map parameters

This field...	Displays...
policy-map	The name of the policy map whose configuration is being displayed
cir	The value of the Committed Information Rate (CIR) configured for this policy map. Possible values are: 1 - 10000000000 bps.
cbs	The value of the Committed Burst Size (CBS) configured for this policy map. Possible values are: 1250 - 1250000000 bytes.
eir	The value of the Excess Information Rate (EIR) configured for this policy map. Possible values are: 1 - 10000000000 bps.
ebs	The value of the Excess Burst Size (EBS) configured for this policy map. Possible values are: 1250 - 1250000000 bytes.
excess-priority	The priority queue that traffic whose bandwidth requirements exceeds what is available in the CIR bucket and is sent to the EIR bucket is set to. Possible values are 0-3.
excess-dscp	The priority queue that traffic whose bandwidth requirements exceeds what is available in the CIR bucket and is sent to the EIR bucket is set to. Possible values are 0-63.

IPv6 ACL-based rate limiting

Rate-limiting is supported for both inbound and outbound traffic on an interface. Rate-limiting IPv6 traffic on an interface is performed by classifying traffic using an IPv6 access-list.

IPv6 ACL based rate-limiting configuration considerations

- IPv6 ACL based rate-limiting can be configured only on a physical interface.
- IPv6 ACL based rate-limiting can be configured separately for inbound and outbound traffic on an interface.

- Multiple IPv6 ACL based rate-limiting policies can be applied to a single port.
- Once a matching ACL clause is hit, subsequent rules and subsequent rate-limiting bindings on the interface are not evaluated.
- An undefined ACL can be used in a rate-limiting configuration.
- When “force-delete-bound-acl” is enabled, an ACL can be deleted even if in use by a rate-limiting policy.
- Whenever the rules of an ACL used in a rate-limiting binding is modified, the changes are not reflected immediately. You must execute the IPv6 ACL rebind command for the changes to take effect.
- IPv4 and IPv6 ACL based rate-limiting configurations can co-exist on an interface.

IPv6 ACL based rate-limiting command options

The following sections discuss the configuration sequence and commands in detail. The following is the entire command syntax. Each of the following configuration commands provide a detailed description of the specific command.

Syntax: **[no] rate-limit** {input [vrf *VRF_NAME*] | output} **access-group** {acl_id | name {ipv6 | ipv4 | mac} *ACL_NAME*} [{priority *PRIORITY_QUEUE*}] {*AVERAGE_RATE_BPS* | *MAX_BURST_BPS*} | {policy-map *POLICY_MAP_NAME*}} | *strict-acl*

NOTE
The keywords IPv4 and mac are used to configure rate-limiting using named IPv4 and named L2 ACLs.

Configuration Sequence

The configuration sequence to configure rate limiting using IPv6 access list include the following.

1. Create the IPv6 access list.
2. Create a policy map.
3. Configure rate limiting on an interface for inbound/outbound traffic using the IPv6 access-list.
   a. Configure the average and maximum burst rate-limit parameters.
   b. Use the policy-map to apply the rate-limit parameters.

Create IPv6 access-list (ACL)

IPv6 access-lists are named access-lists. The following example is an access-list that blocks all Telnet traffic received from IPv6 host 2000:2382:e0bb::2.

```
Brocade(config)# ipv6 access-list fdry
Brocade(config-ipv6-access-list-fdry)# deny tcp host 2000:2382:e0bb::2 any eq
telnet
Brocade(config-ipv6-access-list-fdry)# permit ipv6 any any
Brocade(config-ipv6-access-list-fdry)# exit
```

Create Policy-map

The following example configures the traffic policing policy-map map5 to limit CIR rate to 1000000 the CBS rate to 2000000, the EIR rate to 1000000 and the EBS to 2000000.

```
Brocade(config)# policy-map map5
Brocade(config-policymap map5)# cir 1000000 cbs 2000000 eir 1000000 ebs 2000000
excess-dp 2 excess-dscp 37
```

Configure inbound rate-limiting on an interface

Configure average and maximum burst sizes

The command configures rate-limiting for inbound traffic on the interface using the IPv6 access-list “fdry”, the average rate as 1000000 and the maximum burst size as 2000000.

```
Brocade(config-if-1/1)# rate-limit input access-group name ipv6 fdry 1000000
2000000
```

Syntax: **[no] rate-limit** {{input [vrf *VRF_NAME*]} **access-group** {name {ipv6} *ACL_NAME*} [*AVERAGE_RATE_BPS* | *MAX_BURST_BPS*]

Configure using policy-map

The command configures rate-limiting for inbound traffic on the interface using the IPv6 access-list “fdry” and the policy-map “map5”.

```
Brocade(config-if-1/1)# rate-limit input access-group name ipv6 fdry policy-map
map5
```

Syntax: **[no] rate-limit** {input [vrf *VRF_NAME*] } access-group {name {ipv6} *ACL_NAME*} {policy-map *POLICY_MAP_NAME*}

Configure for a specific priority-queue

The command configures rate-limiting for inbound traffic on priority-queue “q0” on the interface using the IPv6 access-list “fdry”, the average rate as 1000000 and the maximum burst size as 2000000.

```
Brocade(config-if-1/1)# rate-limit input access-group name ipv6 fdry priority q0
1000000 2000000
```

Syntax: **[no] rate-limit** {input [vrf *VRF_NAME*] } **access-group** {name {ipv6 } *ACL_NAME*} [{priority *PRIORITY_QUEUE*}] {*AVERAGE_RATE_BPS* | *MAX_BURST_BPS*}

The command configures rate-limiting for inbound traffic on priority-queue “q0” on the interface using the IPv6 access-list “fdry” and the policy-map “map5”.

```
Brocade(config-if-1/1)# rate-limit input access-group name ipv6 fdry priority q0
policy-map map5
```

Configure VRF specific rate-limit

IPv6 access-list based rate-limiting can be configured for a specific VRF. Rate-limiting is applied to the inbound traffic for the interfaces which are part of the configured VRF. The following command configures rate-limiting for inbound traffic on the VRF “data” using the access-list “fdry”.

```
Brocade(config-if-1/1)# rate-limit input vrf data access-group name ipv6 fdry
1000000 2000000
```

Syntax: **[no] rate-limit input** [vrf *VRF_NAME*] **access-group** {name {ipv6} *ACL_NAME}* {*AVERAGE_RATE_BPS* | *MAX_BURST_BPS*}

NOTE
This feature is not supported in Brocade NetIron CES and Brocade NetIron CER.

Configure outbound rate-limiting on an interface

Use the following steps to configure outbound rate-limiting on an interface.

Configure average and maximum burst sizes

The command configures rate-limiting for outbound traffic on the interface using the IPv6 access-list “fdry”, the average rate as 1000000 and the maximum burst size as 2000000.

```
Brocade(config-if-1/1)# rate-limit output access-group name ipv6 fdry 1000000
2000000
```

Syntax: **[no] rate-limit output access-group** {acl_id | name {ipv6} *ACL_NAME*} {*AVERAGE_RATE_BPS* | *MAX_BURST_BPS*}

Configure using policy-map

The command configures rate-limiting for outbound traffic on the interface using the IPv6 access-list “fdry” and the policy-map “map5”.

```
Brocade(config-if-1/1)# rate-limit output access-group name ipv6 fdry policy-map
map5
```

Syntax: **[no] rate-limit output access-group** {name {ipv6} *ACL_NAME*} {policy-map *POLICY_MAP_NAME*}

Configure for a specific priority-queue

The command configures rate-limiting for outbound traffic on priority-queue “q0” on the interface using the IPv6 access-list “fdry”, the average rate as 1000000 and the maximum burst size as 2000000.

```
Brocade(config-if-1/1)# rate-limit output access-group name ipv6 fdry priority q0
1000000 2000000
```

Syntax: **[no] rate-limit output access-group** { name {ipv6} *ACL_NAME*} [{priority *PRIORITY_QUEUE*}] {*AVERAGE_RATE_BPS* | *MAX_BURST_BPS*}

The command configures rate-limiting for outbound traffic on priority-queue “q0” on the interface using the IPv6 access-list “fdry” and the policy-map “map5”.

```
Brocade(config-if-1/1)# rate-limit output access-group name ipv6 fdry priority q0
policy-map map5
```

Configure strict-ACL rate-limiting on the interface

By default, rate-limiting is applied to traffic that matches a permit clause. If the traffic does not match any clause or if the traffic matches a deny clause, it is forwarded normally (neither dropped nor rate-limited). You can choose to drop packets that do not match any clause or that match the deny clause by configuring the strict ACL option under an interface.

NOTE
The strict ACL option is independent of ACL type (Layer 2/IPv4/IPv6).

When Strict ACL is enabled without any option, it applies to Layer-2, IPv4, and IPv6 ACL based rate-limiting configured on that port. The following command enables strict-ACL rate-limiting on an interface.

```
Brocade(config-if-1/1)# rate-limit strict-acl
```

Syntax: [no] rate-limit *strict-acl*

The following IPv6 ACL v6_permit_h2 has a permit and a deny clause.

```
Brocade(config)# ipv6 access-list v6_permit_h2
Brocade(config-ipv6-access-list-v6_permit_h2)# permit ipv6 host 1000:2382:e0bb::1
any
Brocade(config-ipv6-access-list-v6_permit_h2)# deny ipv6 host 3000::1 any
```

The following configuration enables strict ACL option on interface 2/1.

```
Brocade(config)# interface ethernet 2/1
Brocade(config-if-e1000-2/1)# rate-limit strict-acl
Brocade(config-if-e1000-2/1)# rate-limit input access-group name ipv6
v6_permit_h2 policy-map 1mbps
```

Traffic matching the permit clause will be rate-limited as per the rate values of the policy-map 1mbps.

Traffic which does not match any clause or that matches the deny clause will be dropped.

Deleting an IPv6 Access-List which is bound to rate-limit

When user attempts to delete an access-list which is bound to rate-limit profile, an error is thrown to the user that the ACL is in use.

```
Brocade (config)# no ipv6 access-list sample_v6
IPv6 ACL sample_v6 attached to an interface : error - ACL In Use.
```

To delete an IPv6 access-list which is bound to rate-limit profile, use the following configuration.

```
Brocade (config)# acl-policy
Brocade (config-acl-policy)#force-delete-bound-acl
```

After the force-delete-bound-acl configuration is enabled, you can delete any ACL even if it is bound.

Configuring rate-limit using non-existing access-list

Rate-limiting can be configured using a non-existing or empty IPv6 access-list. When the access-list is created or when filters are added to the access-list and an explicit rebind is performed, the rate-limit parameters will be programmed on the interface.

Output of show commands to verify output

Following is the output of the debug commands to confirm the configuration and functionality.

Display Rate-Limit Configuration

```
Brocade# show rate-limit

interface e 1/1
rate-limit input access-group name ipv6 fdry 993568 2000000
```

Display CAM contents for IPv6 access-list

```
Brocade#show cam v6acl 4/2
LP Index Src IP Addr                                SPort IFL/VLAN ID
         Dst IP Addr                                DPort Pro Age Out IF PRAM
4  b4000 ::/0                                       0     N/A
         ::/0                                       0     0   Dis Pass   000a8
4  b4008 ::/0                                       34816 N/A
         ::/0                                       0     58  Dis Pass   000a9
4  b4010 ::/0                                       34560 N/A
         ::/0                                       0     58  Dis Pass   000aa
4  b4018 ::/0                                       0     N/A
         ::/0                                       0     0   Dis Drop   000ab
```

Display rate-limit configurations

The following show command displays the rate limit configuration on an interface.

Syntax: show rate-limit interface *slot/port* **input | output access-group** {*acl_id* | name {mac | ipv4 | ipv6}} *acl_name*

Display Rate-limit counters

The following show command displays rate-limit counters on an interface.

Syntax: show rate-limit counters interface *slot/port* **input | output** access-group {*acl_id* | name {mac | ipv4 | ipv6 } *acl_name*}

Display Access-list accounting for rate-limiting

The following show command displays rate-limit accounting for an IPv6 ACL.

```
Brocade(config)# show ipv6 access-list accounting eth 1/1 in rate-limit
```

Syntax: show ipv6 access-list accounting *slot/port* **input | output in rate-limit**

Clearing rate-limit counters

You can clear rate-limit counters using the following command:

```
Brocade# clear rate-limit counters ipv6-subnet
```

Syntax: clear rate-limit counters ipv6-subnet

Layer 2 ACL-based rate limiting

Layer 2 ACL-based rate limiting enables devices to limit the rate of incoming traffic in hardware, without CPU intervention. Rate limiting in hardware enables the device to manage bandwidth at line-rate speed.

In general, Layer 2 ACL-based rate limiting works along the same lines as hardware-based rate limiting feature. All the rules and regulations that apply to hardware-based rate limiting also apply to this feature.

Configuration rules and notes

- You can apply Layer 2 ACL-based rate limiting on a physical port. You cannot apply it to a virtual interface or a LAG port.
- You cannot use IPv4 ACL-based filtering and Layer 2 ACL-based rate limiting on the same port. You can, however, configure one port on the device to use IP ACLs and another port on the same device to use Layer 2 ACL-based rate limiting.
- You cannot use IPv4 ACL-based rate limiting and Layer 2 ACL-based rate limiting on the same port. You can, however, configure one port on the device to use IPv4 ACL-base rate limiting and another port on the same device to use Layer 2 ACL-based rate limiting.
- You can bind multiple rate limiting policies to a single port. However, once a matching ACL clause is found for a packet, the device does not evaluate subsequent clauses in that rate limiting ACL and subsequent rate limiting ACLs.
- Only number ACLs support rate limiting
- Layer 2 rate limiting ACLs will function with vlan-cpu-protection, broadcast and multicast limiting features. If incoming traffic matches an inbound Layer 2 rate limiting ACL, it is first rate-limited based on the policy. If packets are not dropped due to rate limiting, they are forwarded either to the CPU or flooded in the VLAN according to the vlan-cpu-protection feature.

NOTE
The above behavior applies only for the Brocade NetIron XMR and Brocade MLX series, not for the Brocade NetIron CES and Brocade NetIron CER. For the Brocade NetIron CES and Brocade NetIron CER platforms, once the broadcast and multicast limiting features are enabled, these limits will take precedence over the defined port rate-limit.

- The broadcast and multicast packet limiting feature limits packets in the CPU, while the Layer 2 ACL RL is a network processing (NP) RL feature. Packets are first subjected to the Layer 2 ACL RL at the NP. Once packets are forwarded to CPU, the broadcast and multicast limiting feature begins functioning and packets may be dropped in the CPU if the rate exceeds the limit.

Editing a Layer 2 ACL Table

You can make changes to the Layer 2 ACL table definitions without unbinding and rebinding the rate limit policy. For example, you can add a new clause to the ACL table, delete a clause from the table, or delete the ACL table that is used by a rate limit policy.

Define rate limiting parameters

To define rate limiting parameters, enter commands such as the following:

```
Brocade(config)#policy-map map1
Brocade(config-policymap map1)#cir 1000000 cbs 2000000 eir 1000000 ebs 2000000
excess-dp 2
```

Binding Layer 2 ACL-based rate limiting policy to a port

To bind an Layer 2 ACL based rate-limiting policy on a specific port, enter commands such as the following:

```
Brocade(config-policymap map1)#int eth 14/1
Brocade(config-if-e10000-14/1)# rate-limit input access-group 400 policy-map map1
```

Syntax: [no] rate-limit [input|output] access-group *num* **policy-map** *map-name*

Specifying rate limiting parameters without a policy map

To specify rate-limiting without using a policy map, enter a command such as the following:

```
Brocade(config-if-e10000-14/1)# rate-limit input access-group 400 49999998416
75000000000
```

Syntax: [no] rate-limit [input|output] access-group *acl-id average-rate maximum-burst*

The *acl-id* for Layer 2 ACLs can range from 400 to 499.

The *average-rate* is the maximum number of bits the policy allows during one second.

The *maximum-burst* parameter specifies the extra bits above the average-rate that traffic can have. Refer to the section "Maximum burst" on page 17 for more details. This command is only used when configuring traffic policing directly to a port as described in "Applying traffic policing parameters directly to a port" on page 16.

Display accounting

To display access list accounting, enter a command such as the following.

```
Brocade#show access-list accounting eth 14/1 in rate-limit
Collecting L2 ACL accounting for 400 on port 14/1  ...  Completed successfully.
RL ACL Accounting Information:
Inbound: ACL 400
    0:   permit 0000.0000.0021 ffff.ffff.ffff any any etype any
       Hit count: (1 sec)                    0   (1 min)                    0
                  (5 min)                    0   (accum)                    0
```

Rate limiting protocol traffic using Layer 2 inbound ACLs

Using interface level Layer 2 inbound ACLs, you can rate limit the following types of protocol traffic by explicitly configuring a filter to match the traffic:

- STP/RSTP/BPDU
- MRP
- VSRP
- LACP
- GARP
- UDLP

To rate-limit all such control traffic enter commands such as the following:

```
Brocade(config)#access-list 402 permit any 0180.c200.0000 ffff.ffff.ffff any
etype any
Brocade(config)#access-list 402 permit any 0304.8000.0000 ffff.ffff.ffff any
etype any
Brocade(config)#access-list 402 permit any 0304.8000.0100 ffff.ffff.ff00 any
etype any
Brocade(config)#access-list 402 permit any 0180.c200.0002 ffff.ffff.ffff any
etype any
Brocade(config)#access-list 402 permit any 0180.c200.0020 ffff.ffff.fff0 any
etype any
Brocade(config)#access-list 402 permit any 00e0.5200.0000 ffff.ffff.ffff any
etype any
Brocade(config)#access-list 402 deny any any any etype any
```

Table 12 lists the protocols and their corresponding filters.

TABLE 12 Filters for protocols

Protocol	Filter
STP/RSTP/BPDU	access-list 402 permit any 0180.c200.0000 ffff.ffff.ffff any etype any
MRP	access-list 402 permit any 0304.8000.0000 ffff.ffff.ffff any etype any
VSRP	access-list 402 permit any 0304.8000.0100 ffff.ffff.ff00 any etype any
LACP	access-list 402 permit any 0180.c200.0002 ffff.ffff.ffff any etype any
GARP	access-list 402 permit any 0180.c200.0020 ffff.ffff.fff0 any etype any
UDLP	access-list 402 permit any 00e0.5200.0000 ffff.ffff.ffff any etype any

NOTE
The filters must have the specific destination MAC address as shown above in the configuration. You can filter all protocols as shown in the previous configuration example above, or only specific protocols.

Example of Layer 2 ACL to rate limit broadcast traffic

To define an ACL that rate limits broadcast traffic and forwards all other traffic without rate limiting, enter commands such the following:

```
Brocade(config)#access-list 411 permit any ffff.ffff.ffff ffff.ffff.ffff
Brocade(config)#access-list 411 deny any any
```

To bind an ACL that rate limits broadcast traffic and forwards all other traffic without rate limiting, enter commands such the following

```
Brocade(config)#int eth 14/1
Brocade(config-if-e10000-14/1)#rate-limit in access-gr 411 8144 100
```

Rate limiting ARP packets

You can limit the rate of ARP traffic that requires CPU processing on Brocade devices, such as ARP request traffic, and ARP response addressed to the device. The feature is set globally and applies to all ARP traffic received at the device. With this feature you can apply a defined policy map to all ARP traffic bound for the CPU.

When the **vlan-cpu-protection** command is configured, ARP request packets are switched within a VLAN by the hardware and thus cannot be rate-limited by the **ip rate limit arp policy-map** command. To limit the rate of ARP packets that are forwarded by hardware, use interface-level, layer-2 inbound ACLs with the “etype arp” option.

Configuring rate limiting of ARP packets

To rate limit ARP packets bound for the CPU using a policy map named “limitarp”, enter the following command.

```
Brocade(config)# ip rate-limit arp policy-map limitarp
```

Syntax: [no] ip rate-limit arp policy-map *map-name*

The *map-name* variable is the name of the policy map to be used to provide parameters for rate limiting CPU-bound ARP packets. If the policy map specified has not been defined, the rate limit values are initialized to the line rate values.

Displaying statistics for ARP rate limiting

You can display ARP Rate Limiting Statistics using the following command.

```
Brocade# show rate-limit arp
 Fwd:        1865392                Drop:  867731400 bytes
 Re-mark:    1864800                Total: 871461592 bytes
```

Syntax: show rate-limit arp

This display shows the following information.

TABLE 13 Rate limit ARP display parameters

Parameter	Description
Fwd	The ARP traffic in bytes that has been sent to the CPU as a result of the ARP rate limit policy since the device was started up or the counter was reset.
Drop	The ARP traffic in bytes that has been dropped as a result of the ARP rate limit policy since the device was started up or the counter was reset.

TABLE 13 Rate limit ARP display parameters

Parameter	Description
Re-mark	The ARP traffic in bytes whose priority have been remarked as a result of exceed the bandwidth available in the CIR bucket for the ARP rate limit policy since the device was started up or the counter was reset.
Total	The total ARP traffic in bytes that has been subjected to the ARP rate limit policy since the device was started up or the counter was reset.

Clearing Statistics for ARP Rate Limiting

You can clear ARP Rate Limiting Statistics using the following command:

```
Brocade# clear rate-limit arp
```

Syntax: clear rate-limit arp

Chapter 3

Configuring Quality of Service (QoS) for the Brocade NetIron CES and Brocade NetIron CER Series

Quality of Service (QoS)

The Quality of Service (QoS) features offer many options for the Brocade NetIron CES and Brocade NetIron CER devices.

Quality of Service (QoS) provides preferential treatment to specific traffic, possibly at the expense of other traffic. Without QoS, the Brocade NetIron CES or Brocade NetIron CER device offers best-effort service to each packet and transmits packets without any assurance of reliability, delay bounds, or throughput. Implementing QoS in a network makes performance more predictable and bandwidth utilization more effective.

QoS implementation in the Brocade NetIron CES or Brocade NetIron CER device complies with the IETF-DiffServ and IEEE 802.1p standards. A typical QoS model deployment is based on the following elements:

- At the network edge, the packet is assigned to a QoS service. The service is assigned based on the packet header information (i.e., packet is trusted) or on the ingress interface configuration (packet is not trusted).
- The QoS service defines the packet's internal QoS handling (e.g., traffic class and drop precedence) and optionally the packet's external QoS marking, through either the IEEE 802.1p User Priority or the IP header DSCP field.
- Subsequent Brocade NetIron CES and Brocade NetIron CER devices within the network core provide consistent QoS treatment to traffic, based on the packet's IEEE 802.1p, or DSCP marking. As a result, an end-to-end QoS behavior is provided.
- A Brocade NetIron CES or Brocade NetIron CER device may modify the assigned service if a packet stream exceeds the configured profile. In this case, the packet may be dropped or reassigned to a lower QoS service.
- The Brocade NetIron CES and Brocade NetIron CER devices incorporate the required QoS features to implement network-edge as well as network-core devices.
- The Brocade NetIron CES and Brocade NetIron CER devices provide flexible mechanisms to classify packets into different service levels.
- The packet header may have its User Priority fields set to reflect the QoS assignment.
- Service application mechanism is based on eight egress priority queues per port (including the CPU port), on which congestion-avoidance and congestion-resolution policies are applied.
- QoS encode policies are not supported on the NetIron CES and NetIron CER devices due to Hardware limitations.

QoS model

This chapter describes how QoS is implemented and configured in the Brocade NetIron CES and Brocade NetIron CER devices. The chapter contains the following sections.

Traffic types

Data - Data packets can be either Network-to-Network traffic or traffic from the CPU. Network-to-Network traffic is considered Data traffic.Qos parameters can be assigned and modified for data traffic.

Control - Packets to and from the CPU is considered control traffic. The QoS parameters fro this traffic are preassigned and not configurable.

Setting packet header QoS fields

The device supports setting or modifying the packet header IEEE 802.1p User Priority or IP-DSCP.

Packet QoS attributes

Every packet classified as Data is assigned a set of QoS attributes that can be modified by each ingress pipeline engine.

Each of the ingress pipeline engines contain several Initial QoS Markers that assign the packet's initial QoS attribute.

The ingress pipeline engine also contains a QoS Remarker that can modify the initial QoS attributes.

Even though Brocade NetIron CES and Brocade NetIron CER devices support four drop precedence values 0,1,2 and 3 internally 1 and 2 are assigned the same drop precedence level. The four levels are kept for CLI compatibility with other Brocade devices. Three internal level of drop precedence are 0, {1,2} and 3. in terms of commonly used color based terminology: 0 represents Green (lowest drop precedence}, 1 and 2 represents yellow (higher drop precedence) and 3 represents Red (highest drop precedence).

TABLE 14 Packet QoS attributes

QoS parameter	Description
TC (Traffic Class)	This is the priority level assigned to the packet. When the TxQ enqueues the packet, it uses this field to select the appropriate priority queue.
DP (Drop Precedence)	The TxQ uses this field for congestion resolution. Packets with higher drop precedence are more likely to be discarded in the event of congestion.

Ingress Traffic processing through a device

The QoS operation on Ingress traffic of a Brocade device involves reception and processing of packets based upon priority information contained within the packet. As the packets are processed through the device, there are several opportunities to influence the processing by configuration as described in the following steps.

1. Derive priority and drop precedence from the packets PCP (802.1p) value. The Priority Code Point (PCP) is a 3-bit field within an 802.1Q tagged frame that is used to convey the priority of the frame. By using a mapping table, the 3-bit PCP field can be decoded to derive priority and drop precedence information. Note: the PCP field was formerly called 802.1p.
2. Derive priority and drop precedence from the packets DSCP value.

3. Force the priority and drop precedence value based on the value configured for the physical port.
4. Force the priority value based on an ACL look-up. This is used for setting a a specific priority for and L2, L3 or L4 traffic flow.

NOTE
DEI value will remain 0 regardless of PCP or DSCP value.

Recognizing inbound packet priorities and mapping to internal priority

Stage 1
Collect priority and drop precedence information from various portions of the packet header

- If a packet's EtherType matches 8100 or the port's EtherType, derive a priority value and drop precedence by decoding the PCP value.
- For IPv4 packets, derive a priority value and drop precedence by decoding the DSCP bits.
- The derived values for PCP, and DSCP are mapped to a default map.
- To assist the device in the decoding process described in "stage 1" decode-map tables are defined.

Stage 2
Determine if a priority value should be forced

- If a packet's EtherType matches 8100 or the port's EtherType, derive a priority value and drop precedence by decoding the PCP value
- If the **qos pcp force** command is configured on the port, the priority and drop precedence values are set to the value read from the PCP bits.
- If the **qos dscp force** command is configured on the port, the priority and drop precedence values are set to the value read from the DSCP bits.
- If none of the qos force commands are configured, the priority and drop precedence values of a given packet is obtained in the following manner:
  - For Tagged and Untagged IPv4Packets: Priority and drop precedence values obtained from decoded DSCP values.
  - For Tagged Non-IPv4 Packets: Priority and drop precedence values obtained from decoded PCP values.
  - For Untagged, Non-IPv4 Packets: Priority and drop precedence values obtained from priority and drop precedence assigned on a port. If no priority and drop precedence is assigned on a port default value of Priority 0 and Drop Precedence 0 is picked.

Forcing the priority of a packet

Once a packet's ingress priority has been mapped, the values that will be used for processing on the device are determined by either forcing or merging.

There are a variety of commands to "force" the priority of a packet based on the following criteria:

- Forced to a priority configured for a specific ingress port. The **priority force** command is configured at the interface where you want it to be applied.
- Forced to a priority that is obtained from the DSCP priority bits. The **qos- dscp force** command is configured at the interface where you want it to be applied.
- Forced to a priority that is obtained from the PCP priority bits. The **qos- pcp force** command is configured at the interface where you want it to be applied.
- Forced to a priority that is based on an ACL match. The **priority-force** keyword can be used within an ACL to apply a priority to specified traffic.

NOTE
Commands to "force" the priority of a packet are not supported for the **vlan priority** command in the Brocade NetIron CES and Brocade NetIron CER platforms.

If multiple commands containing the **priority force** keyword are specified, the command with the highest precedence will take effect as determined by the following order.

1. ACL match
2. Physical port priority value
3. DSCP value in an incoming IPv4 packet
4. PCP value in a tagged frame or PCP

Details of how to configure the force commands are provided in "Configuring a force priority" on page 46.

ACL QoS Considerations

The following should be considered when configuring QoS.

- All packets with ACL match where action is Priority/Priority force will have BOTH UP and EXP bit encoded.
- Regardless of the Port Encoding settings.The effect will be user visible only if the outgoing packet is MPLS for non-MPLS, packets the functionality will remain same as before.
- The EXP bit is carried to the Remote VPLS/VLL in the inner label (VC Label). QoS assignment is on the remote.
- PE device will be based on this EXP. The outgoing packet's PCP value will be encoded based on QoS assigned by the EXP.
- At Ingress:
  - ACL-->TC Assignment ---> TC Encodes {UP, EXP}
- At Remote VPLS:
  - EXP Based QoS Assignment-->TC--> TC Encodes {UP}

Custom decode support

User defined decode maps are supported on the Brocade NetIron CES and Brocade NetIron CER. The custom decode maps have the following implication for QoS handling:

- Per port custom decode maps are not supported. Only a single global QoS map is supported.
- A number of custom decode maps can be defined in the Multi-Service IronWare, but only one can be active at any time in the hardware.
- A user defined map can be applied to replace any existing map including the default map.
- User defined maps are supported for PCP, EXP and DSCP.
- Custom decode maps are supported. This means that packet can be decoded by any one of the many user defined map currently active in the hardware but there will be one and only encode map. Encoding will always be based on default encode map.
- The packet will be decoded as per the decode map, but during encoding NOT all the values will be encoded correctly.
- If the packet is decoded based on DSCP value, the Brocade NetIron CES and Brocade NetIron CER will decode the packet to correct internal priority (Traffic Class) and Drop Precedence. This QoS will be internally respected and packet will be treated in accordance with the assigned Priority and DP. During encoding (if enabled) however, the Brocade NetIron CES and Brocade NetIron CER will encode all the packet QoS attributes as shown below:
  - Assume trust mode is DSCP and encoding policy is turned on for UP and DSCP
  - Assume custom Decode Map: DSCP 56 to Priority 0 and DP 0
  - Encode map is always default
  - IPv4 tagged packet comes with DSCP 56 and UP 5
  - The packet will be assigned Internal Priority of 0 and DP of 0
  - The packet going out will have 802.1p value of 0
  - The packet will NOT have expected DSCP value of 0, instead it will go out unchanged (56)
- If the packet is decoded based on PCP value (which is possible if qos trust mode is PCP force or no trust mode is forced but packet is Tagged and payload is not IPv4), the Brocade NetIron CES will decode the packet to correct internal priority (Traffic Class) and Drop Precedence. This QoS will be internally respected and packet will be treated in accordance with the assigned Priority and DP. During encoding (if enabled) however, Brocade NetIron CES and Brocade NetIron CER will encode all the packet QoS attributes as shown below:
  - Assume trust mode is DSCP and encoding policy is turned on for UP and DSCP
  - Assume custom Decode Map: PCP 5 to Priority 0 and DP 0
  - Encode map is always default
  - Assume an IPv4 tagged packet comes with DSCP 56 and UP 5
  - The packet will be assigned Internal Priority of 0 and DP of 0
  - The packet going out will have DSCP value of 0
  - The packet will not have expected PCP value of 0, instead it will go out unchanged (5)

Forcing the drop precedence of a packet

Once a packet's ingress drop precedence has been mapped, the values that will be used for processing on the device are determined by either forcing or merging.

There are a variety of commands to "force" the drop precedence of a packet based on the following criteria:

- Forced to a drop precedence configured for a specific ingress port. The **drop-precedence force** command is configured at the interface where you want it to be applied.
- Forced to a drop precedence that is obtained from the DSCP priority bits. The **qos dscp force** command is configured at the interface where you want it to be applied.
- Forced to a drop precedence that is obtained from the PCP priority bits. The **qos pcp force** command is configured at the interface where you want it to be applied.
- Forced to a drop precedence that is based on an ACL match. The **drop-precedence-force** keyword can be used within an ACL to apply a priority to specified traffic.

If multiple commands containing the **force** keyword are specified, the command with the highest precedence will take effect as determined by the following order.

1. ACL match
2. Physical port's drop precedence value
3. DSCP value in an incoming IPv4 packet
4. PCP value in a tagged frame or PCP

Details of how to configure the force commands are provided in "Configuring a force priority" on page 46.

Configuring QoS

The QoS configuration process involves separate procedures for Ingress and Egress QoS Processing as described in the following major sections.

Configuring QoS procedures applicable to Ingress and Egress

The following procedures are required to configure procedures applicable to both Ingress and Egress QoS Processing on a Brocade device:

- Support for QoS Configurations on LAG Ports – If you are configuring the enhanced QoS feature on ports within a LAG, refer to "Configuring port-level QoS commands on LAG ports" on page 49.

Configuring a force priority

In situations where there are conflicting priority values for packets on an Ingress port, that conflict can be resolved by performing a priority merge or by using a **force** command to direct the device to use a particular value above other values. A **force** command can be configured for each of the following:

- Force to the values configured on a port
- Force to the value in the DSCP bits
- Force to the value in the PCP bits
- Force to a value specified within an ACL

Configuring a force priority for a port

You can configure an ingress port with a priority to apply to packets that arrive on it using the **priority** command.

To configure an ingress port with a priority, use the **priority** command as shown in the following.

```
Brocade(config)# interface ethernet 10/1
Brocade(config-if-e10000-10/1)priority 6
```

Syntax: priority *priority-value*

The *priority-value* variable is a value between 0 and 7. The default value is 0.

Once a port has been configured with a priority using the **priority** command, you can then configure the port (using the **priority force** command) to force the configured priority when determining the priority relative to other priority values of incoming packets.

To configure an ingress port to force the port-configured priority, use the **priority force** command as shown in the following.

```
Brocade(config)# interface ethernet 10/1
Brocade(config-if-e10000-10/1)priority force
```

Syntax: [no] priority force

The priority will be forced to zero if no value is configured.

Configuring a force drop precedence for a port

You can configure an ingress port with a drop precedence to apply to packets that arrive on it using the **drop-precedence** command.

To configure an ingress port with a drop precedence, use the **drop-precedence** command as shown in the following.

```
Brocade(config)# interface ethernet 10/1
Brocade(config-if-e10000-10/1)drop-precedence 3
```

Syntax: [no] drop-precedence *dp-value*

The *dp-value* variable is a value between 0 and 3.

Once a port has been configured with a drop precedence using the **drop-precedence** command, you can then configure the port (using the **drop-precedence force** command) to force the configured drop precedence when determining the priority relative to other priority values of incoming packets.

To configure an ingress port to force the port-configured drop precedence, use the **drop-precedence force** command as shown in the following.

```
Brocade(config)# interface ethernet 10/1
Brocade(config-if-e10000-10/1)drop-precedence force
```

Syntax: [no] drop-precedence force

Configuring force priority to the DSCP value

You can configure an ingress port (using the **qos dscp force** command) to force the configured DSCP value when determining the priority relative to other priority values of incoming packets.

To configure an ingress port to force the DSCP value, use the **qos dscp force** command as shown in the following.

```
Brocade(config)# interface ethernet 10/1
Brocade(config-if-e10000-10/1)qos dscp force
```

Syntax: [no] qos dscp force

Configuring force priority to the PCP value

You can configure an ingress port (using the **qos pcp force** command) to force the configured PCP value when determining the priority relative to other priority values of incoming packets.

To configure an ingress port to force the PCP value, use the **qos pcp force** command as shown in the following.

```
Brocade(config)# interface ethernet 10/1
Brocade(config-if-e10000-10/1)qos pcp force
```

Syntax: [no] qos pcp force

Configuring force priority to a value specified by an ACL

You can use the **priority-force** keyword within an ACL to apply a priority to specified traffic as described in “Filtering and priority manipulation based on 802.1p priority” in the *Multi-Service IronWare Security Configuration Guide*.

Configuring extended-qos-mode

The **extended-qos-mode** command should only be turned on when deploying CES/CER as MPLS PE devices, if preserving passenger DSCP is required, when terminating VPLS/VLL traffic at the egress end point.

NOTE
You must write this command to memory and perform a system reload for this command to take effect.

NOTE
This command will reduce the hardware table size by half. If the existing configuration has already used more the half of the hardware table size, this command will not succeed. This command will only succeed if there is sufficient space in the hardware table.

You can revert to normal mode using the **no extended-qos-mode** command without losing functionality or behavior.

```
Brocade(config)# extended-qos-mode
```

Syntax: [no] extended-qos-mode

Configuring port-level QoS commands on LAG ports

When applying port-level QoS commands to ports in a LAG, the rules can be different according the configuration as described in the following:

- Port-level QoS configurations where QoS values are applied directly to the port. These commands include the followings: **priority**, **priority-force**, **drop-precedence**, **drop-precedence force**.
- Port-level QoS configurations using commands that begin with the **qos** keyword. These commands include: **qos dscp decode-policy**, **qos pcp decode-policy**, **qos exp decode-policy**, **qos dscp force**, **qos pcp force**, **qos exp force**, **qos dscp encode on**, and **qos pcp encode on**.

LAG configuration rules where QoS values are applied directly to the port

In port-level QoS Configurations where QoS Values are applied directly to the port, the considerations listed below must be followed.

1. Each port that is configured into the LAG, must have the same **priority**, **priority-force**, **drop-precedence**, and **drop-precedence force** configuration.

   If you try to configure a LAG with ports that have a different configuration for these commands, the LAG deployment will fail and you will get an error message as shown in the following.

   ```
   Brocade(config)# lag mylag static
   Brocade(config-lag-mylag)# ports eth 10/1 to 10/2
   Brocade(config-lag-mylag)# primary 10/1
   Brocade(config-lag-mylag)# deploy
   port 10/1 priority is 5, but port 10/2 priority is 0
   Error: port 10/1 and port 10/2 have different configurations
   LAG mylag deployment failed!
   Brocade(config-lag-mylag)#
   ```

2. If you have already formed a LAG with the same configuration, you can change the configuration by making changes to the LAG's primary port.
3. If the LAG configuration is deleted, each of the port in the LAG (primary and secondary) will inherit the QoS configuration of the primary port.

LAG configuration rules for QoS configurations using commands that begin with the qos keyword

NOTE
Due to hardware considerations on the Brocade NetIron CER and Brocade NetIron CES devices, the encode policy must be applied on the ingress interface.

In port-level QoS configurations where QoS configurations using commands that begin with the **qos** keyword are used, the considerations listed below must be followed.

1. The secondary ports configured in the LAG must not have any QoS values configured on them.
2. The **qos** commands that are configured on the primary port are applied to all ports in the LAG.
3. Once the LAG is formed, you can change the QoS configuration for all ports in the LAG by making changes to the LAG's primary port and you cannot make changes to the QoS configurations directly to any of the secondary ports.
4. If the LAG is deleted, the QoS configuration will only be retained on the primary port.

Due to hardware considerations on the Brocade NetIron CER and Brocade NetIron CES devices, the encode policy must be applied on the ingress interface. This restriction results in the encode policies being applied to all packets coming in on the ingress port, regardless of egress port. See example configuration below.

```
!
Brocade(config)# interface ethernet 2/1
Brocade(config)# qos dscp encode-policy on
Brocade(config)# enable
Brocade(config)# priority force
Brocade(config)# priority 5
Brocade(config)# drop-precedence force
Brocade(config)# drop-precedence 3
Brocade(config)# mac access-group 1302 in
!
access-list 1302 permit any any 11 etype any dscp-marking 60
!
```

With the **qos dscp encode-policy** on and access-list 1302 applied, all packets coming in on interface e 2/1 and VLAN 11 will have a modified dscp value of 60 on the egress regardless of the egress port.

In the example, packets not matching the ACL would have the port value set, per "Forcing the drop precedence of a packet" on page 46.

Configuring port-level QoS commands on CPU ports

The control packets destined to the CPU are assigned fixed priorities. The data and control packets that are processed by the CPU are prioritized, scheduled, and rate-shaped so that the higher priority control packets are handled before any lower priority control and data packets. The enhanced control packet prioritization and scheduling scheme ensures the proper transmission or reception of time-sensitive control and protocol packets.

Table 15 lists the protocol and data packets based on their priorities.

TABLE 15 Prioritized protocol and data packets

Priority categorization	Protocols
P7	LACP, UDLD, STP, RSTP, BPDU, VSRP, MRP, LLDP, VRRP, VRRP-E, 802.1x, FDP, CDP, BFD, ERP, 802.1ag.
P6	OSPF, IS-IS, RIP, RIPNG, BGP, IPv6 Neighbor Discovery, CCP, LDP, RSVP.
P5	PIM, PIM-DM, IGMP, DVMRP, MLD.
P4	ARP, DHCP, BOOTP.
P3	Telnet, SNMP.
P2	Reserved.
P1	sFlow.
P0	Data packets.

To configure a CPU port, enter the following command.

```
Brocade(config)# cpu-port
Brocade(config-cpu-port)#
```

Syntax: [no] cpu-port

The **no** option is used to disable QoS on the CPU port.

Configuring port-based rate shaping

You can limit the amount of bandwidth available on a CPU port by configuring the rate shaping value for that port. To set the rate shaping value, enter the following command.

```
Brocade(config-cpu-port)# shaper 1000000
```

Syntax: [**no**] **shaper** *rate*

The *rate* parameter sets the rate shaping value for the CPU port. The acceptable value can be from 1 through 1000000 kilobits per second (Kbps).

The **no** option is used to reset the port shaping rate.

Configuring traffic scheduling

Traffic scheduling can be configured on a per-port and per-queue basis. Traffic scheduling affects the outgoing traffic on the configured CPU port when bandwidth congestion occurs on that port. One strict profile, three Weighted Round Robin (WRR) profiles, and four mixed profiles define the scheduling attributes.The following sections describe how to configure each of the traffic scheduling schemes:

- "Configuring strict priority-based traffic scheduling"
- "Configuring WRR weight-based traffic scheduling"
- "Configuring mixed strict priority- and weight-based traffic scheduling"

Configuring strict priority-based traffic scheduling

To configure strict priority-based scheduling, enter the following command.

```
Brocade(config-cpu-port)# scheduler strict
```

Syntax: [**no**] **scheduler strict**

Strict priority-based scheduling is the default traffic scheduling scheme.

The **no** option is used to disable the strict scheduler.

Configuring WRR weight-based traffic scheduling

When configuring a CPU port scheduler to WRR, one of eight global scheduler profiles is taken and you can assign weights to each priority queue in global configuration mode.

To configure WRR weight-based scheduling., enter the following command.

```
Brocade(config-cpu-port)# scheduler profile WRR0 weighted 40 10 20 30 40 50 60 10
```

Syntax: [**no**] **scheduler profile** *profile name* **weighted** [*queue7-weight queue6-weight queue5-weight queue4-weight queue3-weight queue2-weight queue1-weight queue0-weight*]

The *profile name* parameter specifies the name of the weighted scheduler.

The **weighted** keyword defines the weighted scheduler.

The *queue7-weight* parameter defines the value for queue 7 in calculating the allocated bandwidth of queue 7.

The *queue6-weight* parameter defines the value for queue 6 in calculating the allocated bandwidth of queue 6.

The *queue5-weight* parameter defines the value for queue 5 in calculating the allocated bandwidth of queue 5.

The *queue4-weight* parameter defines the value for queue 4 in calculating the allocated bandwidth of queue 4.

The *queue3-weight* parameter defines the value for queue 3 in calculating the allocated bandwidth of queue 3.

The *queue2-weight* parameter defines the value for queue 2 in calculating the allocated bandwidth of queue 2.

The *queue1-weight* parameter defines the value for queue 1 in calculating the allocated bandwidth of queue 1.

The *queue0-weight* parameter defines the value for queue 0 in calculating the allocated bandwidth of queue 0.

The **no** option is used to return to the default traffic scheduler.

Configuring mixed strict priority- and weight-based traffic scheduling

When configuring the mixed strict priority- and weight-based scheduling scheme, queue 5 to queue 7 are allocated to strict priority-based scheduling and queue 0 to queue 4 are allocated to weight-based scheduling.

To configure mixed strict priority- and weight-based scheduling., enter the following command.

```
Brocade(config-cpu-port)# scheduler profile MIXED0 mixed 20 30 15 5 10
```

Syntax: [**no**] **scheduler profile** *profile name* **mixed** [*queue4-weight queue3-weight queue2-weight queue1-weight queue0-weight*]

The *profile name* parameter specifies the name of the mixed scheduler.

The **mixed** keyword defines the mixed scheduler.

The *queue4-weight* parameter defines the value for queue 4 in calculating the allocated bandwidth of queue 4.

The *queue3-weight* parameter defines the value for queue 3 in calculating the allocated bandwidth of queue 3.

The *queue2-weight* parameter defines the value for queue 2 in calculating the allocated bandwidth of queue 2.

The *queue1-weight* parameter defines the value for queue 1 in calculating the allocated bandwidth of queue 1.

The *queue0-weight* parameter defines the value for queue 0 in calculating the allocated bandwidth of queue 0.

The **no** option is used to return to the default traffic scheduler.

Displaying QoS information

You can display the following QoS information as described:

- **QoS Configuration Information** – Using the **show qos-map decode-map** and **show qos-map encode-map** commands, you can display the priority and drop-precedence values mapped between values internal to the device and values that are received at the device or marked on packets leaving the device. This is described in "Displaying QoS Configuration information" on page 54.
- **QoS Packet and Byte Statistics** – Using the **show qos-map decode-map** and **show qos-map encode-map** commands, you can enable and display the contents of the QoS Packet and Byte Counters as described in "Clearing the QoS packet and byte counters" on page 54.

Displaying QoS Configuration information

You can display the following QoS Configuration information:

- QoS Decode Policy Map Configurations
- QoS Policy Map Binding Configurations

Clearing the QoS packet and byte counters

You can clear the QoS counters whose display is generated using the **show np statistics** command as shown in the following.

Syntax: clear np statistics

Scheduling traffic for forwarding

If the traffic being processed by a Brocade device is within the capacity of the device, all traffic is forwarded as received. Once we reach the point where the device is bandwidth constrained, it becomes subject to traffic scheduling as described in this section.

The Brocade devices classify packets into one of eight internal priorities. Traffic scheduling allows you to selectively forward traffic according to the forwarding queue that is mapped to according to one of the following schemes:

- **Strict priority-based scheduling** – This scheme guarantees that higher-priority traffic is always serviced before lower priority traffic. The disadvantage of strict priority-based scheduling is that lower-priority traffic can be starved of any access.
- **WRR (Weighted Round Robin)** – With WRR destination-based scheduling enabled, some weight-based bandwidth is allocated to all queues. With this scheme, the configured weight distribution is guaranteed across all traffic leaving an egress port and an input port is guaranteed allocation in relationship to the configured weight distribution.
- **Mixed strict priority and weight-based scheduling** – This scheme provides a mixture of strict priority for the three highest priority queues and WRR for the remaining priority queues.

Configuring traffic scheduling

Traffic scheduling can be configured on a per-port basis. It affects the outgoing traffic on the configured port when bandwidth congestion occurs on that port. The following sections describe how to configure each of the traffic scheduling schemes:

- "Configuring strict priority-based traffic scheduling" This option is the default traffic scheduling method if traffic scheduling is not configured on a port.
- "Calculating the values for WRR weight-based traffic scheduling"
- "Configuring WRR weight-based traffic scheduling"
- "Configuring mixed strict priority and weight-based scheduling"

Configuring strict priority-based traffic scheduling

To configure strict priority-based scheduling use a command such as the following.

```
Brocade(config)# interface ethernet 1/1
Brocade(config-if-e1000-1/1)# qos scheduler strict
```

Syntax: [no] qos scheduler strict

This is the default when traffic scheduling is not configured.

Calculating the values for WRR weight-based traffic scheduling

WRR (Weighted Round Robin) scheduling is configured to be a percentage of available bandwidth using the following formula. Remember weight is a relative value.

$$\text{Weight of } q(x) = \frac{q(x)}{q0 + q1 + q2 + q3 + q4 + q5 + q6 + q7}$$

Weight of q (x) = the calculated weight as a percentage of the port's total bandwidth.
For example if you assign the following values to queues 0 to 7:

- Queue 0 =10, Queue 1 = 15, Queue 2 = 20, Queue 3 = 25, Queue 4 = 30, Queue 5 = 35, Queue 6 = 40, and Queue 7 = 45

Where:

q (x) = The value of the queue that you want to determine the weight for. It can be the value of any queue (0 - 7).

q0 - q7 = the assigned values of the eight queues.

Weight of q (x) = the calculated weight as a percentage of the port's total bandwidth.

Determining the correlation between weight and bandwidth example

To determine the correlation between weight and bandwidth use following calculation example.

Example

If you assign the following values to queues 0 to 7:

- Queue 0 =10, Queue 1 = 15, Queue 2 = 20, Queue 3 = 25, Queue 4 = 30, Queue 5 = 35, Queue 6 = 40, and Queue 7 = 45,

To determine the weight of **q3**

$$\text{Weight of q3} = \frac{25}{10 + 15 + 20 + 25 + 30 + 35 + 40 + 45}$$

The weight of q3 is 11.4%. Consequently, q3 will get 11.4% of the port's total bandwidth.

The values of the remaining queues are calculated to be the following: q7 = 20.5%, q6 = 18.2%, q5 = 15.9%, q4 = 13.6%, q3 = 11.4%, q2 = 9.1%, q1 = 6.8%, and q0 = 4.5%

NOTE
The easiest way to calculate the correlation between weight and bandwidth is to have the total weight of all queues sum up to 100, then the weight itself will be the % bandwidth available for that queue.

Configuring WRR weight-based traffic scheduling

To configure WRR weight-based scheduling use a command such as the following at the global configuration level.

```
Brocade(config)# qos scheduler profile MIXED0 mixed 20 30 15 5 10
```

Syntax: **qos scheduler profile** *name* **weighted** *queue7-weight queue6-weight queue5-weight queue4-weight queue3-weight queue2-weight queue1-weight queue0-weight*]

The **weighted** variable defines the weighted scheduler.

The **mixed** variable defines the mixed scheduler.

The *queue7-weight* variable defines the relative value for queue7 in calculating queue7's allocated bandwidth.

The *queue6-weight* variable defines the relative value for queue6 in calculating queue6's allocated bandwidth.

The *queue5-weight* variable defines the relative value for queue5 in calculating queue5's allocated bandwidth.

The *queue4-weight* variable defines the relative value for queue4 in calculating queue4's allocated bandwidth.

The *queue3-weight* variable defines the relative value for queue3 in calculating queue3's allocated bandwidth.

The *queue2-weight* variable defines the relative value for queue2 in calculating queue2's allocated bandwidth.

The *queue1-weight* variable defines the relative value for queue1 in calculating queue1's allocated bandwidth.

The *queue0-weight* variable defines the relative value for queue0 in calculating queue0's allocated bandwidth

Refer to "Calculating the values for WRR weight-based traffic scheduling" for information on assigning queue0-weight to queue4-weight values.

Configuring mixed strict priority and weight-based scheduling

When configuring the mixed strict priority and weight-based scheduling option, queues 5 - 7 are allocated to strict priority-based scheduling and queues 0 - 4 are allocated to weight-based scheduling.

To configure mixed priority and weight-based scheduling use a command such as the following.

```
Brocade(config)# qos scheduler profile Mixed0 mixed 100 80 60 40 20
```

Syntax: **[no] qos scheduler profile name mixed** *queue4-weight queue3-weight queue2-weight queue1-weight queue0-weight*

The *queue4-weight* variable defines the relative value for queue4 in calculating queue4's allocated bandwidth.

The *queue3-weight* variable defines the relative value for queue3 in calculating queue3's allocated bandwidth.

The *queue2-weight* variable defines the relative value for queue2 in calculating queue2's allocated bandwidth.

The *queue1-weight* variable defines the relative value for queue1 in calculating queue1's allocated bandwidth.

The *queue0-weight* variable defines the relative value for queue0 in calculating queue0's allocated bandwidth

The acceptable range for *queuex-weight* variables is 1-128.

Refer to "Calculating the values for WRR weight-based traffic scheduling" for information on assigning queue0-weight to queue4-weight values.

Egress port and priority based rate shaping

Rate shaping is a mechanism to smooth out the variations in traffic above a certain rate. The primary difference between rate shaping and rate limiting is that in rate limiting, traffic exceeding a certain threshold is dropped. In rate shaping, the traffic that exceeds a threshold is buffered so that the output from the buffer follows a more uniform pattern. Rate shaping is useful when burstiness in the source stream needs to be smoothed out and a more uniform traffic flow is expected at the destination.

NOTE
Because excess traffic is buffered, rate shaping must be used with caution. In general, it is not advisable to rate shape delay-sensitive traffic.

Brocade devices support egress rate shaping. Egress rate shaping is supported per port or for each priority queue on a specified port.

Configuring port-based rate shaping

When setting rate shaping for a port, you can limit the amount of bandwidth available on a port within the limits of the port's rated capacity.

NOTE
The egress rate shaping on a port-based and priority based rate shaper is configured in increments of 1Kbps

These limits provide a minimum and maximum rate that the port can be set to. They also provide the increments at which the port capacity can be set. In operation, you can set any number between the minimum and maximum values. The device will automatically round-up the value to the next higher increment.

To set a 10 Gbps port to the incremental port capacity over 2 Gbps, use the following command.

```
Brocade(config)# interface ethernet 2/2
Brocade(config-if-e10000-2/2)# qos shaper 2000000
```

Syntax: [no] qos shaper *rate*

The *rate* variable sets the rate you want to set for the port within the limits available.

Configuring port and priority-based rate shaping

When setting rate shaping for a priority queue, you can limit the amount of bandwidth available for a specified priority within the limits of the capacity of the port that the priority is configured on. You can set the limit for the priority to any value from one to the port's maximum rating and the device will automatically round-up the value to the next increment supported. This will be a slightly higher value than what you specify with the command. For example, if you set the rate for priority 2 on a 10G port to 2,000,000,100, the actual rate would be slightly higher.

NOTE
The egress rate shaping burst size for a port and priority-based shaper is 4096 bytes.

To set the capacity for priority 2 traffic on a 10 Gbps port to the incremental capacity over 2 Gbps, use the following command.

```
Brocade(config)# interface ethernet 2/2
Brocade(config-config-cpu-port)# qos shaper priority 2 2000000
```

Syntax: [no] qos shaper priority *priority-level rate*

The *priority-level* variable specifies the priority that you want to set rate shaping for on the port being configured.

The *rate* variable sets the rate you want to set for the priority.

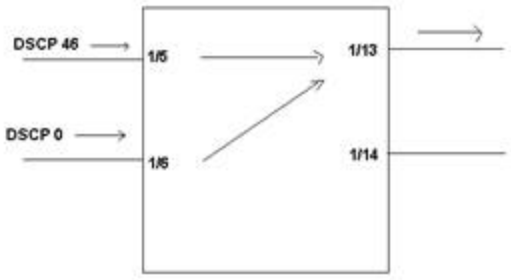
The priority can be from 0 through 7.

The *rate* parameter sets the rate shaping value for the priority queue configured on the CPU port. The acceptable value can be from 1 through 1000000 Kbps.

The **no** option is used to reset the port shaping rate for the priority queue.

Example of configuring Prioritized Voice over Data

When configuring **Prioritize Voice over Data**, use the strict priority method.



In the example below, the DSCP 46 (Voice) is assigned to high priority queue on Ingress port, and goes out of Egress port without changing the DSCP value in the packet.

```
Brocade(config)#int e 1/5
Brocade(config-if-e1000-1/5)#qos scheduler strict
Brocade(config-if-e1000-1/5)#int e 1/6
Brocade(config-if-e1000-1/6)#qos scheduler strict
Brocade(config-if-e1000-1/6)#int e 1/13
Brocade(config-if-e1000-1/13)#qos dscp encode-policy off
Brocade(config-if-e1000-1/13)#qos scheduler strict
```

Use the **show qos-map dscp decode-map default-map command** to verify DSCP 46 is going to the high priority queue as highlighted below.

```
Brocade(config)#show qos-map dscp decode-map default-map
DSCP decode default-map
  DSCP  0 to priority  0 drop-precedence  0
  DSCP  1 to priority  0 drop-precedence  0
  DSCP  2 to priority  0 drop-precedence  1
  DSCP  3 to priority  0 drop-precedence  1
  DSCP  4 to priority  0 drop-precedence  2
  DSCP  5 to priority  0 drop-precedence  2
  DSCP  6 to priority  0 drop-precedence  3
  DSCP  7 to priority  0 drop-precedence  3
  DSCP  8 to priority  1 drop-precedence  0
  DSCP  9 to priority  1 drop-precedence  0
  DSCP 10 to priority  1 drop-precedence  1
  DSCP 11 to priority  1 drop-precedence  1
  DSCP 12 to priority  1 drop-precedence  2
  DSCP 13 to priority  1 drop-precedence  2
  DSCP 14 to priority  1 drop-precedence  3
  DSCP 15 to priority  1 drop-precedence  3
  DSCP 16 to priority  2 drop-precedence  0
  DSCP 17 to priority  2 drop-precedence  0
  DSCP 18 to priority  2 drop-precedence  1
  DSCP 19 to priority  2 drop-precedence  1
  DSCP 20 to priority  2 drop-precedence  2
  DSCP 21 to priority  2 drop-precedence  2
  DSCP 22 to priority  2 drop-precedence  3
  DSCP 23 to priority  2 drop-precedence  3
```

```
  DSCP 24 to priority  3 drop-precedence  0
  DSCP 25 to priority  3 drop-precedence  0
  DSCP 26 to priority  3 drop-precedence  1
  DSCP 27 to priority  3 drop-precedence  1
  DSCP 28 to priority  3 drop-precedence  2
  DSCP 29 to priority  3 drop-precedence  2
  DSCP 30 to priority  3 drop-precedence  3
  DSCP 31 to priority  3 drop-precedence  3
  DSCP 32 to priority  4 drop-precedence  0
  DSCP 33 to priority  4 drop-precedence  0
  DSCP 34 to priority  4 drop-precedence  1
  DSCP 35 to priority  4 drop-precedence  1
  DSCP 36 to priority  4 drop-precedence  2
  DSCP 37 to priority  4 drop-precedence  2
  DSCP 38 to priority  4 drop-precedence  3
  DSCP 39 to priority  4 drop-precedence  3
  DSCP 40 to priority  5 drop-precedence  0
  DSCP 41 to priority  5 drop-precedence  0
  DSCP 42 to priority  5 drop-precedence  1
  DSCP 43 to priority  5 drop-precedence  1
  DSCP 44 to priority  5 drop-precedence  2
  DSCP 45 to priority  5 drop-precedence  2
  DSCP 46 to priority  5 drop-precedence  3
  DSCP 47 to priority  5 drop-precedence  3
  DSCP 48 to priority  6 drop-precedence  0
  DSCP 49 to priority  6 drop-precedence  0
  DSCP 50 to priority  6 drop-precedence  1
  DSCP 51 to priority  6 drop-precedence  1
  DSCP 52 to priority  6 drop-precedence  2
  DSCP 53 to priority  6 drop-precedence  2
  DSCP 54 to priority  6 drop-precedence  3
  DSCP 55 to priority  6 drop-precedence  3
  DSCP 56 to priority  7 drop-precedence  0
  DSCP 57 to priority  7 drop-precedence  0
  DSCP 58 to priority  7 drop-precedence  1
  DSCP 59 to priority  7 drop-precedence  1
  DSCP 60 to priority  7 drop-precedence  2
  DSCP 61 to priority  7 drop-precedence  2
  DSCP 62 to priority  7 drop-precedence  3
  DSCP 63 to priority  7 drop-precedence  3
Brocade(config)#
```

Syntax: show qos-map dscp decode-map default-map

If you want to change from strict to any other policy for weighted or mixed on the Brocade NetIron CES and Brocade NetIron CER, enter the **show qos scheduler profile** command as show below.

```
Brocade#show qos scheduler profile
Index  | Scheme   Type   Pri7   Pri6   Pri5   Pri4   Pri3   Pri2   Pri1   Pri0
-------+----------------+------+------+------+------+------+------+------+----
     0 | Strict
     1 | WRR0     Weight 40     10     20     30     40     50     60     10
     2 | WRR1     Weight 10     10     10     10     10     10     10     10
     3 | WRR2     Weight 10     10     10     10     10     10     10     10
     4 | MIXED0   Weight                    100    80     60     40     20
     5 | MIXED1   Weight                    20     40     60     80     100
     6 | MIXED2   Weight                    30     30     30     30     30
     7 | MIXED3   Weight                    128    128    1      1      1
```

Syntax: show qos scheduler profile

To change the profile from strict to weighted WRR0, use the **qos scheduler WRR0** command on the interface (in this case you would change it on the Outgoing interface ie 1/13. Remember, on the Brocade NetIron CES and Brocade NetIron CER the QOS Scheduler is on Egress interface only.

```
Brocade(config)#sh run int e 1/13
interface ethernet 1/13
 qos scheduler profile WRR0
 enable
```

To change the settings of weighted WRR0, use the **qos scheduler profile WRR0** command in global config mode.

```
Brocade(config)# qos scheduler profile WRRO weighted 40 10 20 30 40 50 60 10
```

Syntax: qos scheduler profile

Issue the **show qos scheduler profile** command to verify the change.

```
Brocade#show qos scheduler profile
Index   | Scheme   Type   Pri7   Pri6   Pri5   Pri4   Pri3   Pri2   Pri1   Pri0
--------+----------------+------+------+------+------+------+------+------+----
     0  | Strict
     1  | WRR0     Weight 40     10     20     30     40     50     60     10
     2  | WRR1     Weight 10     10     10     10     10     10     10     10
     3  | WRR2     Weight 10     10     10     10     10     10     10     10
     4  | MIXED0   Weight                    100    80     60     40     20
     5  | MIXED1   Weight                    20     40     60     80     100
     6  | MIXED2   Weight                    30     30     30     30     30
     7  | MIXED3   Weight                    128    128    1      1      1
```

Syntax: show qos scheduler profile

Clearing traffic manager statistics

You can clear statistic for a Brocade device as shown in the following.

```
Brocade# clear statistics
```

Syntax: clear statistics

Network processor counters displayed

Output from the **show interface** command has been enhanced to provide the following related information:

- The number of packets received at the network processor (NP)
- The number of packets sent from the NP
- The Number of ingress packets dropped at the NP
- The number of packets transmitted from the NP
- The number of packets received by the NP

The following is an example of the new output from the **show interface** command.

```
Brocade(config)# show interface ethernet 3/3
GigabitEthernet3/3 is up, line protocol is up
Hardware is GigabitEthernet, address is 0004.80a0.4052 (bia 0004.80a0.4052)
Configured speed auto, actual 1Gbit, configured duplex fdx, actual fdx
Member of L2 VLAN ID 1, port is untagged, port state is Forwarding
```

```
STP configured to ON, Priority is level0, flow control enabled
mirror disabled, monitor disabled
Not member of any active trunks
Not member of any configured trunks
No port name
MTU 1544 bytes, encapsulation ethernet
300 second input rate: 754303848 bits/sec, 1473249 packets/sec, 89.57% utilization
300 second output rate: 754304283 bits/sec, 1473250 packets/sec, 89.57%
utilization
1015230949 packets input, 64974783168 bytes, 0 no buffer
Received 0 broadcasts, 0 multicasts, 1015230949 unicasts
0 input errors, 0 CRC, 0 frame, 0 ignored
0 runts, 0 giants
NP Ingress dropped 0 packets
1015231660 packets output, 64974824768 bytes, 0 underruns
Transmitted 0 broadcasts, 0 multicasts, 1015231660 unicasts
0 output errors, 0 collisions
```

Multicast queue size, flow control, rate shaping and egress buffer threshold

There are four internal priorities for multicast or broadcast traffic. These four priorities are mapped from the device's eight internal priorities as described inTable 16

TABLE 16 Mapping between multicast or broadcast and internal forwarding priorities

Internal Forwarding Priority	0,1	2,3	4,5	6,7
Multicast Internal Priority	0	1	2	3

The internal forwarding priority of a multicast or broadcast packet is determined from the packet's IEEE 802.1p priority, incoming port priority or IP ToS or DSCP as described in the "Default QoS mappings" on page 79. Four multicast queue types (0 to 3) are used for multicast internal priorities 0 to 3 respectively.

NOTE
Instead of ACL priority, use VLAN or port priority to prioritize multicast traffic.

Configuring multicast queue size

The following example configures a 2 MByte queue size for queue 0.

```
Brocade(config)# qos multicast-queue-type 0 max-queue-size 2048
```

Syntax: [no] qos multicast-queue-type *queue-number* **max-queue-size** *queue-size*

The *queue-number* variable specifies the queue that you want to configure a maximum size for. Possible values are 0 - 3.

The *queue-size* variable specifies size in KBytes that you want to set as the maximum value for the specified multicast queue. Possible values are 1 - 32768 KBytes. The default queue size is 1 Mbyte. This command is applied per device and takes effect on all Traffic Managers within the configured device.

Configuring multicast flow control

Flow controls are available from egress to Ingress, and from fabric to Ingress. At the egress of each Traffic Manager, there are pre-determined thresholds for consumed resources and available resources and separate thresholds for guaranteed multicast or broadcast traffic and best-effort multicast or broadcast traffic. When a threshold is crossed, flow control can be triggered and multicast or broadcast traffic of the corresponding class is stopped at Ingress until resources are below the threshold again. Flow control is disabled by default and can be enabled on an interface using the command shown in the following.

```
Brocade(config)# interface ethernet 2/2
Brocade(config-if-e10000-2/2)# qos multicast flow-control
```

Syntax: [no] qos multicast flow-control

This command changes the flow control setting on the Traffic Manager where the interface resides.

Configuring multicast rate shaping

You can specify either guaranteed or best effort multicast rate shaping for a port in Kilobits per second. Multicast rate shaping is configured per-port to the Ingress port.

The following example changes the best-effort multicast traffic rate to 10 Mbps.

```
Brocade(config)# interface ethernet 2/2
Brocade(config-if-e10000-2/2)# qos multicast shaper best-effort rate 10000
```

Syntax: [no] qos multicast shaper [guaranteed | best-effort] rate *bandwidth*

The **guaranteed** option specifies that the multicast or broadcast shaper applies only to internal multicast priority 3 (the highest multicast priority) traffic.

The **best-effort** option specifies that the multicast or broadcast shaper applies to internal multicast priority 0, 1 and 2 traffic only.

The *bandwidth* variable specifies the maximum bandwidth in Kbps for best effort or guaranteed multicast traffic scheduled by the Traffic Manager across the switch fabric.

Configuration considerations for multicast rate shaping

When applied to a port, the **qos multicast shaper** configuration is applied to all ports on the Interface module that use the same Traffic Manager as the configured port. This is unlike the behavior of rate shaping applied for unicast traffic. The relationship between ports and Traffic Managers is defined in the following tables: Table 35 on page 141 and Table 36 on page 141.

Example

In the following example, multicast rate shaping is applied to port 1 of a Brocade NetIron CER or Brocade NetIron CES.

```
Brocade(config)# interface ethernet 1/1
Brocade(config-if-e10000-1/1)# qos multicast shaper best-effort rate 10000
```

In this example, the configuration will apply to Ingress traffic that arrives on either port 1 or port 2 of the device.

NOTE
When a **qos multicast shaper** command is configured for a port, the configuration command is placed in the running config for all ports that belong to the same Traffic Manager. In the example, that would mean that the **qos multicast shaper best-effort rate 10000** command would appear in the interface configuration section for ports 1 and 2 on the Interface Module.

Configuring multicast egress buffer threshold

With the current configuration egress buffer threshold per port are set to 50% of total egress buffer size, with the new command you can set multicast egress buffer threshold up to 95% of total egress buffer size which helps in reducing egress packet drops when there is a high multicast traffic.

NOTE
It is recommended to use this command when a sudden burst is seen in multicast traffic and not for general use.

The following example explains how to set the thresholds for individual ports so that the port can use the total egress buffer. This is useful when the multicast traffic is very high.

```
Brocade#config terminal
Brocade(config)#interface ethernet 1/1
Brocade(config-if-e10000-1/1)# qos multicast egress-max-buffer port 95% 90%
```

Syntax: qos multicast egress-max-buffer port { [*guaranteed_max_buffer*] [*best-effort_max_buffer*] }

The **guaranteed_max_buffer** specifies the maximum buffer size allowed per port for guaranteed traffic flow (multicast port priority 3). Specified as percentage of total buffer size.

The **best-effort_max_buffer** specifies the maximum buffer size allowed per port for guaranteed traffic flow (multicast port priority 2-0). Specified as percentage of total buffer size.

Additionally you can also have the threshold configured for individual ports so that each port has its dedicated buffer space.

The example uses a 8x10 line card and has four ports per TM. The buffer size is divided into 4 times the total buffer size so that each port has its dedicated buffer space.

```
Brocade#config terminal
Brocade(config)#interface ethernet 1/1
Brocade(config-if-e10000-1/1)# qos multicast egress-max-buffer port 24% 23%
```

Syntax: qos multicast egress-max-buffer port { [*guaranteed_max_buffer*] [*best-effort_max_buffer*] }

The **guaranteed_max_buffer** specifies the maximum buffer size allowed for guaranteed traffic flow (multicast port priority 3). Specified as percentage of total buffer size.

The **best-effort_max_buffer** specifies the maximum buffer size allowed for guaranteed traffic flow (multicast port priority 2-0). Specified as percentage of total buffer size.

Ingress traffic shaping per multicast stream

Internet Protocol Television (IPTV) multicast streams on an individual inbound physical port are rate shaped to a specified rate and are prioritized over the broadcast or unknown-unicast traffic. Each IPTV multicast stream is queued separately and is scheduled independently to the outbound ports. The IPTV rate shaping reduces burstiness in the source stream.

NOTE
The number of active IPTV multicast streams for which per stream ingress shaping can be applied is limited to 512.

Implementation considerations

The considerations for implementing the multicast rate shaping are as follows:

- At least the rate or the priority value must be specified.
- A maximum of 32 profiles and 64 profile-ACL bindings are allowed for multicast traffic.
- A single profile can be bound to multiple ACLs.
- An ACL can be associated only to one profile at a time.
- Either standard or extended ACLs can be used for multicast traffic shaping.
  - When a standard ACL is used, the address specified is treated as a group address and not as a source address.
  - When an extended ACL is used, the source and the destination addresses are treated as the source and group address of the multicast stream, respectively.

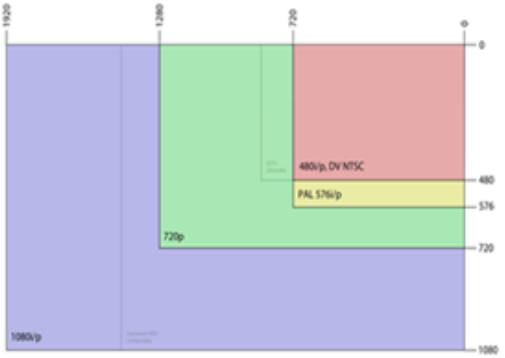
Figure 1 shows the type of IPTV channels and the bandwidth requirements for each type of channel. The following conventions are used in the figure:

- SDTV denotes Standard Definition Television
- HDTV denotes High Definition Television

NOTE
This feature is supported only on the 4x10 and 24x1 interface modules.

FIGURE 1 IPTV Bandwidth Requirements

Channel Encoding	Per Channel	Resolution @ Frames/sec
MPEG-2 SDTV	3-6 Mbps	480i @ 29 576i @ 25
MPEG-2 HDTV	14-20 Mbps	1080i @ 25,29,30 720p @ 50,59,60
H.264/MPEG-4 SDTV	1-4 Mbps	1,280×720@68.3 1,920×1,080@30.1 2,048×1,024@30.0
H.264/MPEG-4 HDTV	6-14 Mbps	1280×720 pixels (720p) 1920×1080 pixels (1080i/1080p)



Channel Encoding Relative TV Format

Configuring multicast traffic policy maps

You can define profiles to match the IPTV multicast traffic of the individual ingress streams. To configure a policy map for the multicast streams, enter the following command.

```
Brocade(config)# policy-map multicast sd_prof rate 2000 burst-size 2500 priority 2
queue-type 3
```

Syntax: [**no**] **policy-map multicast** *profile_name* [**rate** *r*] [**burst-size** *b*] [**priority** *0-7*] [**queue-type** *0-3*]

The *profile_name* is used to provide the parameters for traffic policing the multicast traffic.

The **rate** *r* variable specifies the shaping rate in bits per second. The value ranges from 100 kilobits per second (Kbps) through 20 gigabits per second (Gbps).

The **priority** *0-7* variable specifies the multicast traffic priority. The default value is four.

The **burst-size** *b* variable specifies the maximum number of bytes the multicast traffic is allowed to burst. The value ranges from 3 kilobytes (KB) through 128 KB. The default value is four KB.

The **queue-type** *0-3* variable specifies the queue type for which you want to set the priority. The default value is two. Optionally, you can specify the **burst-size** *b* and the **queue-type** *0-3* value to define a profile.

To delete a defined profile, enter the following command with the profile name.

```
Brocade(config)# policy-map multicast sd_prof
```

The **no** form of the command resets the parameters to their default values.

NOTE
The rate and the priority value cannot be reset for a defined profile.

Binding multicast traffic policy maps

NOTE
A profile must exist in the configuration before it can be used for binding.

A standard or an extended ACL is used to define the IPTV streams that can be bound to a defined profile. The profile binding associates the properties of the profile to all the IPTV streams identified by the ACL. Binding of multicast streams can be done for Layer 3 multicast routing and Layer 2 multicast snooping.

Profile binding for Layer 3 multicast routing

You can bind a defined profile to a defined ACL for the default VRF or a specific VRF.

To bind the profile to the ACL in the default VRF, enter the following commands.

```
Brocade(config)# ip multicast-routing policy-map r1 sd-1
Brocade(config)# ipv6 multicast-routing policy-map r1q0 sdv61
```

To bind the profile to the ACL for a specified VRF, enter the following commands within the VRF "red" configuration context.

```
Brocade(config)# ip vrf red
Brocade(config-vrf-red)# address-family ipv4
Brocade(config-vrf-red-ipv4)# ip multicast-routing policy-map r1 sd-1
Brocade(config-vrf-red-ipv4)# exit-address-family
Brocade(config-vrf-red)# exit-vrf

Brocade(config)# ip vrf red
Brocade(config-vrf-red)# address-family ipv6
Brocade(config-vrf-red-ipv6)# ipv6 multicast-routing policy-map r1q0 sdv61
Brocade(config-vrf-red-ipv6)# exit-address-family
Brocade(config-vrf-red)# exit-vrf
```

Syntax: [**no**] **ip multicast-routing policy-map** *profile_name acl_id* | *acl_name*

The **ip multicast-routing policy-map** specifies ACL binding for IPv4 multicast routing.

The *profile_name* variable specifies the profile name of the multicast stream.

The *acl_id* | *acl_name* variable specifies the number and name of the standard ACL or an extended ACL. Enter a number from 1 through 99 for a standard ACL, and a number from 100 through 199 for an extended ACL.

Syntax: [**no**] **ipv6 multicast-routing policy-map** *profile_name acl_id* | *acl_name*

The **ipv6 multicast-routing policy-map** specifies ACL binding for IPv6 multicast routing.

The **no** form of the command removes the profile binding with the ACL in default VRF.

Profile binding for Layer 2 multicast snooping

You can bind a defined profile to a defined ACL per VLAN or per VPLS instance. To bind the profile to the ACL on VLAN 10, enter the following commands.

```
Brocade(config)# vlan 10
Brocade(config-vlan-10)# ip multicast policy-map r2q1 2
Brocade(config-vlan-10)# ipv6 multicast policy-map r4q3 sdv64
```

Syntax: [**no**] **ip multicast policy-map** *profile_name acl_id* | *acl_name*

The **ip multicast policy-map** specifies the ACL binding for IPv4 multicast snooping.

Syntax: [**no**] **ipv6 multicast policy-map** *profile_name acl_id | acl_name*

The **ipv6 multicast policy-map** specifies the ACL binding for IPv6 multicast snooping.

The **no** form of the command removes the profile binding with the ACL on the VLAN or VPLS.

In the following example, binding for Layer 2 multicast snooping is applied to VPLS instance V1.

```
Brocade(config)# router mpls
Brocade(config-mpls)# vpls v1 10
Brocade(config--mpls-vpls-v1)# multicast policy-map r2q1 2
Brocade(config--mpls-vpls-v1)# multicast policy-map r4q3 sdv64
```

Syntax: [**no**] **multicast policy-map** *profile_name acl_id | acl_name*

NOTE
A profile that is bound cannot be deleted.

Configuration example for rate shaping IPTV multicast stream

The following example shows how to rate shape a multicast stream. In this example, to rate shape a multicast stream, profiles (sd_prof and hd_prof) are defined with rate and priority, ACLs (hd_streams and sd_streams) are configured which permit packets from four host IP addresses and denies all packets that are not explicitly permitted by the first four ACL entries, and then the profiles are bound to the defined ACLs.

```
Brocade(config)# ip access-list standard hd_streams
Brocade(config-std-nacl)# permit host 239.1.1.200
Brocade(config-std-nacl)# permit host 239.1.1.201
Brocade(config-std-nacl)# permit host 239.1.1.202
Brocade(config-std-nacl)# permit host 239.1.1.203
Brocade(config-std-nacl)# deny any
Brocade(config-std-nacl)# exit

Brocade(config)# ip access-list extended sd_streams
Brocade(config-ext-nacl)# permit ip any 239.1.1.1/32
Brocade(config-ext-nacl)# permit ip any 239.1.1.2/32
Brocade(config-ext-nacl)# permit ip any 239.1.1.3/32
Brocade(config-ext-nacl)# permit ip any 239.1.1.5/32
Brocade(config-ext-nacl)# deny ip any any
Brocade(config-ext-nacl)# exit

Brocade(config)# policy-map multicast profile sd_prof rate 2000
Brocade(config)# policy-map multicast profile hd_prof rate 14000 queue-type 2

Brocade(config)# ip multicast-routing policy-map sd_prof sd_streams
Brocade(config)# ip multicast-routing policy-map hd_prof hd_streams
```

Naming convention used in example

- sd_prof = Standard Definition profile
- hd_prof = High Definition profile
- sd_streams = Standard Definition TV stream
- hd_stream = High Definition TV stream

Tuning multicast parameters

NOTE

The following commands are specific to the Brocade NetIron CER and Brocade NetIron CES devices:

- Mutilcast to Unicast descriptor ratio
- Multicast weight
- Multicast descriptor limit
- Multicast Traffic class mapping to Fabric Traffic class

Multicast to Unicast descriptor ratio

The *qos-multicast mcast-unicast-desc-ratio* command defines the number of Multicast Descriptors Duplications per Unicast Descriptor. The default value of this field is 1 which means there is one multicast descriptor replication per unicast. In multicast intensive environment user may want this parameter to be increased to 2 or 4 to give a better replication ratio to multicast traffic.

```
Brocade(config)# qos-multicast mcast-ucast-desc 1
```

Syntax: [**no**] **qos-multicast mcast-ucast-desc-ratio** *ratio-value* [ppcr <ppcr_id>]

The *ratio_value* parameter has a default value of 1. Valid values include 1, 2, and 4.

The *ppcr_id* parameter is an optional keyword. Valid values: 0 - for ports e1/1 - e1/24; 1 - for ports e1/25 - e1/28; 2 - for ports e2/1 - e2/2.

Without the *ppcr id* the command will set the descriptor ratio on all packet processors. If the *ppcr id* is specified then the command will set the descriptor ratio on only that packet processor.

Multicast weight

The Unicast/Multicast Arbiter in Hardware is WRR (weighted round robin). The defaults weight of Multicast and Unicast is assigned as 1. In a multicast intensive environment you may want to assign a higher multicast weight.

```
Brocade(config)# qos-multicast mcast-weight 15
```

Syntax: [**no**] **qos-multicast mcast-weight** *weight-value* [ppcr <ppcr_id>]

The *weight_value* parameter has a default value of 1. Valid values include 0 - 15.

The *ppcr_id* parameter is an optional keyword. Valid values: 0 - for ports e1/1 - e1/24; 1 - for ports e1/25 - e1/28; 2 - for ports e2/1 - e2/2.

Without the *ppcr id* the command will set the *weight value* on all packet processors. If the *ppcr id* is specified then the command will set the *weight value* on only that packet processor.

Multicast descriptor limit

Brocade NetIron CES and Brocade NetIron CER require a descriptor to place the packet in queue. If due to traffic conditions the descriptor limit is reached, the subsequent packets are dropped until the descriptors are available again. This parameter specifies the maximum number of descriptors that can be allocated for multicast packets. In a multicast intensive environment you may want to assign a higher value for the multicast descriptor limit. The default value of this is 2048.

```
Brocade(config)# qos-multicast mcast-desc-limit 8000
```

Syntax: [**no**] **qos-multicast mcast-desc-limit** *limit-value* [ppcr <ppcr_id>]

The *limit_value* parameter has a default value of 2048. Valid values include 0 - 8192.

The *ppcr_id* parameter is an optional keyword. Valid values: 0 - for ports e1/1 - e1/24; 1 - for ports e1/25 - e1/28; 2 - for ports e2/1 - e2/2.

Without the *ppcr id* the command will set the *limit value* on all packet processors. If the *ppcr id* is specified then the command will set the *limit value* on only that packet processor.

Multicast Traffic class mapping to Fabric Traffic class

The multicast traffic class is mapped to Fabric traffic class. Multicast TC 0, 1 is mapped to Fabric TC 0, Multicast TC 2,3 is mapped to Fabric TC 1, Multicast TC 4, 5 is mapped to Fabric TC 2, Multicast TC 6,7 is mapped to fabric TC 3. In the Egress Pipeline if Fabric TC is 2 and it is a multicast packet, the buffering of the descriptors occurs in a deep queue. This is a queue generally used in IPTV applications. You may want to change this mapping to assign all the multicast Traffic class to Fabric Traffic class 2 to utilize the deep buffer queue. This will improve the performance of IPTV applications.

```
Brocade(config)# qos-multicast mcast-tc-to-iptv all
```

Syntax: [**no**] **qos-multicast mcast-tc-to-iptv** *tc-value* | *all*

The *tc_value* parameter is the Multicast traffic class that needs to mapped to Fabric traffic class 2. Valid values include 0 - 7. All will map all multicast traffic classes to Fabric class 2.

Chapter

Configuring Quality of Service for the Brocade NetIron XMR and Brocade MLX series 4

This chapter describes how QoS is implemented and configured in the Brocade device. The chapter contains the following sections:

- **Ingress Traffic Processing through a device** – This section describes the QoS operation on ingress traffic of a Brocade device. Refer to "Ingress Traffic processing through a device" on page 72.
- **Creating an Ingress Decode Policy Map** – This section describes the policy maps used to determine the priority and drop precedence values that will be assigned to the packet within the device. Refer to "Creating an Ingress decode policy map" on page 74.
- **Forcing or Merging Priority of a Packet** – This section describes how once a packet's Ingress priority has been mapped, the values used for processing on the device are determined by either forcing or merging. Refer to "Forcing or merging the priority of a packet" on page 74.
- **Forcing or Merging the Drop Precedence of a Packet** – This section describes how once a packet's Ingress drop precedence has been mapped, the values used for processing on the device are determined by either forcing or merging. Refer to "Forcing or merging the drop precedence of a packet" on page 75.
- **Egress Traffic Processing Exiting a device** – This section describes the QoS operation on Egress Traffic of a Brocade device. This process involves marking packets as they leave a device on the Egress port. Refer to "Egress Traffic processing exiting a device" on page 76.
- **Creating an Egress Decode Policy Map** – This section describes the policy maps used to determine the priority and drop precedence values that a packet will carry when it exits the device. Refer to "Creating an egress encode policy map" on page 76.
- **Backward Compatibility with Pre-03.8.00 QoS Configurations** – This section describes the how previous configurations of the QoS feature on Brocade devices work with the enhanced QoS feature. Refer to "Backward compatibility with pre-03.8.00" on page 77.
- **Default QoS Mappings** – This section describes the default mappings for the following PCP Encode, PCP Decode, DSCP Encode, DSCP Decode, EXP Encode and EXP Decode. Refer to "Default QoS mappings" on page 79.
- **Configuring QoS** – This section describes all of the procedures required for both Ingress and Egress QoS Configuration. Refer to "Configuring QoS" on page 86.
- **Displaying QoS Information** – This section describes how to enable and display the following QoS information: QoS Configuration Information and QoS Packet and Byte Statistics. Refer to "Displaying QoS information" on page 113.
- **Configuring Port-level QoS Commands on LAG Ports** – When applying port-level QoS commands to ports in a LAG, the rules can be different according the configuration as described in "Configuring port-level QoS commands on LAG ports" on page 112.
- **Determining Packet Drop Priority using WRED** – Weighted Random Early Detection (WRED) provides a mechanism for determining which packets to drop in a congested network. This section describes how WRED works. Refer to "Weighted Random Early Discard (WRED)" on page 118.

- **Configuring Packet Drop Priority using WRED** – This section describes how to configure Weighted Random Early Detection (WRED). Refer to "Configuring packet drop priority using WRED" on page 120.
- **Scheduling Traffic for Forwarding** – The Brocade supports six different schemes for prioritizing traffic for forwarding in a congested network. This section describes each of these schemes and how to configure them. Refer to "Scheduling traffic for forwarding" on page 126.
- **Traffic Manager Statistics Display -** Counters have been introduced to track the packets and bytes that enter the Ingress traffic manager and exit the egress traffic manager. Refer to QoS for NI-MLX-1Gx48T modules - "Traffic manager statistics display" on page 138.
- **QoS for NI-MLX-1Gx48-T modules** - The NI-MLX-1Gx48-T module supports 48 1G port. In a fully loaded 32 slot chassis, there are only 8 queues supported on the TM port. Refer to "QoS for NI-MLX-1Gx48-T modules" on page 147.
- **Aggregated TM VOQ Statistics Collection** - Refer to "Aggregated TM VOQ statistics collection" on page 148.
- **QoS Commands affected by Priority Queues** - Refer to "QoS commands affected by priority queues" on page 157.
- **Enhanced Packet Buffering** – This section describes how to configure Enhanced Packet Buffering for prioritizing the buffer pool and VOQ Queue size. This section describes each of these schemes and how to configure them. Refer to "Enhanced buffer management for NI-MLX-10Gx8 modules and NI-X-100Gx2 modules" on page 159.

Ingress Traffic processing through a device

The QoS operation on Ingress Traffic of a Brocade device involves reception and processing of packets based upon priority information contained within the packet. As the packets are processed through the device, there are several opportunities to influence the processing by configuration as described in the steps below.

1. Derive priority and drop precedence from the packets PCP (IEEE 802.1p) value. The Priority Code Point (PCP) is a 3-bit field within an IEEE 802.1Q tagged frame that is used to convey the priority of the frame. By using a mapping table, the 3-bit PCP field can be decoded to derive priority and drop precedence information.

   NOTE
   The PCP field was formerly called IEEE 802.1p.

2. Derive priority and drop precedence from the packets EXP value.
3. Derive priority and drop precedence from the packets DSCP value.

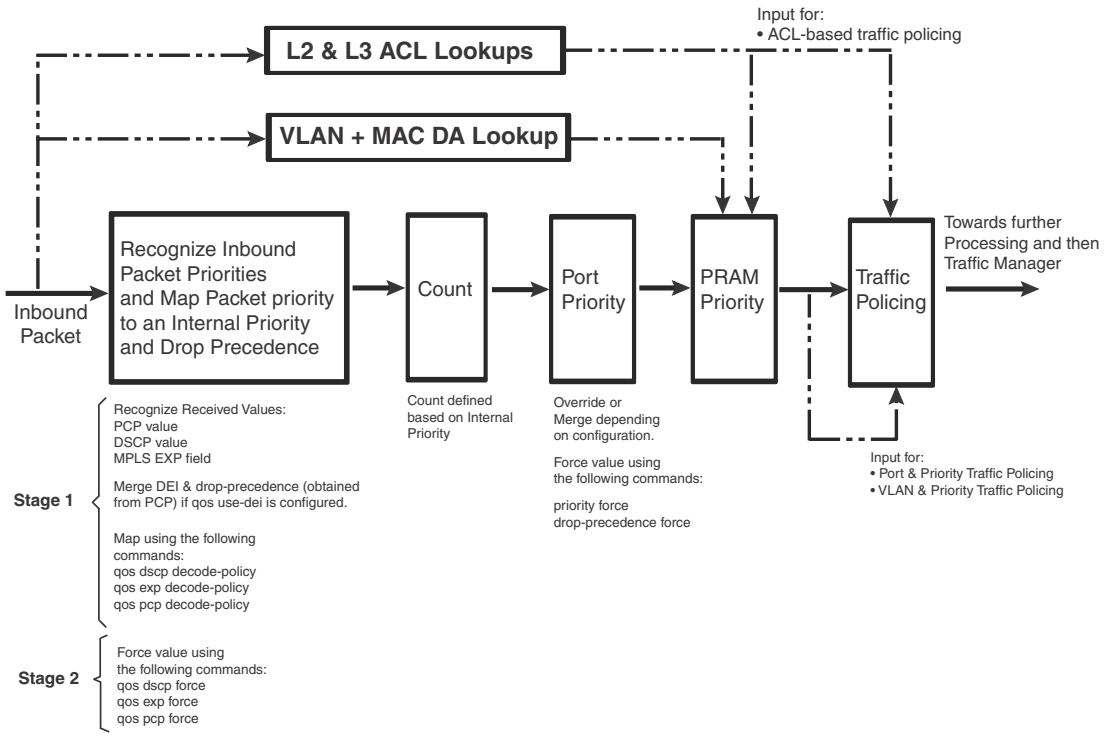
   NOTE
   DSCP encoding and decoding are not supported for MPLS packets.

4. Merge or force the priorities described in steps 1 through 3.
5. Merge or force the priority and drop precedence value based on the value configured for the physical port.
6. Merge or force the priority value based on the value configured for the VLAN.

7. Merge or force the priority value based on an ACL look-up. This is used for setting a a specific priority for and L2, L3 or L4 traffic flow.

This process is described in Figure 2.

FIGURE 2 Logic flow of Ingress QoS processing



Recognizing inbound packet priorities and mapping to internal priority

The processes performed in the first block of Figure 2 can be described in two stages as described in the following:

Stage 1
Collect priority and drop precedence information from various portions of the packet header

- If a packet's EtherType matches 8100 or the port's EtherType, derive a priority value and drop precedence by decoding the PCP value.
- If the **qos use-dei** command is configured, the bit between the VLAN ID and PCP in the VLAN tag will be interpreted as a drop precedence and priority value.
- For MPLS packets, derive a priority value and drop precedence by decoding the EXP bits.
- For IPv4 or v6 packets, derive a priority value and drop precedence by decoding the DSCP bits.
- The derived values for PCP, EXP and DSCP are mapped to either a default map or a configured Ingress Decode Policy Map.

- To assist the device in the decoding process described in “stage 1” decode-map tables are defined.

Stage 2
Determine if a priority value should be forced or merged

- If a packet’s EtherType matches 8100 or the port’s EtherType, derive a priority value and drop precedence by decoding the PCP value.
- If the **qos pcp force** command is configured on the port, the priority and drop precedence values are set to the value read from the PCP bits.
- If the **qos exp force** command is configured on the port, the priority and drop precedence values are set to the value read from the MPLS EXP bits.
- If the **qos dscp force** command is configured on the port, the priority and drop precedence values are set to the value read from the DSCP bits.
- If none of the qos force commands are configured, the priority and drop precedence values are set for IPv4 or v6 packets and MPLS packets as described in the following:

  For IPv4 or v6 Packets: Priority and drop precedence values obtained as a merge of the decoded PCP and decoded DSCP values.

  For MPLS Packets: Priority and drop precedence values obtained as a merge of the decoded PCP and decoded EXP values.

Creating an Ingress decode policy map

Once a packet’s Ingress priority has been recognized for the PCP, DSCP and EXP values, those values are matched against a policy map to determine the priority and drop precedence values that will be assigned to the packet within the device. The maps used can be either:

- Default policy maps described in “Default QoS mappings” on page 79.
- User-configured policy maps that are defined as described:

  dscp decode-map *decode-map-name* – This command allows you to map a recognized DSCP value to a value that you define.

  pcp decode-map *decode-map-name* – This command allows you to map a recognized PCP value to a value that you define.

  exp decode-map *decode-map-name* – This command allows you to map a recognized MPLS EXP value to a value that you define.

Forcing or merging the priority of a packet

Once a packet’s Ingress priority has been mapped, the values that will be used for processing on the device are determined by either forcing or merging.

There are a variety of commands to “force” the priority of a packet based on the following criteria:

- Forced to a priority configured for a specific Ingress port. The **priority force** command is configured at the interface where you want it to be applied.

- Forced to a priority configured for a specific VLAN. The **priority force** command is configured at the VLAN where you want is to be applied.
- Forced to a priority that is obtained from the DSCP priority bits. The **qos- dscp force** command is configured at the interface where you want it to be applied.
- Forced to a priority that is obtained from the EXP priority bits. The **qos- exp force** command is configured at the interface where you want it to be applied.
- Forced to a priority that is obtained from the PCP priority bits. The **qos- pcp force** command is configured at the interface where you want it to be applied.
- Forced to a priority that is based on an ACL match. The **priority-force** keyword can be used within an ACL to apply a priority to specified traffic.

If multiple commands containing the **priority force** keyword are specified, the command with the highest precedence will take effect as determined by the following order.

1. ACL match (if the **qos-tos mark cos** command is configured, it has the same #1 priority precedence as ACL match). Refer to "Specifying the trust level and enabling marking" on page 108 for details.
2. VLAN priority

NOTE
VLAN priority works differently for Layer 2 and Layer 3 traffic. If you apply VLAN priority to a physical port for layer 2 traffic then it will apply VLAN priority and will change traffic accordingly. However, if you apply VLAN priority to a physical port for layer 3 traffic then it will not apply and will not change traffic. VLAN priority will only apply to layer 3 traffic if the port is a VE port.

3. Physical port priority value.
4. DSCP value in an incoming IPv4 or v6 packet.
5. EXP value in an incoming MPLS packet.
6. PCP value in a tagged frame or PCP field or .1ad DE

Details of how to configure the force commands are provided in "Configuring a force priority" on page 96.

Forcing or merging the drop precedence of a packet

Once a packet's Ingress drop precedence has been mapped, the values that will be used for processing on the device are determined by either forcing or merging.

There are a variety of commands to "force" the drop precedence of a packet based on the following criteria:

- Forced to a drop precedence configured for a specific Ingress port. The **drop-precedence force** command is configured at the interface where you want it to be applied.
- Forced to a drop precedence that is obtained from the DSCP priority bits. The **qos dscp force** command is configured at the interface where you want it to be applied.
- Forced to a drop precedence that is obtained from the EXP priority bits. The **qos exp force** command is configured at the interface where you want it to be applied.
- Forced to a drop precedence that is obtained from the PCP priority bits. The **qos pcp force** command is configured at the interface where you want it to be applied.

- Forced to a drop precedence that is based on an ACL match. The **drop-precedence-force** keyword can be used within an ACL to apply a priority to specified traffic.

If multiple commands containing the **force** keyword are specified, the command with the highest precedence will take effect as determined by the following order.

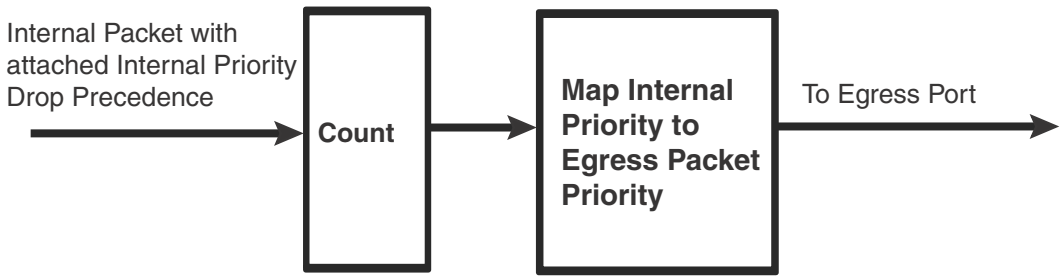
1. ACL match
2. Physical port's drop precedence value
3. DSCP value in an incoming IPv4v6 packet
4. EXP value in an incoming MPLS packet
5. PCP value in a tagged frame or PCP field or .1ad DE

Details of how to configure the force commands are provided in "Configuring a force priority" on page 96.

Egress Traffic processing exiting a device

The QoS operation on Egress Traffic of a Brocade device involves marking packets as they leave a device on the egress port. As the packets are prepared to exit the device you can set the PCP, DSCP, and EXP values in the packet headers. This process is described in Figure 3.

FIGURE 3 Logic flow of Egress QoS processing



Creating an egress encode policy map

The QoS value that a packet carries in its header when it exits a Brocade device on an egress interface is determined by a specified mapping. Unless configured, this value once determined is placed in an internal queue by using one of the default maps described in "Default QoS mappings" on page 79. Alternately, the following commands can be used to define an alternate mapping:

- **pcp encode-map** *encode-map-name* – This command allows you to map the internal priority and drop precedence values of a packet into the PCP code point.
- **dscp encode-map** *encode-map-name* – This command allows you to map the internal priority and drop precedence values of a packet into the DSCP code point.

NOTE
DSCP encoding and decoding are not supported for MPLS packets.

- **exp encode-map** *encode-map-name* – This command allows you to map the internal priority and drop precedence values of a packet into the EXP code point.

Backward compatibility with pre-03.8.00

A number of the commands used in prior releases for QoS configuration have been deprecated and the functions performed by them has been taken over by new commands. The **qos-tos-trust** and **qos-tos mark** commands are still operative although their use is discouraged. Additionally, the **qos-tos map dscp-priority** commands that are in a current configuration are converted to new commands during a software upgrade. For details concerning each of these compatibility issues, refer to the following sections:

- "Commands deprecated in version 03.8.00"
- "qos-tos trust and qos-tos mark commands"
- "qos-tos trust and qos-tos mark commands"

Commands deprecated in version 03.8.00

Table 17 describes each of the commands that were deprecated with version 3.8.00, the purpose of the command, and the equivalent functionality available in versions 03.8.00 and later.

TABLE 17 Deprecated commands from pre-03.8.00 releases

Pre-03.8.00 command	Purpose of the pre-03.8.00 command	Equivalent in advanced QoS infrastructure
`port-priority`	Initializes the global TOS or DSCP table. Once executed, the priority from the DSCP bits is merged with other priorities **NOTE:** By default, DSCP processing is ignored.	• Default behavior is now to process DSCP. • The global DSCP table is now extended into a per-port DSCP table. • If not desired use the following command: `qos dscp encode-policy all-zero-map`
`qos-tos` at the interface level	Initializes the global TOS or DSCP table. Once executed, the priority from the DSCP bits is merged with other priorities	• Default behavior is now to process DSCP. • The global DSCP table is now extended into a per-port DSCP table. • If not desired use the following command at the physical interface level: `qos dscp encode-policy all-zero-map`
`merge-egress-priorities`	Allows a packet to be sent out with the higher of two possible values: • the initial IEEE 802.1p value that the packet arrived with • a new (higher) priority that the packet has been forced to.	This feature is no longer supported because it is not useful. Typically, if a packet is remarked, we want the remarked priority value to be reflected in the outgoing packet and not be a merge of two priorities.
`qos-tos map dscp-priority` at the global configuration level	Initializes entries in the global TOS or DSCP table.	These commands are deprecated and converted into a special decode map named: USER_DSCP_MAP. This process is described in detail in "DSCP-priority mapping commands" on page 78.

qos-tos trust and qos-tos mark commands

The **qos-tos trust** and **qos-tos mark** commands are retained in 03.8.00 and later versions of the Multi-Service IronWare software as described in the following:

- The primary use of these commands is for packet remarking (without changing the internal priority of the packet if desired)

  qos-tos trust indicates the priority to be trusted on the Ingress interface (cos, dscp or ip-prec).

  qos-tos mark indicates which priority is to be marked on the Egress interface (cos or dscp).

- **qos-tos trust** and **qos-tos mark** commands are both applied at the Ingress interface (physical or virtual).

For instructions concerning the configuration of these commands, refer to "Specifying the trust level and enabling marking" on page 108.

NOTE
Because these commands use L4 CAM entries, this may not be used in conjunction with L2 or L3 Multicast Forwarding, or other ACL's.

DSCP-priority mapping commands

In releases prior to 03.8.00, there is a command that globally maps the DSCP priorities in the incoming packets to an internal priority. This command: "**qos-tos map dscp-priority**" is not available in release 03.8.00 and later. If you have configured the **qos-tos map dscp-priority** command in a previous release, the operation of your device will not be changed by upgrading but the configuration commands in the configuration file will be automatically updated to the commands introduced in release 03.8.00. This will be done by the method described in the following:

- the mapping defined by the **qos-tos map dscp-priority** commands is converted to a qos-mapping configuration of a decode map titled USER_DSCP_MAP.
- If the **port-priority** command is configured, the **qos dscp encode-policy** command is included in the configuration and the decode map titled: USER_DSCP_MAP is applied globally on the device.

Mapping from qos-tos mapping to USER_DSCP_MAP

As described previously, in release 03.8.00 and later, a configuration containing **qos-tos map dscp-priority** commands is converted to a **qos-mapping** configuration of a decode map titled USER_DSCP_MAP. The following **qos-tos map dscp-priority** commands map various DSCP priority values to internal priorities from 0 to 7.

```
Brocade(config)# port-priority
Brocade(config)# qos-tos map dscp-priority 0 2 3 4 to 1
Brocade(config)# qos-tos map dscp-priority 8 to 5
Brocade(config)# qos-tos map dscp-priority 16 to 4
Brocade(config)# qos-tos map dscp-priority 24 to 2
Brocade(config)# qos-tos map dscp-priority 32 to 0
Brocade(config)# qos-tos map dscp-priority 40 to 7
Brocade(config)# qos-tos map dscp-priority 48 to 3
Brocade(config)# qos-tos map dscp-priority 56 to 6
```

The following example displays the USER_DSCP_MAP configuration and binding.

```
qos-mapping
  dscp decode-map USER_DSCP_MAP
    dscp-value 32 to priority 0
    dscp-value 0 to priority 1
    dscp-value 2 3 to priority 1
    dscp-value 4 to priority 1
    dscp-value 24 to priority 2
    dscp-value 48 to priority 3
    dscp-value 16 to priority 4
    dscp-value 8 to priority 5
    dscp-value 56 to priority 6
    dscp-value 40 to priority 7
qos dscp decode-policy USER_DSCP_MAP
```

NOTE
If the **port-priority** command was not configured in the pre-converted configuration, the **qos dscp encode-policy USER_DSCP_MAP** command will not be added to the converted file and you will have to configure the **qos dscp encode-policy** command to bind "**USER_DSCP_MAP**" either globally or to a specified port.

Default QoS mappings

If a user defined map is not created or applied to Ingress or Egress traffic, the Brocade device uses a default map to assign PCP, DSCP and EXP priority and drop precedence values. The following tables describe the default QoS mapping values:

- PCP Encode Table
- PCP Decode Table
- DSCP Encode Table
- DSCP Decode Table
- EXP Encode Table
- EXP Decode Table

Table 18 lists the default PCP Encode mappings.

NOTE
The encode commands for QOS mapping such as **qos pcp encode** on | off, **qos dscp encode** on| off and **qos exp encode** on | off are available only at the physical port level. There are no direct commands at global level to enable or disable QOS encode mapping.

TABLE 18 PCP encode table

Priority & Drop Eligibility (DE)	7	7DE	6	6DE	5	5DE	4	4DE	3	3DE	2	2DE	1	1DE	0	0DE

TABLE 18 PCP encode table

	8P0D (default)	7	7	6	6	5	5	4	4	3	3	2	2	1	1	0	0
PCP	7P1D	7	7	6	6	5	4	5	4	3	3	2	2	1	1	0	0
	6P2D	7	7	6	6	5	4	5	4	3	2	3	2	1	1	0	0
	5P3D	7	7	6	6	5	4	5	4	3	2	3	2	1	0	1	0

Table 19 lists the default PCP Decode mappings.

TABLE 19 PCP decode table

PCP		7	6	5	4	3	2	1	0
	8P0D (default)	7	6	5	4	3	2	1	0
PCP	7P1D	7	6	4	4DE	3	2	1	0
	6P2D	7	6	4	4DE	2	2DE	1	0
	5P3D	7	6	4	4DE	2	2DE	0	0DE

Table 20 lists the default DSCP Encode mappings.

TABLE 20 Default DSCP encode table

Priority decimal (binary)	Drop-precedence decimal (binary)	DSCP decimal (binary)	Priority decimal (binary)	Drop-precedence decimal (binary)	DSCP decimal (binary)
0 (000)	0 (00)	0 (000000)	4 (100)	0 (00)	32 (100000)
0 (000)	1 (01)	2 (000010)	4 (100)	1 (01)	34 (100010)
0 (000)	2 (10)	4 (000100)	4 (100)	2 (10)	36 (100100)
0 (000)	3 (11)	6 (000110)	4 (100)	3 (11)	38 (100110)
1 (001)	0 (00)	8 (001000)	5 (101)	0 (00)	40 (101000)
1 (001)	1 (01)	10 (001010)	5 (101)	1 (01)	42 (101010)
1 (001)	2 (10)	12 (001100)	5 (101)	2 (10)	44 (101100)
1 (001)	3 (11)	14 (001110)	5 (101)	3 (11)	46 (101110)
2 (010)	0 (00)	16 (010000)	6 (110)	0 (00)	48 (110000)
2 (010)	1 (01)	18 (010010)	6 (110)	1 (01)	50 (110010)
2 (010)	2 (10)	20 (010100)	6 (110)	2 (10)	52 (110100)
2 (010)	3 (11)	22 (010110)	6 (110)	3 (11)	54 (110110)
3 (011)	0 (00)	24 (011000)	7 (111)	0 (00)	56 (111000)
3 (011)	1 (01)	26 (011010)	7 (111)	1 (01)	58 (111010)
3 (011)	2 (10)	28 (011100)	7 (111)	2 (10)	60 (111100)
3 (011)	3 (11)	30 (011110)	7 (111)	3 (11)	62 (111110)

Table 21 and Table 22 list the default DSCP Decode mappings.

TABLE 21 Default DSCP decode table

DSCP decimal (binary)	Priority decimal (binary)	Drop-precedence decimal (binary)	DSCP decimal (binary)	Priority decimal (binary)	Drop-precedence decimal (binary)
0 (000000)	0 (000)	0 (00)	16 (010000)	2 (010)	0 (00)
1 (000001)	0 (000)	0 (00)	17(010001)	2 (010)	0 (00)
2 (000010)	0 (000)	1 (01)	18 (010010)	2 (010)	1 (01)
3 (000011)	0 (000)	1 (01)	19 (010011)	2 (010)	1 (01)
4 (000100)	0 (000)	2 (10)	20 (010100)	2 (010)	2 (10)
5 (000101)	0 (000)	2 (10)	21(010101	2 (010)	2 (10)
6 (000110)	0 (000)	3 (11)	22 (010110)	2 (010)	3 (11)
7 (000111)	0 (000)	3 (11)	23 (010111)	2 (010)	3 (11)
8 (001000)	1 (001)	0 (00)	24 (011000)	3 (011)	0 (00)
9 (001001)	1 (001)	0 (00)	25 (011001)	3 (011)	0 (00)
10 (001010)	1 (001)	1 (01)	26 (011010)	3 (011)	1 (01)
11 (001011)	1 (001)	1 (01)	27 (011011)	3 (011)	1 (01)
12 (001100)	1 (001)	2 (10)	28 (011100)	3 (011)	2 (10)
13 (001101	1 (001)	2 (10)	29 (011101)	3 (011)	2 (10)
14 (001110)	1 (001)	3 (11)	30 (011110)	3 (011)	3 (11)
15 (001111)	1 (001)	3 (11)	31 (011111)	3 (011)	3 (11)

TABLE 22 Default DSCP decode table (cont.)

DSCP decimal (binary)	Priority decimal (binary)	Drop-precedence decimal (binary)	DSCP decimal (binary)	Priority decimal (binary)	Drop-precedence decimal (binary
32 (100000)	4 (100)	0 (00)	48 (110000)	6 (110)	0 (00)
33 (100001)	4 (100)	0 (00)	49 (110001)	6 (110)	0 (00)
34 (100010)	4 (100)	1 (01)	50 (110010)	6 (110)	1 (01)
35 (100011)	4 (100)	1 (01)	51(110011)	6 (110)	1 (01)
36 (100100)	4 (100)	2 (10)	52(110100)	6 (110)	2 (10)
37 (100101)	4 (100)	2 (10)	53(110101)	6 (110)	2 (10)
38 (100110)	4 (100)	3 (11)	54 (110110)	6 (110)	3 (11)
38 (100111)	4 (100)	3 (11)	55 (110111)	6 (110)	3 (11)
40 (101000)	5 (101)	0 (00)	56 (111000)	7 (111)	0 (00)
41 (101001)	5 (101)	0 (00)	57 (111001)	7 (111)	0 (00)
42 (101010)	5 (101)	1 (01)	58 (111010)	7 (111)	1 (01)
43 (101011)	5 (101)	1 (01)	58 (111011)	7 (111)	1 (01)
44 (101100)	5 (101)	2 (10)	60 (111100)	7 (111)	2 (10)
45 (101101)	5 (101)	2 (10)	61 (111101)	7 (111)	2 (10)

TABLE 22 Default DSCP decode table (cont.)

DSCP decimal (binary)	Priority decimal (binary)	Drop-precedence decimal (binary)	DSCP decimal (binary)	Priority decimal (binary)	Drop-precedence decimal (binary
46 (101110)	5 (101)	3 (11)	62 (111110)	7 (111)	3 (11)
47(101111)	5 (101)	3 (11)	63 (111111)	7 (111)	3 (11)

Table 23 lists the default EXP Encode mappings. Please note that software forwarded VPLS packets do not use the EXP encode table.

TABLE 23 Default EXP encode table

Priority decimal (binary)	Drop-precedence decimal (binary)	EXP value	Priority decimal (binary)	Drop-precedence decimal (binary)	DSCP decimal (binary)
0 (000)	0 (00)	0	4 (100)	0 (00)	4
0 (000)	1 (01)	0	4 (100)	1 (01)	4
0 (000)	2 (10)	0	4 (100)	2 (10)	4
0 (000)	3 (11)	0	4 (100)	3 (11)	4
1 (001)	0 (00)	1	5 (101)	0 (00)	5
1 (001)	1 (01)	1	5 (101)	1 (01)	5
1 (001)	2 (10)	1	5 (101)	2 (10)	5
1 (001)	3 (11)	1	5 (101)	3 (11)	5
2 (010)	0 (00)	2	6 (110)	0 (00)	6
2 (010)	1 (01)	2	6 (110)	1 (01)	6
2 (010)	2 (10)	2	6 (110)	2 (10)	6
2 (010)	3 (11)	2	6 (110)	3 (11)	6
3 (011)	0 (00)	3	7 (111)	0 (00)	7
3 (011)	1 (01)	3	7 (111)	1 (01)	7
3 (011)	2 (10)	3	7 (111)	2 (10)	7
3 (011)	3 (11)	3	7 (111)	3 (11)	7

Table 24 lists the default EXP Encode mappings

TABLE 24 Default EXP encode table

Priority decimal (binary)	Drop-precedence decimal (binary)	EXP value	Priority decimal (binary)	Drop-precedence decimal (binary)	EXP value
0 (000)	0 (00)	0	4 (100)	0 (00)	4
0 (000)	1 (01)	0	4 (100)	1 (01)	4
0 (000)	2 (10)	0	4 (100)	2 (10)	4
0 (000)	3 (11)	0	4 (100)	3 (11)	4
1 (001)	0 (00)	1	5 (101)	0 (00)	5

TABLE 24 Default EXP encode table

Priority decimal (binary)	Drop-precedence decimal (binary)	EXP value	Priority decimal (binary)	Drop-precedence decimal (binary)	EXP value
1 (001)	1 (01)	1	5 (101)	1 (01)	5
1 (001)	2 (10)	1	5 (101)	2 (10)	5
1 (001)	3 (11)	1	5 (101)	3 (11)	5
2 (010)	0 (00)	2	6 (110)	0 (00)	6
2 (010)	1 (01)	2	6 (110)	1 (01)	6
2 (010)	2 (10)	2	6 (110)	2 (10)	6
2 (010)	3 (11)	2	6 (110)	3 (11)	6
3 (011)	0 (00)	3	7 (111)	0 (00)	7
3 (011)	1 (01)	3	7 (111)	1 (01)	7
3 (011)	2 (10)	3	7 (111)	2 (10)	7
3 (011)	3 (11)	3	7 (111)	3 (11)	7

Table 25 lists the default EXP Decode mappings

TABLE 25 Default EXP decode table

EXP value	Priority decimal (binary)	Drop-precedence decimal (binary)
7	7 (111)	0
6	6 (110)	0
5	5 (101)	0
4	4 (100)	0
3	3 (011)	0
2	2 (010)	0
1	2 (001)	0
0	0 (000)	0

Protocol Packet Prioritization

Certain control packets are handled with certain priorities by default and hence those priorities cannot be lowered with any of the QoS configuration commands or the priority force command. The list of these control packets are listed below.

Table 26 on page 84 lists the protocol packets that are internally and automatically prioritized for IPv4, L2, and IPv6.

TABLE 26 Default prioritized protocol table

Protocol Packets
IPv4/L2
ARP
STP/RSTP/BPDU
MRP
VSRP
LACP
GARP
UDLD
IGMP
OSPF / OSPF over GRE
BGP / BGP over GRE
RIP
IS-IS
ES-IS
VRRP
VRRPE
PIM / PIM over GRE
DVMRP
MSDP / MSDP over GRE
RSVP
LDP basic
LDP extended
BOOTP/DHCP
IPv4 Router Alert
ISIS over GRE or GRE
Keep Alive Packets
BFD (Bidirectional Forwarding Detection)
IPv6
OSPF / OSPF in 6to4
BGP / BGP in 6to4
RIPNG
MLD
ND6 / ND6 in 6to4
VRRP

TABLE 26 Default prioritized protocol table

Protocol Packets
VRRPE PIM / PIM in 6to4 BFD (Bidirectional Forwarding Detection)
PIM / PIM in 6to4
BFD (Bidirectional Forwarding Detection)

Enhanced control packet prioritization

The Traffic Manager (TM) allows prioritization and scheduling of packets destined for the CPU to guarantee optimal control packet processing and to reduce protocol flapping. The TM achieves physical separation of CPU-bound data and control packets. The hierarchical structure supports four sets of eight priority queues. The four sets are as follows:

- Protocol set - Protocol packets that are prioritized by the network processor.
- Management set - Packets destined for the router; for example, Ping and Telnet.
- Flow set - Flow-driven packets to the CPU; for example, Unknown DA, DPA, Layer 2 broadcast and multicast, Multicast, VPLS SA Learning packets.
- Snoop set - CPU copy packets; for example, Regular Layer 2 SA learning, sFlow, ACL Logging, RPF Logging. The **rl-cpu-copy** command defines the rate shaping value for the snoop queues.

For each set of the eight priority queues, the priority queue 7 is given the highest rate, the priority queue 6 to queue 2 are given the same rate, and the priority queue 1 and queue 0 are given a lower rate. Each of the four sets of CPU queues is given equal weights. With the flow-driven packets given the same weight as the other packets, the protocol packets and flow-driven packets are queued separately so that the protocol packets are processed at a faster rate compared to the flow-driven packets.

Table 27 lists the protocol packets of the network processor prioritized protocol set based on their priorities.

TABLE 27 Network processor prioritized protocol packets

Priority categorization	Protocols
P7	LACP, UDLD (802.3ah), STP, RSTP, BPDU, VSRP, MRP, BFD, GRE-KA, IS-IS over GRE, G.8032, LLDP, non-CCM 802.1ag (Ethernet + MPLS Encapsulated), BFD (Single-hop, Multi-hop, and MPLS), IS-IS Hello, OSPFv2 Hello (GRE + Ethernet), OSPFv3 Hello (6to4 + Ethernet).
P6	IS-IS Non-Hello, OSPFv2 and OSPFv3 Non-Hello (Ethernet, GRE, 6to4), IPsec ESP Packets (for OSPFv3 over IPsec), OSPF, OSPF over GRE or 6to4, IS-IS, RIP, RIPNG, VRRP (Version 4 and Version 6), VRRP-E (Version 4 and Version 6).
P5	BGP, BGP over GRE or 6to4, PIM, PIM over GRE or 6to4, LDP (basic and extended), RSVP, CCP (MCT), 802.1ag CCM (Ethernet + MPLS Encapsulated).
P4	VPLS Encapsulated PIM, DVMRP, MSDP, MSDP over GRE, MSDP over VPLS.
P3	IGMP, VPLS Encapsulated IGMP, GRE Encapsulated IGMP, ARP, MLD, DHCP, BOOTP, ND6 and ND6 in 6to4.
P2	IPv4 Router Alert.

TABLE 27 Network processor prioritized protocol packets (Continued)

Priority categorization	Protocols
P1	New unassigned protocols.
P0	Existing unassigned protocols: GARP, L2-Trace.

Configuring QoS

The QoS configuration process involves separate procedures for Ingress and Egress QoS Processing as described in the following major sections.

Configuring Ingress QoS procedures

The following procedures are required to configure a Brocade device for Ingress QoS processing:

- **Creating Ingress Decode Policy Maps** – If you want the priority and drop precedence values used within the device to be mapped to a specified value, you must create a Decode Priority map as described in "Configuring Ingress decode policy maps" on page 87.
- **Binding Ingress Decode Policy Maps** – If you want to apply an Ingress Policy Map other than the default, you must bind the Ingress Policy Map either globally or to a specified interface as described in "Binding Ingress decode policy maps" on page 93.
- **Configuring a Force priority** – Where there are multiple QoS values that can be used to determine the QoS level used on the device, the default policy is to determine the value used by performing a merge as described in "Stage 2 Determine if a priority value should be forced or merged" on page 74. Otherwise, you can specify a value that you want used from either the port or VLAN configured value or the DSCP, EXP or PCP values in the packets as described in "Configuring a force priority" on page 96.

Configuring Egress QoS procedures

The following procedures are required to configure a Brocade device for Egress QoS processing:

- **Creating Egress Encode Policy Maps** – If you want the priority and drop precedence values of packets leaving the device to be marked with a specified value, you must create an Encode Priority map as described in "Configuring Egress encode policy maps" on page 99.
- **Binding Egress Encode Policy Maps** – If you want to apply an Egress Policy Map other than the default, you must bind the Egress Policy Map either globally or to a specified interface as described in "Binding an Egress encode EXP policy map" on page 102.

Configuring QoS procedures applicable to Ingress and Egress

The following procedures are required to configure procedures applicable to both Ingress and Egress QoS Processing on a Brocade device:

- **Enabling a Port to Use the DEI bit** – You can configure the device to use the DEI bit when computing the drop precedence value for an incoming packet or encoding the DEI bit for transmitted frame as described in "Enabling a port to use the DEI bit for Ingress and Egress processing" on page 107.
- **Specifying the Trust Level and Enabling Marking** – If you want to use the **qos-tos trust** and **qos-tos mark** commands from pre-03.8.00 versions of the QoS feature, refer to "Specifying the trust level and enabling marking" on page 108.
- **Support for Super Aggregate VLANS** – If you want to use the enhanced QoS feature with Super Aggregate VLANs, refer to "Configuring support for super aggregate VLANs" on page 112.
- **Support for QoS Configurations on LAG Ports** – If you are configuring the enhanced QoS feature on ports within a LAG, refer to "Configuring port-level QoS commands on LAG ports" on page 112.

Configuring Ingress decode policy maps

Ingress Decode Policy Maps are created globally and are applied later either globally for all ports on a device or locally to specific port. To create an Ingress Decode Policy Map, you must first enter the QoS mapping configuration level of the command interface using the **qos-mapping** command, as shown in the following.

```
Brocade(config)# qos-mapping
```

Configuration of each of the decode and encode mappings is described in the following sections:

- Configuring Ingress Decode DSCP Policy Maps
- Configuring Ingress Decode PCP Policy Maps
- Configuring Ingress Decode EXP Policy Maps

Configuring Ingress decode DSCP policy maps

The following procedures are used when configuring an Ingress Decode DSCP Policy Map:

- Naming an Ingress Decode DSCP Policy Map
- Configuring an Ingress Decode DSCP Policy Map

Naming an Ingress decode DSCP policy map

Once you are in the QoS configuration level, can define the name of a Ingress Decode DSCP Policy Map using the **pcp decode-map** command, as shown in the following.

```
Brocade(config)# qos-mapping
Brocade(config-qos-mapping)# dscp decode-map Customer1
```

Syntax: [no] dscp decode-map *map-name*

The **no** option is used to delete a currently configured Ingress Decode DSCP Policy Map. If the Ingress Decode DSCP Policy Map is currently in use the **no** command will be rejected and an error message will be displayed.

The *map-name* variable specifies the name of the Ingress Decode DSCP Policy Map that you are defining. It can be up to 64 characters in length. You can specify the same *map-name* for different types of policy maps. For example, you can use the same name for an Ingress Decode DSCP Policy Map and a Ingress Decode EXP Policy Map.

NOTE
The name "default-map" cannot be used because it is reserved for standard mappings as described in "Default QoS mappings" on page 79.

Configuring an Ingress decode DSCP policy map

Once you have named an Ingress Decode DSCP Policy Map using the **dscp decode-map** command, you can set the values of the named Ingress Decode DSCP Policy Map. Setting the values in an Ingress Decode DSCP Policy Map involves specifying the value of the DSCP bits of an incoming packet and setting them to correspond to a value of 0 to 7 of the device's internal priority. Optionally, you can set a drop precedence value of 0 to 3 in addition to the internal priority value.

To set the values of an Ingress Decode DSCP Policy Map, first specify name of the policy map and then populate the values in the policy map using the **dscp-value** command as shown in the following.

```
Brocade(config)# qos-mapping
Brocade(config-qos-mapping)# dscp decode-map Customer1
Brocade(config-qos-mapping-dscp-decode)# dscp-value 32 to priority 5
drop-precedence 2
```

Syntax: **[no] dscp-value** *dscp-value* **[***dscp-value* **] to priority** *priority-value* **[drop-precedence** *dp-value***]**

The *dscp-value* variable specifies the value of the DSCP bits within the packet header of the incoming packets. You can optionally specify multiple *dscp-value* variables if you want to specify more than one value to map to the same internal priority and drop precedence. Where DSCP values within a policy map are unspecified, the default mapping will be used.

The **priority** keyword together with the *priority-value* variable specifies the internal priority that the packets with the previously specified *dscp-value* value will be mapped to. The *priority-value* variable can be a value between 0 and 7. Please note, when generating the configuration file, a configured priority value that is the same as the value in the default priority map will not be shown.

The **drop-precedence** keyword is an optional parameter that allows you to specify a *dp-number* variable that represents the drop precedence value that you want to assign to incoming packets with the previously specified *dp-value* value. This value is specified in addition to a **priority** *priority-value* value. The *dp-number* variable can be a value between 0 and 3. The default value is the value described in the default DSCP table. Please note, when generating the configuration file, a value for drop precedence will only be shown for non-default values.

When using the **[no]** option to negate a previously configured value of this command, observe the considerations described below.

1. You can negate both the **priority** and **drop-precedence** values (returning them to their default values) by using the **[no]** option with the original command only up to the **priority** value.

   For example: the following command has been used to set the map to assign an internal priority of "4" and a drop precedence of "2" to Ingress packets that have a DSCP value of "40".

   ```
   Brocade(config-qos-mapping-dscp-decode)# dscp-value 40 to priority 4
   drop-precedence 2
   ```

   To set the priority and **drop-precedence** values back to the default values, use the **[no]** option with the previous command up to where the **priority** value is configured, as shown in the following.

   ```
   Brocade(config-qos-mapping-dscp-decode)# no dscp-value 40 to priority 4
   ```

After this command is executed, the **priority** and **drop-precedence** values for **dscp-value 40** will be returned to their default values as described in the default map tables that are defined in "Default QoS mappings" on page 79.

2. You can negate the **drop-precedence** value (returning it to its default value) without changing the currently configured **priority** value. This is done by using the **[no]** option with the original command that includes both the **priority** and **drop-precedence** values.

   For example: the following command has been used to set the priority map to assign an internal priority of "5" and a drop precedence of "1" to Ingress packets that have a DSCP value of "60".

   ```
   Brocade(config-qos-mapping-dscp-decode)# dscp-value 60 to priority 5
   drop-precedence 1
   ```

   To set the **drop-precedence** value back to the default value, use the **[no]** option with the previous command, as shown in the following.

   ```
   Brocade(config-qos-mapping-dscp-decode)# no dscp-value 60 to priority 5
   drop-precedence 1
   ```

   After this command is executed, the **priority** value will remain at 5 and the **drop-precedence** value will be returned to the default **drop-precedence** value for **dscp-value 60**, as described in the default map tables that are defined in "Default QoS mappings" on page 79.

Configuring Ingress decode PCP policy maps

The following procedures are used when configuring an Ingress Decode PCP Policy Map:

- Naming an Ingress Decode PCP Policy Map
- Configuring an Ingress Decode PCP Policy Map

Naming an Ingress decode PCP policy map

Once you are in the QoS configuration level, can define the name of an Ingress Decode PCP Policy Map using the **dscp decode-map** command, as shown in the following.

```
Brocade(config)# qos-mapping
Brocade(config-qos-mapping)# pcp decode-map Customer1
```

Syntax: [no] pcp decode-map *map-name*

The **no** option is used to delete a currently configured Ingress decode PCP policy map. If the policy map is currently in use, the **no** command will be rejected and an error message will be displayed.

The *map-name* variable specifies the name of the Ingress decode PCP policy map that you are defining. It can be up to 64 characters in length. You can specify the same map name for different types of maps. For example, you can use the same name for an Ingress decode PCP policy map and an Ingress decode DSCP policy map.

NOTE
The name "default-map" cannot be used because it is reserved for standard mappings as described in "Default QoS mappings" on page 79.

Configuring an Ingress decode PCP policy map

Once you have named an Ingress PCP Decode Policy Map using the **pcp decode-map** command, you can set the values of the named policy map. Setting the values in a policy map involves specifying the value of the PCP bits of an incoming packet and setting them to correspond to a value of 0 to 7 of the device's internal priority. Optionally, you can set a drop precedence value of 0 to 3 in addition to the internal priority value.

To set the values of an Ingress Decode PCP Policy Map, first specify name of the policy map and then populate the values in the Ingress Decode PCP Policy Map using the **pcp-value** command as shown in the following.

```
Brocade(config)# qos-mapping
Brocade(config-qos-mapping)# pcp decode-map Customer1
Brocade(config-qos-mapping-pcp-decode)# pcp-value 7 to priority 3 drop-precedence
2
```

Syntax: **[no] pcp-value** *pcp-value* [*pcp-value*] **to priority** *priority-value* **[drop-precedence** *dp-value*]

The *pcp-value* variable specifies the value of the PCP bits within the packet header of the incoming packets. You can optionally specify multiple *pcp-value* variables if you want to specify more than one value to map to the same internal priority and drop precedence. Where PCP values within a policy map are unspecified, the default mapping will be used.

The **priority** keyword together with the *priority-value* variable specifies the internal priority that the packets with the previously specified *pcp-value* value will be mapped to. The *priority-value* variable can be a value between 0 and 7. Please note, when generating the configuration file a configured priority value that is the same as the value in the default map will not be shown.

The **drop-precedence** keyword is an optional parameter that allows you to specify a *dp-number* variable that represents the drop precedence value that you want to assign to incoming packets with the previously specified *dp-value* value. This value is specified in addition to a **priority** *priority-value* value. The *dp-number* variable can be a value between 0 and 3. The default value is 0. Please note, when generating the configuration file a value for drop precedence will only be shown for non-zero values.

When using the **[no]** option to negate a previously configured value of this command, observe the considerations described below.

1. You can negate both the **priority** and **drop-precedence** values (returning them to their default values) by using the **[no]** option with the original command only up to the **priority** value.

   For example: the following command has been used to set the map to assign an internal priority of "3" and a drop precedence of "2" to Ingress packets that have a PCP value of "7".

   ```
   Brocade(config-qos-mapping-pcp-decode)# pcp-value 7 to priority 3
   drop-precedence 2
   ```

   To set the **priority** and **drop-precedence** values back to the default values, use the **[no]** option with the previous command up to where the **priority** value is configured, as shown in the following.

   ```
   Brocade(config-qos-mapping-pcp-decode)# no pcp-value 7 to priority 3
   ```

   After this command is executed, the **priority** and **drop-precedence** values for **pcp-value 7** will be returned to their default values as described in the default map tables that are defined in "Default QoS mappings" on page 79.

2. You can negate the **drop-precedence** value (returning it to its default value) without changing the currently configured **priority** value. This is done by using the **[no]** option with the original command that includes both the **priority** and **drop-precedence** values.

For example: the following command has been used to set the priority map to assign an internal priority of “4” and a drop precedence of “2” to Ingress packets that have a PCP value of “6”.

```
Brocade(config-qos-mapping-pcp-decode)# pcp-value 6 to priority 4
drop-precedence 2
```

To set the **drop-precedence** value back to the default value, use the **[no]** option with the previous command, as shown in the following.

```
Brocade(config-qos-mapping-pcp-decode)# no pcp-value 6 to priority 4
drop-precedence 2
```

After this command is executed, the **priority** value will remain at 4 and the **drop-precedence** value will be returned to the default **drop-precedence** value for **pcp-value 6**, as described in the default map tables that are defined in “Default QoS mappings” on page 79.

Configuring Ingress decode EXP Policy maps

The following procedures are used when configuring an Ingress Decode EXP Policy Map:

- Naming an Ingress Decode EXP Policy Map
- Configuring an Ingress Decode EXP Policy Map

Naming an Ingress decode EXP policy map

Once you are in the QoS configuration level, can define the name of a Ingress Decode EXP Policy Map using the **exp decode-map** command, as shown in the following.

```
Brocade(config)# qos-mapping
Brocade(config-qos-mapping)# exp decode-map Customer1
```

Syntax: [no] exp decode-map *map-name*

The **no** option is used to delete a currently configured Ingress Decode EXP Policy Map. If the Ingress Decode EXP Policy Map is currently in use, the **no** command will be rejected and an error message will be displayed.

The *map-name* variable specifies the name of the Ingress Decode EXP Policy Map that you are defining. It can be up to 64 characters in length. You can specify the same Ingress Decode EXP Policy Map for different types of policy maps. For example, you can use the same name for an Ingress Decode DSCP Policy Map and an Ingress Decode EXP Policy Map.

NOTE
The name “default-map” cannot be used because it is reserved for standard mappings as described in “Default QoS mappings” on page 79.

Configuring an Ingress decode EXP policy map

Once you have named an Ingress Decode EXP Policy Map using the **exp decode-map** command, you can set the values of the named policy map. Setting the values in a policy map involves specifying the value of the EXP bits of an incoming packet and setting them to correspond to a value of 0 to 7 of the device’s internal priority value. Optionally, you can set a drop precedence value of 0 to 3 in addition to the internal priority value.

To set the values of an Ingress Decode EXP Policy Map, first specify name of the policy map and then populate the values in the policy map using the **exp-value** command as shown in the following.

```
Brocade(config)# qos-mapping
Brocade(config-qos-mapping)# exp decode-map Customer1
Brocade(config-qos-mapping-exp-decode)# exp-value 7 to priority 5 drop-precedence
2
```

Syntax: [no] **exp-value** *exp-value* [*exp-value*] **to priority** *priority-value* [**drop-precedence** *dp-value*]

The *exp-value* variable specifies the value of the EXP bits within the packet header of the incoming packets. You can optionally specify multiple *exp-value* variables if you want to specify more than one value to map to the same internal priority and drop precedence values. Where EXP values within a policy map are unspecified, the default mapping will be used.

The **priority** keyword together with the *priority-value* variable specifies the internal priority value that the packets with the previously specified *exp-value* value will be mapped to. The *priority-value* variable can be a value between 0 and 7. Please note, when generating the configuration file a configured priority value that is the same as the value in the default priority map will not be shown.

The **drop-precedence** keyword is an optional parameter that allows you to specify a *dp-number* variable that represents the drop precedence value that you want to assign to incoming packets with the previously specified *dp-value* value. This value is specified in addition to a **priority** *priority-value* value. The *dp-number* variable can be a value between 0 and 3. The default value is the value described in the default EXP table. Please note, when generating the configuration file a value for drop precedence will only be shown for non-default values.

When using the **[no]** option to negate a previously configured value of this command, observe the considerations described below.

1. You can negate both the **priority** and **drop-precedence** values (returning them to their default values) by using the **[no]** option with the original command only up to the **priority** value.

   For example: the following command has been used to set the map to assign an internal priority of “5” and a drop precedence of “2” to Ingress packets that have an EXP value of “7”.

   ```
   Brocade(config-qos-mapping-exp-decode)# exp-value 7 to priority 5
   drop-precedence 2
   ```

   To set the **priority** and **drop-precedence** values back to the default values, use the **[no]** option with the previous command up to where the **priority** value is configured, as shown in the following.

   ```
   Brocade(config-qos-mapping-exp-decode)# no exp-value 7 to priority 5
   ```

   After this command is executed, the **priority** and **drop-precedence** values for **exp-value 7** will be returned to their default values as described in the default map tables that are defined in “Default QoS mappings” on page 79.

2. You can negate the **drop-precedence** value (returning it to its default value) without changing the currently configured **priority** value. This is done by using the **[no]** option with the original command that includes both the **priority** and **drop-precedence** values.

   For example: the following command has been used to set the priority map to assign an internal priority of “5” and a drop precedence of “2” to Ingress packets that have a EXP value of “7”.

   ```
   Brocade(config-qos-mapping-exp-decode)# exp-value 7 to priority 5
   drop-precedence 2
   ```

   To set the **drop-precedence** value back to the default value, use the **[no]** option with the previous command, as shown in the following.

```
Brocade(config-qos-mapping-exp-decode)# no exp-value 7 to priority 5
drop-precedence 2
```

After this command is executed, the **priority** value will remain at 5 and the **drop-precedence** value will be returned to the default **drop-precedence** value for **exp-value 7**, as described in the default map tables that are defined in "Default QoS mappings" on page 79.

Binding Ingress decode policy maps

You can bind an Ingress decode policy map globally or per-port using either the default policy map, an all zero policy map, or a user defined policy map. Additionally, for PCP, you can bind the following pre-defined policy maps: 7P1D, 6P2D, and 5P3D. The following procedures describe how to bind Ingress decode policy maps:

- Binding Ingress Decode DSCP Policy Maps
- Binding Ingress Decode PCP Policy Maps
- Binding Ingress Decode EXP Policy Maps

Binding Ingress decode DSCP policy maps

The following procedures describe how to configure the binding of Ingress Decode DSCP Policy Maps:

- Globally Binding an Ingress Decode DSCP Policy Map
- Binding an Ingress Decode DSCP Policy Map to a Port

Globally Binding an Ingress decode DSCP policy map

You can bind an Ingress Decode DSCP Policy Map globally for a Brocade device using the **qos dscp encode-policy** command as shown in the following.

```
Brocade(config)# qos dscp encode-policy Customer1
```

Syntax: [no] qos dscp encode-policy *encode-map-name* **| default-map | all-zero-map**

The *encode-map-name* variable is the name assigned to the Ingress Decode DSCP Policy Map that you want applied globally on the device. If you try to apply an *decode-map-name* value that has not been defined, the configuration will be rejected. If the *decode-map-name* value that has been defined but the policy has not been configured, the configuration will be accepted and the **default-map** will be applied.

The **default-map** option assigns the default Ingress Decode DSCP Policy Map globally on the device. Since the default Ingress Decode DSCP Policy Map is the default setting, this option is only required when the device has been previously set to a different Ingress Decode DSCP Policy Map.

The **all-zero-map** option assigns a Ingress Decode DSCP Policy Map where all DSCP values are mapped to priority 0 and drop precedence 0. This is useful if you do not want to process any DSCP information in the incoming packet.

Binding an Ingress decode DSCP policy map to a port

You can bind an Ingress Decode DSCP Policy Map to a specified port on a Brocade device using the **qos dscp encode-policy** command within an interface configuration, as shown in the following.

```
Brocade(config)# interface ethernet 10/1
Brocade(config-if-e10000-10/1)qos dscp encode-policy Customer1
```

Syntax: [no] qos dscp encode-policy *encode-map-name* **| default-map | all-zero-map**

The *encode-map-name* variable is the name assigned to the Ingress Decode DSCP Policy Map that you want applied to the port whose configuration this is under.

The **default-map** option assigns the default Ingress Decode DSCP Policy Map to the port whose configuration this is under. Since the default Ingress Decode DSCP Policy Map is the global default setting, this option is only required when the device's global map has been set to a Ingress Decode DSCP Policy Map other than the default.

The **all-zero-map** option assigns an Ingress Decode DSCP Policy Map where all DSCP values are mapped to priority 0 and drop precedence 0. This is useful if you do not want to process any DSCP information in the incoming packet

Binding an Ingress decode PCP policy map

The following procedures describe how to configure the binding of an Ingress Decode PCP Policy Map:

- Globally Binding an Ingress Decode PCP Policy Map
- Binding an Ingress Decode PCP Policy Map to a Port

Globally binding an Ingress decode PCP policy map

You can bind an Ingress Decode PCP Policy Map globally for a Brocade device using the **qos pcp decode-policy** command as shown in the following.

```
Brocade(config)# qos pcp decode-policy Customer1
```

Syntax: [no] **qos pcp decode-policy** *decode-map-name* | **default-map** | **all-zero-map** | **7P1D** |**6P2D** | **5P3D**

The *decode-map-name* variable is the name assigned to the Ingress Decode PCP Policy Map that you want applied globally on the device. If you try to apply an *decode-map-name* value that has not been defined, the configuration will be rejected. If the *decode-map-name* value that has been defined but the policy has not been configured, the configuration will be accepted and the **default-map** will be applied.

The **default-map** option assigns the default Ingress Decode PCP Policy Map globally on the device. The default policy map for PCP is the 8P0D decode map. Since the default Ingress Decode PCP Policy Map is the default setting, this option is only required when the device has been previously set to a different Ingress Decode PCP Policy Map.

The **all-zero-map** option assigns an Ingress Decode PCP Policy Map where all PCP values are mapped to priority 0 and drop precedence 0. This is useful if you do not want to process any PCP information in the incoming packet.

The **7P1D** option assigns the 7P1D Ingress Decode PCP Policy Map globally on the device.

The **6P2D** option assigns the 6P2D Ingress Decode PCP Policy Map globally on the device.

The **5P3D** option assigns the 5P3D Ingress Decode PCP Policy Map globally on the device.

NOTE
7P1D, 6P2D, and **5P3D** are as defined in the IEEE 802.1ad specification and described in Table 19.

Binding an Ingress decode PCP policy map to a port

You can bind an Ingress Decode PCP Policy Map to a specified port on a Brocade device using the **qos pcp decode-policy** command as shown in the following.

```
Brocade(config)# qos pcp decode-policy Customer1
Brocade(config)# interface ethernet 10/1
Brocade(config-if-e10000-10/1)qos pcp decode-policy Customer1
```

Syntax: [no] qos pcp decode-policy *decode-map-name* **| default-map | all-zero-map | 7P1D |6P2D | 5P3D**

The *decode-map-name* variable is the name assigned to the Ingress Decode PCP Policy Map that you want applied to the port whose configuration this is under.

The **default-map** option assigns the default Ingress Decode PCP Policy Map to the port whose configuration this is under. Since the default Ingress Decode PCP Policy Map is the default setting, this option is only required when the device's global map has been set to an Ingress Decode PCP Policy Map other than the default.

The **all-zero-map** option assigns an Ingress Decode PCP Policy Map where all PCP values are mapped to priority 0 and drop precedence 0. This is useful if you do not want to process any PCP information in the incoming packet.

The **7P1D** option assigns the 7P1D Ingress Decode PCP Policy Map to the port whose configuration this is under.

The **6P2D** option assigns the 6P2D Ingress Decode PCP Policy Map to the port whose configuration this is under.

The **5P3D** option assigns the 5P3D Ingress Decode PCP Policy Map to the port whose configuration this is under.

NOTE
7P1D, 6P2D, and **5P3D** are as defined in the IEEE 802.1ad specification and described in Table 19.

Binding an Ingress decode EXP policy map

The following procedures describe how to configure the binding of an Ingress Decode EXP Policy Map:

- Globally Binding an Ingress Decode EXP Policy Map.
- Binding an Ingress Decode EXP Policy Map to a Port.

Globally binding an Ingress decode EXP policy map

You can bind an Ingress Decode EXP Policy Map globally for a Brocade device using the **qos exp decode-policy** command as shown in the following.

```
Brocade(config)# qos exp decode-policy Customer1
```

Syntax: [no] qos exp decode-policy *decode-map-name* **| default-map | all-zero-map**

The *decode-map-name* variable is the name assigned to the Ingress Decode EXP Policy Map that you want applied globally on the device. If you try to apply a *decode-map-name* value that has not been defined, the configuration will be rejected. If the *decode-map-name* value that has been defined but the policy has not been configured, the configuration will be accepted and the **default-map** will be applied.

The **default-map** option assigns the default Ingress Decode EXP Policy Map globally on the device. Since the default Ingress Decode EXP Policy Map is the default setting, this option is only required when the device has been previously set to a different Ingress Decode EXP Policy Map.

The **all-zero-map** option assigns an Ingress Decode EXP Policy Map where all EXP values are mapped to priority 0 and drop precedence 0.

Binding an Ingress decode EXP policy map to a port

You can bind an Ingress Decode EXP Policy Map to a specified port on a Brocade device using the **qos exp decode-policy** command as shown in the following.

```
Brocade(config)# interface ethernet 10/1
Brocade(config-if-e10000-10/1)qos exp decode-policy Customer1
```

Syntax: [no] qos exp decode-policy *decode-map-name* **| default-map | all-zero-map**

The *decode-map-name* variable is the name assigned to the Ingress Decode EXP Policy Map that you want applied to the port whose configuration this is under:

The **default-map** option assigns the default Ingress Decode EXP Policy Map to the port whose configuration this is under. Since the default Ingress Decode EXP Policy Map is the default setting, this option is only required when the device's global map has been set to an Ingress Decode EXP Policy Map other than the default.

The **all-zero-map** option assigns an Ingress Decode EXP Policy Map where all EXP values are mapped to priority 0 and drop precedence 0. This is useful if you do not want to process any EXP information in the incoming packet.

Configuring a force priority

In situations where there are conflicting priority values for packets on an Ingress port, that conflict can be resolved by performing a priority merge or by using a **force** command to direct the device to use a particular value above other values. A **force** command can be configured for each of the following:

- Force to the values configured on a port
- Force to the value configured for a VLAN
- Force to the value in the DSCP bits
- Force to the value in the EXP bits
- Force to the value in the PCP bits
- Force to a value specified within an ACL

Configuring a force priority for a port

You can configure an Ingress port with a priority to apply to packets that arrive on it using the **priority** command.

To configure an Ingress port with a priority, use the **priority** command as shown in the following.

```
Brocade(config)# interface ethernet 10/1
Brocade(config-if-e10000-10/1)priority 6
```

Syntax: [no] priority *priority-value*

The *priority-value* variable is a value between 0 and 7. The default value is 0.

Once a port has been configured with a priority using the **priority** command, you can then configure the port (using the **priority force** command) to force the configured priority when determining the priority relative to other priority values of incoming packets.

To configure an Ingress port to force the port-configured priority, use the **priority force** command as shown in the following:

```
Brocade(config)# interface ethernet 10/1
Brocade(config-if-e10000-10/1)priority force
```

Syntax: [no] priority force

Configuring a force drop precedence for a port

You can configure an Ingress port with a drop precedence to apply to packets that arrive on it using the **drop-precedence** command.

To configure an Ingress port with a drop precedence, use the **drop-precedence** command as shown in the following.

```
Brocade(config)# interface ethernet 10/1
Brocade(config-if-e10000-10/1)drop-precedence 3
```

Syntax: [no] drop-precedence *dp-value*

The *dp-value* variable is a value between 0 and 3.

Once a port has been configured with a drop precedence using the **drop-precedence** command, you can then configure the port (using the **drop-precedence force** command) to force the configured drop precedence when determining the priority relative to other priority values of incoming packets.

To configure an Ingress port to force the port-configured drop precedence, use the **drop-precedence force** command as shown in the following.

```
Brocade(config)# interface ethernet 10/1
Brocade(config-if-e10000-10/1)drop-precedence force
```

Syntax: [no] drop-precedence force

Configuring a force priority for a VLAN

By default, VLANs have priority 0. To change a port-based VLAN's QoS priority, use the following method. The priority applies to outbound traffic on ports in the VLAN.

To change the QoS priority of port-based VLAN 20 on a Chassis device to priority queue 7, enter the following commands.

```
Brocade(config)# vlan 20
Brocade(config-vlan-20)# priority 7
```

Syntax: [no] priority *num*

The *num* parameter can be from 0 – 7 and specifies one of the eight QoS queues.

NOTE
When you apply the VLAN priority command with running traffic, it may drop packets for a short period of time. This is normal.

Once a VLAN has been configured with a priority using the **priority** command, you can then configure the VLAN (using the **priority force** command) to force the configured priority when determining the priority relative to other priority values of incoming packets.

To configure an Ingress port to force the VLAN-configured priority, use the **priority force** command as shown in the following.

```
Brocade(config)# vlan 20
Brocade(config-vlan-20) priority force
```

Syntax: [no] priority force

Configuring force priority to the DSCP value

You can configure an Ingress port (using the **qos dscp force** command) to force the configured DSCP value when determining the priority relative to other priority values of incoming packets.

To configure an Ingress port to force the DSCP value, use the **qos dscp force** command as shown in the following.

```
Brocade(config)# interface ethernet 10/1
Brocade(config-if-e10000-10/1)qos dscp force
```

Syntax: [no] qos dscp force

Configuring force priority to the EXP value

You can configure an Ingress port (using the **qos exp force** command) to force the configured EXP value when determining the priority relative to other priority values of incoming packets.

To configure an Ingress port to force the EXP value, use the **qos exp force** command as shown in the following.

```
Brocade(config)# interface ethernet 10/1
Brocade(config-if-e10000-10/1)qos exp force
```

Syntax: qos exp force

Configuring force priority to the PCP value

You can configure an Ingress port (using the **qos pcp force** command) to force the configured PCP value when determining the priority relative to other priority values of incoming packets.

To configure an Ingress port to force the PCP value, use the **qos pcp force** command as shown in the following.

```
Brocade(config)# interface ethernet 10/1
Brocade(config-if-e10000-10/1)qos pcp force
```

Syntax: qos pcp force

Configuring force priority to a value specified by an ACL

You can use the **priority-force** keyword within an ACL to apply a priority to specified traffic as described in “Filtering and priority manipulation based on 802.1p priority” in the *Multi-Service IronWare Security Configuration Guide.*

Configuring Egress encode policy maps

Egress Encode Policy Maps are created globally and are applied later either globally for all ports on a device or locally to specific port. To create an Egress Encode Policy Map, you must first enter the QoS mapping configuration level of the command interface using the **qos-mapping** command, as shown in the following.

```
Brocade(config)# qos-mapping
```

Configuration of each of the Egress Encode Policy Maps is described in the following sections:

- Configuring Egress Encode DSCP Policy Maps
- Configuring Egress Encode PCP Policy Maps
- Configuring Egress Encode EXP Policy Maps

Configuring Egress encode DSCP policy maps

The following procedures are used when configuring an Egress Encode DSCP Policy Map:

- Naming an Egress Encode DSCP Policy Map
- Configuring an Egress Encode DSCP Policy Map

Naming an Egress encode DSCP policy map

Once you are in the QoS configuration level, can define the name of an Egress Encode DSCP Policy Map using the **dscp encode-map** command, as shown in the following.

```
Brocade(config)# qos-mapping
Brocade(config-qos-mapping)# dscp encode-map Customer1
```

Syntax: [no] dscp encode-map *map-name*

The **no** option is used to delete a currently configured Egress Encode DSCP Policy Map. If the policy map is currently in use, the **no** command will be rejected and an error message will be displayed.

The *map-name* variable specifies the name of the Egress Encode DSCP Policy Map that you are defining. It can be up to 64 characters in length. You can specify the same policy map name for different types of maps. For example, you can use the same policy pam name for an Egress Encode DSCP Policy Map and an Egress Encode EXP Policy map.

NOTE
The name “default-map” cannot be used because it is reserved for standard mappings as described in “Default QoS mappings” on page 79.

Configuring an Egress encode DSCP policy map

Once you have named an Egress Encode DSCP Policy Map using the **dscp encode-map** command, you can set the values of the named encode policy map. Setting the values in an Egress Encode DSCP Policy Map involves specifying a DSCP value to be marked in outgoing packets for a specified priority value (0 - 7) and optionally a drop precedence value (0 - 3).

To set the values of an Egress Encode DSCP Policy Map, first specify name of the policy map and then populate the values in the Egress Encode DSCP Policy Map using the **priority** command as shown in the following.

```
Brocade(config)# qos-mapping
Brocade(config-qos-mapping)# dscp encode-map Customer1
Brocade(config-qos-mapping-dscp-encode)# priority 7 drop-precedence 2 to
dscp-value 3
```

Syntax: [no] **priority** *priority-value* [**drop-precedence** *dp-value*] [*dp-value*] **to dscp-value** *dscp-value*

The **priority** keyword together with the *priority-value* variable specifies the internal priority value that egress packets will be marked from. The *priority-value* variable can be a value between 0 and 7. For unspecified priority values, the default mapping values are used.

The **drop-precedence** keyword is an optional parameter that allows you to specify a *dp-number* variable that represents the drop precedence value that you specify in addition to a **priority** *priority-value* value. The *dp-number* variable can be a value between 0 and 3. Multiple *dp-number* variables can be configured in a single command. The default value is "any" which means that drop priorities 0 - 3 are assigned. If a drop precedence value is specified only for a subset of values, the entries with unspecified values will be initialized as specified in the default mapping.

The *dscp-value* variable specifies the value that will be marked onto the DSCP bits within the packet header of the outgoing packets. This applies to packets that match the **priority** and **drop precedence** values specified in this command.

The **[no]** option allows you to negate a previously configured value and return to the default mapping for the specified **priority** and **drop precedence** values.

Configuring an Egress encode PCP policy map

The following procedures are used when configuring an Egress Encode PCP Policy Maps:

- Naming an Egress Encode PCP Policy Map
- Configuring an Egress Encode PCP Policy Map

Naming an Egress encode PCP policy map

Once you are in the QoS configuration level, can define the name of an Egress Encode PCP Policy Map using the **pcp encode-map** command, as shown in the following.

```
Brocade(config)# qos-mapping
Brocade(config-qos-mapping)# pcp encode-map Customer1
```

Syntax: [no] **pcp encode-map** *map-name*

The **no** option is used to delete a currently configured Egress Encode PCP Policy Map. If the policy map is currently in use, the **no** command will be rejected and an error message will be displayed.

The *map-name* variable specifies the name of the Egress Encode PCP Policy Map that you are defining. It can be up to 64 characters in length. You can specify the same policy map name for different types of policy maps. For example, you can use the same name for an Egress Encode PCP Policy Map and an Egress Encode EXP Policy Map.

NOTE
The name "default-map" cannot be used because it is reserved for standard mappings as described in "Default QoS mappings" on page 79.

Configuring an Egress encode PCP policy map

Once you have named an Egress Encode PCP Policy Map using the **pcp encode-map** command, you can set the values of the named policy map. Setting the values in an Egress Encode PCP Policy Map involves specifying a PCP value to be marked in outgoing packets for a specified internal priority value (0 - 7) and optionally a drop precedence value (0 - 3).

To set the values of an Egress Encode PCP Policy Map, first specify name of the policy map and then populate the values in the policy map using the **priority** command as shown in the following.

```
Brocade(config)# qos-mapping
Brocade(config-qos-mapping)# pcp encode-map Customer1
Brocade(config-qos-mapping-pcp-encode)# priority 7 drop-precedence 2 to pcp-value
3
```

Syntax: [no] priority *priority-value* **[drop-precedence** *dp-value*] [*dp-value*] **to pcp-value** *pcp-value*

The **priority** keyword together with the *priority-value* variable specifies the priority value that the egress packets will be marked with. The *priority-value* variable can be a value between 0 and 7. For unspecified priority values, the default mapping values are used.

The **drop-precedence** keyword is an optional parameter that allows you to specify a *dp-number* variable that represents the drop precedence value that you specify in addition to a **priority** *priority-value* value. The *dp-number* variable can be a value between 0 and 3. Multiple *dp-number* variables can be configured in a single command. The default value is “any” which means that drop priorities 0 - 3 are assigned. If a drop precedence value is specified only for a subset of values, the entries with unspecified values will be initialized as specified in the default mapping.

The *pcp-value* variable specifies the value that will be marked onto the PCP bits within the packet header of the outgoing packets. This applies to packets that match the **priority** and **drop precedence** values specified in this command.

The **[no]** option allows you to negate a previously configured value and return to the default mapping for the specified **priority** and **drop precedence** values.

Configuring an Egress Encode EXP policy map

The following procedures are used when configuring an Egress Encode EXP Policy Map:

- Naming an Egress Encode EXP Policy Map
- Configuring an Egress Encode EXP Policy Map

Naming an Egress encode EXP policy map

Once you are in the QoS configuration level, can define the name of an Egress Encode EXP Policy Map using the **exp encode-map** command, as shown in the following.

```
Brocade(config)# qos-mapping
Brocade(config-qos-mapping)# exp encode-map Customer1
```

Syntax: [no] exp encode-map *map-name*

The *map-name* variable specifies the name of the Egress Encode EXP Policy Map that you are defining. It can be up to 64 characters in length. You can specify the same policy map name for different types of policy maps. For example, you can use the same name for an Egress Encode EXP Policy and an Egress Encode DSCP Policy Map.

The **no** option is used to delete a currently configure Egress Encode EXP Policy Map. If the policy map is currently in use, the **no** command will be rejected and an error message will be displayed.

NOTE
The name “default-map” cannot be used because it is reserved for standard mappings as described in “Default QoS mappings” on page 79.

Configuring an Egress encode EXP policy map

Once you have named an Egress Encode EXP Policy Map using the **exp encode-map** command, you can set the values of the named encode policy map. Setting the values in an Egress Encode EXP Policy Map involves specifying an EXP value to be marked in outgoing packets for a specified priority value (0 - 7) and optionally a drop precedence value (0 - 3).

To set the values of an Egress Encode EXP Policy Map, first specify name of the policy map and then populate the values in the policy map using the **priority** command as shown in the following.

```
Brocade(config)# qos-mapping
Brocade(config-qos-mapping)# exp encode-map Customer1
Brocade(config-qos-mapping-exp-encode)# priority 7 drop-precedence 2 to exp-value
3
```

Syntax: **[no] priority** *priority-value* **[drop-precedence** *dp-value***] [***dp-value***] to exp-value** *exp-value*

The **priority** keyword together with the *priority-value* variable specifies the internal forwarding value of the egress packets. The *priority-value* variable can be a value between 0 and 7. For unspecified priority values, the default mapping values are used.

The **drop-precedence** keyword is an optional parameter that allows you to specify a *dp-number* variable that represents the drop precedence value that you specify in addition to a **priority** *priority-value* value. The *dp-number* variable can be a value between 0 and 3. Multiple *dp-number* variables can be configured in a single command. The default value is “any” which means that drop priorities 0 - 3 are assigned. If a drop precedence value is specified only for a subset of values, the entries with unspecified values will be initialized as specified in the default mapping.

The *exp-value* variable specifies the value that will be marked onto the EXP bits within the packet header of the outgoing packets. This applies to packets that match the **priority** and **drop precedence** values specified in this command.

The **[no]** option allows you to negate a previously configured value and return to the default mapping for the specified **priority** and **drop precedence** values.

Binding an Egress encode EXP policy map

You can bind an Egress Encode Policy map globally or per-port using either the default policy map, an all zero policy map, or a user defined policy map. Additionally, for PCP, you can bind the following pre-defined policy maps: 7P1D, 6P2D, and 5P3D. The following procedures describe how to bind Egress Encode Policy Maps:

- Binding an Egress Encode DSCP Policy Map
- Binding an Egress Encode PCP Policy Map
- Binding an Ingress Encode EXP Policy Map

Binding an Egress encode DSCP policy map

The following procedures describe how to configure the binding of an Egress Encode DSCP Policy Map:

- Globally Binding an Egress Encode DSCP Policy Map
- Binding an Egress Encode DSCP Policy Map to a Port

Globally binding an Egress encode DSCP policy map

You can bind an Egress Encode DSCP Policy Map globally for a Brocade device using the **qos dscp encode-policy** command as shown in the following.

```
Brocade(config)# qos dscp encode-policy Customer1
```

Syntax: [no] qos dscp encode-policy *encode-map-name* **| default-map | all-zero-map**

The *encode-map-name* variable is the name assigned to the Egress Encode DSCP Policy Map that you want applied globally on the device. If you try to apply a *encode-map-name* value that has not been defined, the configuration will be rejected. If the *encode-map-name* value that has been defined but the policy has not been configured, the configuration will be accepted and the **default-map** will be applied.

The **default-map** option assigns the default Egress Encode DSCP Policy Map globally on the device. Since the default Egress Encode DSCP Policy Map is the default setting, this option is only required when the device has been previously set to a different Egress Encode DSCP Policy Map. When configured globally, the **qos dscp encode-policy default-map** command will not be displayed within the configuration even if it is explicitly configured.

The **all-zero-map** option assigns an Egress Encode DSCP Policy Map where all 32 combinations of priority and drop precedence are mapped to 0.

The **no** option allows you to withdraw a previously configured encode policy. If the **qos dscp encode-policy** command is not configured, then the **no qos pcp encode-policy** command will generate an error message.

The **no** option allows you to withdraw a previously configured encode policy. If the **qos dscp encode-policy default-map** command is not configured, then the **no qos dscp encode-policy default-map** command will still be allowed because the **qos dscp encode-policy default-map** is the default configuration.

Binding an Egress encode DSCP policy map to a port

You can bind an Egress Encode DSCP Policy Map to a specified port on a Brocade device using the **qos dscp encode-policy** command within an interface configuration, as shown in the following.

```
Brocade(config)# interface ethernet 10/1
Brocade(config-if-e10000-10/1)qos dscp encode-policy Customer1
Brocade(config-if-e10000-10/1)qos dscp encode-policy on
```

Syntax: [no] qos dscp encode-policy *encode-map-name* **| default-map | all-zero-map**

NOTE
The **qos dscp encode-policy on** command is shown in this example because unlike PCP or EXP, the DSCP encode policy is off by default.

The *encode-map-name* variable is the name assigned to the Egress Encode DSCP Policy Map that you want applied to the port whose configuration this is under.

The **default-map** option assigns the default Egress Encode DSCP Policy Map to the port whose configuration this is under. Since the default Egress Encode DSCP Policy Map is the global default setting, this option is only required when the device's global map has been set to an Egress Encode DSCP Policy Map other than the default. The **qos dscp encode-policy** command will not be displayed within the configuration unless it is explicitly configured.

The **all-zero-map** option assigns an Egress Encode DSCP Policy Map where all 32 combinations of priority and drop precedence are mapped to 0.

The **no** option allows you to withdraw a previously configured encode policy. If the **qos pcp encode-policy** command is not configured, then the **no qos pcp encode-policy** command will generate an error message.

The **no** option allows you to withdraw a previously configured Egress Encode DSCP Policy Map. If the **qos dscp encode-policy default-map** command is not configured, then the **no qos dscp encode-policy default-map** command will generate an error message because the **qos dscp encode-policy default-map** command was never configured on the port.

NOTE
The Egress Encode DSCP Policy is applied to the egress port, allowing different egress policies to be configured on each port.

Enabling and disabling an Egress Encode DSCP Policy Map on a port

To enable or disable an Egress Encode DSCP Policy Map on a port, use the **qos dscp encode-policy** command as shown in the following.

```
Brocade(config)# interface ethernet 10/1
Brocade(config-if-e10000-10/1)qos dscp encode-policy on
```

Syntax: [no] qos dscp encode-policy on | off

The **on** option enables DSCP encode on the port. The **qos dscp encode-policy on** command will not be displayed within the configuration unless it is explicitly configured.

The **off** option disables DSCP encode on the port. This is the default setting.

NOTE
Enable Encode DSCP Policy Map on the corresponding mirror port as well by using the **qos dscp encode-policy o**n command.

Binding Egress encode PCP policy map

The following procedures describe how to configure the binding of an Egress Encode PCP Policy Map:

- Globally Binding an Egress Encode PCP Policy Map Policy
- Binding an Egress Encode PCP Policy Map to a Port

Globally binding an Egress Encode PCP Policy Map

You can bind an Egress Encode PCP Policy Map globally for a Brocade device using the **qos pcp encode-policy** command as shown in the following.

```
Brocade(config)# qos pcp encode-policy Customer1
```

Syntax: [no] qos pcp encode-policy *encode-map-name* **| default-map | all-zero-map | 7P1D |6P2D | 5P3D**

The *encode-map-name* variable is the name assigned to the Egress Encode PCP Policy Map that you want applied globally on the device. If you try to apply a *encode-map-name* value that has not been defined, the configuration will be rejected. If the *encode-map-name* value has been defined but the policy has not been configured, the configuration will be accepted and the **default-map** will be applied.

The **default-map** option assigns the default Egress Encode PCP Policy Map globally on the device. Since the default Egress Encode PCP Policy Map is the default setting, this option is only required when the device has been previously set to a different Egress Encode PCP Policy Map. When configured globally, the **qos pcp encode-policy default-map** command will not be displayed within the configuration even if it is explicitly configured.

The **all-zero-map** option assigns an Egress Encode PCP Policy Map where all 32 combinations of priority and drop precedence are mapped to 0.

The **7P1D** option assigns the 7P1D Egress Encode PCP Policy Map globally on the device.

The **6P2D** option assigns the 6P2D Egress Encode PCP Policy Map globally on the device.

The **5P3D** option assigns the 5P3D Egress Encode PCP Policy Map globally on the device.

NOTE
7P1D, 6P2D, and **5P3D** are as defined in the IEEE 802.1ad specification and described in Table 18.

The **no** option allows you to withdraw a previously configured encode policy. If the **qos pcp encode-policy default-map** command is not configured, then the **no qos pcp encode-policy default-map** command will still be allowed because the **qos pcp encode-policy default-map** is the default configuration.

Binding an Egress encode PCP policy map to a port

You can bind an Egress Encode PCP Policy Map to a specified port on a Brocade device using the **qos pcp encode-policy** command as shown in the following.

```
Brocade(config)# interface ethernet 10/1
Brocade(config-if-e10000-10/1)qos pcp encode-policy Customer1
```

Syntax: [no] **qos pcp encode-policy** *encode-map-name* **| default-map | all-zero-map | 7P1D |6P2D | 5P3D**

The *encode-map-name* variable is the name assigned to the Egress Encode PCP Policy Map that you want applied to the port whose configuration this is under.

The **default-map** option assigns the default Egress Encode PCP Policy Map to the port whose configuration this is under. Since the default Egress Encode PCP Policy Map is the default setting, this option is only required when the device's global map has been set to an Egress Encode PCP Policy Map other than the default. The **qos pcp encode-policy default-map** command will not be displayed within the configuration unless it is explicitly configured.

The **all-zero-map** option assigns an Egress Encode PCP Policy Map where all 32 combinations of priority and drop precedence are mapped to 0.

The **7P1D** option assigns the 7P1D Egress Encode PCP Policy Map to the port whose configuration this is under.

The **6P2D** option assigns the 6P2D Egress Encode PCP Policy Map to the port whose configuration this is under.

The **5P3D** option assigns the 5P3D Egress Encode PCP Policy Map to the port whose configuration this is under.

NOTE
7P1D, 6P2D, and **5P3D** are as defined in the IEEE 802.1ad specification and described in Table 18.

The **no** option allows you to withdraw a previously configured Egress Encode PCP Policy Map. If the **qos pcp encode-policy** command is not configured, then the **no qos pcp encode-policy** command will generate an error message.

The **no** option allows you to withdraw a previously configured Egress Encode PCP Policy Map. If the **qos pcp encode-policy default-map** command is not configured, then the **no qos pcp encode-policy default-map** command will generate an error message because the **qos pcp encode-policy default-map** command was never configured on the port.

Enabling and disabling an Egress Encode PCP Policy Map on a port

To enable or disable an Egress Encode DSCP Policy Map on a port, use the **qos pcp encode-policy** command as shown in the following.

```
Brocade(config)# interface ethernet 10/1
Brocade(config-if-e10000-10/1)qos pcp encode-policy on
```

Syntax: qos pcp encode-policy on | off

The **on** option enables PCP encode-policy on the port. This is the default setting. The **qos pcp encode-policy on** command will not be displayed within the configuration unless it is explicitly configured.

The **off** option disables PCP encode-policy on the port.

Binding Egress encode EXP policy maps

The following procedures describe how to configure the binding of an Egress Encode EXP Policy Map:

- Globally Binding an Egress Encode EXP Policy Map
- Binding an Egress Encode EXP Policy Map to a Port

Globally binding an Egress Encode EXP Policy Map

You can bind an Egress Encode EXP Policy Map globally for a Brocade device using the **qos exp encode-policy** command as shown in the following.

```
Brocade(config)# qos exp encode-policy Customer1
```

Syntax: [no] qos exp encode-policy *encode-map-name* **| default-map | all-zero-map**

The *encode-map-name* variable is the name assigned to the Egress Encode EXP Policy Map that you want applied globally on the device. If you try to apply an *encode-map-name* value that has not been defined, the configuration will be rejected. If the *encode-map-name* value that has been defined but the policy has not been configured, the configuration will be accepted and the **default-map** will be applied.

The **default-map** option assigns the default Egress Encode EXP Policy Map globally on the device. Since the default Egress Encode EXP Policy Map is the default setting, this option is only required when the device has been previously set to a different Egress Encode EXP Policy Map. When configured globally, the **qos exp encode-policy default-map** command will not be displayed within the configuration even if it is explicitly configured.

The **all-zero-map** option assigns an Egress Encode EXP Policy Map where all 32 combinations of priority and drop precedence are mapped to 0.

The **no** option allows you to withdraw a previously configured Egress Encode EXP Policy Map. If the **qos exp encode-policy** command is not configured, then the **no qos exp encode-policy** command will generate an error message.

The **no** option allows you to withdraw a previously configured Egress Encode EXP Policy Map. If the **qos exp encode-policy default-map** command is not configured, the **no qos exp encode-policy default-map** command will still be allowed because **qos exp encode-policy default-map** is the default configuration.

Binding an Egress Encode EXP Policy Map to a port

You can bind an Egress Encode EXP Policy Map to a specified port on a Brocade device using the **qos exp encode-policy** command as shown in the following.

```
Brocade(config)# interface ethernet 10/1
Brocade(config-if-e10000-10/1)qos exp encode-policy Customer1
```

Syntax: [no] qos exp encode-policy *encode-map-name* **| default-map | all-zero-map**

The *encode-map-name* variable is the name assigned to the Egress Encode EXP Policy Map that you want applied to the port whose configuration this is under:

The **default-map** option assigns the default Egress Encode EXP Policy Map to the port whose configuration this is under. Since the default Egress Encode EXP Policy Map is the default setting, this option is only required when the device's global policy map has been set to an Egress Encode EXP Policy Map other than the default. The **qos exp encode-policy default-map** command will not be displayed within the configuration unless it is explicitly configured.

The **all-zero-map** option assigns an Egress Encode EXP Policy Map where all 32 combinations of priority and drop precedence are mapped to 0.

The **no** option allows you to withdraw a previously configured encode policy. If the **qos exp encode-policy default-map** command is not configured, then the **no qos exp encode-policy default-map** command will generate an error message because the **qos exp encode-policy default-map** command was never configured on the port.

Enabling and disabling an Egress Encode EXP Policy Map on a port

To enable or disable an Egress Encode EXP Policy Map on a port, use the **qos exp encode-policy** command as shown in the following.

```
Brocade(config)# interface ethernet 10/1
Brocade(config-if-e10000-10/1)qos exp encode-policy on
```

Syntax: [no] qos exp encode-policy on | off

The **on** option enables EXP encode on the port. This is the default setting. The **qos exp encode-policy on** command will not be displayed within the configuration unless it is explicitly configured.

The **off** option disables EXP encode on the port.

Enabling a port to use the DEI bit for Ingress and Egress processing

In the IEEE 802.1ad specification, two types of tag are defined:

- Customer VLAN tag (C-TAG)
- Service VLAN tag (S-TAG)

The semantics and structure of the S-TAG is identical to that of the C-TAG, with the exception that bit 5 in octet 1, the Drop Eligible Indicator (DEI) bit, dis used to indicate if the packet is drop eligible. This allows all 3 bits in the PCP ID to be used for indicating priority of the packet with the drop precedence indicated by the DEI bit. The IEEE 802.1ad requires that if this capability is provided, it must be independently manageable for each port.

On the Brocade device the **qos use-dei** command can be configured at the port level to allow a drop-precedence value for incoming packet to be computed based on the DEI bit. Additionally, if this command is configured, then a drop-eligible parameter will be encoded in the DEI bit of transmitted frames. If the internal drop precedence of the packet is 2 or 3, the DEI will be transmitted as 1; otherwise it will be transmitted as 0.

This command is configured as described in the following.

```
Brocade(config)# interface ethernet 10/1
Brocade(config-if-e10000-10/1)qos use-dei
```

Syntax: [no] qos use-dei

NOTE
This command applies for both Ingress and Egress processing.

Specifying the trust level and enabling marking

The following commands were retained from pre-03-8.00 versions of Multi-Service IronWare software:

- qos-tos trust
- qos-tos mark

These commands operate on the QoS values within the packets as they arrive on the device. The **qos-tos trust** command specifies which value among the following to use to classify the packet for marking: **cos, ip-prec,** and **dscp**. The **qos-tos mark** command specifies a CoS or DSCP value to mark on outgoing packets as specified by the mappings described in "Packet mapping commands" on page 110.

NOTES: You cannot use these commands and other L4 features such as:

- IPv4 ACLs and IPv4 ACL-based rate-limiting
- L2 ACLs and L2 ACL-based rate-limiting
- PBR
- VLAN ID and Inner VLAN ID translation on the same interface

NOTE
The design of this feature requires that the **qos-tos trust** and **qos-tos mark** commands be used together.

NOTE
In versions of the Multi-Service IronWare prior to 03.8.00, before configuring the **qos-tos trust** and **qos-tos** mark commands, you had to configure the **port-priority** command at global CONFIG level. Beginning with version 03.8.00, the **port-priority** command is no longer supported. You can now directly configure the **qos-tos trust** and **qos-tos mark** commands at the interface-level. However without the **port-priority** command configured, the per-port DSCP decode map is not initialized as previously. For information concerning the upgrade of a previously configured DSCP decode map, refer to "DSCP-priority mapping commands" on page 78.

Specifying the trust level

The trust level specifies where you want the device to get the QoS value for a packet received on the interface.

To set the trust level for an interface to IP Precedence, enter the following command at the configuration level for the interface.

```
Brocade(config-if-1/1)# qos-tos trust ip-prec
```

Syntax: [no] qos-tos trust cos | ip-prec | dscp

The **cos | ip-prec | dscp** parameter specifies the trust level:

- **cos** – The device uses the IEEE 802.1p (CoS) priority value in the packet's Ethernet frame header. Use this trust option when you plan to mark the packet's DSCP value based on the incoming IEEE 802.1p value.
- **ip-prec** – The device uses the three most-significant bits in the packet's ToS field and interprets them as an IP precedence value. Use this trust option when the incoming packet is from a device that does not support DSCP and you need to mark the packet for QoS on DSCP devices.
- **dscp** – The device uses the six most-significant bits in the packet's ToS field and interprets them as a DSCP value.

Enabling marking

Marking changes the value of an outbound packet's IEEE 802.1p priority field, DSCP field, or both to match the results of the QoS mappings performed by the device. When you enable marking on an interface, the marking applies to packets that enter the device through that interface.

The following example enables marking for traffic that arrives on port 1/1 and enables the **qos pcp encode-policy on** command on egress port 1/14, as shown.

```
Brocade(config-if-e10000-1/1)# qos-tos mark cos
Brocade(config-if-e10000-1/1)# interface ethernet 1/14
Brocade(config-if-e10000-1/1)# qos pcp encode-policy on
```

This command enables marking of the IEEE 802.1p field in the Ethernet frame.

Syntax: [no] qos-tos mark cos | dscp

The **cos | dscp** parameter specifies the type of marking:

- **cos** – The device changes the outbound packet's IEEE 802.1p priority value to match the results of the device's QoS mapping from the specified trust level.
- **dscp** – The device changes the outbound packet's IEEE 802.1p priority value to match the results of the device's QoS mapping from the specified trust level.

NOTE
In release 03.8.00 and later, the **qos pcp encode-policy on** command must be configured when the **qos-tos mark cos** command is configured. The **qos pcp encode-policy** command is on by default and does not require explicit configuration unless it has been configured to be **off**.

NOTE
You can't apply an ACL to an interface in the outbound direction to change the priority of certain types of traffic.

Packet mapping commands

The **qos-tos trust** command, that is retained from pre-03.8.00 versions of Multi-Service IronWare software, described in the proceeding section, specifies that a COS, IP-Precedence, or DSCP value received on an Ingress port will be used to determine the QoS value that is marked on an outgoing packet on an egress port. The **qos-tos mark** command that is also retained from pre-03.8.00 versions, directs the device to mark outgoing packets with a COS or DSCP value as specified in the command. The value to be marked is determined by a mapping between the value received on the Ingress port and another value that you set using one of the following procedures:

- Changing the CoS -> DSCP Mappings
- Changing the IP Precedence –> DSCP Mappings
- Changing the DSCP –> DSCP Mappings

Changing the CoS –> DSCP mappings

The CoS –> DSCP mappings are used if the trust level is CoS as set by the **qos-tos trust** command.

To change the CoS –> DSCP mappings, enter commands such as the following at the global CONFIG level of the CLI.

```
Brocade(config)# qos-tos map cos-dscp 0 33 25 49 17 7 55 41
Brocade(config)# ip rebind-acl all
```

This command configures the mappings displayed in the COS-DSCP map portion of the QoS information display.

```
Brocade(config-if-1/1)# show qos-tos
```

...portions of table omitted for simplicity...

```
COS-DSCP map:

     COS: 0 1 2 3 4 5 6 7
  ----------------------------------------------------------
     dscp: 0 33 25 49 17 7 55 41
```

Syntax: [no] qos-tos cos-dscp *dscp1 dscp2 dscp3 dscp4 dscp5 dscp6 dscp7 dscp8*

The *dscp1 ... dscp8* parameters specify the DSCP values you are mapping to the eight CoS values. You must enter DSCP values for all eight CoS values, in order from CoS value 0 – 7.

NOTE
To place a qos-tos mapping change into effect, you must enter the **ip rebind-acl all** command at the global CONFIG level of the CLI after making the mapping change. This applies to mappings that are configured using the **qos-tos map** command.

Changing the IP precedence –> DSCP mappings

The IP precedence –> DSCP mappings are used if the trust level is IP Precedence as set by the **qos-tos trust** command.

To change the IP precedence –> DSCP mappings, enter commands such as the following at the global CONFIG level of the CLI.

```
Brocade(config)# qos-tos map ip-prec-dscp 0 32 24 48 16 8 56 40
Brocade(config)# ip rebind-acl all
```

This command configures the mappings displayed in the IP Precedence-DSCP map portion of the QoS information display.

```
Brocade(config-if-1/1)# show qos-tos
```

...portions of table omitted for simplicity...

```
IP Precedence-DSCP map:

  ip-prec:     0     1     2     3     4     5     6     7
  ---------------------------------------------------------
     dscp:     0    32    24    48    16     8    56    40
```

For information about the rest of this display, refer to "Displaying QoS configuration information".

Syntax: [no] qos-tos map ip-prec-dscp *dscp1 dscp2 dscp3 dscp4 dscp5 dscp6 dscp7 dscp8*

The *dscp1 ... dscp8* parameters specify the DSCP values you are mapping to the IP precedence values. You must enter DSCP values for all eight IP precedence values, in order from IP precedence value 0 – 7.

NOTE
To place a qos-tos mapping change into effect, you must enter the **ip rebind-acl all** command at the global CONFIG level of the CLI after making the mapping change. This applies to mappings that are configured using the **qos-tos map** command.

Changing the DSCP –> DSCP mappings

To change a DSCP –> DSCP mapping, enter a command such as the following at the global CONFIG CLI level.

```
Brocade(config)# qos-tos map dscp-dscp 0 10
Brocade(config)# ip rebind-acl all
```

This command changes the mapping of DSCP value 0 from 0 to 10.

Syntax: [no] qos-tos map dscp-dscp *old-dscp-value* [*old-dscp-value...*]
to *new-dscp-value* [*new-dscp-value...*]

You can change up to eight DSCP values in the same commend. Make sure you enter the old values and their new values in the same order.

NOTE
To place a qos-tos mapping change into effect, you must enter the **ip rebind-acl all** command at the global CONFIG level of the CLI after making the mapping change. This applies to mappings that are configured using the **qos-tos map** command.

Configuring support for super aggregate VLANs

In a super-aggregate VLAN application, you can optionally configure an untagged interface to copy the QOS bits from the tag value set by the edge device to the tag value set by the core device. This is only supported if the incoming packet has ETYPE 0x8100. This can be configured using the **qos decode-cvlan-pcp** command as shown in the following.

```
Brocade(config)# interface ethernet 10/1
Brocade(config-if-e10000-10/1)qos decode-cvlan-pcp
```

Syntax: [no] qos decode-cvlan-pcp

NOTE
The command **aggregated-vlan-copy-cos** is available at the physical interface level to copy the COS value from the internal to the external VLAN tag (for SAV). This command will be automatically migrated to the new command **qos decode-cvlan-pcp**.

Configuring port-level QoS commands on LAG ports

When applying port-level QoS commands to ports in a LAG, the rules can differ according the following:

- For port-level QoS Configurations where QoS Values are Applied Directly to the Port. These commands include the followings: **priority**, **priority force**, **drop-precedence**, **drop-precedence force**.
- For Port-level QoS configurations using commands that begin with the **qos** keyword. These commands include: **qos use-dei, qos dscp decode-policy**, **qos pcp decode-policy**, **qos exp decode-policy**, **qos dscp force**, **qos pcp force**, **qos exp force**, **qos dscp encode-policy**, **qos pcp encode-policy**, and **qos exp encode-policy**.

LAG configuration rules where QoS values are applied directly to the port

In port-level QoS Configurations where QoS values are applied directly to the port, the considerations listed below must be followed.

1. Each port that is configured into the LAG, must have the same **priority**, **priority force**, **drop-precedence**, and **drop-precedence force** configuration.

   If you try to configure a LAG with ports that have a different configuration for these commands, the LAG deployment will fail and you will get an error message as shown in the following.

   ```
   Brocade(config)# lag mylag static
   Brocade(config-lag-mylag)# ports eth 10/1 to 10/2
   Brocade(config-lag-mylag)# primary 10/1
   Brocade(config-lag-mylag)# deploy
   port 10/1 priority is 5, but port 10/2 priority is 0
   Error: port 10/1 and port 10/2 have different configurations
   LAG mylag deployment failed!
   Brocade(config-lag-mylag)#
   ```

2. If you have already formed a LAG with the same configuration, you can change the configuration by making changes to the LAG's primary port.
3. If the LAG configuration is deleted, each of the port in the LAG (primary and secondary) will inherit the QoS configuration of the primary port.

LAG configuration rules for QoS configurations using commands that begin with the qos keyword

In port-level QoS Configurations where QoS Configurations Using Commands that begin with the **qos** keyword are used, the considerations listed below must be followed.

1. The secondary ports configured in the LAG must not have any QoS values configured on them.
2. The **qos** commands that are configured on the primary port are applied to all ports in the LAG.
3. After the LAG is formed, you can change the QoS configuration for all ports in the LAG by making changes to the LAG's primary port, but you cannot change the QoS configurations directly on any of the secondary ports.
4. If the LAG is deleted, the QoS configuration will be retained on the primary and secondary ports.

Displaying QoS information

You can display the following QoS information as described:

- **QoS Configuration Information** – Using the **show qos-map decode-map** and **show qos-map encode-map** commands, you can display the priority and drop-precedence values mapped between values internal to the device and values that are received at the device or marked on packets leaving the device. This is described in "Displaying QoS configuration information" on page 113.
- **QoS Packet and Byte Statistics** – Using the **show qos-map decode-map** and **show qos-map encode-map** commands, you can enable and display the contents of the QoS Packet and Byte Counters as described in "Displaying QoS packet and byte counters" on page 116.

Displaying QoS configuration information

You can display the following QoS Configuration information:

- QoS Decode Policy Map Configurations
- QoS Policy Map Binding Configurations

Displaying QoS Decode Policy Map configurations

To display QoS Decode Policy Map configuration information, enter the following command at any level of the CLI.

```
Brocade(config)# Show qos-map dscp decode-map test1
DSCP decode map test1
  DSCP  0 to priority  0 drop-precedence  0
  DSCP  1 to priority  0 drop-precedence  0
  DSCP  2 to priority  0 drop-precedence  1
  DSCP  3 to priority  0 drop-precedence  1
  DSCP  4 to priority  0 drop-precedence  2
  DSCP  5 to priority  0 drop-precedence  2
  DSCP  6 to priority  0 drop-precedence  3
  DSCP  7 to priority  0 drop-precedence  3
  DSCP  8 to priority  7 drop-precedence  0
  DSCP  9 to priority  1 drop-precedence  0
  DSCP 10 to priority  6 drop-precedence  1
  DSCP 11 to priority  1 drop-precedence  1
  DSCP 12 to priority  1 drop-precedence  2
  DSCP 13 to priority  1 drop-precedence  2
  DSCP 14 to priority  1 drop-precedence  3
  DSCP 15 to priority  1 drop-precedence  3
  DSCP 16 to priority  2 drop-precedence  0
  DSCP 17 to priority  2 drop-precedence  0
  DSCP 18 to priority  2 drop-precedence  1
  DSCP 19 to priority  2 drop-precedence  1
  DSCP 20 to priority  2 drop-precedence  2
  DSCP 21 to priority  7 drop-precedence  2
  DSCP 22 to priority  2 drop-precedence  3
  DSCP 23 to priority  2 drop-precedence  3
  DSCP 24 to priority  3 drop-precedence  0
  DSCP 25 to priority  3 drop-precedence  0
  DSCP 26 to priority  3 drop-precedence  1
  DSCP 27 to priority  3 drop-precedence  1
  DSCP 28 to priority  3 drop-precedence  2
  DSCP 29 to priority  3 drop-precedence  2
  DSCP 30 to priority  2 drop-precedence  1
  DSCP 31 to priority  3 drop-precedence  3
   ....
```

Syntax: show qos-map dscp | exp | pcp decode-map *map-name* **| all-zero-map | default-map**

The **dscp** option is used to display an Ingress DSCP Policy Map configuration.

The **exp** option is used to display an Ingress EXP Policy Map configuration.

The **pcp** option is used to display an Ingress PCP Policy Map configuration.

The *map-name* variable is the name of the Ingress Policy Map whose configuration you want to display.

The **all-zero-map** option is used to display the specified Ingress Policy Map's all-zero-map configuration.

The **default-map** option is used to display the specified Ingress Policy Map's default configuration.

Displaying QoS Egress Encode Policy Map configurations

To display QoS Egress Encode Policy Map configuration information, enter the following command at any level of the CLI.

```
Brocade(config)# show qos-map dscp encode-map test2
DSCP encode map test2
  Priority  0 drop-precedence  0 to DSCP  0
  Priority  0 drop-precedence  1 to DSCP  2
  Priority  0 drop-precedence  2 to DSCP  4
  Priority  0 drop-precedence  3 to DSCP  6
  Priority  1 drop-precedence  0 to DSCP 44
  Priority  1 drop-precedence  1 to DSCP 44
  Priority  1 drop-precedence  2 to DSCP 44
  Priority  1 drop-precedence  3 to DSCP 44
  Priority  2 drop-precedence  0 to DSCP 20
  Priority  2 drop-precedence  1 to DSCP 25
  Priority  2 drop-precedence  2 to DSCP 20
  Priority  2 drop-precedence  3 to DSCP 20
  Priority  3 drop-precedence  0 to DSCP 55
  Priority  3 drop-precedence  1 to DSCP 55
  Priority  3 drop-precedence  2 to DSCP 55
  Priority  3 drop-precedence  3 to DSCP 55
  Priority  4 drop-precedence  0 to DSCP 32
  Priority  4 drop-precedence  1 to DSCP 34
  Priority  4 drop-precedence  2 to DSCP 36
  Priority  4 drop-precedence  3 to DSCP 38
  Priority  5 drop-precedence  0 to DSCP 54
  Priority  5 drop-precedence  1 to DSCP 54
  Priority  5 drop-precedence  2 to DSCP 54
  Priority  5 drop-precedence  3 to DSCP 54
  Priority  6 drop-precedence  0 to DSCP 48
  Priority  6 drop-precedence  1 to DSCP 50
  Priority  6 drop-precedence  2 to DSCP 52
  Priority  6 drop-precedence  3 to DSCP 54
  Priority  7 drop-precedence  0 to DSCP 27
  Priority  7 drop-precedence  1 to DSCP 27
  Priority  7 drop-precedence  2 to DSCP 27
  Priority  7 drop-precedence  3 to DSCP 27
```

Syntax: show qos-map dscp | exp | pcp encode-map *map-name* **| all-zero-map | default-map**

The **dscp** option is used to display an Egress DSCP Policy Map configuration.

The **exp** option is used to display an Egress EXP Policy Map configuration.

The **pcp** option is used to display an Egress PCP Policy Map configuration.

The *map-name* variable is the name of the Egress Policy Map whose configuration you want to display.

The **all-zero-map** option is used to display the specified Egress Policy Map's all-zero-map configuration.

The **default-map** option is used to display the specified Egress Policy Map's default configuration.

Displaying QoS Binding configurations

To display QoS Binding configuration information, enter the following command at any level of the CLI.

```
Brocade(config)# show qos-map binding global
 qos pcp decode-policy pcp-t2
 qos exp decode-policy exp-t1
 qos dscp decode-policy dscp-t3
 qos dscp encode-policy dscp-d3
```

Syntax: show qos-map binding global | *slot/port*

The **global** option is used to display all QoS Policy Map bindings configured on the device.

The *slot/port* variable is used to display all QoS Policy Map bindings configured on the device.

Displaying QoS packet and byte counters

You can enable and display the collection of statistics for Ingress and Egress packet priorities as described in the following sections:

- Enabling QoS Packet and Byte Counters
- Displaying QoS Packet and Byte Counters
- Clearing QoS Packet and Byte Counters

Enabling QoS packet and byte counters

You can enable the collection of statistics for Ingress and Egress packet priorities using the **enable-qos-statistics** command as shown in the following.

```
Brocade(config)# enable-qos-statistics
Brocade#
```

Syntax: [no] enable-qos-statistics

The default for this command is disabled.

Using the **no** option returns a previous enabled configuration to the default disabled state.

Displaying QoS packet and byte counters

You can enable the collection of statistics for Ingress and Egress packet priorities using the **enable-qos-statistics** command. Once the collection of statistics is enabled, the **show np qos statistics** command can be used to display a count of the packet priorities of Ingress and Egress packets as shown in the following.

```
Brocade# show np qos statistics eth 1/1
Port 1/1
  Ingress counters:
    COS 0: packets 0                    bytes 0
    COS 1: packets 0                    bytes 0
    COS 2: packets 0                    bytes 0
    COS 3: packets 0                    bytes 0
    COS 4: packets 0                    bytes 0
    COS 5: packets 0                    bytes 0
    COS 6: packets 0                    bytes 0
    COS 7: packets 1122084909           bytes 134650189080
  Egress counters:
    COS 0: packets 0                    bytes 0
    COS 1: packets 0                    bytes 0
    COS 2: packets 0                    bytes 0
    COS 3: packets 0                    bytes 0
    COS 4: packets 4056756685           bytes 486810801752
    COS 5: packets 0                    bytes 0
    COS 6: packets 0                    bytes 0
    COS 7: packets 453                  bytes 49490
```

Syntax: show np qos statistics ethernet *slot/port* **| slot** *slot-number*

The **ethernet** option is used to display all QoS counters for the ethernet interface specified by the *slot/port* variable.

The **slot** option is used to display all QoS counters for the interface module whose location is specified by the *slot-number* variable.

Table 28 describes the parameters displayed for the show np qos statistics command.

TABLE 28 QoS counter information

This field...	Displays...
Ingress Counters	Statistics displayed below this heading are for packets arriving on the Ingress port (or ports) before any overrides or merging of packet priorities have been performed.
COS <*num*>: packets	The number of packets that have arrived on the specified port or module with a DSCP, EXP, or PCP value equal to the value of the <*num*> variable.
COS <*num*>: bytes	The number of bytes contained in the packets that have arrived on the specified port or module with a DSCP, EXP, or PCP value equal to the value of the <*num*> variable.
Egress Counters	Statistics displayed below this heading are for packets leaving the device on the Egress port (or ports) accounting for all priority modifications that have been performed on them. These statistics accurately reflect the values for packets that are forwarded out of the device.

TABLE 28 QoS counter information (Continued)

This field...	Displays...
COS <*num*>: packets	The number of packets leaving the device on the specified port or module with a DSCP, EXP, or PCP value equal to the value of the <*num*> variable.
COS <*num*>: bytes	The number of bytes contained in the packets leaving the device on the specified port or module with a DSCP, EXP, or PCP value equal to the value of the <*num*> variable.

Clearing the QoS packet and byte counters

You can clear the QoS counters whose display is generated using the **show np qos statistics** command as shown in the following.

```
Brocade(config)#clear np qos statistics ethernet 2/5
```

Syntax: clear np qos statistics ethernet *slot/port*

The **ethernet** option is used to clear all QoS counters for the ethernet interface specified by the *slot/port* variable.

Weighted Random Early Discard (WRED)

On the Brocade device, queues are provided to buffer traffic levels that exceed the bandwidth of individual ports. For each output port, a set of eight priority queues is allocated on each inbound traffic manager. When traffic exceeds the bandwidth of a port, packets are dropped randomly as long as the congestion persists. Under these conditions, traffic of greater priority can be dropped instead of traffic with a lesser priority.

Instead of being subject to this random process, you can configure a Brocade device to monitor traffic congestion and drop packets according to a WRED (Weighted Random Early Discard) algorithm. This algorithm enables the system to detect the onset of congestion and take corrective action. In practice, WRED causes a device to start dropping packets as traffic in the device starts to back up. WRED provides various control points that can be configured to change a system's reaction to congestion. The following variables are used when calculating whether to drop or forward packets:

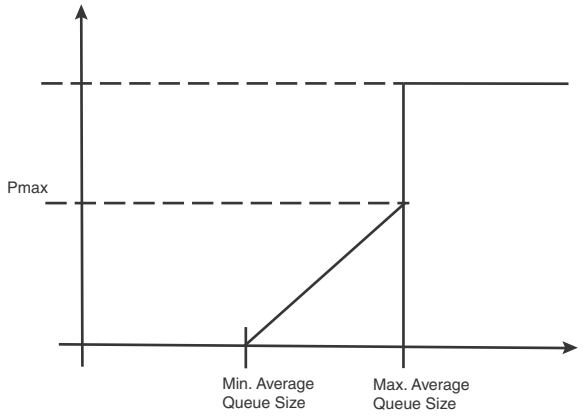
- **Statistical Average-Q-Size** – The statistical average size of the queue calculated over time on the device.
- **Current-Q-Size** – The current size of the queue as calculated on the device.
- **Wq** – This variable specifies the weights that should be given to the current queue size and the statistical average-q-size when calculating the size for WRED calculations.
- **Max-Instantaneous-Q-Size** – The maximum size up to which a queue is allowed to grow. Packets that cause the queue to grow beyond this point are unconditionally dropped. This variable is user configured.
- **Min-Average-Q-Size** – The average queue size below which all packets are accepted. This variable is user configured.
- **Max-Average-Q-Size** – The average queue size above which all packets are dropped. This variable is user configured.
- **Pmax** – The maximum drop probability when queue-size is at Max-Average-Q-Size. This variable is user configured.

- **Pkt-Size-Max** – The packet size to which the current packet's size is compared as shown in the algorithm below. This variable is user configured.

How the WRED algorithm operates

The graph in Figure 4 describes the interaction of the previously described variables in the operation of WRED. When a packet arrives at a device, the average queue size (**q-size**) is calculated (note that this is not the statistical average queue size - refer to "Calculating avg-q-size" on page 119). If **q-size** as calculated is below the configured Min. Average Queue Size, then the packet is accepted. If the average queue size is above the Max. configured Average Queue Size threshold, the packet is dropped. If the instantaneous queue size exceeds the value configured for the Max-Instantaneous-Q-Size, the packet is dropped. If the Average Queue size falls between the Min. Average Queue Size and the Max. Average Queue Size, packets are dropped according to the calculated probability described in "Calculating packets that are dropped" on page 119.

FIGURE 4 WRED operation graph



Calculating avg-q-size

The algorithm first calculates the **avg-q-size** through the following equation.

avg-q-size = ((1 - Wq) * Statistical Average-Q-Size) + (Wq * Current-Q-Size)

The user-configured **Wq** value is instrumental to the calculation and can be:

- equal to the statistical average queue size (**Wq** == 0), or
- equal to the current queue size (**Wq** == 1) or
- be between 0 and 1 (0 < **Wq** < 1).

Lower Wq values cause the **avg-q-size** to lean towards the statistical average queue size, reducing WRED's sensitivity to the current state of the queue and thus reducing WRED's effectiveness. On the other hand, higher **Wq** values cause the **avg-q-size** to lean towards the instantaneous queue size, which exposes WRED to any change in the instantaneous queue size and thus may cause WRED to overreact in cases of bursts. Thus, the value of **Wq** should be carefully chosen according to the application at hand.

Calculating packets that are dropped

The **Pdrop** value, as calculated in the following equation, is the probability that a packet will be dropped in a congested device.

$$Pdrop = \frac{pkt\text{-}size}{pkt\text{-}size\text{-}max} * Pmax * \frac{(avg\text{-}q\text{-}size - min\text{-}avg\text{-}q\ size)}{(max\text{-}avg\text{-}q\text{-}size - min\text{-}avg\text{-}q\ size)}$$

Applying the WRED algorithm to device traffic

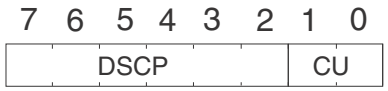
Packets are assigned to an Ingress queue type based on their individual destination port and one of the 8 (0 - 7) internal priorities. Each of these priorities is assigned a queue type from 0 - 7 according to the internal priority it belongs to as shown in Table 29.

TABLE 29 Internal priority to queue type mapping

Internal priority	0	1	2	3	4	5	6	7
Queue type	0	1	2	3	4	5	6	7

The WRED algorithm is applied to traffic on these individual queues based upon parameters configured for its assigned queue type. When traffic arrives at a queue, it is passed or dropped as determined by the WRED algorithm. Packets in an individual queue are further differentiated by one of four drop precedence values which are determined by the value of bits 3:2 of the TOS or DSCP bits in the IPv4 or IPv6 packet header as shown in Figure 5.

FIGURE 5 TOS or DSCP bits in packet header



DSCP = Differentiated Services Codepoint
CU = currently unused

The user configurable values applied per queue type and per drop precedence value are:

- Maximum Drop Probability
- Minimum and Maximum Average Queue Size
- Maximum Packet Size

Configuring packet drop priority using WRED

For a description of WRED, refer to "Weighted Random Early Discard (WRED)" on page 118. This section describes how to configure the parameters described in that section to enable the use of WRED on a Brocade device. In addition, there is a default configuration that can be enabled that sets the parameters to the values shown in Table 31. If you use the default configuration, you do not need to set the parameters individually.

To configure WRED, you must configure the following parameters:

- "Enabling WRED"
- "Setting the averaging-weight (Wq) parameter" (optional)
- "Configuring the maximum instantaneous queue size" (optional)
- "Configuring the drop precedence parameters" (optional)
- "Restoring default WRED parameters"

Enabling WRED

WRED must be enabled for the queue type of any forwarding queue that you want it to operate on. To enable WRED for the forwarding queues with a queue type of 3, enter the following command.

```
Brocade(config)#qos queue-type 3 wred enable
```

Syntax: [no] qos queue-type *queue-number* **wred enable**

The *queue-type* variable is the number of the forwarding queue that you want to enable WRED for. There are eight forwarding queues on Brocade devices. They are numbered 0 to 7.
Default values are as described in Table 31. You can optionally adjust any of the pre-configured parameters described there.

Setting the averaging-weight (Wq) parameter

The Wq parameter described in "Weighted Random Early Discard (WRED)" on page 118 is configured as the **averaging-weight** parameter. In this implementation, you can set one of 13 (1 - 13) possible values. These values represent a Wq value as described in Table 30

TABLE 30 Possible Wq values

Averaging weight setting	Wq value as a percentage
1	50%
2	25%
3	12.5%
4	6.2%
5	3.12%
6	1.56%
7	0.78%
8	0.4%
9	0.2%
10	0.09%
11	0.05%
12	0.02%
13	0.01%

To set the wq parameter for queues with a queue type of 1 to 25%, use the following command.

```
Brocade(config)#qos queue-type 1 wred averaging-weight 2
```

This gives the current queue size a weight of 25% over the statistical average queue size.

Syntax: [no] qos queue-type *queue-type* **wred averaging-weight** *avg-weight-value*

The *queue-type* variable is the number of the forwarding queue type that you want to configure the **averaging-weight** (Wq) parameter for. There are eight forwarding queue types on Brocade devices. They are numbered 0 to 7.

The *avg-weight-value* variable is the weight-ratio between instantaneous and average queue sizes. It is described as the Wq parameter in "Weighted Random Early Discard (WRED)" on page 118. It can be one of the 13 values expressed as 1 to 13 described in Table 30. The default value is 9 which maps to a Wq value of .19%.

Configuring the maximum instantaneous queue size

You can set the maximum size to which a queue is allowed to grow. Packets that cause the queue to grow beyond this setting are unconditionally dropped. To set the maximum instantaneous queue size for queues with a queue type of 1 to 32000 KBytes, use the following command.

```
Brocade(config)#qos queue-type 1 max-queue-size 32
```

Syntax: **[no] qos queue-type** *queue-number* **max-queue-size** *max-queue*

The *queue-type* variable is the number of the forwarding queue type that you want to configure the **instantaneous-queue-size** parameter for. There are eight forwarding queue types on Brocade devices. They are numbered 0 to 7.

The *max-queue* variable is the maximum size to which a queue is allowed to grow. It is defined in Kbytes. The default values are shown in Table 31.

Configuring the drop precedence parameters

The DSCP or TOS bits in packets are used to prioritize packet delivery for specified queue types. These values are from 0 to 4. Packets with a DSCP or TOS value of 0 are least likely to be dropped and packets with a DSCP or TOS of 3 are most likely to be dropped.

NOTE
In addition to bits in the DSCP, the DP option can use other fields (in the PCP header or the EXP bit header) to control WRED in the priority queues.

In addition, the maximum drop probability, the minimum and maximum average queue size, and the maximum packet size can be configured to apply selectively to packets with a specified queue type and DSCP or TOS value. The following sections describe how to set the following drop precedence parameters for each of the four DSCP or TOS values for each of the four queue types:

- "Setting the maximum drop probability"
- "Setting the minimum and maximum average queue size"
- "Setting the maximum packet size"

NOTE
Packets that do not have the DSCP or TOS value set are assigned a drop precedence equal to the DSCP or TOS level of 0.

Setting the maximum drop probability

To set the maximum drop probability for queue type 1 and drop precedence 0 when the queue size reaches the Max-average-q-size value to 20% use the following command.

```
Brocade(config)#qos queue-type 1 wred drop-precedence 0 drop-probability-max 20%
```

Syntax: **[no] qos queue-type** *queue-type* **wred drop-precedence** *drop-precedence-value* **drop-probability-max** *p-max*

The *queue-type* variable is the number of the forwarding queue type that you want to configure drop-precedence for. There are eight forwarding queue types on Brocade devices. They are numbered 0 to 7.

The *drop-precedence-value* variable for the drop-precedence parameter is the TOS or DSCP value in the IPv4 or IPv6 packet header. It determines drop precedence on a scale from **0 - 3**. Packets than contain a DSCP value of **0** are least likely to be dropped and packets with a value of **3** are most likely to be dropped. The default value is **0**.

The *p-max* variable defines the maximum drop probability when the queue size is at the value configured for **max-avg-q-size**. This value is expressed as a percentage. Use the % sign after you type the drop-probability-max value. The default values are shown in Table 31.

Setting the minimum and maximum average queue size

Configuration Considerations

When setting the minimum and maximum average queue size, consider the following

- If a user enters a min-avg-queue-size that is equal to what is currently configured for the max-avg-queue-size, then the min-avg-queue-size is decremented by 64. The min-avg-queue-size is decremented by 64 because the value must be different from the max-avg-queue-size that is currently configured. The following example is a warning message that is displayed on the console.

```
Warning: The min-avg-queue-size is decreased to(min-avq-queue-size-64)as min and
max should be different to be effective.
```

- If a user enters a max-avg-queue-size that is equal to what is currently configured for the min-avg-queue-size, then the max-avg-queue-size is incremented by 64. The max-avg-queue-size is incremented by 64 because the value must be different from the min-avg-queue-size that is currently configured. The following example is a warning message that is displayed on the console.

```
Warning: The max-avg-queue-size is increased to(max-avq-queue-size+64)as min and
max should be different to be effective.
```

- If a user enters a min-avg-queue-size equal to the max-avg-queue-size, then the max-avg-queue-size is incremented by 64. The max-avg-queue-size is incremented by 64 because the value must be different from the min-avg-queue-size The following example is a warning message that is displayed on the console.

```
Warning: The max-avg-queue-size is increased to(max-avq-queue-size+64)as min and
max should be different to be effective.
```

However if a user enters a max-avg-queue-size and min-avg-queue-size equal to 32768, then the min-avg-queue-size is decremented.

To set the maximum average queue size for queue type 1 and drop precedence 0 to the maximum size of 32768 Kbytes, use the following command.

```
Brocade(config)#qos queue-type 1 wred drop-precedence 0 max-avg-queue-size 32768
```

Syntax: **[no] qos queue-type** *queue-type* **wred drop-precedence** *drop-precedence-value* **max-avg-queue-size** *max-size*

To set the minimum average queue size to the maximum size of 16 Kbytes, use the following command.

```
Brocade(config)#qos queue-type 1 wred drop-precedence 0 min-avg-queue-size 16
```

Syntax: **[no] qos queue-type** *queue-type* **wred drop-precedence** *drop-precedence-value* **min-avg-queue-size** *min-size*

The *queue-type* variable is the number of the forwarding queue type that you want to configure drop-precedence for. There are eight forwarding queue types on Brocade devices. They are numbered **0** to **7**.

The *drop-precedence-value* variable for the drop-precedence parameter is the TOS or DSCP value in the IPv4 or IPv6 packet header. It determines drop precedence on a scale from **0 - 3**. Packets than contain a DSCP value of **0** are least likely to be dropped and packets with a value of **3** are most likely to be dropped. The default value is **0**.

The *min-size* variable is the average queue size below which all packets are accepted. Possible values are 1 - 32768 KBytes. It must be set in multiples of 64K. The default values are shown in Table 31.

The *max-size* variable is the average queue size above which all packets are dropped. (1 - 32768) (KBytes) in multiples of 64K. The default values are shown in Table 31.

Setting the maximum packet size

To set the maximum packet size to 16 bytes for queue type 1 and drop precedence 0, use the following command.

```
Brocade(config)#qos queue-type 1 wred drop-precedence 0 packet-size-max 16
```

Syntax: **[no] qos queue-type** *queue-type* **wred drop-precedence** *drop-precedence-value* **packet-size-max** *pkt-size*

The *queue-type* variable is the number of the forwarding queue type that you want to configure drop-precedence for. There are eight forwarding queue types on Brocade devices. They are numbered 0 to 7.

The *drop-precedence-value* variable for the drop-precedence parameter is the TOS or DSCP value in the IPv4 or IPv6 packet header. It determines drop precedence on a scale from 0 - 3. Packets than contain a DSCP value of 0 are least likely to be dropped and packets with a value of 3 are most likely to be dropped. The default value is 0.

The *pkt-size* variable is the pkt-size-max variable used in the equation described in "Calculating packets that are dropped" on page 119. Permissible values are an even number of bytes between 16 and 32768. The default values are shown for each queue type and drop precedence value in Table 31.

Restoring default WRED parameters

Table 31 describes all of the default values for each of the WRED parameters. If you change any of the values from the default values, you can restore the defaults per queue type. To reset the queue type 1 with default values for the WRED parameters, use the following command.

```
Brocade(config)#qos queue-type 1 wred default-params
```

Syntax: **[no] qos queue-type** *queue-number* **default-params**

The *queue-number* variable is the number of the forwarding queue that you want to configure drop-precedence for. There are eight forwarding queues on Brocade devices. They are numbered 0 to 7

TABLE 31 WRED default settings

Queue type	Drop precedence	Minimum average queue size (KByte)	Maximum average queue size (KByte)	Maximum packet size (Byte)	Maximum drop probability	Maximum instantaneous queue size (Kbyte)	Average weight
0	0	320	1024	16384	2%	1024	6.25%
	1	256	1024	16384	4%		
	2	256	1024	16384	9%		
	3	192	1024	16384	10%		
1	0	320	1024	16384	2%	1024	6.25%
	1	256	1024	16384	4%		
	2	256	1024	16384	9%		
	3	192	1024	16384	9%		
2	0	384	1024	16384	2%	1024	6.25%
	1	320	1024	16384	4%		
	2	256	1024	16384	9%		
	3	256	1024	16384	9%		
3	0	384	1024	16384	2%	1024	6.25%
	1	320	1024	16384	4%		
	2	256	1024	16384	9%		
	3	256	1024	16384	9%		
4	0	384	1024	16384	2%	1024	6.25%
	1	320	1024	16384	4%		
	2	256	1024	16384	9%		
	3	256	1024	16384	9%		
5	0	384	1024	16384	2%	1024	6.25%
	1	320	1024	16384	4%		
	2	256	1024	16384	9%		
	3	256	1024	16384	9%		
6	0	1024	1088	16384	0%	1024	6.25%
	1	448	832	16384	2%		
	2	384	832	16384	5%		
	3	320	832	16384	6%		
7	0	1024	1088	16384	0%	1024	6.25%
	1	448	832	16384	2%		
	2	384	832	16384	5%		
	3	320	832	16384	6%		

NOTES:

- If you enter the **min-avg-queue-size** equal to what is already configured as the **max-avg-queue-size**, then the **min-avg-queue-size** will be decremented by 64 to make it different from the **max-avg-queue-size**, the following warning is displayed: "Warning - **min-avg-queue-size** is decreased to (**min-avg-queue-size** - 64) as min and max should be different to be effective."
- If you enter the **max-avg-queue-size** equal to what is already configured as the min-avg-queue-size, then the **max-avg-queue-size** will be incremented by 64 to make it different from the **min-avg-queue-size**, the following warning is displayed: "Warning - max-avg-queue-size is increased to (max-avg-queue-size + 64) as the min & max should be different to be effective."
- If you enter the min-average-queue-size equal to the **max-avg-queue-size**, the **max-avg-queue-size** will be incremented by 64 to make it different from **min-avg-queue-size**, the following warning is displayed: "Warning - **max-avg-queue-size** increased to (**max-avg-queue-size + 64**) as min & max should be different to be effective." Unless you enter the **max-avg-queue-size** and **min-avg-queue-size** equal to 32768, the **min-avg-queue-size** will be decremented.

Displaying the WRED configuration

To view a WRED configuration, use the following command.

```
Brocade# show qos wred
QType Enable AverWt    MaxQSz DropPrec MinAvgQSz MaxAvgQSz MaxDropProb MaxPktSz
0     Yes    9(0.19%)  16384  0           5696     16384          2%      16384
                               1           4864     16384          4%      16384
                               2           4096     16384          9%      16384
                               3           3264     16384         10%      16384
1     No
2     No
3     Yes    9(0.19%)  16384  0           6528     16384          2%      16384
                               1           5696     16384          4%      16384
                               2           4864     16384          9%      16384
                               3           4096     16384          9%      16384
4     No
5     No
6     No
7     No
```

Scheduling traffic for forwarding

If the traffic being processed by a Brocade device is within the capacity of the device, all traffic is forwarded as received. Once we reach the point where the device is bandwidth constrained, it becomes subject to drop priority if configured as described in "Configuring packet drop priority using WRED" on page 120 or traffic scheduling as described in this section.

The Brocade devices classify packets into one of eight internal priorities. Traffic scheduling allows you to selectively forward traffic according to the forwarding queue that is mapped to according to one of the following schemes:

- **Strict priority-based scheduling** – This scheme guarantees that higher-priority traffic is always serviced before lower priority traffic. The disadvantage of strict priority-based scheduling is that lower-priority traffic can be starved of any access.

- **WFQ weight-based traffic scheduling** – With WFQ destination-based scheduling enabled, some weight-based bandwidth is allocated to all queues. With this scheme, the configured weight distribution is guaranteed across all traffic leaving an egress port and an input port is guaranteed allocation in relationship to the configured weight distribution.
- **Mixed strict priority and weight-based scheduling** – This scheme provides a mixture of strict priority for the three highest priority queues and WFQ for the remaining priority queues.

Configuring traffic scheduling

Traffic scheduling can be configured on a per-port basis. It affects the outgoing traffic on the configured port when bandwidth congestion occurs on that port. The following sections describe how to configure each of the traffic scheduling schemes:

- "Configuring strict priority-based traffic scheduling" This option is the default traffic scheduling method if traffic scheduling is not configured on a port.
- "Calculating the values for WFQ Weight-based traffic scheduling"
- "Configuring WFQ weight-based traffic scheduling"
- "Configuring mixed strict priority and weight-based scheduling"
- "Configuring egress unicast and multicast traffic scheduling,"

Configuring strict priority-based traffic scheduling

To configure strict priority-based scheduling use a command such as the following.

```
Brocade(config)# interface ethernet 1/1
Brocade(config-if-e1000-1/1)# qos scheduler strict
```

Syntax: qos scheduler strict

This is the default when traffic scheduling is not configured.

Calculating the values for WFQ Weight-based traffic scheduling

Weighted Fair Queueing (WFQ) scheduling is configured to be a percentage of available bandwidth using the following formula.

$$\text{Weight of q (x)} = \frac{q(x)}{q0 + q1 + q2 + q3 + q4 + q5 + q6 + q7}$$

Where

q (x) = The value of the queue that you want to determine the weight for. It can be the value of any queue (0 - 7).

q0 - q7 = the assigned values of the eight queues.

Weight of q (x) = the calculated weight as a percentage of the port's total bandwidth.

For example if you assign the following values to queues 0 to 7:

- Queue 0 =10, Queue 1 = 15, Queue 2 = 20, Queue 3 = 25, Queue 4 = 30, Queue 5 = 35, Queue 6 = 40, and Queue 7 = 45,

To determine the weight of **q3**, use the following formula.

$$\text{Weight of q3} = \frac{25}{10 + 15 + 20 + 25 + 30 + 35 + 40 + 45}$$

The weight of q3 is 11.4%. Consequently, q3 will get 11.4% of the port's total bandwidth.

The values of the remaining queues are calculated to be the following: q7 = 20.5%, q6 = 18.2%, q5 = 15.9%, q4 = 13.6%, q3 = 11.4%, q2 = 9.1%, q1 = 6.8%, and q0 = 4.5%

Configuring WFQ weight-based traffic scheduling

To configure WFQ weight-based scheduling use a command such as the following.

```
Brocade(config)# interface ethernet 1/1
Brocade(config-if-e1000-1/1)# qos scheduler weighted 5 10 15 20 30 15 5 10
```

Syntax: **qos scheduler weighted** *queue7-weight queue6-weight queue5-weight queue4-weight queue3-weight queue2-weight queue1-weight queue0-weight*

The *queue7-weight* variable defines the relative value for queue7 in calculating queue7's allocated bandwidth.

The *queue6-weight* variable defines the relative value for queue6 in calculating queue6's allocated bandwidth.

The *queue5-weight* variable defines the relative value for queue5 in calculating queue5's allocated bandwidth.

The *queue4-weight* variable defines the relative value for queue4 in calculating queue4's allocated bandwidth.

The *queue3-weight* variable defines the relative value for queue3 in calculating queue3's allocated bandwidth.

The *queue2-weight* variable defines the relative value for queue2 in calculating queue2's allocated bandwidth.

The *queue1-weight* variable defines the relative value for queue1 in calculating queue1's allocated bandwidth.

The *queue0-weight* variable defines the relative value for queue0 in calculating queue0's allocated bandwidth

The acceptable range for *queuex-weight* variables is 1-128.

Refer to "Calculating the values for WFQ Weight-based traffic scheduling" for information on assigning queue0-weight to queue7-weight values.

Configuring mixed strict priority and weight-based scheduling

When configuring the mixed strict priority and weight-based scheduling option, queues 5 - 7 are allocated to strict priority-based scheduling and queues 0 - 4 are allocated to weight-based scheduling.

To configure mixed priority and weight-based scheduling use a command such as the following.

```
Brocade(config)# interface ethernet 1/1
Brocade(config-if-e1000-1/1)# qos scheduler mixed 100 80 60 40 20
```

Syntax: **qos scheduler mixed** *queue4-weight queue3-weight queue2-weight queue1-weight queue0-weight*

The *queue4-weight* variable defines the relative value for queue4 in calculating queue4's allocated bandwidth.

The *queue3-weight* variable defines the relative value for queue3 in calculating queue3's allocated bandwidth.

The *queue2-weight* variable defines the relative value for queue2 in calculating queue2's allocated bandwidth.

The *queue1-weight* variable defines the relative value for queue1 in calculating queue1's allocated bandwidth.

The *queue0-weight* variable defines the relative value for queue0 in calculating queue0's allocated bandwidth

The acceptable range for *queuex-weight* variables is 1-128.

Refer to "Calculating the values for WFQ Weight-based traffic scheduling" for information on assigning queue0-weight to queue4-weight values.

Configuring egress unicast and multicast traffic scheduling

You can schedule the egress unicast and multicast traffic based on the allocated bandwidth. To configure WFQ weight-based scheduling for an interface, enter the following commands.

```
Brocade(config)# interface ethernet 1/1
Brocade(config-if-e1000-1/1)# qos egress-weight unicast 1 multicast 3
```

Syntax: [**no**] **qos egress-weight unicast** *unicast-weight-val* **multicast** *multicast-weight-val*

The **unicast** *unicast-weight-val* variable specifies the allocated bandwidth for the egress unicast traffic. The value ranges from 1 through 255.

The **multicast** *multicast-weight-val* variable specifies the allocated bandwidth for the egress multicast traffic. The value ranges from 1 through 255.

Egress port and priority based rate shaping

Rate shaping is a mechanism to smooth out the variations in traffic above a certain rate. The primary difference between rate shaping and rate limiting is that in rate limiting, traffic exceeding a certain threshold is dropped. In rate shaping, the traffic that exceeds a threshold is buffered so that the output from the buffer follows a more uniform pattern. Rate shaping is useful when burstiness in the source stream needs to be smoothed out and a more uniform traffic flow is expected at the destination.

NOTE
Because excess traffic is buffered, rate shaping must be used with caution. In general, it is not advisable to rate shape delay-sensitive traffic.

Brocade devices support egress rate shaping. Egress rate shaping is supported per port or for each priority queue on a specified port.

Configuring port-based rate shaping

When setting rate shaping for a port, you can limit the amount of bandwidth available on a port within the limits of the port's rated capacity. Within that capacity, you can set the bandwidth at increments within the ranges described in Table 32.

TABLE 32 Port-based rate shaping interval table

Range	Increment supported within the range
0 - 10M	8,333
10M - < 100M	20,833
100 M - < 1G	208,333
1G - 10G	2,083,333

NOTE
The egress rate shaping burst size for a port-based shaper is 10000 bytes.

These limits provide a minimum and maximum rate that the port can be set to. They also provide the increments at which the port capacity can be set. In operation, you can set any number between the minimum and maximum values. The device will automatically round-up the value to the next higher increment.

For example, if you set the rate of a 10G port to 2,000,000,000, the actual rate would be 2,002,083,173. This is because it is the next highest increment above 2,000,000,000.

To set a 10 Gbps port to the incremental port capacity over 2 Gbps, use the following command.

```
Brocade(config)# interface ethernet 2/2
Brocade(config-if-e10000-2/2)# qos shaper 2000000000
```

Syntax: [no] qos shaper *rate*

The *rate* variable sets the rate you want to set for the port within the limits available as described in Table 32. The rate is set in bps.

NOTE
When the rate shaping of a port is enabled, you can expect a deviation of +3% or -3% actual port rate from the configured port rate. The deviation may vary between different types of TM devices within the ranges mentioned in Table 32. In addition, if you configure port shaper on 24x10, 2x100, and 8x10 modules for 8-400 MB size, then the result will not be deterministic and priorities may not be guaranteed for the bandwidth configured. The shaper configuration less than 8 MB is not supported.

Configuring port and priority-based rate shaping

When setting rate shaping for a priority queue, you can limit the amount of bandwidth available for a specified priority within the limits of the capacity of the port that the priority is configured on. You can set the limit for the priority to any value from one to the port's maximum rating and the device will automatically round-up the value to the next increment supported. This will be a slightly higher value than what you specify with the command. For example, if you set the rate for priority 2 on a 10G port to 2,000,000,100, the actual rate would be slightly higher.

NOTE
The egress rate shaping burst size for a port and priority-based shaper is 3072 bytes.

To set the capacity for priority 2 traffic on a 10 Gbps port to the incremental capacity over 2 Gbps, use the following command.

```
Brocade(config)# interface ethernet 2/2
Brocade(config-if-e10000-2/2)# qos shaper priority 2 2000000000
```

Syntax: [no] qos shaper priority *priority-level rate*

The *priority-level* variable specifies the priority that you want to set rate shaping for on the port being configured.

The *rate* variable sets the rate you want to set for the priority. The rate is set in bps.

Multicast queue size, flow control, rate shaping and egress buffer threshold

There are four internal priorities for multicast or broadcast traffic. These four priorities are mapped from the device's eight internal priorities as described inTable 33

TABLE 33 Mapping between multicast or broadcast and internal forwarding priorities

Internal Forwarding Priority	0,1	2,3	4,5	6,7
Multicast Internal Priority	0	1	2	3

The internal forwarding priority of a multicast or broadcast packet is determined from the packet's IEEE 802.1p priority, incoming port priority or IP ToS or DSCP as described in the "Default QoS mappings" on page 79. Four multicast queue types (0 to 3) are used for multicast internal priorities 0 to 3 respectively.

NOTE
Instead of ACL priority, use VLAN or port priority to prioritize multicast traffic.

Configuring multicast queue size

The following example configures a 2 MByte queue size for queue 0.

```
Brocade(config)# qos multicast-queue-type 0 max-queue-size 2048
```

Syntax: [no] qos multicast-queue-type *queue-number* **max-queue-size** *queue-size*

The *queue-number* variable specifies the queue that you want to configure a maximum size for. Possible values are 0 - 3.

The *queue-size* variable specifies size in KBytes that you want to set as the maximum value for the specified multicast queue. Possible values are 1 - 32768 KBytes. The default queue size is 1 Mbyte.

This command is applied per device and takes effect on all Traffic Managers within the configured device.

Configuring multicast flow control

Flow controls are available from egress to Ingress, and from fabric to Ingress. At the egress of each Traffic Manager, there are pre-determined thresholds for consumed resources and available resources and separate thresholds for guaranteed multicast or broadcast traffic and best-effort multicast or broadcast traffic. When a threshold is crossed, flow control can be triggered and multicast or broadcast traffic of the corresponding class is stopped at Ingress until resources are below the threshold again. Flow control is disabled by default and can be enabled on an interface using the command shown in the following.

```
Brocade(config)# interface ethernet 2/2
Brocade(config-if-e10000-2/2)# qos multicast flow-control
```

Syntax: [no] qos multicast flow-control

This command changes the flow control setting on the Traffic Manager where the interface resides.

Configuring multicast rate shaping

You can specify either guaranteed or best effort multicast rate shaping for a port in Kilobits per second. Multicast rate shaping is configured per-port to the Ingress port.

The following example changes the best-effort multicast traffic rate to 10 Mbps.

```
Brocade(config)# interface ethernet 2/2
Brocade(config-if-e10000-2/2)# qos multicast shaper best-effort rate 10000
```

Syntax: [no] qos multicast shaper [guaranteed | best-effort] rate *bandwidth*

The **guaranteed** option specifies that the multicast or broadcast shaper applies only to internal multicast priority 3 (the highest multicast priority) traffic.

The **best-effort** option specifies that the multicast or broadcast shaper applies to internal multicast priority 0, 1 and 2 traffic only.

The *bandwidth* variable specifies the maximum bandwidth in Kbps for best effort or guaranteed multicast traffic scheduled by the Traffic Manager across the switch fabric.

Configuration considerations for multicast rate shaping

When applied to a port, the **qos multicast shaper** configuration is applied to all ports on the Interface module that use the same Traffic Manager as the configured port. This is unlike the behavior of rate shaping applied for unicast traffic. The relationship between ports and Traffic Managers is defined in the following tables: "Ethernet ports per traffic manager" on page 141.

Example 1

In the following example, multicast rate shaping is applied to port 1/18 on a 20-port, 10/100/1000 Copper Ethernet Interface module (NI-XMR-1Gx20-GC).

```
Brocade(config)# interface ethernet 1/18
Brocade(config-if-e1000-1/18)# qos multicast shaper best-effort rate 10000
```

In this example, the configuration will apply to Ingress traffic that arrives on any port of the Interface module.

Example 2

In the following example, multicast rate shaping is applied to port 1/1 of a 4-port, 10 GbE Ethernet Interface module (NI-XMR-10Gx4).

```
Brocade(config)# interface ethernet 1/1
Brocade(config-if-e10000-1/1)# qos multicast shaper best-effort rate 10000
```

In this example, the configuration will apply to Ingress traffic that arrives on either port 1/1 or port 1/2 of the Interface module.

NOTE
When a **qos multicast shaper** command is configured for a port, the configuration command is placed in the running config for all ports that belong to the same Traffic Manager. In Example 1, that would mean that the **qos multicast shaper best-effort rate 10000** command would appear in the interface configuration section for all ports (1 to 20) on the Interface Module. In Example 2, that would mean that the **qos multicast shaper best-effort rate 10000** command would appear in the interface configuration section for ports 1 and 2 on the Interface Module.

Configuring multicast egress buffer threshold

With the current configuration egress buffer threshold per port are set to 50% of total egress buffer size, with the new command you can set multicast egress buffer threshold up to 95% of total egress buffer size which helps in reducing egress packet drops when there is a high multicast traffic.

NOTE
It is recommended to use this command when a sudden burst is seen in multicast traffic and not for general use.

The following example explains how to set the thresholds for individual ports so that the port can use the total egress buffer. This is useful when the multicast traffic is very high.

```
Brocade#config terminal
Brocade(config)#interface ethernet 1/1
Brocade(config-if-e10000-1/1)# qos multicast egress-max-buffer port 95% 90%
```

Syntax: qos multicast egress-max-buffer port { [*guaranteed_max_buffer*] [*best-effort_max_buffer*] }

The **guaranteed_max_buffer** specifies the maximum buffer size allowed per port for guaranteed traffic flow (multicast port priority 3). Specified as percentage of total buffer size.

The **best-effort_max_buffer** specifies the maximum buffer size allowed per port for guaranteed traffic flow (multicast port priority 2-0). Specified as percentage of total buffer size.

Additionally you can also have the threshold configured for individual ports so that each port has its dedicated buffer space.

The example uses a 8x10 line card and has four ports per TM. The buffer size is divided into 4 times the total buffer size so that each port has its dedicated buffer space.

```
Brocade#config terminal
Brocade(config)#interface ethernet 1/1
Brocade(config-if-e10000-1/1)# qos multicast egress-max-buffer port 24% 23%
```

Syntax: qos multicast egress-max-buffer port { [*guaranteed_max_buffer*] [*best-effort_max_buffer*] }

The **guaranteed_max_buffer** specifies the maximum buffer size allowed for guaranteed traffic flow (multicast port priority 3). Specified as percentage of total buffer size.

The **best-effort_max_buffer** specifies the maximum buffer size allowed for guaranteed traffic flow (multicast port priority 2-0). Specified as percentage of total buffer size.

Ingress traffic shaping per multicast stream

Internet Protocol Television (IPTV) multicast streams on an individual inbound physical port are rate shaped to a specified rate and are prioritized over the broadcast or unknown-unicast traffic. Each IPTV multicast stream is queued separately and is scheduled independently to the outbound ports. The IPTV rate shaping reduces burstiness in the source stream.

NOTE
The number of active IPTV multicast streams for which per stream ingress shaping can be applied is limited to 512.

NOTE
Internet Protocol Television (IPTV) multicast streams are not supported on Brocade NetIron CES and Brocade NetIron CER devices.

Implementation considerations

The considerations for implementing the multicast rate shaping are as follows:

- At least the rate or the priority value must be specified.
- A maximum of 32 profiles and 64 profile-ACL bindings are allowed for multicast traffic.
- A single profile can be bound to multiple ACLs.
- An ACL can be associated only to one profile at a time.
- Either standard or extended ACLs can be used for multicast traffic shaping.
  - When a standard ACL is used, the address specified is treated as a group address and not as a source address.
  - When an extended ACL is used, the source and the destination addresses are treated as the source and group address of the multicast stream, respectively.

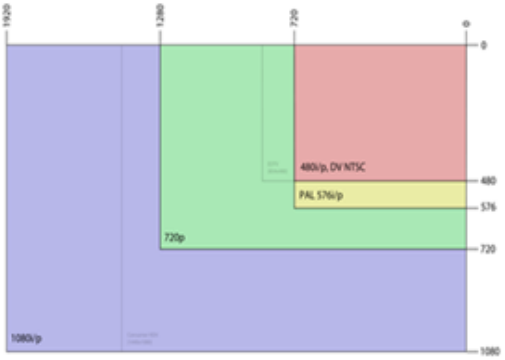
Figure 6 shows the type of IPTV channels and the bandwidth requirements for each type of channel. The following conventions are used in the figure:

- SDTV denotes Standard Definition Television
- HDTV denotes High Definition Television

NOTE
This feature is supported only on the 4x10 and 24x1 interface modules.

FIGURE 6 IPTV Bandwidth Requirements

Channel Encoding	Per Channel	Resolution @ Frames/sec
MPEG-2 SDTV	3-6 Mbps	480i @ 29 576i @ 25
MPEG-2 HDTV	14-20 Mbps	1080i @ 25,29,30 720p @ 50,59,60
H.264/MPEG-4 SDTV	1-4 Mbps	1,280×720@68.3 1,920×1,080@30.1 2,048×1,024@30.0
H.264/MPEG-4 HDTV	6-14 Mbps	1280×720 pixels (720p) 1920×1080 pixels (1080i/1080p)



Channel Encoding Relative TV Format

Configuring multicast traffic policy maps

You can define profiles to match the IPTV multicast traffic of the individual ingress streams. To configure a policy map for the multicast streams, enter the following command.

```
Brocade(config)# policy-map multicast sd_prof rate 2000 burst-size 2500 priority 2
queue-type 3
```

Syntax: [**no**] **policy-map multicast** *profile_name* [**rate** *r*] [**burst-size** *b*] [**priority** *0-7*] [**queue-type** *0-3*]

The *profile_name* is used to provide the parameters for traffic policing the multicast traffic.

The **rate** *r* variable specifies the shaping rate in bits per second. The value ranges from 100 kilobits per second (Kbps) through 20 gigabits per second (Gbps).

The **priority** *0-7* variable specifies the multicast traffic priority. The default value is four.

The **burst-size** *b* variable specifies the maximum number of bytes the multicast traffic is allowed to burst. The value ranges from 3 kilobytes (KB) through 128 KB. The default value is four KB.

The **queue-type** *0-3* variable specifies the queue type for which you want to set the priority. The default value is two. Optionally, you can specify the **burst-size** *b* and the **queue-type** *0-3* value to define a profile.

To delete a defined profile, enter the following command with the profile name.

```
Brocade(config)# policy-map multicast sd_prof
```

The **no** form of the command resets the parameters to their default values.

NOTE
The rate and the priority value cannot be reset for a defined profile.

Binding multicast traffic policy maps

NOTE
A profile must exist in the configuration before it can be used for binding.

A standard or an extended ACL is used to define the IPTV streams that can be bound to a defined profile. The profile binding associates the properties of the profile to all the IPTV streams identified by the ACL. Binding of multicast streams can be done for Layer 3 multicast routing and Layer 2 multicast snooping.

Profile binding for Layer 3 multicast routing

You can bind a defined profile to a defined ACL for the default VRF or a specific VRF.

To bind the profile to the ACL in the default VRF, enter the following commands.

```
Brocade(config)# ip multicast-routing policy-map r1 sd-1
Brocade(config)# ipv6 multicast-routing policy-map r1q0 sdv61
```

To bind the profile to the ACL for a specified VRF, enter the following commands within the VRF "red" configuration context.

```
Brocade(config)# ip vrf red
Brocade(config-vrf-red)# address-family ipv4
Brocade(config-vrf-red-ipv4)# ip multicast-routing policy-map r1 sd-1
Brocade(config-vrf-red-ipv4)# exit-address-family
Brocade(config-vrf-red)# exit-vrf

Brocade(config)# ip vrf red
Brocade(config-vrf-red)# address-family ipv6
Brocade(config-vrf-red-ipv6)# ipv6 multicast-routing policy-map r1q0 sdv61
Brocade(config-vrf-red-ipv6)# exit-address-family
Brocade(config-vrf-red)# exit-vrf
```

Syntax: [**no**] **ip multicast-routing policy-map** *profile_name acl_id* | *acl_name*

The **ip multicast-routing policy-map** specifies ACL binding for IPv4 multicast routing.

The *profile_name* variable specifies the profile name of the multicast stream.

The *acl_id* | *acl_name* variable specifies the number and name of the standard ACL or an extended ACL. Enter a number from 1 through 99 for a standard ACL, and a number from 100 through 199 for an extended ACL.

Syntax: [**no**] **ipv6 multicast-routing policy-map** *profile_name acl_id* | *acl_name*

The **ipv6 multicast-routing policy-map** specifies ACL binding for IPv6 multicast routing.

The **no** form of the command removes the profile binding with the ACL in default VRF.

Profile binding for Layer 2 multicast snooping

You can bind a defined profile to a defined ACL per VLAN or per VPLS instance. To bind the profile to the ACL on VLAN 10, enter the following commands.

```
Brocade(config)# vlan 10
Brocade(config-vlan-10)# ip multicast policy-map r2q1 2
Brocade(config-vlan-10)# ipv6 multicast policy-map r4q3 sdv64
```

Syntax: [**no**] **ip multicast policy-map** *profile_name acl_id* | *acl_name*

The **ip multicast policy-map** specifies the ACL binding for IPv4 multicast snooping.

Syntax: [**no**] **ipv6 multicast policy-map** *profile_name acl_id | acl_name*

The **ipv6 multicast policy-map** specifies the ACL binding for IPv6 multicast snooping.

The **no** form of the command removes the profile binding with the ACL on the VLAN or VPLS.

In the following example, binding for Layer 2 multicast snooping is applied to VPLS instance V1.

```
Brocade(config)# router mpls
Brocade(config-mpls)# vpls v1 10
Brocade(config--mpls-vpls-v1)# multicast policy-map r2q1 2
Brocade(config--mpls-vpls-v1)# multicast policy-map r4q3 sdv64
```

Syntax: [**no**] **multicast policy-map** *profile_name acl_id | acl_name*

NOTE
A profile that is bound cannot be deleted.

Configuration example for rate shaping IPTV multicast stream

The following example shows how to rate shape a multicast stream. In this example, to rate shape a multicast stream, profiles (sd_prof and hd_prof) are defined with rate and priority, ACLs (hd_streams and sd_streams) are configured which permit packets from four host IP addresses and denies all packets that are not explicitly permitted by the first four ACL entries, and then the profiles are bound to the defined ACLs.

```
Brocade(config)# ip access-list standard hd_streams
Brocade(config-std-nacl)# permit host 239.1.1.200
Brocade(config-std-nacl)# permit host 239.1.1.201
Brocade(config-std-nacl)# permit host 239.1.1.202
Brocade(config-std-nacl)# permit host 239.1.1.203
Brocade(config-std-nacl)# deny any
Brocade(config-std-nacl)# exit

Brocade(config)# ip access-list extended sd_streams
Brocade(config-ext-nacl)# permit ip any 239.1.1.1/32
Brocade(config-ext-nacl)# permit ip any 239.1.1.2/32
Brocade(config-ext-nacl)# permit ip any 239.1.1.3/32
Brocade(config-ext-nacl)# permit ip any 239.1.1.5/32
Brocade(config-ext-nacl)# deny ip any any
Brocade(config-ext-nacl)# exit

Brocade(config)# policy-map multicast profile sd_prof rate 2000
Brocade(config)# policy-map multicast profile hd_prof rate 14000 queue-type 2

Brocade(config)# ip multicast-routing policy-map sd_prof sd_streams
Brocade(config)# ip multicast-routing policy-map hd_prof hd_streams
```

Naming convention used in example

- sd_prof = Standard Definition profile
- hd_prof = High Definition profile
- sd_streams = Standard Definition TV stream
- hd_stream = High Definition TV stream

Traffic manager statistics display

Counters have been introduced to track the packets and bytes that enter the Ingress traffic manager and exit the egress traffic manager. Data from these counters can be displayed as described in the following sections.

Displaying all traffic manager statistics for a device

The following command displays all traffic manager statistics for a device by port groups that belong to each traffic manager.

```
Brocade# show tm statistics
--------- Ports 2/1 - 2/20 ---------
Ingress Counters:
   Total Ingress Pkt Count:                 464418
   EnQue Pkt Count:                         464418
   EnQue Byte Count:                        51904240
   DeQue Pkt Count:                         464418
   DeQue Byte Count:                        51904240
   TotalQue Discard Pkt Count:              0
   TotalQue Discard Byte Count:             0
   Oldest Discard Pkt Count:                0
   Oldest Discard Byte Count:               0
Egress Counters:
   EnQue Pkt Count:                         701812
   EnQue Byte Count:                        78785888
   Discard Pkt Count:                       0
   Discard Byte Count:                      0
--------- Ports 4/1 - 4/20 ---------
Ingress Counters:
   Total Ingress Pkt Count:                 0
   EnQue Pkt Count:                         0
   EnQue Byte Count:                        0
   DeQue Pkt Count:                         0
   DeQue Byte Count:                        0
   TotalQue Discard Pkt Count:              0
   TotalQue Discard Byte Count:             0
   Oldest Discard Pkt Count:                0
   Oldest Discard Byte Count:               0
Egress Counters:
   EnQue Pkt Count:                         0
   EnQue Byte Count:                        0
   Discard Pkt Count:                       0
   Discard Byte Count:                      0
```

Syntax: show tm statistics

Displaying traffic manager statistics for a port group

The following command displays all traffic manager statistics for a specified port group as identified by a slot and port within the group.

```
Brocade#show tm statistics ethernet 2/1
--------- Ports 2/1 - 2/20 ---------
Ingress Counters:
   Total Ingress Pkt Count:                 464454
   EnQue Pkt Count:                         464454
```

```
   EnQue Byte Count:                        51907696
   DeQue Pkt Count:                         464454
   DeQue Byte Count:                        51907696
   TotalQue Discard Pkt Count:              0
   TotalQue Discard Byte Count:             0
   Oldest Discard Pkt Count:                0
   Oldest Discard Byte Count:               0
Egress Counters:
   EnQue Pkt Count:                         701866
   EnQue Byte Count:                        78791072
   Discard Pkt Count:                       0
   Discard Byte Count:                      0
```

Syntax: show tm statistics ethernet *slot/port*

The *slot/port* variable specifies the slot and port number of the port group that you want to display traffic manager statistics for.

NOTE
A traffic manager contains a specific number of ports depending on the Interface module as described in Table 35. Specifying a particular port and slot gathers statistics for all ports that belong to the same port group.

Displaying traffic manager statistics for an interface module

The following command displays all traffic manager statistics for an interface module identified by its slot number.

```
Brocade#show tm statistics slot 4
--------- Ports 4/1 - 4/20 ---------
Ingress Counters:
   Total Ingress Pkt Count:                 0
   EnQue Pkt Count:                         0
   EnQue Byte Count:                        0
   DeQue Pkt Count:                         0
   DeQue Byte Count:                        0
   TotalQue Discard Pkt Count:              0
   TotalQue Discard Byte Count:             0
   Oldest Discard Pkt Count:                0
   Oldest Discard Byte Count:               0
Egress Counters:
   EnQue Pkt Count:                         0
   EnQue Byte Count:                        0
   Discard Pkt Count:                       0
   Discard Byte Count:                      0
```

Syntax: show tm statistics ethernet *slot/port*

The **slot** *slot-number* option specifies an interface module that you want to display traffic manager statistics from.

TABLE 34 Traffic manager statistics

This field...	Displays...
Ingress Statistics	
Total Ingress Pkt Count	A count of all packets entering into this traffic manager. A traffic manager contains a specific number of ports depending on the Interface module as described in Table 35.
EnQue Pkt Count	A count of all packets entering Ingress queues on this traffic manager. A traffic manager contains a specific number of ports depending on the Interface module as described in Table 35.
EnQue Byte Count	A count of all bytes entering Ingress queues on this traffic manager. A traffic manager contains a specific number of ports depending on the Interface module as described in Table 35.
DeQue Pkt Count	A count of all packets dequeued from Ingress queues and forwarded on this traffic manager. A traffic manager contains a specific number of ports depending on the Interface module as described in Table 35.
DeQue Byte Count	A count of all bytes dequeued from Ingress queues and forwarded on this traffic manager.
TotalQue Discard Pkt Count	A count of all packets failing to enter Ingress queues on this traffic manager. This may be due to: • the queue reaching its maximum depth, WRED, or other reasons. • the network processor deciding to drop packets for reasons including: an unknown Layer-3 route, RPF, or segment filtering. A traffic manager contains a specific number of ports depending on the Interface module as described in Table 35.
TotalQue Discard Byte Count	A count of all bytes failing to enter Ingress queues on this traffic manager. This may be due to: • the queue reaching its maximum depth, WRED, or other reasons. • the network processor deciding to drop packets for reasons including: an unknown Layer-3 route, RPF, or segment filtering. A traffic manager contains a specific number of ports depending on the Interface module as described in Table 35.
Oldest Discard Pkt Count	A count of all packets entering Ingress queues on this traffic manager, but deleted afterwards due to buffer full. A traffic manager contains a specific number of ports depending on the Interface module as described in Table 35.
Oldest Discard Byte Count	A count of all bytes entering Ingress queues on this traffic manager, but deleted afterwards due to buffer full. A traffic manager contains a specific number of ports depending on the Interface module as described in Table 35.
Egress statistics	
EnQue Pkt Count	A count of all packets entering egress queues and forwarded out on this traffic manager. A traffic manager contains a specific number of ports depending on the Interface module as described in Table 35.
EnQue Byte Count	A count of all bytes entering egress queues and forwarded out on this traffic manager. A traffic manager contains a specific number of ports depending on the Interface module as described in Table 35.
Discard Pkt Count	A count of all packets failing to enter egress queues on this traffic manager. A traffic manager contains a specific number of ports depending on the Interface module as described in Table 35.
Discard Byte Count	A count of all bytes failing to enter egress queues on this traffic manager. A traffic manager contains a specific number of ports depending on the Interface module as described in Table 35.

NOTE
The byte counts displayed from the **show tm statistics** command incorporate proprietary internal headers of various lengths.

TABLE 35 Ethernet ports per traffic manager

Interface module	Ports per Traffic Manager (TM)	
	TM 1	TM 2
4 X 10 Gbps (Ethernet)	1 - 2	3 - 4
2 X 10 Gbps (Ethernet)	1 - 2	
20 X 1 Gbps (Ethernet)	1 - 20	

TABLE 36 Polling intervals for specified modules

Interface module	Interval (seconds)
4 X 10 Gbps (Ethernet)	180
2 X 10 Gbps (Ethernet)	180
20 X 1 Gbps (Ethernet)	180
8 X 10 Gbps (Ethernet)	180
2 X 100 Gbps (Ethernet)	1 80
4 X 40 Gbps (Ethernet)	1 80
24 x 10 Gbps (Ethernet)	600

Displaying traffic manager statistics for NI-MLX-10Gx8-M and NI-MLX-10Gx8-D modules

The following command displays traffic manager statistics for the NI-MLX-10Gx8-M module, and the NI-MLX-10Gx8-D module identified by its slot number.

```
Brocade#show tm statistics slot 4
--------- Ports 4/1 - 4/4 ---------
Ingress Counters:
   Total Ingress Pkt Count:                  61402830423
   EnQue Pkt Count:                          61402825288
   DeQue Pkt Count:                          61402692118
   TotalQue Discard Pkt Count:               5096
   Oldest Discard Pkt Count:                 0
Egress Counters:
   EnQue Pkt Count:                          95406035820
   Discard Pkt Count:                        0

--------- Ports 4/5 - 4/8 ---------
Ingress Counters:
   Total Ingress Pkt Count:                  64207166485
   EnQue Pkt Count:                          64207161341
   DeQue Pkt Count:                          64207029336
   TotalQue Discard Pkt Count:               5087
   Oldest Discard Pkt Count:                 0
Egress Counters:
   EnQue Pkt Count:                          35018656764
   Discard Pkt Count:                        0
```

Syntax: show tm statistics [slot *slot-number*]

The **slot** *slot-number* option specifies the slot number of the port group that you want to display traffic manager statistics for.

Displaying traffic manager statistics for the 4x10G module

The following command displays traffic manager statistics for the 4x10G module identified by its slot number.

```
Brocade#show tm statistics slot 1
--------- Ports 1/1 - 1/2 ---------
Ingress Counters:
   Total Ingress Pkt Count:                 37145922200
   EnQue Pkt Count:                         37145922200
   EnQue Byte Count:                        5943609079168
   DeQue Pkt Count:                         37145922200
   DeQue Byte Count:                        5943609079168
   TotalQue Discard Pkt Count:              0
   TotalQue Discard Byte Count:             0
   Oldest Discard Pkt Count:                0
   Oldest Discard Byte Count:               0
Egress Counters:
   EnQue Pkt Count:                         83890318963
   EnQue Byte Count:                        13422682341696
   Discard Pkt Count:                       218
   Discard Byte Count:                      34752

--------- Ports 1/3 - 1/4 ---------
Ingress Counters:
   Total Ingress Pkt Count:                 141547098478
   EnQue Pkt Count:                         141547098478
   EnQue Byte Count:                        22647526064544
   DeQue Pkt Count:                         141547098478
   DeQue Byte Count:                        22647526064544
   TotalQue Discard Pkt Count:              0
   TotalQue Discard Byte Count:             0
   Oldest Discard Pkt Count:                0
   Oldest Discard Byte Count:               0
Egress Counters:
   EnQue Pkt Count:                         216769846687
   EnQue Byte Count:                        11606527206560
   Discard Pkt Count:                       0
   Discard Byte Count:                      0
```

Syntax: show tm statistics [slot *slot-number*]

The **slot** *slot-number* option specifies the slot number of the port group that you want to display traffic manager statistics for.

Displaying traffic manager statistics for the 20x1G module

The following command displays traffic manager statistics for the 20x1G module.

```
Brocade#show tm statistics all-counters 0
Ingress Counters:
   LBP Pkt Count:                          0
   QDP EnQue Pkt Count:                    0
   QDP EnQue Byte Count:                   0
   QDP DeQue Pkt Count:                    0
   QDP DeQue Byte Count:                   0
   QDP Head Delete Pkt Count:              0
   QDP Head Delete Byte Count:             0
   QDP Tail Delete Pkt Count:              0
   QDP Tail Delete Byte Count:             0
   Flow Status Message Count:              0
   Transmit Data Cell Count:               0
   TDM_A Pkt Count:                        0
   TDM_B Pkt Count:                        0

Programmable Ingress Counters:
[Queue Select: 8000, Queue Mask 0x0007]
   QDP EnQue Pkt Count:                    0
   QDP EnQue Byte Count:                   0
   QDP DeQue Pkt Count:                    0
   QDP DeQue Byte Count:                   0
   QDP Head Delete Pkt Count:              0
   QDP Head Delete Byte Count:             0
   QDP Tail Delete Pkt Count:              0
   QDP Tail Delete Byte Count:             0
   Flow Status Message Count:              0

Egress Counters:
   EGQ EnQue Pkt Count:                    0
   EGQ EnQue Byte Count:                   0
   EGQ Discard Pkt Count:                  0
   EGQ Discard Byte Count:                 0
   EGQ Segment Error Count:                0
   EGQ Fragment Error Count:               0
   Port63 Error Pkt Count:                 0
   Pkt Header Error Pkt Count:             0
   Pkt Lost Due to Buffer Full Pkt Count:  0
   Reassem Err Discard Pkt Count:          0
   Reassem Err Discard Fragment(32B) Count:  0
   TDM_A Lost Pkt Count:                   0
   TDM_B Lost Pkt Count:                   0

Programmable Egress Counters:
[Port Id for Enque: 0 (Disable), Port Id for Discard: 0 (Disable)]
   EGQ EnQue Pkt Count:                    0
   EGQ EnQue Byte Count:                   0
   EGQ Discard Pkt Count:                  0
   EGQ Discard Byte Count:                 0
```

Syntax: show tm statistics all-counters *dev_id*

The *dev_id* variable specifies the device id that you want to display traffic manager statistics for.

Displaying traffic manager statistics for IPTV multicast queue

The following command displays traffic manager statistics for the IPTV Multicast queue on an Ethernet module.

```
Brocade# show tm-voq-stat src_port eth 3/21 fid 8004
Multicast Queue-Id:8207
-----------------------
Priority =  0/1
   EnQue Pkt Count                          8496
   EnQue Bytes Count                        11758464
   DeQue Pkt Count                          7743
   DeQue Bytes Count                        0
   Total Discard Pkt Count                  0
   Total Discard Bytes Count                0
   Oldest Discard Pkt Count                 0
   Oldest Discard Bytes Count               0
   Current Queue Depth                      0
   Maximum Queue Depth since Last read      0
```

Syntax: show tm-voq-stat src_port eth *slot/port* **fid** *fid-id*

The **eth** *slot/port* variable displays TM statistics for an individual port.

The **fid** *fid-id* variable displays TM statistics for a specified fid.

Table 37 describes the output parameters of the **show tm-voq-stat src_port eth** command.

TABLE 37 Output parameters of the **show tm-voq-stat src_port eth 3/21 fid 8004** command

Field	Description
Multicast Queue-Id	Shows the IPTV multicast queue identifier.
EnQue Pkt Count	Shows the count of all packets entering ingress queues on this traffic manager.
EnQue Bytes Count	Shows the count of all bytes entering ingress queues on this traffic manager.
DeQue Pkt Count	Shows the count of all packets dequeued from ingress queues and forwarded on this traffic manager.
DeQue Bytes Count	Shows the count of all bytes dequeued from ingress queues and forwarded on this traffic manager.
Total Discard Pkt Count	Shows the count of all packets failing to enter ingress queues on this traffic manager. This may be due to: • The queue reaching its maximum depth, WRED, or other reasons. • The network processor deciding to drop packets for reasons including: an unknown Layer 3 route, RPF, or segment filtering.
Total Discard Bytes Count	Shows the count of all bytes failing to enter ingress queues on this traffic manager. This may be due to: • The queue reaching its maximum depth, WRED, or other reasons. • The network processor deciding to drop packets for reasons including: an unknown Layer 3 route, RPF, or segment filtering.
Oldest Discard Pkt Count	Shows the count of all packets entering ingress queues on this traffic manager, but deleted afterwards due to buffer full.
Oldest Discard Bytes Count	Shows the count of all bytes entering ingress queues on this traffic manager, but deleted afterwards due to buffer full.

TABLE 37 Output parameters of the **show tm-voq-stat src_port eth 3/21 fid 8004** command (Continued)

Field	Description
Current Queue Depth	Shows the current queue depth.
Maximum Queue Depth since Last read	Shows the maximum queue depth since last access to read.

Clearing traffic manager statistics

You can clear traffic manager statistics selectively for a specified port group, selectively for an interface module, or for an entire Brocade device as shown in the following.

```
Brocade# clear tm statistics ethernet slot 4
```

Syntax: clear tm statistics [ethernet *slot/port* **| slot** *slot-number* **]**

Executing the **clear tm statistics** command without any options clears all traffic manager statistics on the device.

The **ethernet** *slot/port* option specifies a port group that you want to clear traffic manager statistics from.

The **slot** *slot-number* option specifies an interface module that you want to clear traffic manager statistics from.

New network processor counters displayed for packets to and from traffic manager

Output from the **show interface** command has been enhanced to provide the following traffic manager related information:

- Number of packets received at the network processor (NP)
- Number of packets sent from the NP to the traffic manager (TM)
- Number of Ingress packets dropped at the NP
- Number of packets transmitted from the NP
- Number of packets received by the NP from the TM

The following is an example of the new output from the show interface command with the changed output highlighted in **bold**.

```
Brocade(config)# show interface ethernet 3/3
GigabitEthernet3/3 is up, line protocol is up
Hardware is GigabitEthernet, address is 0004.80a0.4052 (bia 0004.80a0.4052)
Configured speed auto, actual 1Gbit, configured duplex fdx, actual fdx
Member of L2 VLAN ID 1, port is untagged, port state is Forwarding
STP configured to ON, Priority is level0, flow control enabled
mirror disabled, monitor disabled
Not member of any active trunks
Not member of any configured trunks
No port name
MTU 1544 bytes, encapsulation ethernet
300 second input rate: 754303848 bits/sec, 1473249 packets/sec, 89.57% utilization
300 second output rate: 754304283 bits/sec, 1473250 packets/sec, 89.57%
```

```
utilization
1015230949 packets input, 64974783168 bytes, 0 no buffer
Received 0 broadcasts, 0 multicasts, 1015230949 unicasts
0 input errors, 0 CRC, 0 frame, 0 ignored
0 runts, 0 giants
NP received 1039220106 packets, Sent to TM 1039220442 packets
NP Ingress dropped 0 packets
1015231660 packets output, 64974824768 bytes, 0 underruns
Transmitted 0 broadcasts, 0 multicasts, 1015231660 unicasts
0 output errors, 0 collisions
NP transmitted 1039221393 packets, Received from TM 1039221562 packets
```

QoS for NI-MLX-1Gx48-T modules

The NI-MLX-1Gx48-T module supports 48 1G port. In a fully loaded 32 slot chassis, there are only 8 queues supported on the TM port. The Brocade chassis supports 1008 ports with 8 queues per port. Beginning with this release, the Brocade configuration allows you configure more ports in the system by changing the TM port to use 4 queues instead of 8. The Brocade chassis supports 2016 ports using 4 queues per port.

Limitations on TM ports

The TM Port limitations are reached under the following situations.

1. When a new module type is configured.

2. When a new line card is inserted (no configured type).

3. When the user tries to configure the **max-tm-queue** parameter from 4 to 8.

The relationship between max TM queues and max TM ports are supported in the system as follows:

TABLE 38 Maximum TM queues and TM ports

Max TM queue per port	Max TM port
8	1008
4	2016

Configuring priority queues from 8 to 4

The **system-init max-tm-queues** command allows you to configure the maximum number of queues in TM to 4. To configure priority queues from 8 to 4, enter the following command.

```
Brocaden(config)# system-init max-tm-queues 4
```

Syntax: [no] system-init max-tm-queues *num*

The *num* value specifies changing the number of queues to 4.

NOTE
When configuring priority queues from 8 to 4, or vice versa, the system displays the following message: **Reload required. Please write memory and then reload or power cycle. Failure to reload could cause system instability or failure.**

The NP continues to map all inbound packets to 8 internal priorities. If the **system-init max-tm-queues** command is configured, the NP will right shift this priority number by one bit before sending the packet to TM. The TM will en-queue the packets based on the following table:

TABLE 39 Queue type

NP priority	TM queue
7,6	3
5,4	2
3,2	1
1,0	0

Aggregated TM VOQ statistics collection

Supported modules

Traffic Manager queue statistics are only reported on the following interface modules:

- BR-MLX-10Gx8-X, NI-MLX-10Gx8-M, and NI-MLX-10Gx8-D
- BR-MLX-100Gx2-X and BR-MLX-100Gx1-X
- NI-X-OC192x2, NI-X-OC48x8, NI-X-OC48x4, and NI-X-OC48x2
- NI-MLX-48-T-A
- BR-MLX-24x1GF-X-ML, BR-MLX-24x1GC-X-ML, BR-MLX-24x1GF-X, and BR-MLX-24x1GC-X
- BR-MLX-10Gx24-DM (Added in NetIron 5.4.00b)

NOTE
The following modules are not supported NI-X-OC192x2, NI-X-OC48x8, NI-X-OC48x4, and NI-X-OC48x2.

Configuring aggregated TM VOQ statistics collection

When Traffic Manager (TM) Virtual Output Queue (VOQ) statistics collection is enabled, the system will start collecting TM queue statistics. To configure priority queues, refer to “Configuring priority queues from 8 to 4” on page 485 of the *Brocade MLX Series and NetIron Family Configuration Guide*.

Enabling aggregated TM VOQ statistics collection

The **tm-voq-collection** command allows you to enable and disable aggregated TM VOQ statistics collection.

```
Brocade(config)# statistics
Brocade(config-statistics)# tm-voq-collection
```

Syntax: [no] tm-voq-collection

NOTE
If priority queues are configured with system init max-tm-queues 4, TM will queue packets based on Table 40.

TABLE 40 Queue Type

NP priority	TM queue
7,6	3
5,4	2
3,2	1
1,0	0

Enabling SNMP support for brcdTMDestUcastQStatTable

Aggregated TM VOQ statistics per destination physical port per priority can be retrieved via SNMP using brcdTMDestUcastQStatTable. To enable SNMP agent support for brcdTMDestUcastQStatTable, use the **snmp-server enable mib tm-dest-qstat** command. By default, SNMP support for this table is disabled.

```
Brocade(config)# snmp-server enable mib tm-dest-qstat
```

Syntax: [no] snmp-server enable mib tm-dest-qstat

NOTE
The **tm-voq-collection** command must be enabled along with the **snmp-server enable mib tm-dest-qstat** command to enable SNMP support for aggregated TM VOQ statistics.

```
Brocade(config)# statistics
Brocade(config-statistics)# tm-voq-collection
```

Syntax: [no] tm-voq-collection

Enabling SNMP support for brcdNPQosStatTable

NP QOS statistics can be retrieved via SNMP using brcdNPQosStatTable. To enable SNMP agent support for brcdNPQosStatTable, use the **snmp-server enable mib np-qos-stat** command. By default, SNMP support for this table is disabled.

```
Brocade(config)# snmp-server enable mib np-qos-stat
```

Syntax: [no] snmp-server enable mib np-qos-stat

NOTE
The **enable-qos-statistics** command must be enabled along with the **snmp-server enable mib np-qos-stat** command to enable SNMP support for retrieving NP QoS statistics.

NOTE
NP QOS statistics are supported for physical ports only.

```
Brocade(config)# enable-qos-statistics
```

Syntax: [no] enable-qos-statistics

Displaying TM statistics from one queue or all queues

Use the following command to display traffic manager statistics for ethernet.

```
Brocade# show tm-voq-stat src_port eth 2/1 dst_port ethernet
-----ethernet 2/2 - 1/4------------------
EnQue Pkt Count                          4168645330
   EnQue Bytes Count                        1010575722
   DeQue Pkt Count                          0
   DeQue Bytes Count                        0
   Total Discard Pkt Count                  2084322665
   Total Discard Bytes Count                505287857
   Oldest Discard Pkt Count                 0
   Oldest Discard Bytes Count               0
   WRED Dropped Pkt Count                   1594822490
   WRED Dropped Bytes Count                 126321962
   Current Queue Depth                      0
   Maximum Queue Depth since Last read      0
```

Use the following command to display traffic manager statistics for all priorities.

```
Brocade# show tm-voq-stat   src_port  p1/1   dst_port p1/2
--------- Ports 1/1 - 1/4  ---------
Priority  =  0
   EnQue Pkt Count                                    81581531
   EnQue Bytes Count                               2692190523
   DeQue Pkt Count                                 81581531
   DeQue Bytes Count                               2692190523
   Total Discard Pkt Count                         0
   Total Discard Bytes Count                       0
   Oldest Discard Pkt Count                        0
   Oldest Discard Bytes Count                   0
   WRED Dropped Pkt Count                       0
   WRED Dropped Bytes Count                        0
   Current Queue Depth                                0
   Maximum Queue Depth since Last read      2310
Priority =  1
   EnQue Pkt Count                                    0
   EnQue Bytes Count                                  0
   DeQue Pkt Count                                    62
   DeQue Bytes Count                                  1302
   Total Discard Pkt Count                         0
   Total Discard Bytes Count                       0
   Oldest Discard Pkt Count                        0
   Oldest Discard Bytes Count                      0
```

```
   WRED Dropped Pkt Count                          0
   WRED Dropped Bytes Count                         21
   Current Queue Depth                                0
   Maximum Queue Depth since Last read      0
Priority =  2
....
```

Syntax: show tm-voq-stat src_port *source-port* **dst_port ethernet** *destination-port priority*

Specification of a **source-port** and **destination-port** is required.

You can optionally specify a **priority** to limit the display to a single priority.

The output from the TM Q statistics is available only if the src card type is a module listed in the supported modules list

You can optionally specify a **priority** to limit the display to a single priority or use the **all** parameter to display all priorities.

TABLE 41 Traffic Manager statistics

This field...	Displays...
EnQue Pkt Count	A count of all packets entering ingress queues on this traffic manager.
EnQue Byte Count	A count of all bytes entering ingress queues on this traffic manager.
DeQue Pkt Count	A count of all packets dequeued from ingress queues and forwarded on this traffic manager.
DeQue Byte Count	A count of all bytes dequeued from ingress queues and forwarded on this traffic manager.
TotalQue Discard Pkt Count	A count of all packets failing to enter ingress queues on this traffic manager. This may be due to: • the queue reaching its maximum depth, WRED, or other reasons. • the network processor deciding to drop packets for reasons including: an unknown Layer-3 route, RPF, or segment filtering.
TotalQue Discard Byte Count	A count of all bytes failing to enter ingress queues on this traffic manager. This may be due to: • the queue reaching its maximum depth, WRED, or other reasons. • the network processor deciding to drop packets for reasons including: an unknown Layer-3 route, RPF, or segment filtering.
Oldest Discard Pkt Count	A count of all packets entering ingress queues on this traffic manager, but deleted afterwards due to buffer full.
Oldest Discard Byte Count	A count of all bytes entering ingress queues on this traffic manager, but deleted afterwards due to buffer full.
WRED Dropped Pkt Count	A count of all packets entering ingress queues on this traffic manager but dropped due to WRED.
WRED Dropped Bytes Count	A count of all bytes entering ingress queues on this traffic manager but dropped due to WRED.
Maximum Queue Depth since Last read	The maximum queue depth since last access to read.

Displaying TM statistics from the multicast queue

Use the following command to display traffic manager statistics from the Multicast queue for priority 1 on a module.

```
Brocade# show tm-voq-stat src_port eth 4/1 multicast 1
Priority = 0/1
   EnQue Pkt Count                          0
   EnQue Bytes Count                        0
   DeQue Pkt Count                          0
   DeQue Bytes Count                        0
   Total Discard Pkt Count                  0
   Total Discard Bytes Count                0
   Oldest Discard Pkt Count                 0
   Oldest Discard Bytes Count               0
   WRED Dropped Pkt Count                   0
   WRED Dropped Bytes Count                 0
   Current Queue Depth                      0
   Maximum Queue Depth since Last read      0
```

Syntax: show tm-voq-stat src_port *source-port* **multicast** *priority* | **all**

Specification of a **source-port** is required.

You can optionally specify a **priority** to limit the display to a single priority or use the **all** parameter to display all priorities.

Showing collected aggregated TM VOQ statistics

Aggregated counters are the aggregation of counters from all ingress TMs to a specified destination port. Use the **show tm-voq-stats dst_port** command to display aggregated counters for each port.

```
Brocade# show tm-voq-stat dst_port ethernet 8/1 4

   Port     Enqueued (pkts)          Dequeued (pkts)          Dropped (pkts)
             (bytes)                   (bytes)                  (bytes)
--------------------------------------------------------------------------------
    8/1   :
    P4    :    1277236787               1377236787               0
               198322097328             198322097328             0
```

Syntax: show tm-voq-stats dst_port [ethernet *slot/port* | **all] [** *priority* | **all]**

The *slot/port* variable specifies the slot and port number of the port group from which you want to display Traffic Manager statistics. The **all** option displays TM statistics for all ports.

The *P* variable specifies the priority for which you want to display Traffic Manager statistics. The **all** option displays TM statistics for all priorities.

The following combinations of the command can be used to display Traffic Manager statistics:

- Use the **show tm-voq-stats dst_port ethernet** *slot/port P* command to display priority P counters for the slot/port.
- Use the **show tm-voq-stats dst_port ethernet** *slot/port* **all** command to display aggregated and all priorities counters for the *slot/port*.
- Use the **show tm-voq-stats dst_port ethernet** *slot/port* command to display aggregated counters for the *slot/port*.

- Use the **show tm-voq-stats dst_port all** *P* command to display priority *P* counters for all the ports in the system.
- Use the **show tm-voq-stats dst_port all** command to display aggregated counters for all the ports in the system.
- Use the **show tm-voq-stats dst_port all all** command to display all priorities and aggregate counters for all the ports in the system.

NOTE
The statistics are shown only when aggregated TM VOQ statistics collection is enabled.

Table 42 describes the columns displayed when using the **show tm-voq-stats dst_port** command.

TABLE 42 CLI display of show tm-voq-stats dst_port command

Field	Description
Enqueued (pkts) (bytes)	An aggregated count of all enqueued packets (and bytes) per egress port per priority.
Dequeued (pkts) (bytes)	An aggregated count of all dequeued packets (and bytes) per egress port per priority.
Dropped (pkts) (bytes)	An aggregated count of all dropped packets (and bytes) per egress port per priority.

Clearing the TM VOQ statistics

Use the **clear tm-voq-stats dst_port** command to clear all the counters for a specified port, or all ports, when the statistics collection is enabled. This command also clears SNMP statistics reported in brcdTMDestUcastQStatTable.

```
Brocade(config)# clear tm-voq-stats dst_port all
```

Syntax: clear tm-voq-stats dst_port [ethernet *slot/port* **| all]**

The *slot/port* variable specifies the slot and port number of the port group from which you want to clear traffic manager statistics. The **all** option clears traffic manager statistics from all ports.

Displaying TM VOQ depth summary

The **show tm-voq-stat max-queue-depth slot** command provides summary of the maximum queue depth of any queue from TM and provides additional information for debugging purposes. Knowing the maximum queue depth also allows a way to set maximum queue size adjusted for the specific traffic patterns on a system.

The following example displays summary of the maximum queue depth from the TM:

```
Brocade#show tm-voq-stat max-queue-depth slot

    --------- Ports 3/1 - 3/24 ---------
    QType     Max Depth       Max Util        Destination Port
    0         1013804         96%                     3/1
    1         1013848         96%                     3/1
    2         1013666         96%                     3/4
    3         1013794         96%                     3/1
    4         1013564         96%                     3/1
    5         538             0%                      2/7
    6         532             0%                      2/7
```

```
7        0               0%                      NA
--------- Ports 3/25 - 3/48 ---------
QType    Max Depth       Max Util        Destination Port
0        0               0%                      NA
1        0               0%                      NA
2        0               0%                      NA
3        0               0%                      NA
4        0               0%                      NA
5        0               0%                      NA
6        0               0%                      NA
7        0               0%                      NA
```

Syntax: show tm-voq-stat max-queue-depth slot slot number

The **show tm-voq-stat max-queue-depth slot** command displays the following information:

TABLE 43 TM VOQ maximum queue depth summary.

Field	Description
QType	Specifies the queue priority.
Max Depth	Specifies the maximum queue depth of any queue with Qtype in bytes.
Destination Port	Specifies the destination port of queue that had highest max queue depth.
Max Util	Specifies the percentage of maximum queue utilization (max-queue-depth / max-queue-size).

Displaying TM buffer utilization

The **show tm buffer-pool-stats slot** command displays the maximum buffer utilization from the TM and provides additional information for debugging purposes.

The following example displays maximum buffer utilization:

```
Brocade#show tm buffer-pool-stats slot

    --------- Ports 3/1 - 3/4 ---------
    Maximum Buffer Size: 0 (0%)
    Maximum Occupied Buffer Descriptors: 0 (0%)
    --------- Ports 3/5 - 3/8 ---------
    Maximum Buffer Size: 0 (0%)
    Maximum Occupied Buffer Descriptors: 0 (0%)
```

Syntax: show tm buffer-pool-stats slot slot number

The **show tm buffer-pool-stats slot** command displays the following information:

TABLE 44 TM buffer pool statistics.

Field	Description
Maximum Buffer Size	Specifies the maximum buffer size in bytes for both gold and bronze traffic and also shows percentage of buffer used out of maximum packet buffer.
Maximum Occupied Buffer Descriptors	Specifies the maximum buffer pointers used for both gold and bronze traffic and also shows percentage of buffer pointers used out of total buffer points.

Clearing TM VOQ depth summary

Use the **clear tm-voq-stat max-queue-depth slot** command to clear the maximum queue depth summary of any queue.

```
Brocade#clear tm-voq-stat max-queue-depth slot
```

Syntax: clear tm-voq-stat max-queue-depth slot slot number

The slot number specifies the decimal value of the slot number of the group from which you want to clear the traffic manager queue depth summary.

Clearing TM buffer utilization

Use the **clear tm buffer-pool-stats slot** command to clear the maximum buffer utilization from the TM.

```
Brocade#clear tm buffer-pool-stats slot
```

Syntax: clear tm buffer-pool-stats slot slot number

The slot number specifies the decimal value of the slot number of the group from which you want to clear the traffic manager buffer utilization.

Displaying QoS packet and byte counters

You can enable the collection of statistics for Ingress and Egress packet priorities using the **enable-qos-statistics** command. Once the collection of statistics is enabled, the **show np statistics** command can be used to display a count of the packet priorities of Ingress and Egress packets as shown in the following.

```
Brocade# show np statistics
TD: Traffic Despritor. Each TD has size of 512 Bytes

MODULE # 0 PPCR # 0 :
Ingress Counters :
Received packets                                    = 5172
Discarded packets                                   = 0
Received TDs on traffic class 0                     = 0
Received TDs on traffic class 1                     = 0
Received TDs on traffic class 2                     = 0
Received TDs on traffic class 3                     = 0
Received TDs on traffic class 4                     = 0
Received TDs on traffic class 5                     = 0
Received TDs on traffic class 6                     = 0
Received TDs on traffic class 7                     = 10344

Egress Counters :
Transmitted unicast packets                         = 0
Transmitted multicast packets                       = 0
Transmitted broadcast packets                       = 0
Filtered packets due to VLAN spanning tree          = 0
Tail dropped packets                                = 0
Control packets                                     = 10344
Packets filtered due to egress forward restrictions   = 0
Packets dropped due to full multicast egress queue    = 91459

TD: Traffic Despritor. Each TD has size of 512 Bytes

MODULE # 1 PPCR # 0 :
Ingress Counters :
Received packets                                    = 47809289718
Discarded packets                                   = 0
Received TDs on traffic class 0                     = 47809289569
Received TDs on traffic class 1                     = 0
Received TDs on traffic class 2                     = 0
Received TDs on traffic class 3                     = 0
Received TDs on traffic class 4                     = 0
Received TDs on traffic class 5                     = 0
Received TDs on traffic class 6                     = 0
Received TDs on traffic class 7                     = 0

Egress Counters :
Transmitted unicast packets                         = 18561287821
Transmitted multicast packets                       = 0
Transmitted broadcast packets                       = 0
Filtered packets due to VLAN spanning tree          = 0
Tail dropped packets                                = 5910551222
Control packets                                     = 0
Packets filtered due to egress forward restrictions   = 0
Packets dropped due to full multicast egress queue    = 0
```

QoS commands affected by priority queues

- Priority-based Rate Shaping
- Weighted Random Early Discard (WRED)
- Weighted-based Scheduling and Mixed Strict Priority
- CPU Copy Queue
- Traffic Manager Statistics

Priority-based rate shaping

If the user specifies a priority of 4-7 when the max-tm-queues parameter is configured using 4 queues, the **qos shaper priority** command is accepted, but a warning message is displayed.

The following example displays the **qos shaper priority** command configured with a priority 4 shaper.

```
Brocade(config-if-eth-16/1)# qos shaper priority 4 1000000
Warn: current max TM queues is 4-configuration of priority 4-7 will not have any
effect.
```

NOTE
If a priority 5 shaper is already configured, and the max-tm-queues parameter changes from 8 to 4 queues and is reloaded, then the priority 5 shaper configuration line is still displayed. The priority shaper 5 will not take effect.

Weighted Random Early Discard (WRED)

When WRED is enabled for a queue type of any forwarding queue, it will receive a warning message that is similar to when the **priority-based rate shaping** command is configured. Refer to "Priority-based rate shaping" on page 157 for more information.

The following example displays enabling WRED for the forwarding queues with a queue type of 6.

```
Brocade(config)#qos queue-type 6 wred enable
Warn: current max TM queues is 4-configuration of queue-type 4-7 will not have any
effect.
```

NOTE
If the **system-init max-tm-queues 4** command is configured, the user is able to configure similar WRED parameters, such as Average weight, Max Instantaneous queue size, Drop Precedence, etc. for all priorities. The default values of all WRED parameters (refer to Table 39 on page 148) is only effective when queue-type 0-3 is used.

Weighted-based scheduling and mixed strict priority

When the **max-tm-queues** parameter is configured with 8 or 4 queues, the **qos scheduler weighted** command and **qos scheduler mixed** command will still take the same number of weight values, but the unnecessary priority values are ignored.

The following example displays when the **qos scheduler weighted** command is configured using 4 queues.

```
Brocade(config-ethe-1/1)#qos scheduler weighted 7 6 5 4 3 2 11
Current max TM queues is 4 - weights "7", "6", "5", "4" for priority 7-4 will not
have any effect.
```

The following example displays when the **qos scheduler mixed** command is configured using 4 queues.

```
Brocade(config-ethe-1/1)#qos scheduler mixed 4 3 2 1 1
Current max TM queues is 4 - weights "4", "3", "2" for queues 4-2 will not have
any effect.
```

The following table displays how traffic scheduling for Strict Priority-based Scheduling and Weighted-based Scheduling is configured differently between 8 and 4 queues:

TABLE 45 Strict v.s. weighted queues

	8 queues (current)	4 queues (new)
Strict	7,6,5	3,2
Weighted	4,3,2,1,0	1,0

Error messages for CPU copy queue and traffic manager statistics

The following error messages are displayed for CPU copy queue and traffic manager statistics when the incorrect queues are configured.

CPU copy queue

When **system-init max-tm-queues 4** command is configured, the **rl-cpu-copy** command displays a warning message when the user specifies a priority 4-7.

Example

```
Brocade(config)# rl-cpu-copy priority 4 1000000
Warn: current max TM queues is 4-configuration of priority 4-7 will not have any
effect.
```

Traffic manager statistics

When **system-init max-tm-queues 4** command is configured, the **show tm-voq-stat** command will only take a priority 0-3. An error message is displayed when an invalid priority range is enabled.

Example

```
Brocade#show tm-voq-stat src_port eth 9/1 dst_port ethernet 2/3 5
Error: priority range 0 to 3.
```

NOTE
The **show tm-voq-stat** command will print statistics for 4 queues, instead of 8. The output from the TM Q statistics is available only if the src card type is a 48x1GC module or 8x10G module.

Enhanced buffer management for NI-MLX-10Gx8 modules and NI-X-100Gx2 modules

The prioritized buffer-pool feature establishes two buffer-pools, gold buffer-pool for high priority traffic and bronze buffer-pool for low priority traffic. Each internal priority can be associated with either the gold buffer-pool or the bronze buffer-pool. High priority traffic is guaranteed buffers even in the presence of bursty low priority traffic. This is only applicable to 8x10G and 2x100G modules.

The Virtual Output Queue (VOQ) size configuration enables the user to increase the queue size.

Enhanced Packet Buffer Management

Two buffer-pools are established, gold buffer-pool for high priority traffic and bronze buffer-pool for low priority traffic. Each internal priority can be associated with either the gold buffer-pool or the bronze buffer-pool. This guarantees buffers for high priority traffic, even in the presence of bursty low priority traffic. By default the internal priority/queue-type 7 is associated with the gold buffer-pool and the internal priorities/queue-types 6-0 are associated with the bronze buffer-pool.

Enhanced Packet Buffering Considerations

- For the 8x10G family, the buffer-pools apply only to unicast forwarded traffic for the 8x10G family.
- For the 2x100G family, the buffer-pools are common for unicast/unknown-unicast/broadcast/multicast forwarded traffic.

The buffer-pools are calculated as follows: **"Gold buffer-pool minimum guarantee percentage = 100% – Bronze buffer-pool percentage"** and **"Bronze buffer-pool minimum guarantee percentage = 100% – Gold buffer-pool percentage".** If the minimum buffer guarantee percentage is less than "0", then the minimum buffer guarantee percentage is "0".

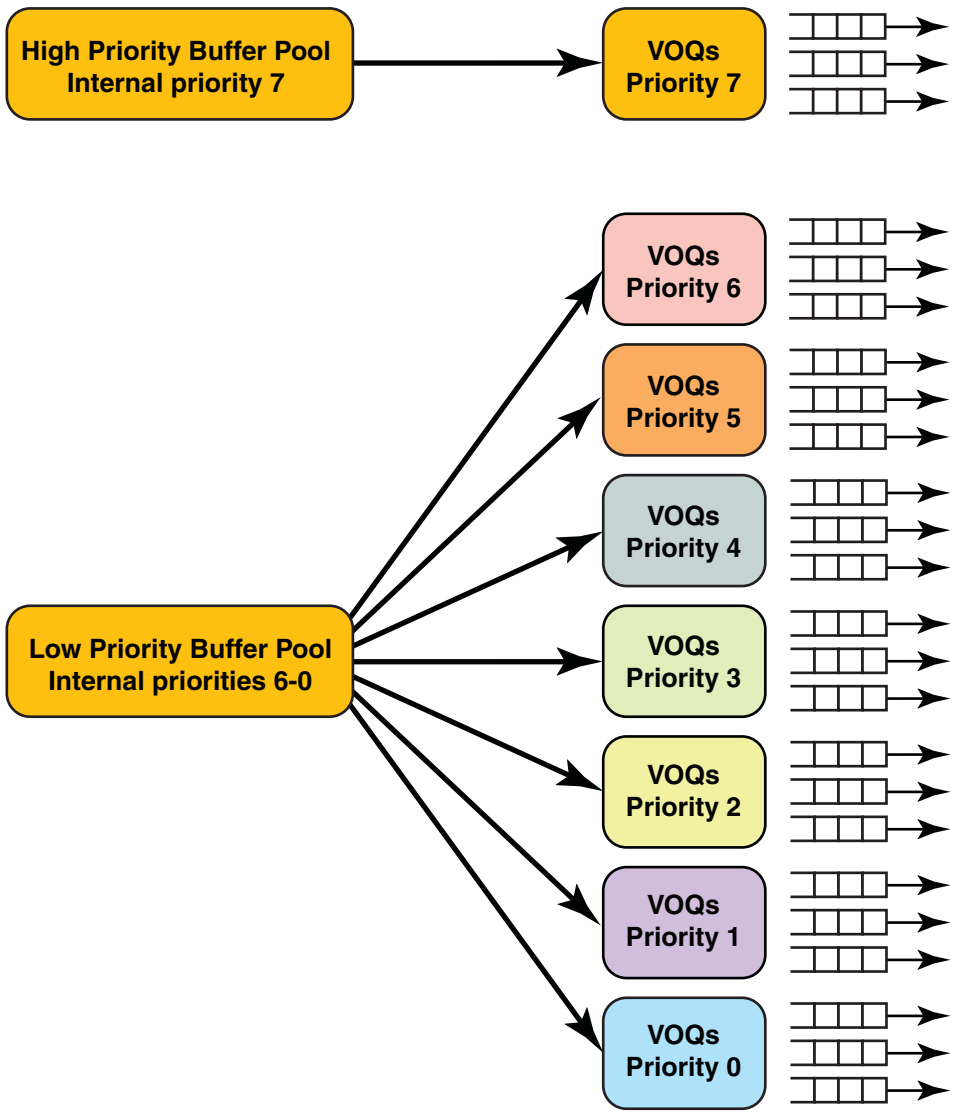
Default Configuration

For the 8x10 family of modules provides a minimum buffer guarantee for unicast control protocol traffic. The Gold buffer-pool has 100% of physical memory and Bronze buffer-pool has 95% of physical memory.

For 2x100G module - The goal is to provide a minimum buffer guarantee for unicast/multicast control protocol traffic. The buffer-pool is shared between Unicast/Broadcast/Unknown-unicast/Multicast. The Gold buffer-pool has 100% of physical memory and Bronze buffer-pool has 95% of physical memory.

The system default is depicted in Figure 1 and Figure 2.

FIGURE 7 System Default Priorities and Corresponding Buffer Pools



Configuration of buffer-pool priority to queue type

The following table displays the traffic types that are associate with each priority. Buffer-pool configuration enables the mapping of priorities to either the gold or bronze buffer-pool. VOQ Priority 7 cannot be removed from the Gold buffer-pool. VOQ Priority 0 cannot be removed from the Bronze buffer-pool.

TABLE 46 Strict v.s. weighted queues

Queue Type (Prioritization Category)	Protocol
P7	LACP, UDLD (802.3ah), STP/RSTP/BPDU, VSRP, MRP, BFD, GRE-KA/IS-IS over GRE, G.8032, LLDP, non-CCM 802.1ag (Eth + MPLS-enc.), BFD(Single-hop/Multi-hop/MPLS), IS-IS Hello, OSPFv2 Hello(GRE+Eth), OSPFv3 Hello(6to4+eth)
P6	IS-IS Non-Hello, OSPFv2/3 Non-Hello (Eth, GRE, 6to4), IPSec ESP Packets (for OSPFv3 over IP-sec), OSPF/OSPF over GRE or 6to4, IS-IS, RIP/RIPNG, VRRP (v4/v6), VRRPE(v4/v6)
P5	BGP/BGP over GRE or 6to4, LDP (basic and extended), RSVP, CCP(MCT), 802.1ag CCM (Eth + MPLS-enc.)
P4	PIM/PIM over GRE or 6to4, VPLS Encapsulated PIM, DVMRP, MSDP/MSDP over GRE/MSDP over VPLS
P3	IGMP, VPLS Encapsulated IGMP, GRE encapsulated IGMP, ARP, MLD, DHCP/BOOTP, ND6/ND6 in 6to4
P2	IPv4 Router Alert
P1	New Unassigned Protocols
P0	Existing Unassigned Protocols: GARP, ESIS, L2-Trace, REMOVE ESIS Prioritization

You can map each queue type to the buffer-pool type as needed. The queue-types are individually selected to be placed in the desired buffer-pool.

To map a queue-type to a buffer-pool enter the following command:

```
Brocade(config)#qos queue-type 5 buffer-pool bronze
```

Syntax: [no] qos queue-type *queue-number* **buffer-pool** *buffer-pool-type* | **gold** | **bronze**

The *queue-number* variable signifies the queue-type priority which will be associated with the buffer-pool. The *queue-number* can vary from 1-6.

The *buffer-pool-type* variable signifies the type of buffer-pool, gold or bronze.

Note: This command applies only to Unicast forwarded traffic. This does not apply to Broadcast/Unknown-unicast/Multicast forwarded traffic.

Configuration considerations

- Queue-type/internal priority 7 cannot be removed from Gold buffer-pool.
- Queue-type/internal priority 0 cannot be removed from Bronze buffer-pool.
- The command is applicable only to 8x10 family and 2x100G.
- This command is available only on switch reload.

Configuring buffer-pool size

```
Brocade(config)#qos buffer-pool bronze 50
```

Syntax: [no] **qos buffer-pool** *buffer-pool-type* | **gold** | **bronze** *max-percentage*

The *buffer-pool-type* variable signifies the type of buffer-pool, gold or bronze.

The *percentage* specifies the maximum percentage of memory allocated for the *buffer-pool-type.*

Configuration considerations

- The percentage of memory allocated for Bronze buffer-pool cannot exceed 95%.
- The percentage of memory allocated for Bronze buffer-pool cannot be below 5%.
- The command is applicable only to 8x10 family and 2x100G.
- Check the configuration as priorities are placed into buffer-pools individually.

Configuration examples

The example below depicts where internal priorities 7-6 are associated with the Gold buffer-pool and internal priorities 5-0 with the Bronze buffer-pool.

```
Brocade(config)# qos queue-type 6 buffer-pool gold
```

The example below depicts where the Bronze buffer-pool is configured with 70% of physical memory.

```
Brocade (config)# qos buffer-pool bronze 70
```

Displaying buffer-pool information

To display the buffer-pool configuration enter the command:

```
Brocade# show qos buffer-pool
```

Syntax: Show qos buffer-pool

```
system34#show qos buffer-pool
Unicast Queue-Type        Buffer Type

                0                BRONZE
                1                BRONZE
                2                BRONZE
                3                BRONZE
                4                BRONZE
                5                BRONZE
                6                BRONZE
                7                GOLD

Multicast Queue-Type        Buffer Type

                3                        GOLD
                2                        BRONZE
                1                        BRONZE
                0                        BRONZE

Buffer Type         Memory(%)     Min. Gurantee(%)
```

```
BRONZE                95                  0
GOLD                  100                  5

Module Type  Total Memory   Max. Gold   Min. Gold   Max. Bronze   Min. Bronze

8x10                     1392 MB       1392  MB     64 MB       1328MB            0
MB
2x100                   1472 MB         1472 MB      73 MB       1398 MB          0
MB
```

Configuring Virtual Output Queue (VOQ) queue size

Modules with 256MB VOQ size support

The following modules support a maximum VOQ size of 256MB. You also can disable the maximum VOQ size to allow the VOQ to grow to the size of the buffer-pool which the internal priority is associated with. This is recommended to be used only when there are two separate buffer-pools – one for high priority traffic and another for low priority traffic.

- NI-MLX-10Gx8-D
- NI-MLX-10Gx8-M
- NI-MLX-10Gx8-X
- NI-X-100Gx2

Configuration

To configure the max queue size for a queue-type (internal priority) enter the following command:

```
Brocade(config)#qos queue-type 1 max-queue-size 260000
```

Syntax: [no] qos queue-type *queue-number* **max-queue-size** *max-queue*

The *queue-number* variable signifies the queue-type priority for which “max-queue-size” is changed. The *queue-number* can vary from 0-7.

The *max-queue* variable signifies the maximum value of the queue size in KB for the queue-type or internal priority. The *max-queue* can vary from 1-262144 KBytes.

Modules with 64MB VOQ size support

The following modules support a maximum VOQ size of 64MB.

- NI-MLX-X-10Gx4
- NI-MLX-X-1Gx24-GC
- NI-MLX-X-1Gx24-SPF
- NI-MLX-1Gx48-T

Configuration

To configure the max queue size for a queue-type (internal priority) enter the following command:

```
Brocade(config)#qos queue-type 1 max-queue-size 64000
```

Syntax: [no] qos queue-type *queue-number* **max-queue-size** *max-queue*

The *queue-number* variable signifies the queue-type priority for which “max-queue-size” is changed. The *queue-number* can vary from 0-7.

The *max-queue* variable signifies the maximum value of the queue size in KB for the queue-type/internal priority. The *max-queue* can vary from 0-65536 KBytes. Setting the “max-queue-size” to “0” implicitly sets the “max-queue-size” to the maximum value.

Legacy Modules VOQ size support

Legacy modules include module older that those listed above. Legacy modules support a maximum VOQ queue size of 32MB.

Configuration

To configure the max queue size for a queue-type (internal priority) enter the following command:

```
Brocade(config)#qos queue-type 1 max-queue-size 15000
```

Syntax: [no] qos queue-type *queue-number* **max-queue-size** *max-queue*

The *queue-number* variable signifies the queue-type priority for which “max-queue-size” is changed. The *queue-number* can vary from 0-7.

The *max-queue* variable signifies the maximum value of the queue size in KB for the queue-type/internal priority. The *max-queue* can vary from 0-32768 KBytes. Setting the “max-queue-size” to “0” implicitly sets the “max-queue-size” to the maximum value.

Commands

- show tm buffer-pool-stats slot
- show tm-voq-stat max-queue-depth slot
- clear tm-voq-stat max-queue-depth slot
- clear tm buffer-pool-stats slot

show tm buffer-pool-stats slot

Displays the maximum buffer utilization from the TM.

Syntax **show tm buffer-pool-stats slot** slot number

Parameters

slot number	The slot number specifies the decimal value of the slot from traffic manager.

Command Modes Privileged EXEC mode

Usage Guidelines The **show tm buffer-pool-stats slot** command displays the maximum buffer utilization from the TM and provides additional information for debugging purposes.

Command Output The **show tm buffer-pool-stats slot** command displays the following information:

Output field	Description
Maximum Buffer Size	Specifies the maximum buffer size in bytes for both gold and bronze traffic and also shows the percentage of buffer used out of maximum packet buffer.
Maximum Occupied Buffer Descriptors	Specifies the maximum buffer pointers used for both gold and bronze traffic and also shows the percentage of buffer pointers used out of total buffer points.

Examples The following example displays maximum buffer utilization:

```
Brocade#show tm buffer-pool-stats slot

--------- Ports 3/1 - 3/4 ---------
Maximum Buffer Size:                    0 (0%)
Maximum Occupied Buffer Descriptors:    0 (0%)

--------- Ports 3/5 - 3/8 ---------
Maximum Buffer Size:                    0 (0%)
Maximum Occupied Buffer Descriptors:    0 (0%)
```

History

Release	Command History
Multi-Service IronWare R05.5c	This command was introduced.

Related Commands **clear tm buffer-pool-stats slot** slot number

show tm-voq-stat max-queue-depth slot

Displays the summary of maximum queue depth of any TM queue.

Syntax **show tm-voq-stat max-queue-depth slot** slot number

Parameters slot number The slot number specifies the decimal value of the slot of the traffic manager.

Command Modes Privileged EXEC mode

Usage Guidelines The **show tm-voq-stat max-queue-depth slot** command provides a summary of the maximum queue depth of any queue from the TM and provides additional information for debugging purposes.

NOTE
Knowing the maximum queue depth also allows a way to set max queue size adjusted for the specific traffic patterns on a system.

Command Output The **show tm-voq-stat max-queue-depth slot** command displays the following information:

Output field	Description
QType	Queue priority
Max Depth	Maximum queue depth of any queue with Qtype in bytes
Max Util	Percentage of maximum queue utilization (max-queue-depth / max-queue-size)
Destination Port	Destination port of queue that had highest max queue depth

Examples The following example displays summary of the maximum queue depth:

```
Brocade#show tm-voq-stat max-queue-depth slot

--------- Ports 3/1 - 3/24 ---------
QType    Max Depth       Max Util        Destination Port
0        1013804         96%                     3/1
1        1013848         96%                     3/1
2        1013666         96%                     3/4
3        1013794         96%                     3/1
4        1013564         96%                     3/1
5        538             0%                      2/7
6        532             0%                      2/7
7        0               0%                      NA
--------- Ports 3/25 - 3/48 ---------
QType    Max Depth       Max Util        Destination Port
0        0               0%                      NA
1        0               0%                      NA
2        0               0%                      NA
3        0               0%                      NA
4        0               0%                      NA
5        0               0%                      NA
6        0               0%                      NA
7        0               0%                      NA
```

History

Release	Command History
Multi-Service IronWare R05.5c	This command was introduced.

Related Commands **clear tm-voq-stat max-queue-depth slot** slot number

clear tm-voq-stat max-queue-depth slot

Clears the summary of maximum queue depth of any TM queue.

Syntax **clear tm-voq-stat max-queue-depth slot** slot number

Parameters

slot number	The slot number specifies the decimal value of the slot of the traffic manager.

Command Modes Privileged EXEC mode

Usage Guidelines The **clear tm-voq-stat max-queue-depth slot** command clears the maximum queue depth of any queue from the TM.

NOTES: When you clear the maximum queue depth statistics, you need to wait for the following amount of time to read and update the max-queue-depth value again:

- 180 seconds for the BR-MLX-10Gx8-X, NI-MLX-10Gx8-M, NI-MLX-10Gx8-D, BR-MLX-100Gx2-X, and 4X40 modules.
- Above 180 seconds for the 10Gx24-DM module.

Command Output The **clear tm-voq-stat max-queue-depth slot** command clears the following information:

Output field	Description
QType	The queue priority.
Max Depth	The maximum queue depth of any queue with Qtype in bytes.
Max Util	The percentage of maximum queue utilization (max-queue-depth / max-queue-size).
Destination Port	The destination port of queue that has highest max queue depth.

Examples The following example clears summary of the maximum queue depth:

```
Brocade#clear tm-voq-stat max-queue-depth slot
```

History

Release	Command History
Multi-Service IronWare R05.5c	This command was introduced.

Related Commands **show tm-voq-stat max-queue-depth slot** slot number

clear tm buffer-pool-stats slot

Clears the maximum buffer utilization from the TM.

Syntax **clear tm buffer-pool-stats slot** slot number

Parameters

slot number	The slot number specifies the decimal value of the slot from the traffic manager.

Command Modes Privileged EXEC mode

Usage Guidelines The **clear tm buffer-pool-stats slot** command clears the maximum buffer utilization from the TM.

NOTES: When you clear the maximum queue depth statistics, you need to wait for the following amount of time to read and update the max-queue-depth value again:

- 180 seconds for the BR-MLX-10Gx8-X, NI-MLX-10Gx8-M, NI-MLX-10Gx8-D, BR-MLX-100Gx2-X, and 4X40 modules.
- Above 180 seconds for the 10Gx24-DM module.

Command Output The **clear tm buffer-pool-stats slot** command clears the following information:

Output field	Description
Maximum Buffer Size	Specifies the maximum buffer size in bytes for both gold and bronze traffic and also shows percentage of buffer used out of maximum packet buffer.
Maximum Occupied Buffer Descriptors	Specifies the maximum buffer pointers used for both gold and bronze traffic and also shows percentage of buffer pointers used out of total buffer points.

Examples The following example clears maximum buffer utilization:

```
Brocade#clear tm buffer-pool-stats slot
```

History

Release	Command History
Multi-Service IronWare R05.5c	This command was introduced.

Related Commands **show tm buffer-pool-stats slot** slot number

Chapter

Hierarchical Quality of Service (HQoS) 5

Table 47 displays the individual Brocade devices that support HQoS.

TABLE 47 Supported Brocade Hierarchical QoS (HQoS) features

Features supported	Brocade NetIron XMR Series	Brocade MLX Series	Brocade NetIron CES 2000 Series BASE package	Brocade NetIron CES 2000 Series ME_PREM package	Brocade NetIron CES 2000 Series L3_PREM package	Brocade NetIron CER 2000 Series Base package	Brocade NetIron CER 2000 Series Advanced Services package
HQoS	Yes	Yes	No	No	No	No	No
PBB	Yes	Yes	No	No	No	No	No
Local VPLS	Yes	Yes	No	No	No	No	No
VPLS	No	No	No	No	No	No	No
IPv4	No	No	No	No	No	No	No
IPv6	No	No	No	No	No	No	No
Hierarchical Levels			No	No	No	No	No
10GX8-M Module	Yes	Yes	No	No	No	No	No
10GX8-X Module	Yes	Yes	No	No	No	No	No
10GX8-D Module	No	No	No	No	No	No	No
10Gx24 Module	No	No	No	No	No	No	No
HQoS Support for VPLS and LAG Interfaces	Yes	Yes	No	No	No	No	No
WRED Queue Management for HQoS	Yes	Yes	No	No	No	No	No

Hierarchical QoS (HQoS) for 8x10G modules

NOTE
HQoS is supported on the egress of 10G ports of the NI-MLX-10GX8-M and BR-MLX-10GX8-X modules.
HQoS is not supported on the NI-MLX-10GX8-D module.

Hierarchical QoS (HQoS) allows a carrier to consolidate different services on the same physical device running on the same physical infrastructure.

HQoS is a valuable tool, especially for networks that support multiple business customers who are running multiple applications with different prioritization and scheduling requirements over the same infrastructure.

HQoS uses an advanced scheduling mechanism, with multiple levels and multiple instances of scheduling and traffic shaping for the different services over a same connection. HQoS allows lower priority traffic to fully utilize the available bandwidth on a port, while ensuring high levels of QoS, e.g., low latency and guaranteed bandwidth, to higher priority traffic classes on that port. In summary, HQoS allows providers to offer improved customer SLAs and optimizes use of network resources.

How HQoS works

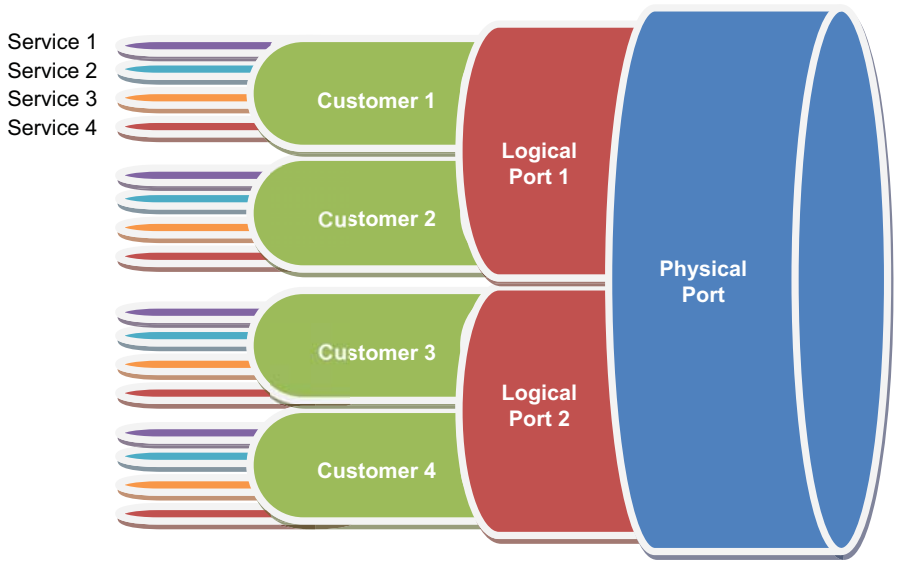
Hierarchical Quality of Service (HQoS) organizes a scheduler policy into a hierarchical tree that consists of a root node, branches nodes, and leaf nodes, where:

- The root node is the convergence point for all traffic and corresponds to a scheduler followed by a traffic shaper. The root node schedules and shapes the aggregated egress traffic of a physical port.
- A branch node is located in the middle of the hierarchy and corresponds to a scheduler followed by a traffic shaper.
- A leaf node corresponds to a scheduling queue.

HQoS scheduling levels do not support packet field matching capabilities. Packets are inspected once before being queued. Once packets go into a queue, everything beyond that point is a sequence of rate shapers and schedulers as defined by the hierarchical scheduling tree for that egress port.

HQoS supports a number of scheduling and shaping levels. Each level performs scheduling and shaping functions. By careful configuration, the different HQoS levels can map directly to a desired hierarchical traffic management model. For example, a hierarchy can be configured where an HQoS level represents the aggregated traffic of individual Customers, another level the individual VLANs of each customer, and another level the traffic following a Logical port downstream.

FIGURE 8 HQoS model



HQoS Components

The following HQoS components are supported in this release.

Supported levels of scheduling

At every scheduling/shaping level, the sum of the shaping rates going into a scheduler element does not need to add up to less than the shaping rate out of that scheduler element. The scheduler scheme used, Strict Priority (SP), Round Robin (RR), Weighted Round Robin (WRR), or mixed scheduling schemes, will determined how much traffic is scheduled for transmission from each scheduler flow. The rate shaper of a scheduler flow will limit the amount of traffic that can be scheduled from that scheduler flow. Therefore, the combination of scheduler flow rate shapers and scheduler element allows for the sharing of bandwidth among the respective scheduler flows in an ordered and pre-determined fashion.

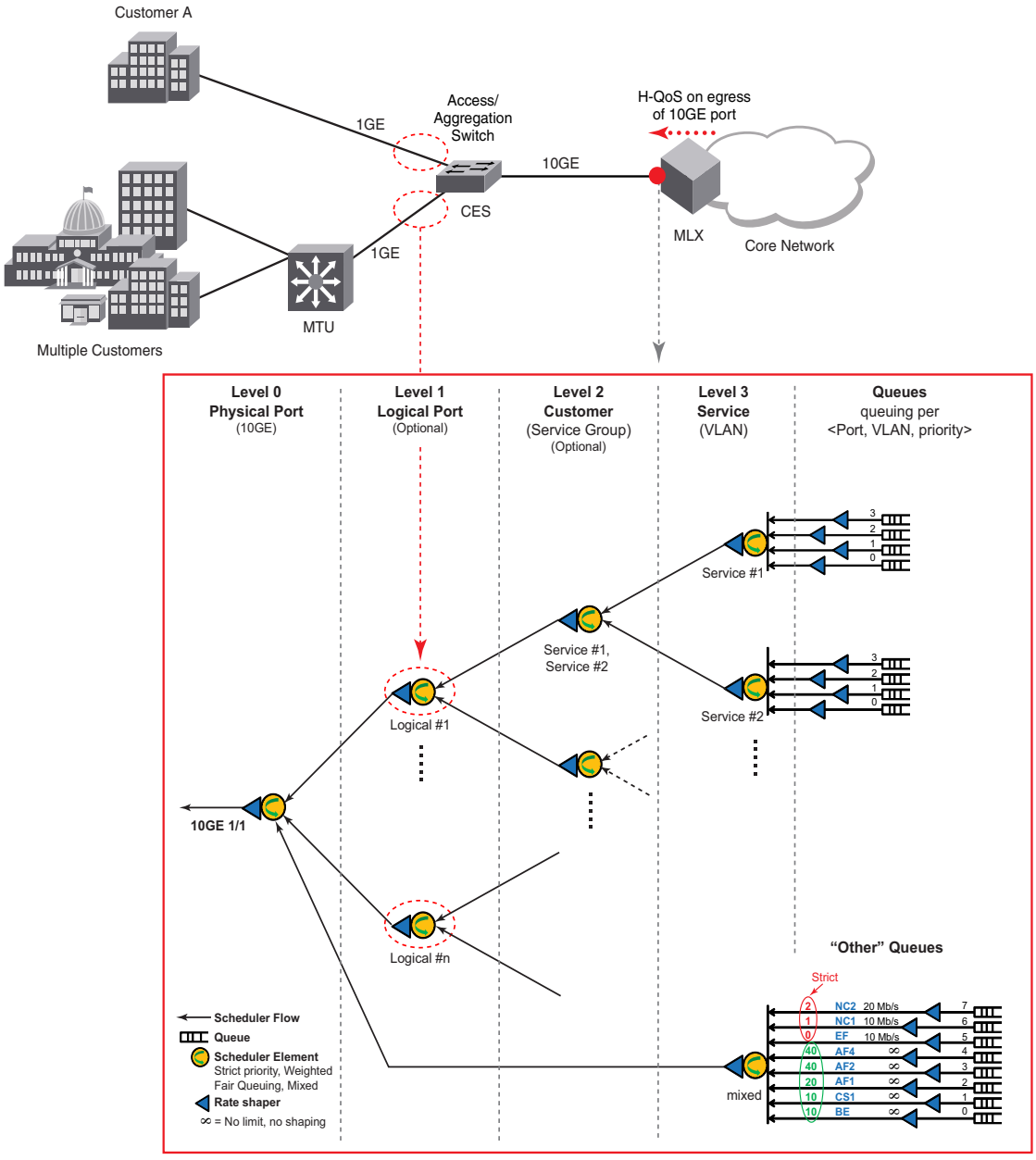
HQoS towards the customers

HQoS can shape the traffic towards the downstream 1GE links using the “Logical” port level of HQoS on Brocade devices.

In Figure 9 on page 174, two logical ports would be defined and shaped to 1Gb/s.

The HQoS policy is configured with Customer traffic connected to the appropriate “Logical” port on the HQoS hierarchy.

FIGURE 9 HQoS towards the Customers

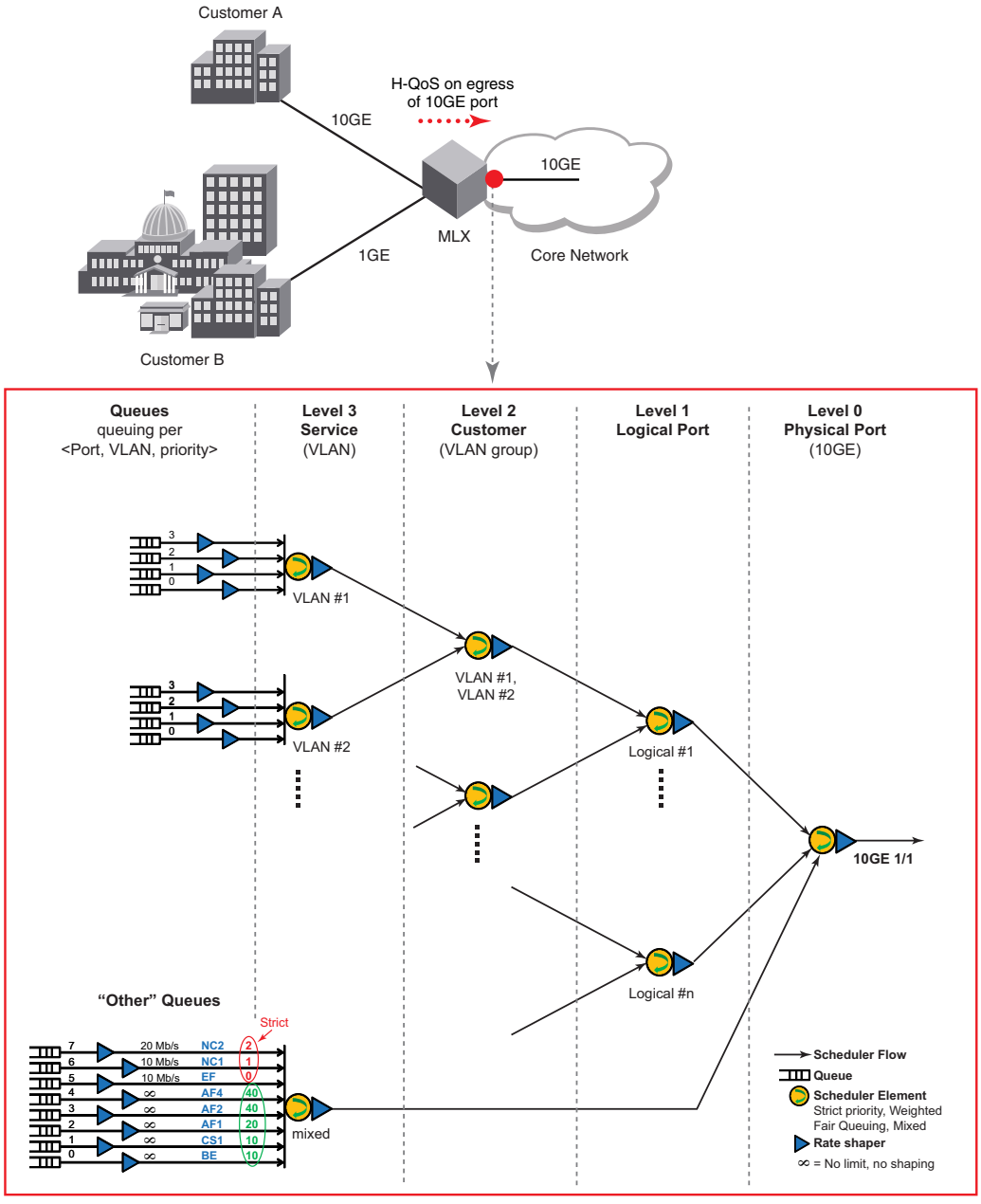


HQoS towards core network

HQoS usage towards the Service Provider core network on 10GE ports ensure high levels of QoS higher priority traffic classes for the customer.

The core network in this case can be a PB or PBB network.

FIGURE 10 HQoS towards Core Network



- **Level 3:** The *Service* level provides the scheduler/shaper for individual customer services, e.g., VLAN. The SLA applied here would apply to the individual service for that customer. For example, a customer may have two distinct point-to-point services identified by a SVLAN sharing the same physical link to the customer. The *Service* level schedules traffic from individual priority levels for a particular SVLAN. Priority levels may be served in Strict Priority (SP), Round Robin (RR), Fair Queuing (FQ), Weighted Round Robin (WRR), or mixed scheduling schemes.
- **Level 2:** The *Customer* level provides the scheduler/shaper for the aggregated services of a customer. For example, if a customer has two VLAN services at Level 3, this level would provide for scheduling/shaping of the combined traffic of the two customer VLANs. VLANs of a same customer are commonly served in Round Robin (RR), Fair Queuing (FQ), or Weighted Round Robin (WRR) mode.
- **Level 1:** The *Logical Port* level provides schedulers/shapers for the traffic that would flow through the egress port of a downstream device. This level allows for scheduling and shaping of traffic to fit a downstream port. Customer scheduler flows are commonly served in Round Robin (RR), Fair Queuing (FQ), or Weighted Round Robin (WRR) mode.
- **Level 0:** The *Physical Port* level provides a scheduler/shaper for the egress port. The Physical Port level schedules traffic from individual Logical Port scheduler flows. Logical Port scheduler flows are commonly served in Round Robin (RR), Fair Queuing (FQ), or Weighted Round Robin (WRR) mode.

Scheduling traffic for forwarding

If the traffic being processed by a Brocade device is within the capacity of the device, all traffic is forwarded as received. Once it reaches the point where the device is bandwidth constrained, it becomes subject to drop priority if configured.

The Brocade devices classify packets into one of eight internal priorities. Traffic scheduling allows you to selectively forward traffic according to one of the following schemes:

- **Strict priority-based scheduling** – This scheme guarantees that higher-priority traffic is always serviced before lower priority traffic. The disadvantage of strict priority-based scheduling is that lower-priority traffic can be starved.
- **WFQ weight-based traffic scheduling** – This scheme services different traffic priorities based on defined weights. Available bandwidth is divided among the different traffic priorities proportionally to the defined weights. For example, two priority levels with a same assigned weight will be served with about the same bandwidth. A priority level with a weight value twice the weight value of another priority level will be served at about twice the bandwidth of the other priority level.
- **Mixed strict priority and weight-based scheduling** – This scheme provides a mixture of strict priority for the three highest priority with levels and WFQ for the remaining priority levels.

For additional information on configuring QoS, refer to "Configuring Quality of Service for the Brocade NetIron XMR and Brocade MLX series" on page 71.

For examples of queue schedulers for HQoS, refer to "HQoS queue scheduler models" on page 205.

Supported deployment models

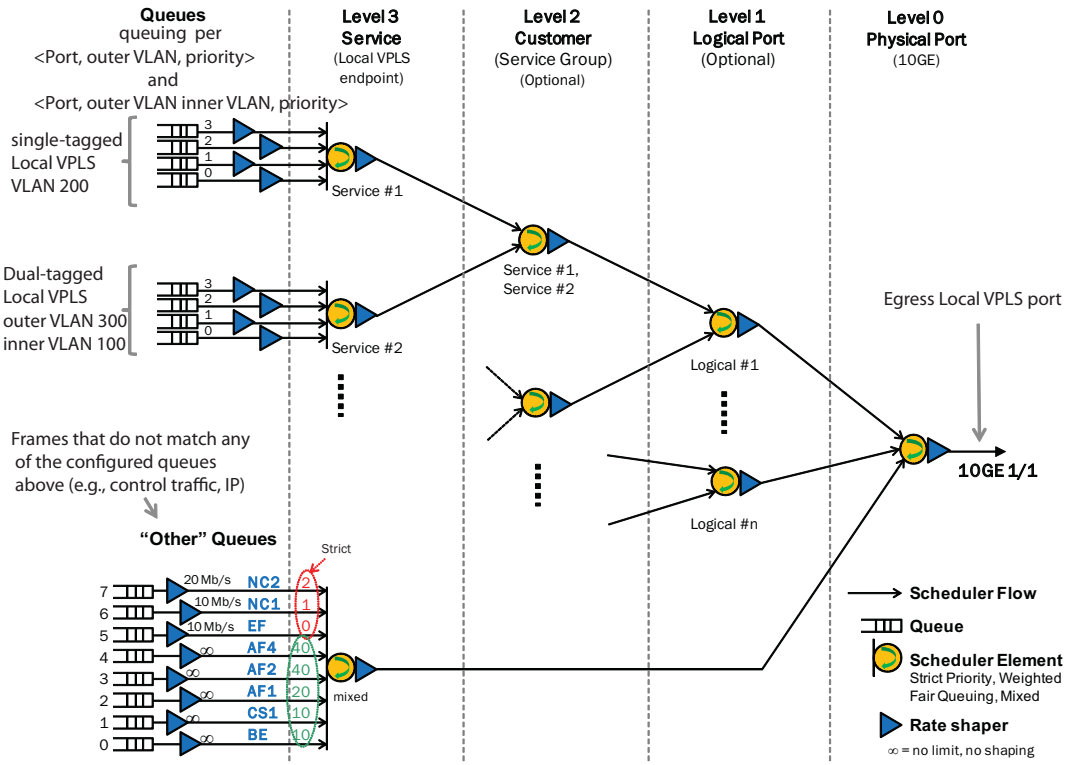
HQoS for Local VPLS

Figure 11 shows a Local VPLS HQoS model. This model can be used for any kind of VLAN, such as Customer 802.1Q VLANs (CVLAN), Provider Bridging 802.1ad VLANs (SVLAN), or Provider Backbone Bridging 802.1ah VLANs (BVLAN). The type of VLAN being used is defined by the port Ethertype configuration.

As Figure 11 shows HQoS for Local VPLS supports single-tagged and dual-tagged endpoint queuing on the same egress port.

Traffic that is not explicitly mapped to one of the VLAN queues is sent to the default "other" queues shown in the figure. The parameters of the other queue are configurable as described in "Other traffic" on page 180.

FIGURE 11 HQoS for Local VPLS with single-tagged and dual-tagged endpoints example

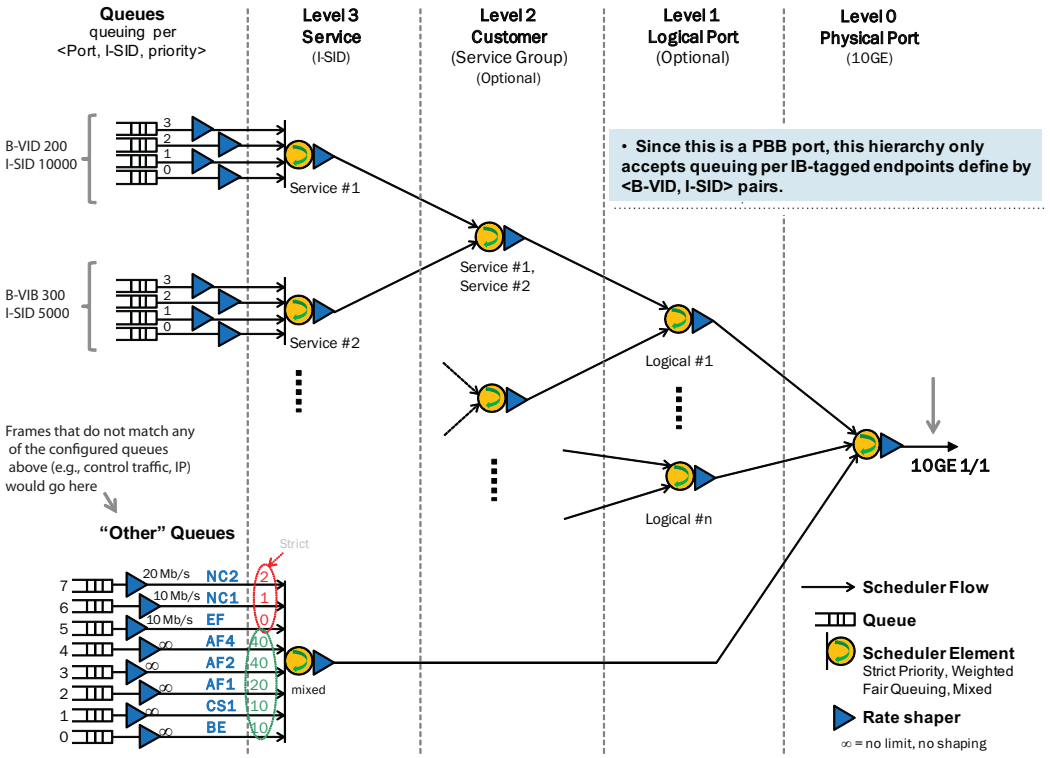


HQoS for PBB traffic

PBB ports can use either BVLAN-based queuing (BVLAN HQoS model, as shown in Figure 11 on page 177) or I-SID-based queuing as shown in Figure 12, where the egress port is an 802.1ah (PBB) port and where packets are queued per I-SID.

A BVLAN may carry a large number of services identified by distinct I-SID values. The BVLAN HQoS model (using HQoS for Local VPLS as shown in Figure 11) differs from the I-SID HQoS model in that BVLANs represent PBB tunnels that carry traffic from many different services, while the I-SID HQoS model represent individual services. The B-VLAN HQoS model would is for Backbone Core Bridges, while the I-SID HQoS model is for Backbone Edge Bridges.

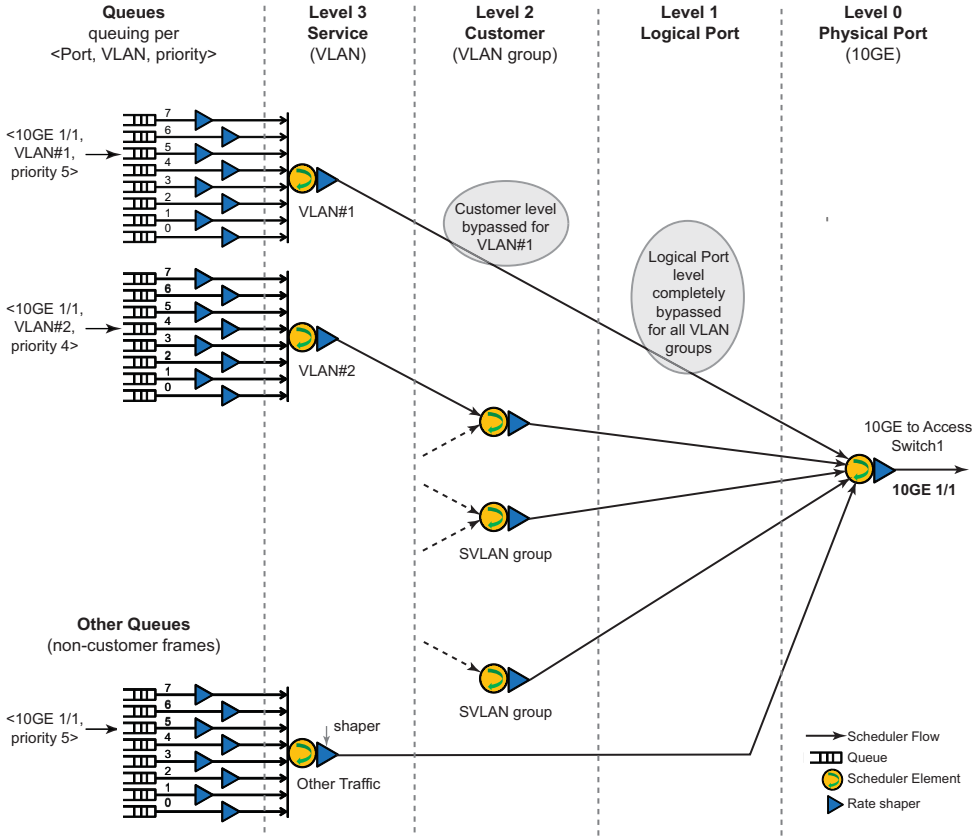
FIGURE 12 PBB HQoS example



Bypassing hierarchy levels

Figure 13 is an example where the Service Provider does not require the Logical Port level.

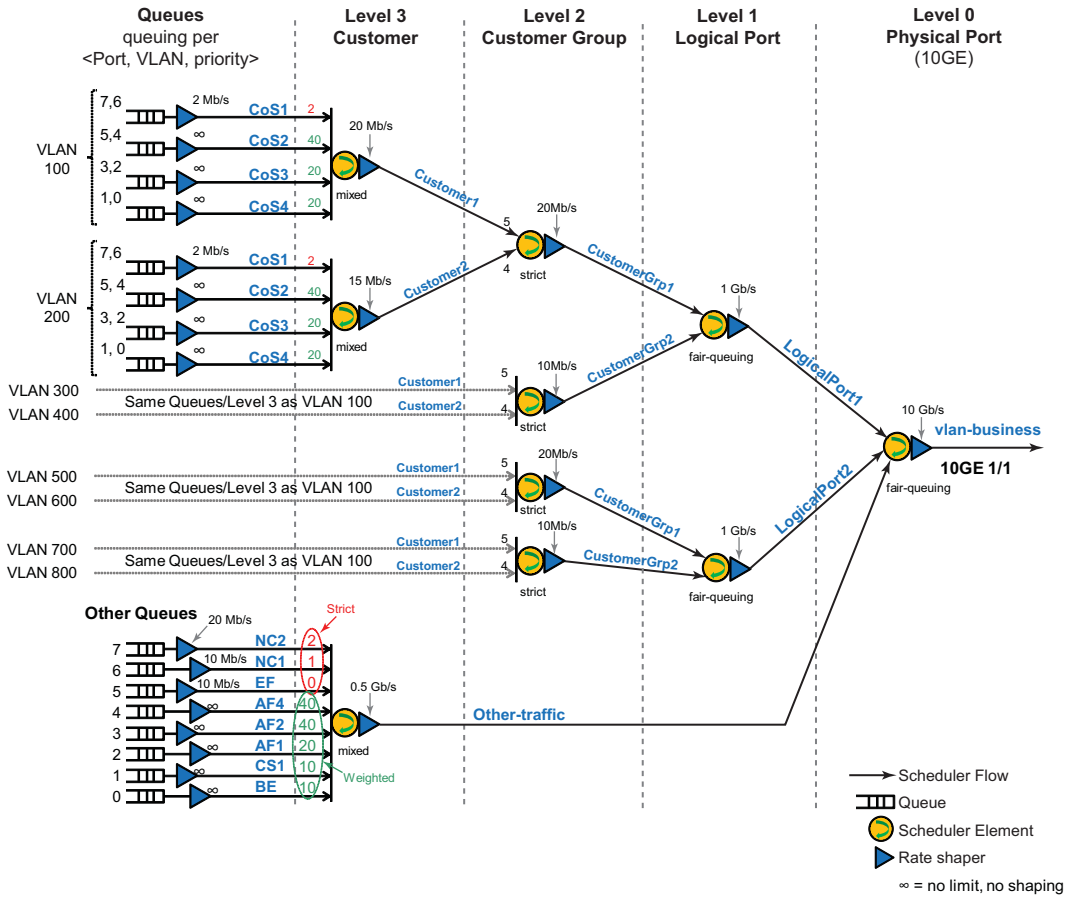
FIGURE 13 Bypassing hierarchy levels example



Other traffic

Figure 14 on page 180 displays a Local VPLS HQoS model concurrently supporting non-customer traffic, which is referred to as "other traffic". The customer traffic will always be queued and scheduled through the customer portion of the Layer 2 HQoS scheduler irrespective of what customer Layer 2 traffic is carrying. The "other queues" is used for non-customer traffic. That is, traffic that is not explicitly mapped to an HQoS queue. The "other queues" supports 8 queues/levels of priority. For this kind of traffic, you can queue packets per priority level, as shown. Note: Each priority level can be shaped independently and the other traffic as a whole can also be shaped to a desired rate.

FIGURE 14 Other traffic example

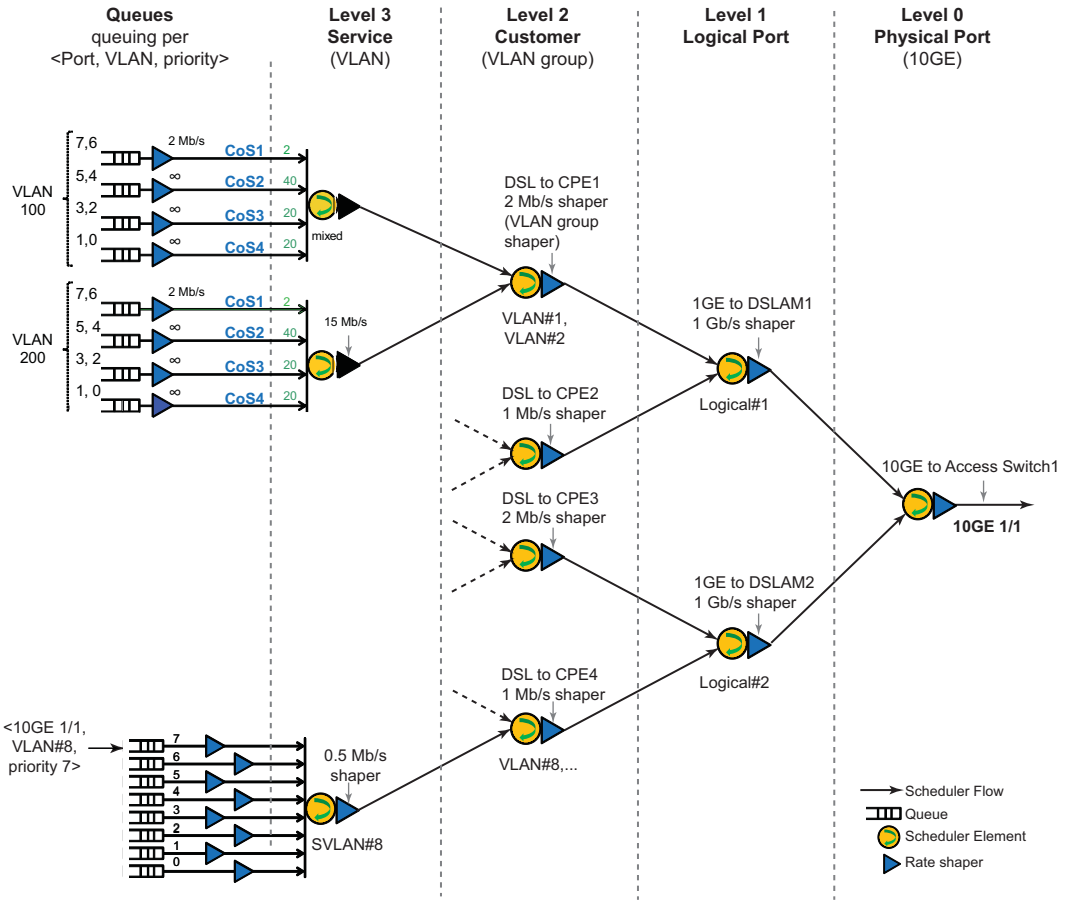


Configuring HQoS

The HQoS configuration procedure goes through the following phases:

- Create scheduling entities and configure forwarding profiles for them.
- Associate the scheduling entities with the forwarding profiles.
- Configure the match criteria for each node.
- Apply the organized scheduler policy to an interface.

FIGURE 15 HQoS example



The rate shaper of a scheduler flow will limit the amount of traffic that can be scheduled from that scheduler flow. Therefore, the combination of scheduler flow rate shapers and scheduler element allows for the sharing of bandwidth among the respective scheduler flows.

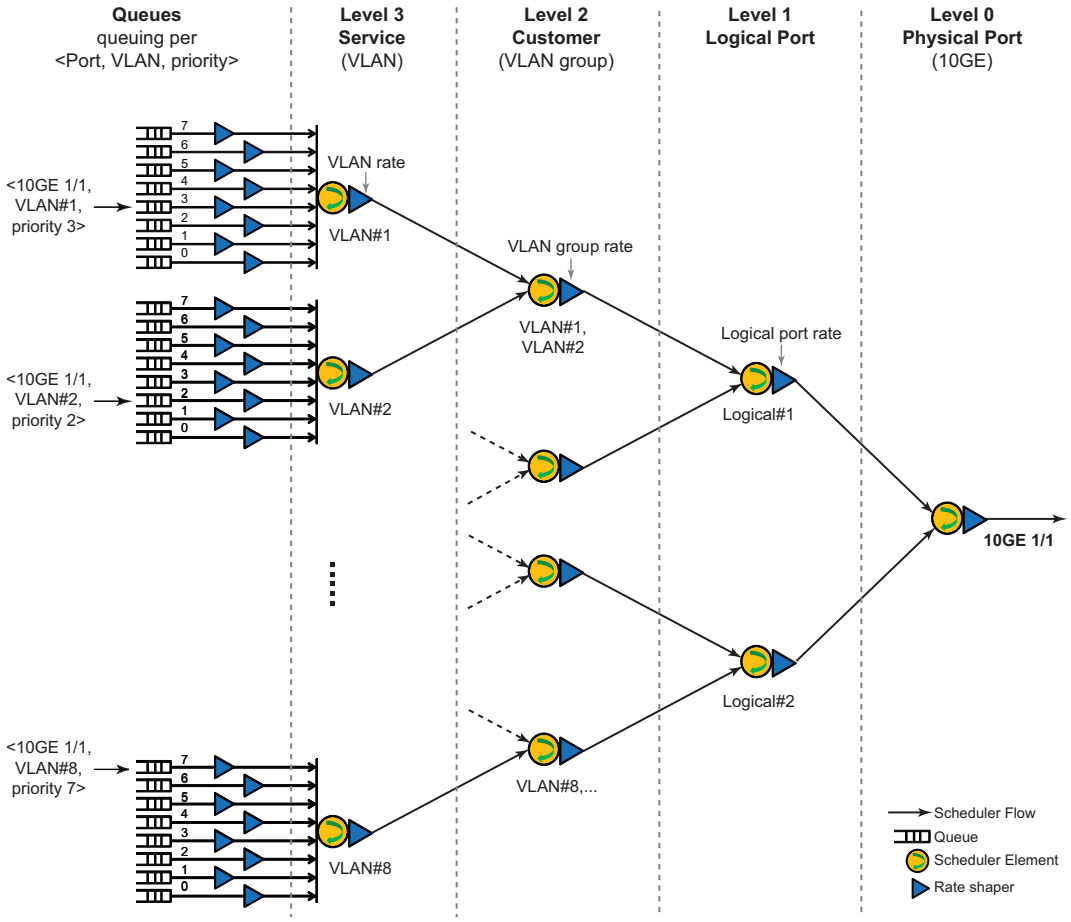
Configuration considerations

- HQoS is not supported for broadcast, multicast, and unknown unicast traffic
- HQoS is not supported on the interface of a LAG
- If HQoS is enabled on a interface, then the port QoS commands (port shaper configurations) cannot be executed
- Egress mirroring is degraded when HQoS is enabled

- Continuous bursts from 2K HQoS customer streams of jumbo packet sizes at line rate cannot be sustained by XPP's small egress FIFOs.

Configuring HQoS for Local VPLS

FIGURE 16 HQoS for Local VPLS model



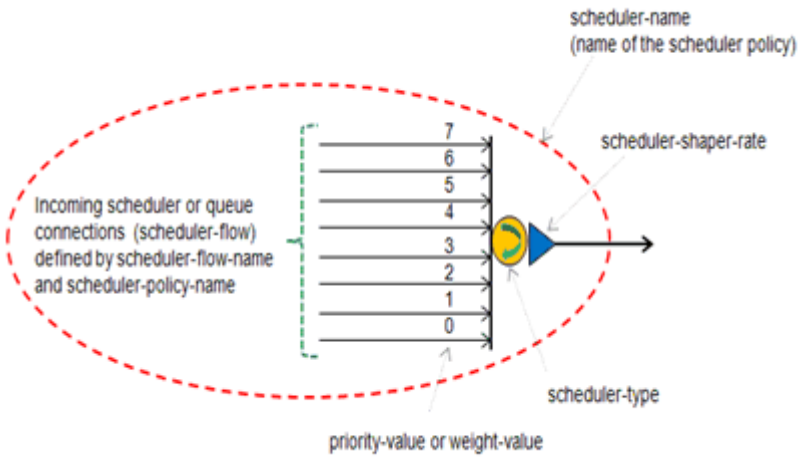
HQoS scheduler policy

HQoS is configured by first defining the HQoS scheduler policies and then mapping them to the physical ports. The same HQoS scheduler policy can be applied to multiple ports to enable provisioning.

To build a complete HQoS scheduler, you must define the types of scheduler elements used by the HQoS hierarchy at each hierarchy level.

Figure 17 shows the elements that must be configured for the HQoS policy.

FIGURE 17 HQoS scheduler policy



The following HQoS scheduler policies examples use Figure 16 on page 182.

Level 0 policy

The following is an example of how to configure a Level 0 policy

```
Brocade(config)# hqos scheduler-policy vlan-business level level-0
Brocade(config-hqos-scheduler-policy vlan-business) #shaper-rate 10000000
Brocade(config-hqos-scheduler-policy vlan-business) #shaper-burst-size 10
Brocade(config-hqos-scheduler-policy vlan-business) #scheduler-type weighted
Brocade(config-hqos-scheduler-policy vlan-business) #scheduler-flow LogicalPort1
scheduler-input 7 scheduler-policy logical-port-type1
Brocade(config-hqos-scheduler-policy vlan-business) #scheduler-flow LogicalPort2
scheduler-input 6 scheduler-policy logical-port-type1
Brocade(config-hqos-scheduler-policy vlan-business) #scheduler-flow Other-traffic
scheduler-input 5 scheduler-policy other-policy
```

Level 1 policy

The following is an example of how to configure a Level 1 policy.

```
Brocade(config)# hqos scheduler-policy logical-port-type1 level level-1
Brocade(config-hqos-scheduler-policy logical-port-type1)# shaper-rate 1000000
Brocade(config-hqos-scheduler-policy logical-port-type1)# shaper-burst-size 10
Brocade(config-hqos-scheduler-policy logical-port-type1)# scheduler-type weighted
Brocade(config-hqos-scheduler-policy logical-port-type1)# scheduler-flow
CustomerGrp1 scheduler-input 3 scheduler-policy customer-group-type1
Brocade(config-hqos-scheduler-policy logical-port-type1)# scheduler-flow
CustomerGrp2 scheduler-input 2 scheduler-policy customer-group-type1
```

Level 2 policy

The following is an example of how to configure a Level 2 policy.

```
Brocade(config)# hqos scheduler-policy customer-group-type1 level level-2
Brocade(config-hqos-scheduler-policy customer-group-type1)# shaper-rate  20000
Brocade(config-hqos-scheduler-policy customer-group-type1)# shaper-burst-size 10
Brocade(config-hqos-scheduler-policy customer-group-type1)# scheduler-type strict
```

```
Brocade(config-hqos-scheduler-policy customer-group-type1)# scheduler-flow
Customer1 scheduler-input 3 scheduler-policy customer-type1
Brocade(config-hqos-scheduler-policy customer-group-type1)#scheduler-flow
Customer2 scheduler-input 2 scheduler-policy customer-type1
```

Level 3 policy

The following is an example of how to configure a Level 2 policy.

```
Brocade(config)# hqos scheduler-policy customer-type1 level  level-3
Brocade(config-hqos-scheduler-policy customer-type1)# shaper-rate 20000
Brocade(config-hqos-scheduler-policy customer-type1)# shaper-burst-size 10
Brocade(config-hqos-scheduler-policy customer-type1)# scheduler-type mixed
Brocade(config-hqos-scheduler-policy customer-type1)# scheduler-flow CoS1
scheduler-input 3 scheduler-policy Q-7-6
Brocade(config-hqos-scheduler-policy customer-type1)# scheduler-flow CoS2
scheduler-input 2 weight 40 scheduler-policy Q-5-4
Brocade(config-hqos-scheduler-policy customer-type1)# scheduler-flow CoS3
scheduler-input 1 weight 20 scheduler-policy Q-3-2
Brocade(config-hqos-scheduler-policy customer-type1)# scheduler-flow CoS4
scheduler-input 0 weight 20 scheduler-policy Q-1-0
```

Syntax: [no] **hqos scheduler-policy** *scheduler-policy-name* **level** *level-number* | **level-0 | level-1**| **level-2 | level-3**

Syntax: [no] [**shaper-rate** *shaper-rate*]

Syntax: [no] [**shaper-burst-size** *shaper-burst-size*]

Syntax: [no] {**scheduler-type | scheduler-type-other}** *scheduler-type* | **strict | weighted | mixed**

Syntax: [no] {**scheduler-flow** *scheduler-flow-name* {**scheduler-input** *scheduler-input-value*} | [**weight** *weight-value*] **scheduler-policy** *scheduler-flow-policy-name*}

The *scheduler-policy-name* parameter is a string up to 128 characters.

The *scheduler-flow-name* parameter is a string up to 128 characters.

The *level-number* parameter is one of the following keywords level-0, level-1, level-1, or level-3

The *shaper-rate* is an optional parameter. The shaping rate is set with the minimum of 1Mbps and a maximum of 10Gbps. If no shaper-rate specified, the traffic will not be subject to shaping.

The *shaper-burst-size* is an optional parameter. The shaper burst size is set with the minimum of 2 Kbytes and a maximum of 256 Kbytes. The default value for the shaper burst size is set to 10 Kbytes.

{**scheduler-type**} is either **strict, mixed,** or **weighted**. This scheduler is used for 4 queue customer traffic schedulers. For fair-queuing, use weighted scheduler with all weights being equal.

{**scheduler-type-other**} is either **strict, mixed,** or **weighted**. This scheduler is only used for 8 queue “other traffic” schedulers. For fair-queuing, use weighted scheduler with all weights being equal.

The *scheduler-input-value* is a number representing the ordering of a flow with respect to a scheduler. The range is 0-7.

The *weight-value* is a number representing the weight of a scheduler flow when a weighted or mixed scheduler is used. The range is 1-64.

The *scheduler-flow-policy-name* is a string up to 128 characters. The scheduler policy used for a particular scheduler flow.

HQoS queue policy

HQoS hierarchy is achieved by creating a set of queues.

- A queue is rate shaped and the resulting traffic is referenced as a scheduler flow.
- The scheduler flow out of a queue is referenced by a scheduler policy.
- A queue stores packets based on a selected matching criteria.

FIGURE 18 H-QoS Queue Policy

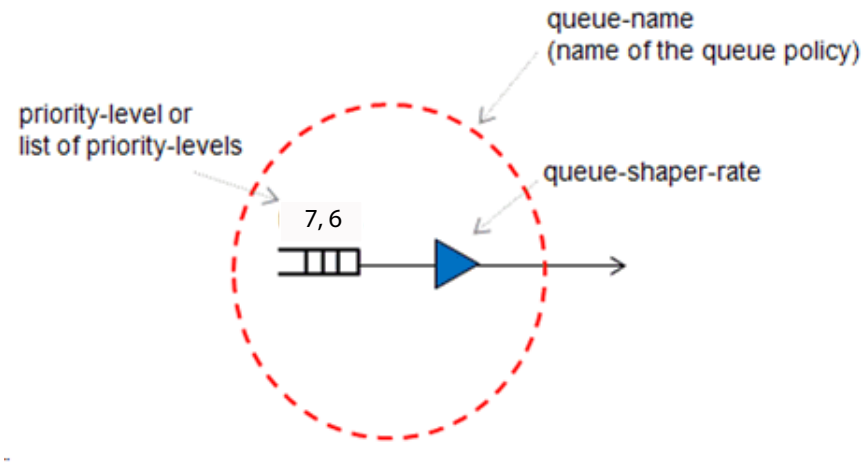


Figure 16 on page 182 defines queue policies named "Q-7-6", "Q-5-4", "Q-3-2", and "Q-1-0". A queue policy defines a queue for traffic of a given set of priority levels. Traffic out of a queue is shaped to a desired rate.

To configure the queue policies, enter commands such as the following.

```
Brocade(config)# hqos queue-policy Q-7-6
Brocade(config)# shaper-rate 2000
Brocade(config)# shaper-burst-size 10

Brocade(config)# hqos queue-policy Q-5-4
Brocade(config)# shaper-rate 10000000; default shaper rate (10G)
Brocade(config)# shaper-burst-size 10; default burst size (10KB)

Brocade(config)# hqos queue-policy Q-3-2
Brocade(config)# shaper-rate 10000000
Brocade(config)# shaper-burst-size 10

Brocade(config)# hqos queue-policy Q-1-0
Brocade(config)# shaper-rate 10000000
Brocade(config)# shaper-burst-size 10
```

Syntax: [no] hqos queue-policy *queue-name*

Syntax: [no] [shaper-rate *shaper-rate*]

Syntax: [no] [shaper-burst-size *shaper-burst-size*]

The *queue-name* is a string up to 128 characters.

The *shaper-rate* is an optional parameter. The shaping rate is set with the minimum of 1Mbps and a maximum of 10Gbps. If no shaper-rate specified, the traffic will not be subject to shaping.

The *shaper-burst-size* is an optional parameter. The shaper burst size is set with the minimum of 2 Kbytes and a maximum of 256 Kbytes. The default value for the shaper burst size is set to 10 Kbytes.

Binding a policy to an interface and applying a mapping condition

The **hqos service-policy** command is used to apply an HQoS scheduler policy to an egress physical interface or port. An 8 input Strict Priority (SP) scheduler will be created on the egress port if no other match condition is specified through the **hqos-map** command.

NOTE
This command will be valid only for interfaces on NI-MLX-10Gx8-M and NI-MLX-10Gx8-X modules. An error will be seen for interfaces on the NI-MLX-10Gx8-D module.

The **hqos-map** command is used to map customer endpoints and profiles to the HQoS scheduler flows. The mapping of customer endpoints and profiles is specified through the match criteria. The **hqos- map** command allows for changing the shaping rate of a scheduler. When configuring the interface, consider the following:

- Within the port configuration, **hqos-map** keyword must be present after **service-policy output** *level-0-scheduler-name*
- Only level 0 policies can be applied to an interface.
- If an HQoS policy is applied to an interface, the following changes to the scheduling tree structure is not supported.
  - Modifying HQoS policies (scheduler or queue)
  - Adding or removing a scheduler or queue policy that is referenced in the bound policy

The following is an example of how to configure the HQoS policy mapping for the VLAN HQoS model defined in Figure 16 on page 182.

Map the vlan-business policy to the Ethernet port 1/1 by using commands such as the following.

```
Brocade(config)# interface ethernet 1/1
Brocade(config-if-e10000-1/1)# hqos service-policy output vlan-business
```

Syntax: interface ethernet *port /slot*

Syntax: [no] hqos service-policy output *level-0-scheduler-name*

The **service-policy output** option defines the *level-0-scheduler-name* and can be up to 128 characters.

Make the necessary VLAN mappings and settings to the shaper rates by using commands such as the following.

```
Brocade(config-if-e10000-1/1)# hqos-map LogicalPort1.CustomerGrp1.Customer1 match
vlan 100
Brocade(config-if-e10000-1/1)# hqos-map LogicalPort1.CustomerGrp1.Customer2 match
vlan 200
Brocade(config-if-e10000-1/1)# hqos-map LogicalPort1.CustomerGrp2.Customer1 match
vlan 300
Brocade(config-if-e10000-1/1)#hqos-map LogicalPort1.CustomerGrp2.Customer2 match
vlan 400
```

```
Brocade(config-if-e10000-1/1)# hqos-map LogicalPort2.CustomerGrp1.Customer1 match
vlan 500
Brocade(config-if-e10000-1/1)# hqos-map LogicalPort2.CustomerGrp1.Customer2 match
vlan 600
Brocade(config-if-e10000-1/1)# hqos-map LogicalPort2.CustomerGrp2.Customer1 match
vlan 700
Brocade(config-if-e10000-1/1)# hqos-map LogicalPort2.CustomerGrp2.Customer2 match
vlan 800
Brocade(config-if-e10000-1/1)# hqos-map Other-traffic match other
Brocade(config-if-e10000-1/1)# hqos-map LogicalPort1.CustomerGrp1.Customer2
shaper-rate 15000
Brocade(config-if-e10000-1/1)# hqos-map LogicalPort1.CustomerGrp2 shaper-rate
10000
Brocade(config-if-e10000-1/1)# hqos-map LogicalPort2.CustomerGrp2 shaper-rate
10000
```

Syntax: [no] hqos-map *hqos-scheduler-node-name*

Syntax: [no] [shaper-rate *shaper-rate*]

Syntax: [no] [shaper-burst-size *shaper-burst-size*]

Syntax: [no] [match other][{**match vlan** *vlan-num*} | **[inner-vlan** *inner-vlan-num* | **isid** *isid-num*]]

The *scheduler-node-name* is a string up to 512 characters. The scheduler-node-name specifies the hqos-scheduler-node in full-path format.

The *shaper-rate* is an optional parameter. If there is no shaper-rate specified, the shaper-rate as specified in the HQoS policy will be used.

The *shaper-burst-size* is an optional parameter. If there is no shaper-burst-size specified, the shaper-burst-size as specified in the HQoS policy will be used.

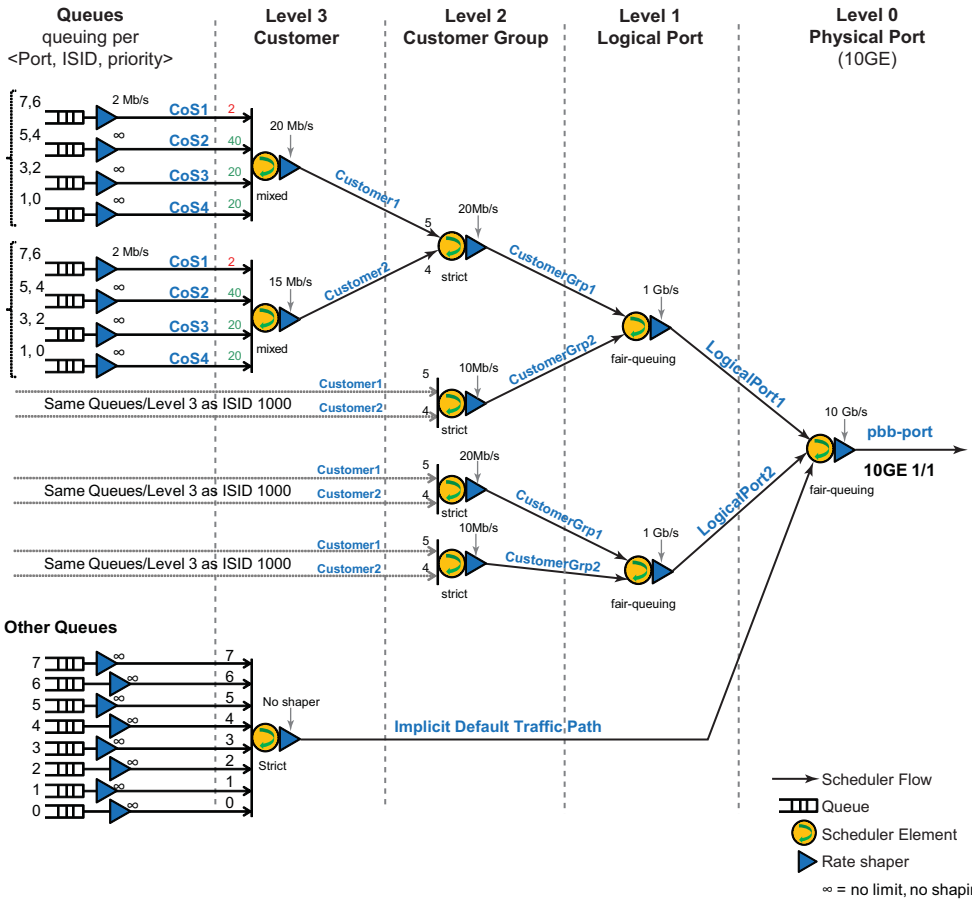
The *match other* is an optional parameter. For traffic that does not match any of the other defined matching rules. This is the default matching criteria for a scheduler flow without any defined matching criteria.

The *match vlan* ia an optional parameter. The **vlan-num** range is 1 through 4094. This option should be the egress VLAN on the HQoS port.

The *inner-vlan* is an optional parameter. The **vlan-num** range is 1 through 4094. This option should be the egress VLAN on the HQoS port.

PBB HQoS

FIGURE 19 PBB HQoS mode



HQoS configuration for PBB

The following configurations define the hierarchy for the PBB HQoS in Figure 19.

Refer to "HQoS scheduler policy" on page 183 for additional information on configuring the HQoS scheduler policies.

At the level-0 scheduler policy configuration, the main difference is the absence of an explicit "default traffic" path. A default path is created implicitly with strict priority scheduling among the 8 default traffic queues. The implicit default traffic path is always lower priority than customer traffic regardless of level-0 scheduler type.

```
Brocade(config)# hqos scheduler-policy pbb-port level level-0
Brocade(config-hqos-scheduler-policy pbb-port)# shaper-rate 10000000
Brocade(config-hqos-scheduler-policy pbb-port)# shaper-burst-size 10
Brocade(config-hqos-scheduler-policy pbb-port)# scheduler-type weighted
Brocade(config-hqos-scheduler-policy pbb-port)# scheduler-flow LogicalPort1
scheduler-input 7 scheduler-policy logical-port-type1
Brocade(config-hqos-scheduler-policy pbb-port)# scheduler-flow LogicalPort2
scheduler-input 6 scheduler-policy logical-port-type1
```

At the level-1 scheduler policy configuration, there are two customer groups competing in a weighted fair queue.

- CustomerGrp1 will get preferential treatment with twice the bandwidth of CustomerGrp2 because of the weight values associated with them.
- CustomerGrp1 can receive up to 666 Mbps and CustomerGrp2 receives up to 333 Mbps from the total 1Gbps shaped at this level.

```
Brocade(config)# hqos scheduler-policy logical-port-type1 level level-1
Brocade(config-hqos-scheduler-logical-port-type1)# shaper-rate 1000000
Brocade(config-hqos-scheduler-logical-port-type1)# shaper-burst-size 10
Brocade(config-hqos-scheduler-logical-port-type1)# scheduler-type weighted
Brocade(config-hqos-scheduler-logical-port-type1)# scheduler-flow CustomerGrp1
scheduler-input 3 weight 2 scheduler-policy customer-group-type1
Brocade(config-hqos-scheduler-logical-port-type1)# scheduler-flow CustomerGrp2
scheduler-input 2 weight 1 scheduler-policy customer-group-type1
```

At the level-2 scheduler policy configuration, the only thing that changes is the scheduling type from strict to fair weighted (when no explicit weight value is set, the default is 1).

```
Brocade(config)# hqos scheduler-policy customer-group-type1 level level-2
Brocade(config-hqos-scheduler-customer-group-type1)# shaper-rate  20000
Brocade(config-hqos-scheduler-customer-group-type1)# shaper-burst-size 10
Brocade(config-hqos-scheduler-customer-group-type1)# scheduler-type weighted
Brocade(config-hqos-scheduler-customer-group-type1)# scheduler-flow Customer1
scheduler-input 3 scheduler-policy customer-type1
Brocade(config-hqos-scheduler-customer-group-type1)# scheduler-flow Customer2
scheduler-input 2 scheduler-policy customer-type1
```

At the last level, each customer employs a strict scheduler amongst its 4 priority queues with no shapers (open shapers set to 10Gbps).

```
Brocade(config)# hqos scheduler-policy customer-type1 level  level-3
Brocade(config-hqos-scheduler-customer-type1)# shaper-rate 20000
Brocade(config-hqos-scheduler-customer-type1)# shaper-burst-size 10
Brocade(config-hqos-scheduler-customer-type1)# scheduler-type strict
Brocade(config-hqos-scheduler-customer-type1)# scheduler-flow CoS1
scheduler-input 3 scheduler-policy Q-7-6
Brocade(config-hqos-scheduler-customer-type1)# scheduler-flow CoS2
scheduler-input 2 scheduler-policy Q-5-4
Brocade(config-hqos-scheduler-customer-type1)# scheduler-flow CoS3
scheduler-input 1 scheduler-policy Q-3-2
Brocade(config-hqos-scheduler-customer-type1)# scheduler-flow CoS4
scheduler-input 0 scheduler-policy Q-1-0

Brocade(config)# hqos queue-policy queue-default
Brocade(config)# shaper-rate 10000000
Brocade(config)# shaper-burst-size 10
```

Binding Policy to interface and applying mapping condition for PBB

Refer to"Binding a policy to an interface and applying a mapping condition" on page 186 for additional information on binding a policy to an interface and applying a mapping condition.

Binding the mappings to HQoS PBB port is similar to the Local VPLS HQoS model. The only difference is you must add the "ISID" as the customer match condition.

```
Brocade(config)# interface ethernet 1/1
Brocade(config-if-e10000-1/1)# hqos service-policy output pbb-port
```

```
Brocade(config-if-e10000-1/1)# hqos-map Logical-Port1.CustomerGrp1.Customer1
match vlan 100 isid 1000
Brocade(config-if-e10000-1/1)# hqos-map Logical-Port1.CustomerGrp1.Customer2
match vlan 200 isid 2000
Brocade(config-if-e10000-1/1)# hqos-map Logical-Port1.CustomerGrp2.Customer1
match vlan 300 isid 3000
Brocade(config-if-e10000-1/1)# hqos-map Logical-Port1.CustomerGrp2.Customer2
match vlan 400 isid 4000
Brocade(config-if-e10000-1/1)# hqos-map Logical-Port2.CustomerGrp1.Customer1
match vlan 500 isid 5000
Brocade(config-if-e10000-1/1)# hqos-map Logical-Port2.CustomerGrp1.Customer2
match vlan 600 isid 6000
Brocade(config-if-e10000-1/1)# hqos-map Logical-Port2.CustomerGrp2.Customer1
match vlan 700 isid 7000
Brocade(config-if-e10000-1/1)# hqos-map Logical-Port2.CustomerGrp2.Customer2
match vlan 800 isid 8000
```

Displaying HQoS information

Use the following commands to show HQoS information.

Displaying the HQoS policy

Use the **show hqos policy** command to display the information of the policy and its associated flows.

```
Brocade# show hqos policy vlan-business
    Scheduler-policy-name vlan-businessLevel level-0
    Scheduler-type weighted  Shaper-rate 2000000 Kbps
    scheduler-flow Logical-Port1 scheduler-input 7 scheduler-policy
logical-port-type1
    scheduler-flow Logical-Port2 scheduler-input 6 scheduler-policy
logical-port-type1
   scheduler-flow Other-traffic  scheduler-input 5 scheduler-policy other-policy
```

Use the **show hqos policy** command to display the information of all the interfaces to which this policy is applied.

```
Brocade# show hqos policy vlan-business applied
            Ethernet Port:  1/1
            Scheduler-node Type:  Root
            Scheduler-node Policy Name:  vlan-business
            Scheduler-node Name:  vlan-business
            Scheduler-node ID:  0x00610000
```

Syntax: show hqos policy *policy-name* [**applied**]

Displaying HQoS interface information

Use the **show hqos interface ethernet** command to display information for the specified interface.

```
Brocade# show hqos interface eth 1/1
    Interface Number: 0x31
    HQOS State: Enabled
    Scheduler Tree State: Download Finish/Active
    Policy-name: vlan-business
```

```
    Scheduler-Node Type: Root
    Scheduler-Node Name: vlan-business
    Scheduler-Node ID: 0x310000
    Scheduler-Node Scheduler Type: Strict
    Scheduler-Node Shaper Rate: 2000000 Kbps
    Scheduler-Node Shaper Rate Burst Size: 128 KB
```

In this example, the HQoS policy tree has been applied to an interface.

```
Brocade# show hqos interface eth 1/1 scheduler-node vlan-business.Logical-Port1

    Scheduler-Node Type: Non-Root
    Scheduler-Node Policy Name:
logical-port-type1
    Scheduler-Node Name: Logical-Port1
    Scheduler-Node ID: 0x310008
    Scheduler-Node Scheduler Type: Strict
    Scheduler-Node Scheduler-input: 1
    Scheduler-Node Shaper Rate: 1000000 Kbps
    Scheduler-Node Shaper Rate Burst Size: 64 KB
    Scheduler-Node-Parent Node Name: vlan-business
    Scheduler-Node-Parent Node ID: 0x310000
```

In this example, the HQoS policy tree has been applied to an interface and shows the parent and child node information in a HQoS policy tree.

```
Brocade# show hqos interface eth 1/1 scheduler-node vlan-business.Logical-Port1
child

    Parent Information:
    Scheduler-Node Type: Non-Root
    Scheduler-Node Policy Name: logical-port-type1
    Scheduler-Node Name: Logical-Port1
    Scheduler-Node ID: 0x310008
    Scheduler-Node Scheduler Type: Strict
    Scheduler-Node Scheduler-input: 1
    Scheduler-Node Shaper Rate: 1000000 Kbps
    Scheduler-Node Shaper Rate Burst Size: 64 KB
    Scheduler-Node-Parent Node Name: vlan-business
    Scheduler-Node-Parent Node ID: 0x310000

    Child Information:
    Scheduler-Node Type: Non-Root
    Scheduler-Node Policy Name: customer-type1
    Scheduler-Node Name: Logical-Port1. Customer1
Scheduler-Node ID: 0x310015
    Scheduler-Node Scheduler Type: Strict
    Scheduler-Node Scheduler-input: 3
    Scheduler-Node Shaper Rate: 20000 Kbps
    Scheduler-Node Shaper Rate Burst Size: 128 KB
    Scheduler-Node-Parent Node Name: Logical-Port1
    Scheduler-Node-Parent Node ID: 0x310008

    Scheduler-Node Type: Non-Root
    Scheduler-Node Policy Name: customer-type1
    Scheduler-Node Name: Logical-Port1.Customer2
    Scheduler-Node ID: 0x310016
    Scheduler-Node Scheduler Type: Strict
    Scheduler-Node Scheduler-input: 3
    Scheduler-Node Shaper Rate: 20000 Kbps
```

```
    Scheduler-Node Shaper Rate Burst Size: 128 KB
    Scheduler-Node-Parent Node Name: Logical-Port1
    Scheduler-Node-Parent Node ID: 0x310008
```

Syntax: show hqos interface ethernet *slot/port* [**scheduler-node** *scheduler-node-name* | **scheduler-node-id**] [**child**]

Displaying the HQoS Max Queue Size

Use the **show hqos max-queue-size** command to display the priority for customer and default traffic queues.

```
Brocade# show hqos max-queue-size

Other Traffic
QType Max-Size (KB)
-----+-------------
  0       1024
  1       1024
  2       1024
  3       1024
  4       1024
  5       1024
  6       1024
  7       1024

Customer Traffic
QType Max-Size (KB)
-----+-------------
  0       1024
  1       1024
  2       1024
  3       1024
```

Syntax: show hqos max-queue-size

Displaying the buffer pool

Use the **show hqos buffer-pool command to** display the HQoS buffer pool configurations.

```
Brocade# show hqos buffer-pool

Other Traffic
QType Buffer Type
-----+-------------
  0   BRONZE
  1   BRONZE
  2   BRONZE
  3   BRONZE
  4   BRONZE
  5   BRONZE
  6   BRONZE
  7   GOLD

Customer Traffic
QType Buffer Type
-----+-------------
  0   BRONZE
  1   BRONZE
```

```
  2   BRONZE
  3   GOLD

Buffer Type    Memory(%)    Min. Gurantee(%)

BRONZE               95    0
GOLD                100    5

Module Type  Total Memory  Max. Gold  Min. Gold  Max. Bronze  Min. Bronze
8x10             1392 MB    1392 MB      69 MB      1322 MB         0 MB
```

Syntax: show hqos buffer-pool

Displaying HQoS global resource information

Use the **show hqos resource global** command to display the HQoS resources for specified slot.

```
Brocade# show hqos resource global
Global Resource:
              Maximum Allocated
Scheduler:     262144  122
    Queue:     131072  8

show the hqos resources for system

show hqos resource slot 1

Port 2/1 - 2/4:
               Maximum Allocated
Scheduler:     7936    122
    Queue:     8192    8

Port 2/5 - 2/8:
               Maximum Allocated
Scheduler:     7936    0
    Queue:     8192    0
```

Syntax: show hqos resource global

Displaying HQoS errors

Use the **show hqos error** command to display the HQoS error counts for a specified slot.

```
Brocade# show hqos error slot 2
Slot 2
-------
      Invalid Input Error Count: 0
    No HW Resources Error Count: 0
      Memory Alloc Failed Count: 0
           Internal Error Count: 0
            LP Busy Error Count: 0
          HW Driver Error Count: 0
            Unknown Error Count: 0
```

Syntax: show hqos error

Displaying HQoS Statistics

Use the **show the hqos statistics** command to display the specified flow information for a specified interface.

```
Brocade# show hqos statistics ethernet 1/1 queue Queue name: Queue name:
vlan-business.Logical-Port1.Customer1.COS1
    Priorities: 7, 6
    EnQue Pkt Count 0
    EnQue Bytes Count 0
    DeQue Pkt Count 0
    DeQue Bytes Count 0
    Total Discard Pkt Count 0
Total Discard Bytes Count         0
Oldest Discard Pkt Count               0
Oldest Discard Bytes Count         0
Current Queue Depth                 0
Maximum Queue Depth since Last read                 0
```

Use the **show the hqos statistics** command to display flow information for the “other” traffic flows.

```
Brocade# show hqos statistics ethernet 1/1 queue default-other

Node: implicit_match_all        Queue index: 0
Priorities: 0
                            EnQueue Packet Count                 0
                                EnQueue Byte Count               0
                            DeQueue Packet Count                 0
                                DeQueue Byte Count               0
                      Total Discard Packet Count                 0
                          Total Discard Byte Count               0
                    Oldest Discard Packet Count                  0
                        Oldest Discard Byte Count                0
                                Current Queue Depth              0
Maximum Queue Depth Since Last Read                              0
```

Use the **show the hqos statistics** command to display flow information for the specified index.

```
Brocade#(config-if-e10000-2/1)#show hqos statistics ethernet 2/1 queue
LogicalPort1.CustomerGrp1.Customer1
Node: LogicalPort1.CustomerGrp1.Customer1       Queue index: 0
Priorities: 1,0
               EnQueue Packet Count                    0
                 EnQueue Byte Count                    0
               DeQueue Packet Count                    0
                 DeQueue Byte Count                    0
         Total Discard Packet Count                    0
           Total Discard Byte Count                    0
        Oldest Discard Packet Count                    0
          Oldest Discard Byte Count                    0
                Current Queue Depth                    0
Maximum Queue Depth Since Last Read                    0

Node: LogicalPort1.CustomerGrp1.Customer1       Queue index: 1
Priorities: 3,2
               EnQueue Packet Count                    0
                 EnQueue Byte Count                    0
               DeQueue Packet Count                    0
                 DeQueue Byte Count                    0
         Total Discard Packet Count                    0
```

```
           Total Discard Byte Count                    0
        Oldest Discard Packet Count                    0
          Oldest Discard Byte Count                    0
                Current Queue Depth                    0
Maximum Queue Depth Since Last Read                    0

Node: LogicalPort1.CustomerGrp1.Customer1       Queue index: 2
Priorities: 5,4
               EnQueue Packet Count                    0
                 EnQueue Byte Count                    0
               DeQueue Packet Count                    0
                 DeQueue Byte Count                    0
         Total Discard Packet Count                    0
           Total Discard Byte Count                    0
        Oldest Discard Packet Count                    0
          Oldest Discard Byte Count                    0
                Current Queue Depth                    0
Maximum Queue Depth Since Last Read                    0

Node: LogicalPort1.CustomerGrp1.Customer1       Queue index: 3
Priorities: 7,6
               EnQueue Packet Count                    0
                 EnQueue Byte Count                    0
               DeQueue Packet Count                    0
                 DeQueue Byte Count                    0
         Total Discard Packet Count                    0
           Total Discard Byte Count                    0
        Oldest Discard Packet Count                    0
          Oldest Discard Byte Count                    0
                Current Queue Depth                    0
Maximum Queue Depth Since Last Read                    0
```

Use the **show the hqos statistics** command to display flow information for the default other flows.

```
Brocade#(config-if-e10000-2/1)#show hqos statistics ethernet 2/1 queue
default-other
Node: implicit_match_all        Queue index: 0
Priorities: 0
               EnQueue Packet Count                    0
                 EnQueue Byte Count                    0
               DeQueue Packet Count                    0
                 DeQueue Byte Count                    0
         Total Discard Packet Count                    0
           Total Discard Byte Count                    0
        Oldest Discard Packet Count                    0
          Oldest Discard Byte Count                    0
                Current Queue Depth                    0
Maximum Queue Depth Since Last Read                    0

Node: implicit_match_all        Queue index: 1
Priorities: 1
               EnQueue Packet Count                    0
                 EnQueue Byte Count                    0
               DeQueue Packet Count                    0
                 DeQueue Byte Count                    0
         Total Discard Packet Count                    0
           Total Discard Byte Count                    0
        Oldest Discard Packet Count                    0
          Oldest Discard Byte Count                    0
                Current Queue Depth                    0
Maximum Queue Depth Since Last Read                    0
```

```
Node: implicit_match_all        Queue index: 2
Priorities: 2
               EnQueue Packet Count                    0
                 EnQueue Byte Count                    0
               DeQueue Packet Count                    0
                 DeQueue Byte Count                    0
         Total Discard Packet Count                    0
           Total Discard Byte Count                    0
        Oldest Discard Packet Count                    0
          Oldest Discard Byte Count                    0
                Current Queue Depth                    0
Maximum Queue Depth Since Last Read                    0

Node: implicit_match_all        Queue index: 3
Priorities: 3
               EnQueue Packet Count                    0
                 EnQueue Byte Count                    0
               DeQueue Packet Count                    0
                 DeQueue Byte Count                    0
         Total Discard Packet Count                    0
           Total Discard Byte Count                    0
        Oldest Discard Packet Count                    0
          Oldest Discard Byte Count                    0
                Current Queue Depth                    0
Maximum Queue Depth Since Last Read                    0

Node: implicit_match_all        Queue index: 4
Priorities: 4
               EnQueue Packet Count                    0
                 EnQueue Byte Count                    0
               DeQueue Packet Count                    0
                 DeQueue Byte Count                    0
         Total Discard Packet Count                    0
           Total Discard Byte Count                    0
        Oldest Discard Packet Count                    0
          Oldest Discard Byte Count                    0
                Current Queue Depth                    0
Maximum Queue Depth Since Last Read                    0

Node: implicit_match_all        Queue index: 5
Priorities: 5
               EnQueue Packet Count                    0
                 EnQueue Byte Count                    0
               DeQueue Packet Count                    0
                 DeQueue Byte Count                    0
         Total Discard Packet Count                    0
           Total Discard Byte Count                    0
        Oldest Discard Packet Count                    0
          Oldest Discard Byte Count                    0
                Current Queue Depth                    0
Maximum Queue Depth Since Last Read                    0

Node: implicit_match_all        Queue index: 6
Priorities: 6
               EnQueue Packet Count                    0
                 EnQueue Byte Count                    0
               DeQueue Packet Count                    0
                 DeQueue Byte Count                    0
         Total Discard Packet Count                    0
```

```
           Total Discard Byte Count                    0
        Oldest Discard Packet Count                    0
          Oldest Discard Byte Count                    0
                Current Queue Depth                    0
Maximum Queue Depth Since Last Read                    0

Node: implicit_match_all        Queue index: 7
Priorities: 7
               EnQueue Packet Count                    0
                 EnQueue Byte Count                    0
               DeQueue Packet Count                    0
                 DeQueue Byte Count                    0
         Total Discard Packet Count                    0
           Total Discard Byte Count                    0
        Oldest Discard Packet Count                    0
          Oldest Discard Byte Count                    0
                Current Queue Depth                    0
Maximum Queue Depth Since Last Read                    0
```

Use the **show the hqos statistics** command to display flow information for the default other flows for an index.

```
Brocade#(config-if-e10000-2/1)#show hqos statistics ethernet 2/1 queue
default-other index 0
Queue name: implicit_match_all
Priorities: 0
               EnQueue Packet Count                    0
                 EnQueue Byte Count                    0
               DeQueue Packet Count                    0
                 DeQueue Byte Count                    0
         Total Discard Packet Count                    0
           Total Discard Byte Count                    0
        Oldest Discard Packet Count                    0
          Oldest Discard Byte Count                    0
                Current Queue Depth                    0
Maximum Queue Depth Since Last Read                    0
```

Use the **show the hqos statistics** command to display flow information for the default other flows for a specific index. The following examples displays index 1.

```
Brocade#(config-if-e10000-2/1)#show hqos statistics ethernet 2/1 queue
default-other index 1
Queue name: implicit_match_all
Priorities: 1
               EnQueue Packet Count                    0
                 EnQueue Byte Count                    0
               DeQueue Packet Count                    0
                 DeQueue Byte Count                    0
         Total Discard Packet Count                    0
           Total Discard Byte Count                    0
        Oldest Discard Packet Count                    0
          Oldest Discard Byte Count                    0
                Current Queue Depth                    0
Maximum Queue Depth Since Last Read                    0
```

Syntax: show hqos statistics ethernet *slot/port* **queue** *hqos-scheduler-node-name* [**index** *index number*]

Clearing HQoS statistics

Use the **clear the hqos statistics** command to clear the statistics for a specified flow.

```
Brocade# clear hqos statistics ethernet ethernet 1/1 queue default-other
```

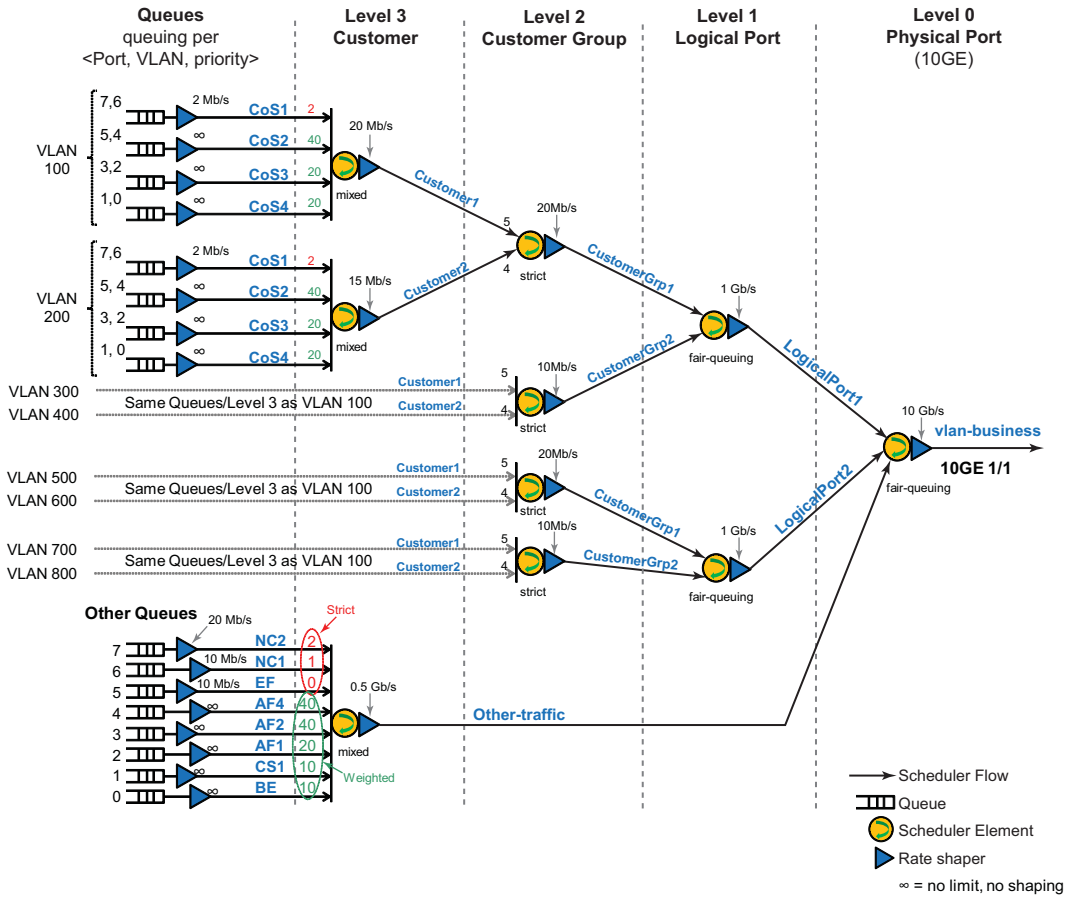
Syntax: Clear hqos statistics ethernet *slot/port* **queue** *hqos-scheduler-node-name* **index** *index*

Sample configurations

NOTE
All VPLS HQoS traffic in TM will drop after changing the loopback IP address in MPLS configuration.

Local VPLS HQoS example

FIGURE 20 VLAN HQoS deployment example



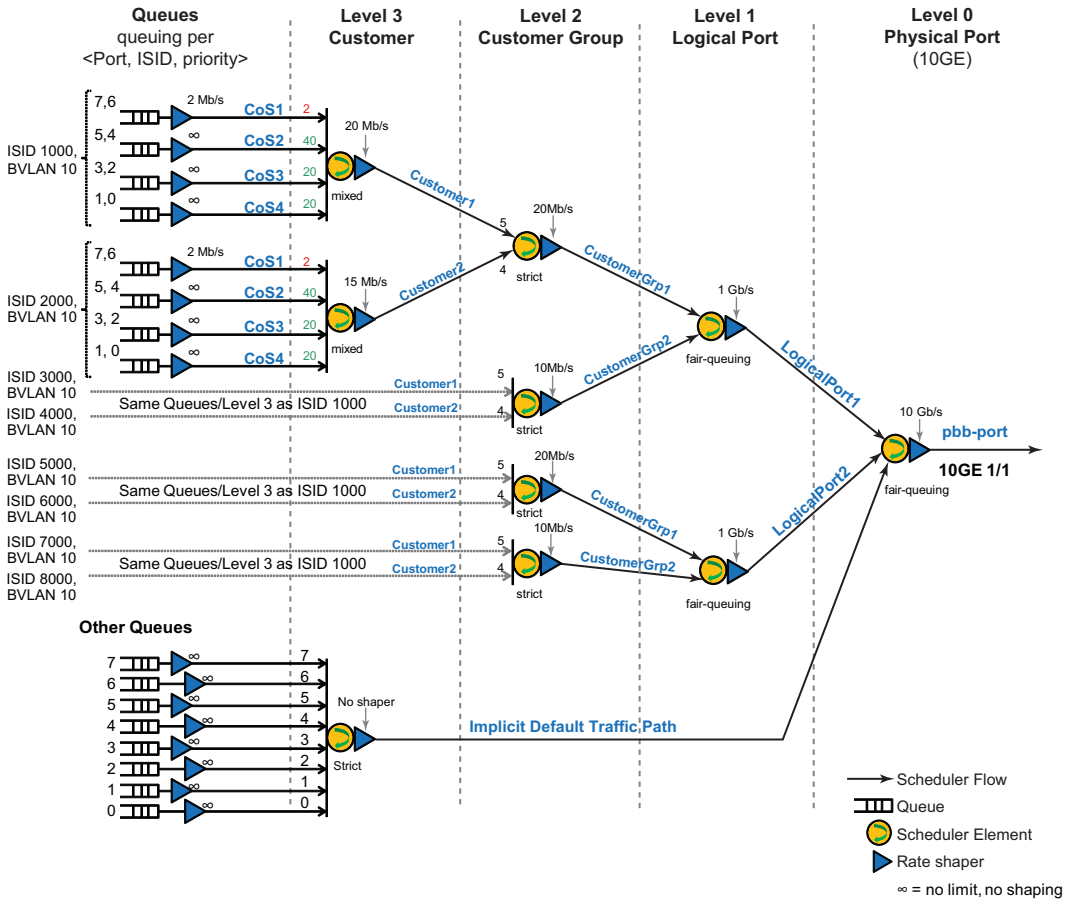

```
hqos scheduler-policy vlan-business level level-0
 shaper-rate 10000000
 shaper-burst-size 10
 scheduler-type weighted
  scheduler-flow LogicalPort1 scheduler-input 7 scheduler-policy
logical-port-type1
```

```
  scheduler-flow LogicalPort2 scheduler-input 6 scheduler-policy
logical-port-type1
  scheduler-flow Other-traffic  scheduler-input 5 scheduler-policy other-policy
!
hqos scheduler-policy logical-port-type1 level level-1
 shaper-rate 1000000
 shaper-burst-size 10
 scheduler-type weighted
  scheduler-flow CustomerGrp1 scheduler-input 3 scheduler-policy
customer-group-type1
  scheduler-flow CustomerGrp2 scheduler-input 2 scheduler-policy
customer-group-type1
!
hqos scheduler-policy customer-group-type1 level level-2
 shaper-rate  20000
 shaper-burst-size 10
 scheduler-type strict
  scheduler-flow Customer1 scheduler-input 3 scheduler-policy customer-type1
  scheduler-flow Customer2 scheduler-input 2 scheduler-policy customer-type1
!
hqos scheduler-policy customer-type1 level  level-3
 shaper-rate 20000
 shaper-burst-size 10
 scheduler-type mixed
  scheduler-flow CoS1 scheduler-input 3 scheduler-policy Q-7-6
  scheduler-flow CoS2 scheduler-input 2 weight 40 scheduler-policy Q-5-4
  scheduler-flow CoS3 scheduler-input 1 weight 20 scheduler-policy Q-3-2
  scheduler-flow CoS4 scheduler-input 0 weight 20 scheduler-policy Q-1-0
!
hqos scheduler-policy other-policy  level level-3
 scheduler-shaper 500000
 scheduler-type-other mixed
  scheduler-flow NC2 scheduler-input 7 scheduler-policy Q-7
  scheduler-flow NC1 scheduler-input 6 scheduler-policy Q-6
  scheduler-flow EF scheduler-input 5 scheduler-policy Q-5
  scheduler-flow AF4 scheduler-input 4 weight 40 scheduler-policy queue-default
  scheduler-flow AF2 scheduler-input 3 weight 40 scheduler-policy queue-default
  scheduler-flow AF1 scheduler-input 2 weight 20 scheduler-policy queue-default
  scheduler-flow CS1 scheduler-input 1 weight 10 scheduler-policy queue-default
  scheduler-flow BE  scheduler-input 0 weight 10 scheduler-policy queue-default
!
hqos queue-policy Q-7-6
    shaper-rate 2000
    shaper-burst-size 10
hqos queue-policy Q-5-4
    shaper-rate 10000000
    shaper-burst-size 10
hqos queue-policy Q-3-2
    shaper-rate 10000000
    shaper-burst-size 10
hqos queue-policy Q-1-0
    shaper-rate 10000000
    shaper-burst-size 10
hqos queue-policy Q-7
   shaper-rate  20000
hqos queue-policy Q-6
    shaper-rate 10000
hqos queue-policy Q-5
    shaper-rate 10000
hqos queue-policy queue-default
```

```
    shaper-rate 10000000
    shaper-burst-size 10
!
!
router mpls
 vpls Customer1 1
  vlan 100
   tagged ethe 2/1 ethe 1/1
 vpls Customer2 2
  vlan 200
   tagged ethe 2/1 ethe 1/1
 vpls Customer3 3
  vlan 300
   tagged ethe 2/1 ethe 1/1
 vpls Customer4 4
  vlan 400
   tagged ethe 2/1 ethe 1/1
 vpls Customer5 5
  vlan 500
   tagged ethe 2/1 ethe 1/1
 vpls Customer6 6
  vlan 600
   tagged ethe 2/1 ethe 1/1
 vpls Customer7 7
  vlan 700
   tagged ethe 2/1 ethe 1/1
 vpls Customer8 8
  vlan 800
   tagged ethe 2/1 ethe 1/1
!
!
interface ethernet 1/1
 hqos service-policy output vlan-business
 hqos-map LogicalPort1.CustomerGrp1.Customer1 match vlan 100
 hqos-map LogicalPort1.CustomerGrp1.Customer2 match vlan 200
 hqos-map LogicalPort1.CustomerGrp2.Customer1 match vlan 300
 hqos-map LogicalPort1.CustomerGrp2.Customer2 match vlan 400
 hqos-map LogicalPort2.CustomerGrp1.Customer1 match vlan 500
 hqos-map LogicalPort2.CustomerGrp1.Customer2 match vlan 600
 hqos-map LogicalPort2.CustomerGrp2.Customer1 match vlan 700
 hqos-map LogicalPort2.CustomerGrp2.Customer2 match vlan 800
 hqos-map Other-traffic match other
 hqos-map LogicalPort1.CustomerGrp1.Customer2 shaper-rate 15000
 hqos-map LogicalPort1.CustomerGrp2 shaper-rate 10000
 hqos-map LogicalPort2.CustomerGrp2 shaper-rate 10000
```

PBB HQoS example configuration

FIGURE 21 PBB HQoS deployment example




```
hqos scheduler-policy pbb-port level level-0
 shaper-rate 10000000
 shaper-burst-size 10
 scheduler-type weighted
  scheduler-flow LogicalPort1 scheduler-input 7 scheduler-policy
logical-port-type1
  scheduler-flow LogicalPort2 scheduler-input 6 scheduler-policy
logical-port-type1
!
hqos scheduler-policy logical-port-type1 level level-1
 shaper-rate 1000000
 shaper-burst-size 10
 scheduler-type weighted
  scheduler-flow CustomerGrp1 scheduler-input 3 weight 2 scheduler-policy
customer-group-type1
  scheduler-flow CustomerGrp2 scheduler-input 2 weight 1 scheduler-policy
customer-group-type1
!
hqos scheduler-policy customer-group-type1 level level-2
 shaper-rate  20000
 shaper-burst-size 10
 scheduler-type weighted
```

```
  scheduler-flow Customer1 scheduler-input 3 scheduler-policy customer-type1
  scheduler-flow Customer2 scheduler-input 2 scheduler-policy customer-type1
!
hqos scheduler-policy customer-type1 level  level-3
 shaper-rate 20000
 shaper-burst-size 10
 scheduler-type strict
  scheduler-flow CoS1 scheduler-input 3 scheduler-policy queue-default
  scheduler-flow CoS2 scheduler-input 2 scheduler-policy queue-default
  scheduler-flow CoS3 scheduler-input 1 scheduler-policy queue-default
  scheduler-flow CoS4 scheduler-input 0 scheduler-policy queue-default
!
hqos queue-policy queue-default
    shaper-rate 10000000
    shaper-burst-size 10
!
!
router mpls
 vpls Customer1 1
  pbb
  vlan 100
   tagged ethe 3/1
  vlan 10 isid 1000
   tagged eth 1/1 ethe 2/8

 vpls Customer2 2
  pbb
  vlan 200
   tagged ethe 3/1
  vlan 10 isid 2000
   tagged eth 1/1 eth 2/8

 vpls Customer3 3
  pbb
  vlan 300
   tagged ethe 3/1
  vlan 10 isid 3000
   tagged eth 1/1 eth 2/8

 vpls Customer4 4
  pbb
  vlan 400
   tagged ethe 3/1
  vlan 10 isid 4000
   tagged eth 1/1 eth 2/8

 vpls Customer5 5
  pbb
  vlan 500
   tagged ethe 3/1
  vlan 10 isid 5000
   tagged eth 1/1 eth 2/8

vpls Customer6 6
  pbb
  vlan 600
   tagged ethe 3/1

  vlan 10 isid 6000
tagged eth 1/1 eth 2/8
```

```
 vpls Customer7 7
  pbb
  vlan 700
   tagged ethe 3/1
  vlan 10 isid 7000
   tagged eth 1/1 eth 2/8

 vpls Customer8 8
  pbb
  vlan 800
   tagged ethe 3/1
  vlan 10 isid 8000
   tagged eth 1/1 eth 2/8
!
!interface ethernet 1/1
 hqos service-policy output pbb-port
 hqos-map Logical-Port1.CustomerGrp1.Customer1 match vlan 10 isid 1000
 hqos-map Logical-Port1.CustomerGrp1.Customer2 match vlan 10 isid 2000
 hqos-map Logical-Port1.CustomerGrp2.Customer1 match vlan 10 isid 3000
 hqos-map Logical-Port1.CustomerGrp2.Customer2 match vlan 10 isid 4000
 hqos-map Logical-Port2.CustomerGrp1.Customer1 match vlan 10 isid 5000
 hqos-map Logical-Port2.CustomerGrp1.Customer2 match vlan 10 isid 6000
 hqos-map Logical-Port2.CustomerGrp2.Customer1 match vlan 10 isid 7000
 hqos-map Logical-Port2.CustomerGrp2.Customer2 match vlan 10 isid 8000
```

Scheduler and queue policy configuration templates

Scheduler and queue policy configuration templates are available for creating the HQOS tree.

The configuration does not come into effect till they are bound to a interface supporting HQOS.

Once a policy is bound to an interface, you cannot make any changes to the policy (except shaper rate and burst size). The policy has to be unbound and then make the changes to the policy and rebind it.

The queues can be connected to any of the following levels (3, 2 or 1). It cannot be connected to level 0 directly.

The same policy can be applied to multiple interfaces.

HQOS scheduler policy configuration template

```
  [no] hqos scheduler-policy <scheduler-policy-name> level <level-number> |
level-0 | level-1| level-2 | level-3
    [no][shaper-rate <shaper-rate>]
    [no] [shaper-burst-size <shaper-burst-size>]
    [no]{scheduler-type | scheduler-type-other} <scheduler-type> | strict |
weighted | mixed
    [no]{scheduler-flow <scheduler-flow-name> {scheduler-input
<scheduler-input-value>} | [weight <weight-value>]  scheduler-policy
<scheduler-flow-policy-name>}
```

HQOS Queue policy configuration template

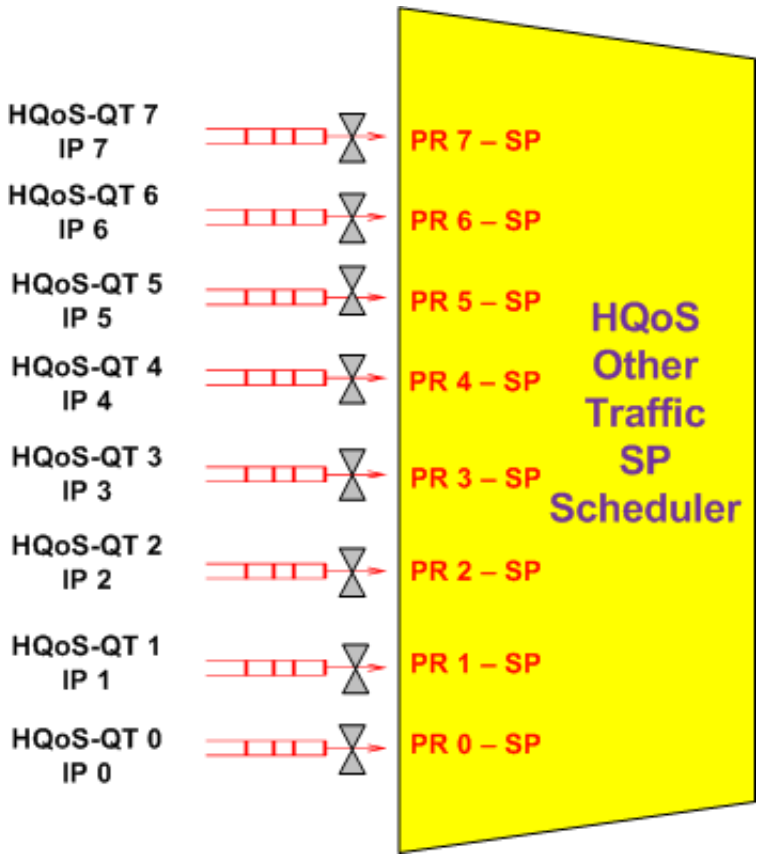
```
[no] hqos queue-policy <queue-name>
    [no] [shaper-rate <shaper-rate>]
    [no] [shaper-burst-size <shaper-burst-size>]

HQOS mapping template.
[no] hqos-map <hqos-scheduler-node-name>
              [no] [shaper-rate <shaper-rate>]
     [no] [shaper-burst-size <shaper-burst-size>]
               [no] [match other] |
              [no] [{match vlan <vlan-num>} | [inner-vlan <inner-vlan-num> | isid
<isid-num> ] ]

HQOS policy binding to an interface
interface ethernet <port >/<slot>
    [no] hqos service-policy output <level-0-scheduler-name>
```

HQoS queue scheduler models

Strict priority (SP)

Figure 22 is an example of scheduling model for HQoS other traffic. All the 8 scheduler inputs are SP.

FIGURE 22 Strict priority scheduler (SP)



HQoS-QT stands for hqos-queue-type. The range is <0-7>.

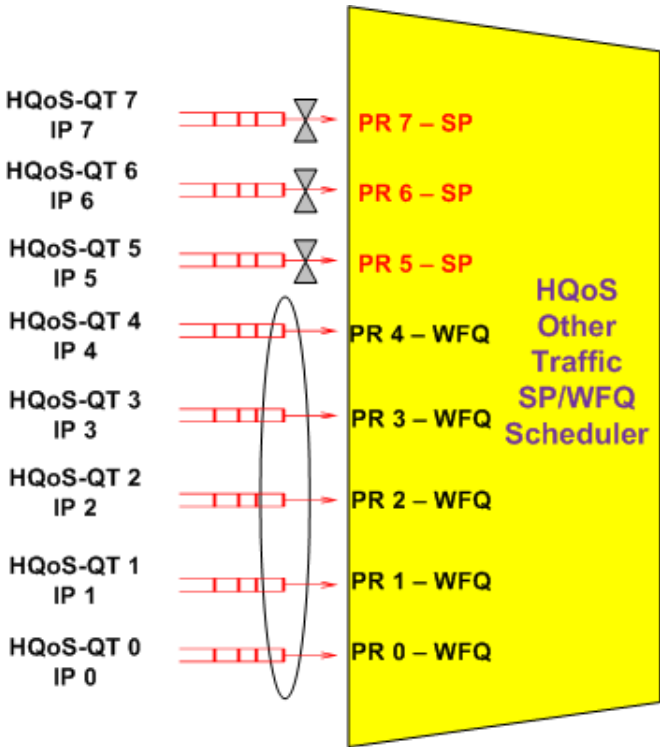
IP stands for internal priority. The range is <0-7>.

PR stands for scheduler-input (the ordering of a flow with respect to a scheduler which is specified in the hqos scheduler policy). The range is <0-7>.

Mixed Strict Priority and Weighted Fair Queue (SP/WFQ)

Figure 23 is an example of mixed SP and WFQ scheduling model for HQoS customer traffic. In this example, the top three scheduler inputs are SP and the bottom five scheduler inputs are WFQ.

FIGURE 23 Mixed Strict Priority and Weighted Fair Queue



The supportable weight range for each input is <1-64>.

HQoS-QT stands for hqos-queue-type. The range is <0-7>.

IP stands for internal priority. The range is <0-7>.

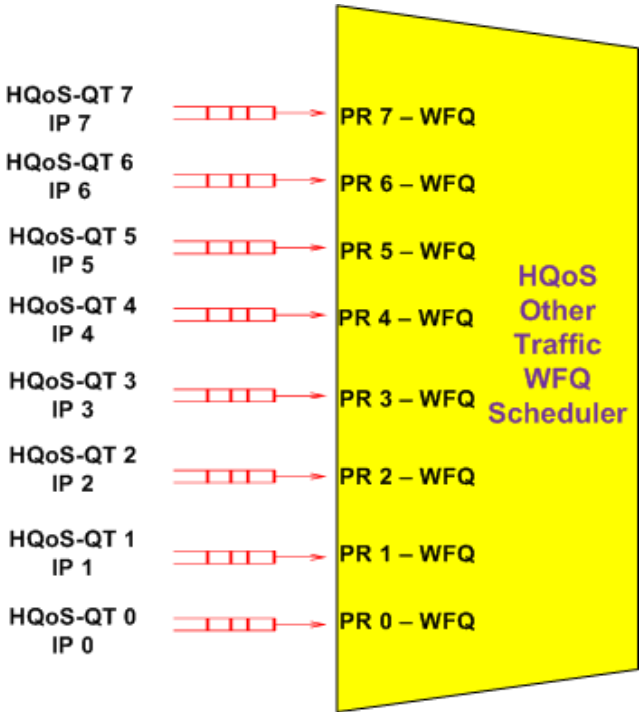
PR stands for scheduler-input (the ordering of a flow with respect to a scheduler which is specified in the hqos scheduler policy). The range is <0-7>.

Weighted Fair Queue and Fair Queue (WFQ/FQ)

Figure 24 is an example the WFQ and FQ scheduling model for HQoS other traffic.

In this example, all 8 scheduler inputs are WFQ. If all 8 scheduler inputs are equal, the scheduling model is Fair Queue (FQ). The supportable weight range for each input is <1-64>.

FIGURE 24 WFQ/FQ scheduling model for HQoS other traffic



HQoS-QT stands for hqos-queue-type. The range is <0-7>.

IP stands for internal priority. The range is <0-7>.

PR stands for scheduler-input (the ordering of a flow with respect to a scheduler which is specified in the hqos scheduler policy). The range is <0-7>.

QoS Queue Types

Table 48 and Table 49 include the system defaults for the HQoS queue-types and internal priority. These 12 queue-types are created in 8x10G modules during the LP initialization.

TABLE 48 HQoS "Other Traffic" queue-type

HQoS "Other Traffic" queue-type	Internal priority	8x10G family buffer-pool	8x10G family default queue size
7	7	Gold	1MB
6	6	Bronze	1MB
5	5	Bronze	1MB
4	4	Bronze	1MB

TABLE 48 HQoS "Other Traffic" queue-type

HQoS "Other Traffic" queue-type	Internal priority	8x10G family buffer-pool	8x10G family default queue size
3	3	Bronze	1MB
2	2	Bronze	1MB
1	1	Bronze	1MB
0	0	Bronze	1MB

TABLE 49 HQoS "Customer Traffic" queue-type

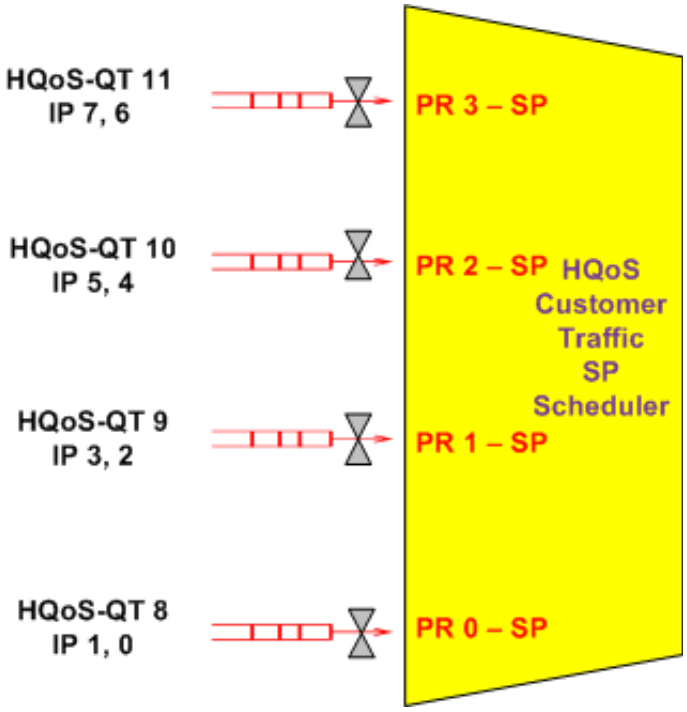
HQoS "Customer Traffic" queue-type	Internal priority	8x10G family buffer-pool	8x10G family default queue size
11	7,6	Gold	1MB
10	5,4	Bronze	1MB
9	3,2	Bronze	1MB
8	0,1	Bronze	1MB

HQoS Queue Schedulers - Customer Traffic

Strict Priority (SP)

Figure 25 depicts the SP scheduling model for HQoS other traffic. All the 4 scheduler inputs are SP.

FIGURE 25 SP scheduling model for HQoS other traffic



HQoS-QT stands for hqos-queue-type. The range is <8-11>.

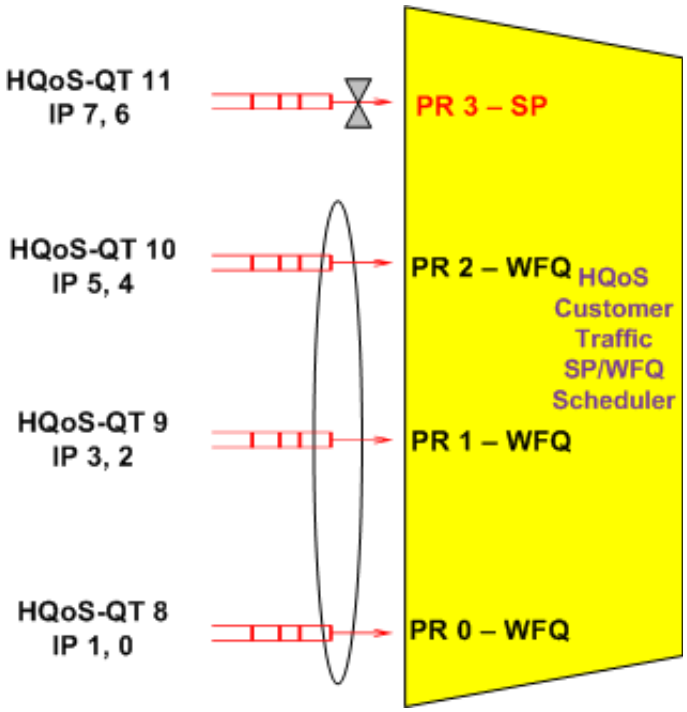
IP stands for internal priority. The range is <0-7>.

PR stands for scheduler-input (the ordering of a flow with respect to a scheduler which is specified in the hqos scheduler policy). The range is <0-3>.

Mixed Strict Priority and Weighted Fair Queue (SP/WFQ)

Figure 23 depicts the mixed SP/WFQ scheduling model for HQoS other traffic. The top scheduler input is SP and the bottom 3 scheduler inputs are WFQ.

FIGURE 26 Mixed Strict Priority and Weighted Fair Queue



The supportable weight range for each input is <1-64>.

HQoS-QT stands for hqos-queue-type. The range is <8-11>.

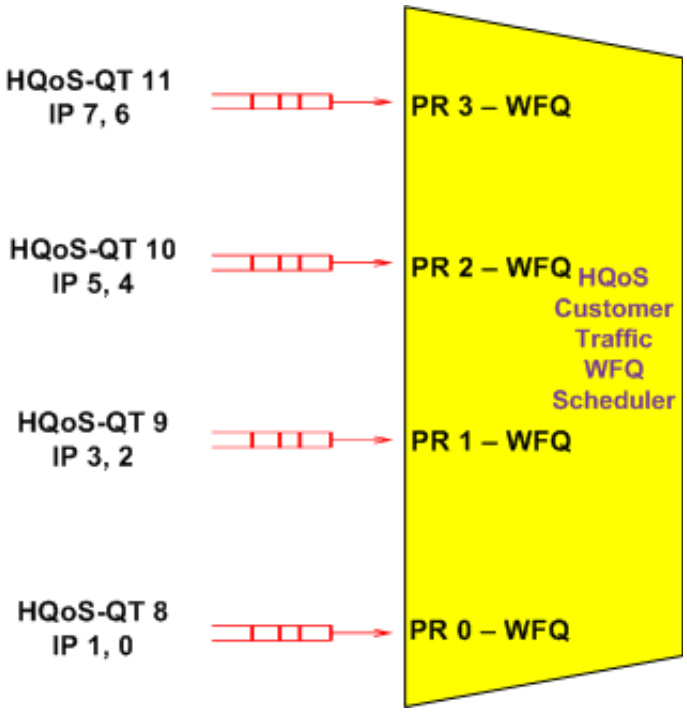
IP stands for internal priority. The range is <0-7>.

PR stands for scheduler-input (the ordering of a flow with respect to a scheduler which is specified in the hqos scheduler policy). The range is <0-3>.

Weighted Fair Queue and Fair Queue (WFQ/FQ)

Figure 27 depicts the mixed Weighted Fair Queue (WFQ) scheduling model for HQoS customer traffic. All the 4 scheduler inputs are WFQ. If all the 4 scheduler inputs are equal, the scheduling model is FQ.

FIGURE 27 Weighted Fair Queue and Fair Queue



The supportable weight range for each input is <1-64>.

HQoS-QT stands for hqos-queue-type. The range is <8-11>.

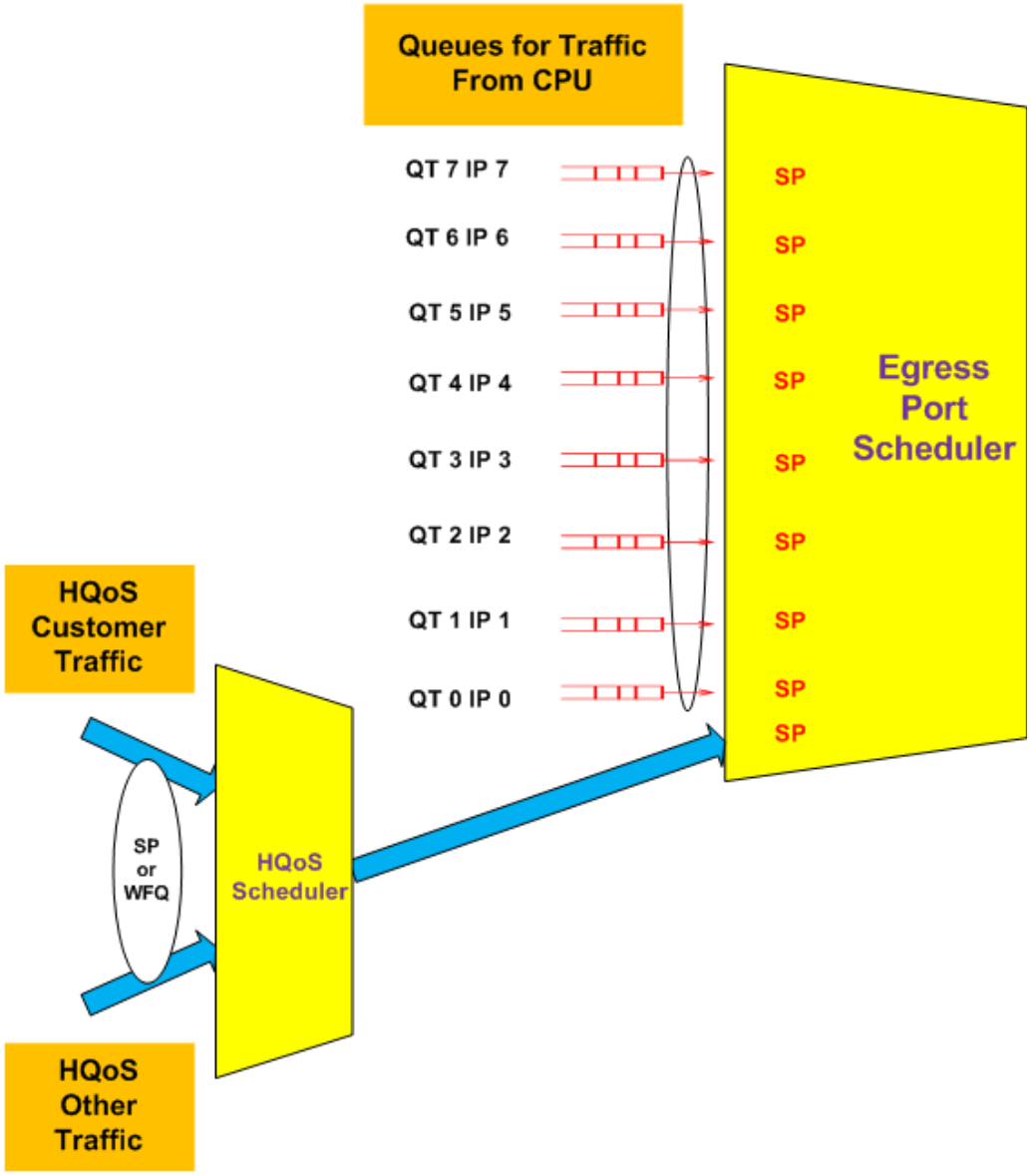
IP stands for internal priority. The range is <0-7>.

PR stands for scheduler-input (the ordering of a flow with respect to a scheduler which is specified in the hqos scheduler policy). The range is <0-3>.

HQoS egress port scheduler

Figure 28 depicts the Egress Port Scheduler for a port for which HQoS is enabled.

FIGURE 28 HQoS egress port scheduler



The Egress Port Scheduler has 9 SP inputs. The scheduler and queue setup for the first 8 inputs is exactly the same as the current egress port scheduler without HQoS. The queues attached to the first 8 inputs are used for scheduling packets from the CPU, which are sent which includes protocol packets and CPU forwarded packets. The 9th (last) input is used for attaching the HQoS scheduler. The scheduling mechanism between the packets sent from CPU and the level 0 HQoS scheduler is strict priority.