



CHAPTER 2

Using Modular QoS CLI

This chapter describes how to configure Modular QoS CLI (MQC) objects that can be used for configuring QoS features using the Cisco Nexus 7000 Series NX-OS software.

This chapter includes the following sections:

- [Information About MQC, page 2-1](#)
- [Licensing Requirements for Using MQC Objects, page 2-2](#)
- [Using an MQC Object, page 2-2](#)
- [Attaching and Detaching a QoS Policy Action from an Interface, page 2-17](#)
- [Session Manager Support for QoS, page 2-19](#)
- [Feature History for Using Modular QoS CLI, page 2-20](#)

Information About MQC

MQC provides a language to define QoS policies.



Note

MQC commands are included in the *Cisco Nexus 7000 Series NX-OS Quality of Service Command Reference, Release 4.2*.

You configure QoS policies using three steps:

1. Define traffic classes.
2. Associate policies and actions with each traffic class.
3. Attach policies to logical or physical interfaces and VLANs.

MQC provides three command types to define traffic classes and policies:

- **class-map**—Defines a class map that represents a class of traffic based on packet-matching criteria. Class maps are referenced in policy maps.



Note

When you configure match all for a QoS class map by entering the *class-map type qos match-all* command, the match-all option does not work. Instead, the match criteria is always treated as match any.

- **table-map**—Defines a table map that represents a mapping from one set of packet field values to another set of packet fields. Table maps are referenced in policy maps.

Send document comments to nexus7k-docfeedback@cisco.com.

- **policy-map**—Defines a policy map that represents a set of policies to be applied on a class-by-class basis to class maps.

You define the following class-map and policy-map object types when you create them:

- **qos**—Defines MQC objects that you can use for marking and policing.
- **queuing**—Defines MQC objects that you can use for queuing and scheduling.



Note

The **qos** type is the default.

You can attach policies to ports, port channels, VLANs, subinterfaces, or tunnels by using the **service-policy** interface configuration command.

You can view all or individual values for MQC objects by using the **show table-map**, **show class-map**, and **show policy-map** commands.



Caution

In the interface configuration mode, the Cisco Nexus 7000 switch might accept QoS and ACL commands irrespective of the line card on which the interface is hosts is up or down. However, you cannot enter interface sub-mode when the line card is down because the Cisco Nexus 7000 switch does not accept any pre-configuration information.

Licensing Requirements for Using MQC Objects

The following table shows the licensing requirements for this feature:

Product	License Requirement
NX-OS	QoS requires no license. Any feature not included in a license package is bundled with the Cisco NX-OS system images and is provided at no extra charge to you. For a complete explanation of the NX-OS licensing scheme, see the <i>Cisco Nexus 7000 Series NX-OS Licensing Guide, Release 4.2</i> .

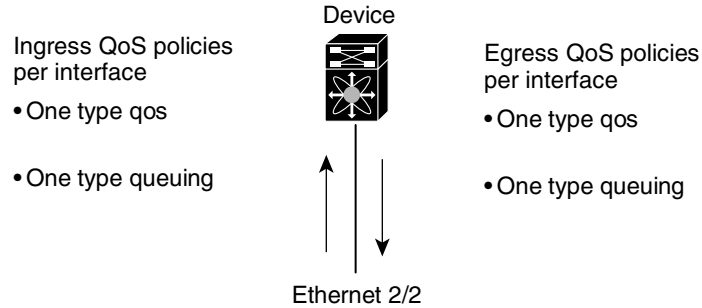
However, using VDCs requires an Advanced Services license.

Using an MQC Object

You configure QoS and queuing policies using the MQC class-map, policy-map, and table-map objects. You cannot use table maps in queuing policies. After you configure class maps and policy maps, you can attach one policy map of each type to the ingress direction of an interface. [Figure 2-1](#) lists the maximum QoS and queuing policies that you can define on each interface.

Send document comments to nexus7k-docfeedback@cisco.com.

Figure 2-1 Maximum QoS Policies Per Interface



A policy map contains either a qos policy or queuing policy. The policy map references the names of class maps that represent traffic classes. For each class of traffic, the device applies the policies on the interface or VLAN that you select.

A packet is matched sequentially to a class of traffic starting from the first traffic class definition. When a match is found, the policy actions for that class are applied to the packet.

The reserved class map receives all traffic that is not matched in type qos policies, and the device applies the policy actions as it would for any other traffic class. You use class-default to perform mutations (mutation is a method for translating QoS values in the packet header prior to traffic classification).



Note

You can access user-defined MQC objects only in the virtual device context (VDC) in which they were created. You can access the system-defined MQC objects in all VDCs.

This section includes the following topics:

- [Type qos Policies, page 2-3](#)
- [Type queuing Policies, page 2-5](#)
- [System-Defined MQC Objects, page 2-6](#)
- [Configuring an MQC Object, page 2-10](#)
- [Applying Descriptions to MQC Objects, page 2-15](#)
- [Verifying an MQC Object, page 2-16](#)

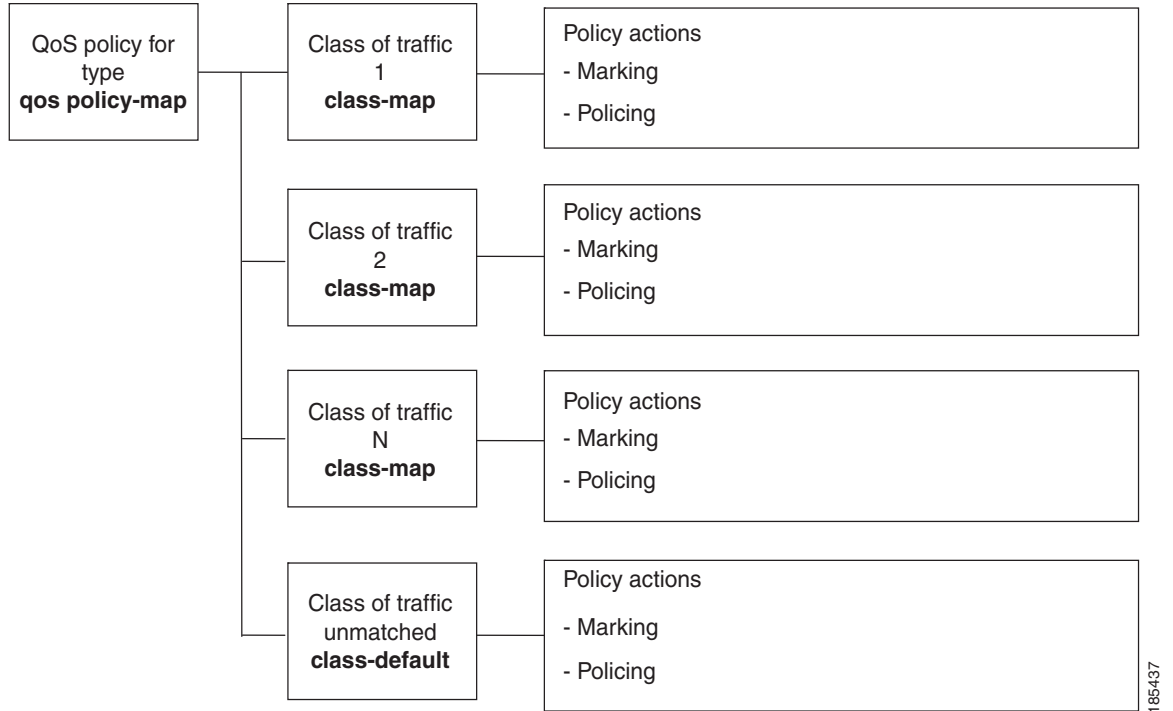
Type qos Policies

You use type qos policies to mark, to apply mutations, to set the ingress port trust state, and to police packets.

[Figure 2-2](#) shows the QoS policy structure with the associated MQC objects of type qos without mutation, and [Figure 2-3](#) shows the QoS policy structure with mutation. The MQC objects are shown in bold.

Send document comments to nexus7k-docfeedback@cisco.com.

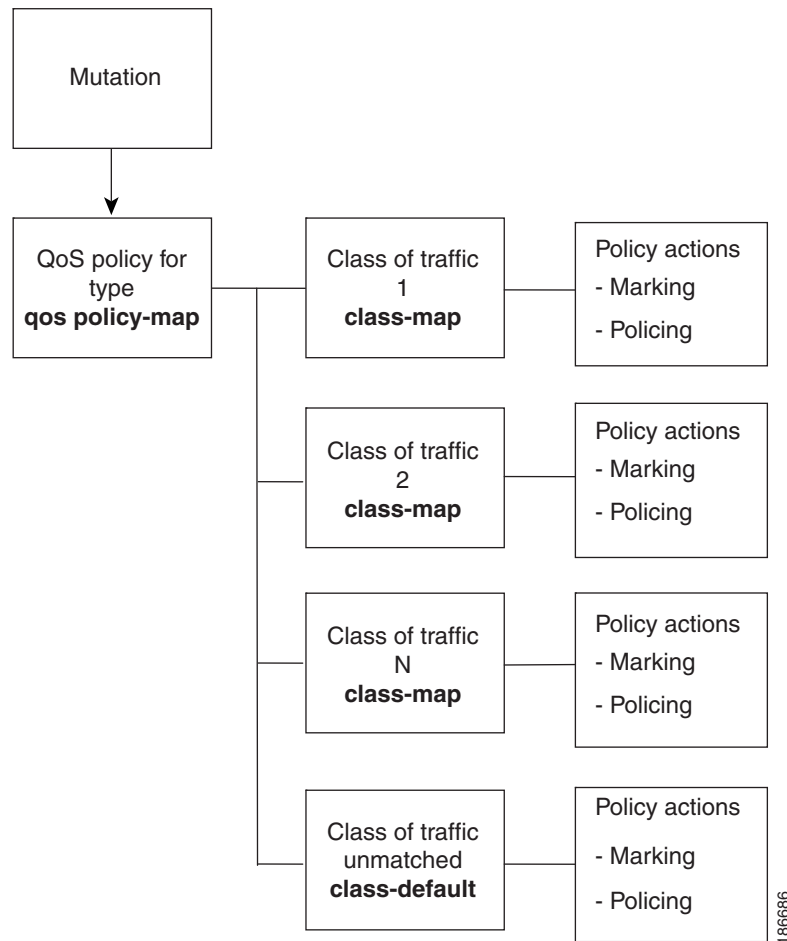
Figure 2-2 QoS Policy Diagram Showing Type qos MQC Object Usage without Mutation



185437

Send document comments to nexus7k-docfeedback@cisco.com.

Figure 2-3 QoS Policy Diagram Showing Type qos MQC Object Usage with Mutation



Type queuing Policies

You use type queuing policies to mark, shape, and queue packets. Marking is limited to the CoS field and does not support the use of table maps.

Figure 2-4 shows the QoS policy structure with associated MQC objects of type queuing. The MQC objects are shown in bold.

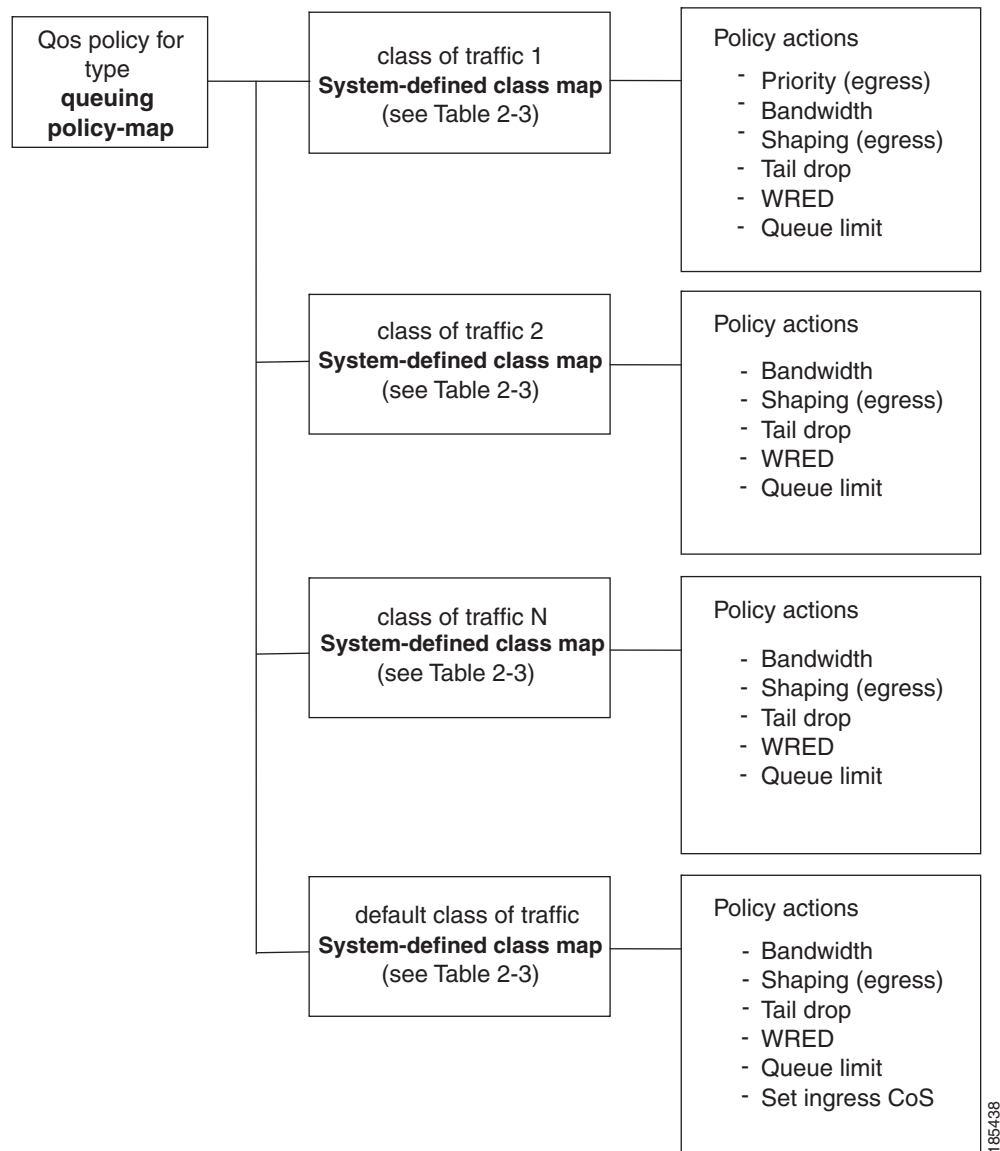


Note

MQC table-map objects cannot be used in policies of type queuing.

Send document comments to nexus7k-docfeedback@cisco.com.

Figure 2-4 QoS Policy Diagram Showing Type queuing MQC Object Usage



Note: See Chapter 5, "Queuing and Scheduling," for information on configuring these parameters.

System-Defined MQC Objects



Note

These are the default MQC objects. All of these values apply across all VDCs.

When you configure QoS features, and the systems requests one of these MQC objects, you can use these system-defined objects. The system-defined MQC objects are shown in [Table 2-1](#). See the tables listed next to the object for information on these system-defined objects.

Send document comments to nexus7k-docfeedback@cisco.com.

Table 2-1 System-Defined MQC Objects

Table and Page	Description
Table 2-2 on page 2-7	Type qos class maps
Table 2-3 on page 2-7	Type queuing class maps
Table 2-4 on page 2-8	Table maps
Table 2-5 on page 2-9	Policy maps

Type qos class maps that are defined by the system are listed in [Table 2-2](#).



Note

You cannot reference the conform-color-in, conform-color-out, exceed-color-in, or exceed-color-out class maps in a policy map.

Table 2-2 System-Defined Type qos Class Maps

Class Map Name	Description
class-default	Type qos class map that is assigned to all packets that match none of the criteria of traffic classes that you define in a type qos policy map. You can use class-default for mutation.
conform-color-in	Type qos conform color class map in the input direction. This color-aware class map makes a policer color-aware for conform action.
conform-color-out	Type qos conform color class map in the output direction. This color-aware class map makes a policer color-aware for conform action.
exceed-color-in	Type qos exceed color class map in the input direction. This color-aware class map makes a policer color-aware for exceed action.
exceed-color-out	Type qos exceed color class map in the output direction. This color-aware class map makes a policer color-aware for exceed action.

Type queuing class maps that are defined by the system are listed in [Table 2-3](#).

Table 2-3 System-Defined Type queuing Class Maps

Class Map Queue Name	Description	Default CoS Values
1 Gigabit Module Ingress: 2 queues with 4 thresholds per queue		
2q4t-in-q1	Ingress queue 1 of 2q4t type	5-7
2q4t-in-q-default	Ingress default queue of 2q4t type	0-4
1 Gigabit Module Egress: 1 strict priority queue and 3 normal queues with 4 thresholds per queue		
1p3q4t-out-pq1 ¹	Egress priority queue of 1p3q4t type	5-7
1p3q4t-out-q2	Egress queue 2 of 1p3q4t type	—

Send document comments to nexus7k-docfeedback@cisco.com.

Table 2-3 System-Defined Type queuing Class Maps (continued)

Class Map Queue Name	Description	Default CoS Values
1p3q4t-out-q3	Egress queue 3 of 1p3q4t type	—
1p3q4t-out-q-default	Egress default queue of 1p3q4t type	0-4
10 Gigabit Module Ingress: 8 queues with 2 thresholds per queue		
8q2t-in-q1	Ingress queue 1 of 8q2t type	5-7
8q2t-in-q2	Ingress queue 2 of 8q2t type	—
8q2t-in-q3	Ingress queue 3 of 8q2t type	—
8q2t-in-q4	Ingress queue 4 of 8q2t type	—
8q2t-in-q5	Ingress queue 5 of 8q2t type	—
8q2t-in-q6	Ingress queue 6 of 8q2t type	—
8q2t-in-q7	Ingress queue 7 of 8q2t type	—
8q2t-in-q-default	Ingress default queue of 8q2t type	0-4
10 Gigabit Module Egress: 1 strict priority queue and 7 normal queues with 4 thresholds per queue		
1p7q4t-out-pq1 ¹	Egress priority queue of 1p7q4t type	5-7
1p7q4t-out-q2	Egress queue 2 of 1p7q4t type	—
1p7q4t-out-q3	Egress queue 3 of 1p7q4t type	—
1p7q4t-out-q4	Egress queue 4 of 1p7q4t type	—
1p7q4t-out-q5	Egress queue 5 of 1p7q4t type	—
1p7q4t-out-q6	Egress queue 6 of 1p7q4t type	—
1p7q4t-out-q7	Egress queue 7 of 1p7q4t type	—
1p7q4t-out-q-default	Egress default queue of 1p7q4t type	0-4

1. These are either priority or normal queues. If you use the priority keyword in your configuration, these are used as priority queues. Otherwise, they are used as normal queues.

Table maps that are defined by the system are listed in [Table 2-4](#). The default mapping of values in the tables maps is contained in RFC 2597. These are not configurable.

Table 2-4 System-Defined Table Maps

Table Map Name	Description
cir-markdown-map	Table map used to mark down packets that exceed the committed information rate (CIR). Note Enter the show table-map command to display the default mapping.
pir-markdown-map	Table map used to mark down packets that violate the peak information rate (PIR). Note Enter the show table-map command to display the default mapping.
cos-discard-class-map	Table map used to map the CoS value to the discard-class value.

Send document comments to nexus7k-docfeedback@cisco.com.

Table 2-4 **System-Defined Table Maps (continued)**

Table Map Name	Description
cos-dscp-map	Table map used to map the CoS value to the DSCP value.
cos-precedence-map	Table map used to map the CoS value to the precedence value.
dscp-cos-map	Table map used to map the DSCP value to the CoS value.
dscp-precedence-map	Table map used to map the DSCP value to the precedence value.
dscp-discard-class-map	Table map used to map the DSCP value to the discard-class value.
precedence-dscp-map	Table map used to map the precedence value to the DSCP value.
precedence-cos-map	Table map used to map the precedence value to the CoS value.
precedence-discard-class-map	Table map used to map the precedence value to the discard-class value.
discard-class-cos-map	Table map used to map the discard-class value to the CoS value.
discard-class-prec-map	Table map used to map the discard-class value to the precedence value.
discard-class-dscp-map	Table map used to map the discard-class value to the DSCP value.

Policy maps that are defined by the system are listed in [Table 2-5](#).

Table 2-5 **System-Defined Queuing Policy Maps**

Queuing Policy Map Name	Description
default-in-policy	Input queuing policy map that is attached to all module ports to which you do not apply a queuing policy map. The default configuration values are as follows: policy-map type queuing default-in-policy class type queuing in-q1 queue-limit percent 50 bandwidth percent 80 class type queuing in-q-default queue-limit percent 50 bandwidth percent 20
default-out-policy	Output queuing policy map that is attached to all module ports to which you do not apply a queuing policy map. The default configuration values are as follows: policy-map type queuing default-out-policy class type queuing out-pq1 priority level 1 queue-limit percent 16 class type queuing out-q2 queue-limit percent 1 class type queuing out-q3 queue-limit percent 1 class type queuing out-q-default queue-limit percent 82 bandwidth remaining percent 25

Send document comments to nexus7k-docfeedback@cisco.com.

Configuring an MQC Object

When you specify a MQC object command, the device creates the object if it does not exist and then enters map mode.

To remove a class-map, table-map, or policy-map object, use the **no** form of the command that you used to create the object.

For the commands that you can use in the MQC object mode, see the following configuration chapters:

- [Chapter 3, “Configuring Classification”](#)
- [Chapter 4, “Configuring Marking”](#)
- [Chapter 5, “Configuring Mutation Mapping”](#)
- [Chapter 6, “Configuring Policing”](#)
- [Chapter 7, “Configuring Queuing and Scheduling”](#)

This section includes the following topics:

- [Configuring or Modifying a Class Map, page 2-10](#)
- [Configuring or Modifying a Table Map, page 2-12](#)
- [Configuring or Modifying a Policy Map, page 2-13](#)

Configuring or Modifying a Class Map

You can create or modify a class map. You can then reference class maps in policy maps.



Note

You cannot create a queuing class map; you must use one of the system-defined queuing class maps listed in [Table 2-3](#).

SUMMARY STEPS

1. **config t**
2. **class-map** [type qos] [match-any | match-all] *class-map-name*
3. **exit**
4. **class-map** [type qos] { conform-color-in | conform-color-out | exceed-color-in | exceed-color-out }
5. **exit**
6. **class-map** type queuing match-any *class-queuing-name*
7. **exit**
8. **show class-map** [type qos] [*class-map-name* | conform-color-in | conform-color-out | exceed-color-in | exceed-color-out]
9. **show class-map** type queuing [*class-queuing-name*]
10. **copy running-config startup-config**

Send document comments to nexus7k-docfeedback@cisco.com.

DETAILED STEPS

	Command	Purpose
Step 1	config t Example: switch# config t switch(config)#	Enters configuration mode.
Step 2	class-map [type qos] [match-any match-all] class-map-name Example: switch(config)# class-map class1 switch(config-cmap-qos)#	Creates or accesses the class map of type qos, and then enters class-map qos mode. Class-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters. Note When you configure match all for a QoS class map by entering the <i>class-map type qos match-all</i> command, the match-all option does not work. Instead, the match criteria is always treated as match any
Step 3	exit Example: switch(config-cmap-qos)# exit switch(config)#	Exits class-map qos mode and enters configuration mode.
Step 4	class-map [type qos] {conform-color-in conform-color-out exceed-color-in exceed-color-out} Example: switch(config)# class-map exceed-color-in switch(config-color-map)#	(Optional) Accesses the class map of type qos for one of the system-defined color maps, and then enters color-map mode. Note This is only used when color-aware policing is required.
Step 5	exit Example: switch(config-color-map)# exit switch(config)#	Exits color-map mode, and then enters configuration mode.
Step 6	class-map type queuing match-any [class-queuing-name] Example: switch(config)# class-map type queuing match-any lp3q4t-out-pql switch(config-cmap-que)#	Creates or accesses the class map of type queuing, and then enters class-map queuing mode. Class queuing names are listed in Table 2-3 .

Send document comments to nexus7k-docfeedback@cisco.com.

	Command	Purpose
Step 7	exit Example: switch(config-cmap-que)# exit switch(config)#	Exits class-map queuing mode and enters configuration mode.
Step 8	show class-map [type qos] [<i>class-map-name</i> conform-color-in conform-color-out exceed-color-in exceed-color-out] Example: switch(config)# show class-map	(Optional) Displays information about all configured class maps or a selected class map of type qos.
Step 9	show class-map type queuing [<i>class-queuing-name</i>] Example: switch(config)# show class-map type queuing	(Optional) Displays information about all configured class maps or a selected class map of type queuing. Class queuing names are listed in Table 2-3 .
Step 10	copy running-config startup-config Example: switch(config)# copy running-config startup-config	(Optional) Saves the running configuration to the startup configuration.

Configuring or Modifying a Table Map

You can create or modify a table map that you can reference in policy maps. See [Chapter 4, “Configuring Marking”](#) for information on configuring table maps.

SUMMARY STEPS

1. **config t**
2. **table-map** *table-map-name*
3. **exit**
4. **table-map** { **cir-markdown-map** | **pir-markdown-map**
5. **exit**
6. **show table-map** [*table-map-name* | **cir-markdown-map** | **pir-markdown-map**]
7. **copy running-config startup-config**

Send document comments to nexus7k-docfeedback@cisco.com.

DETAILED STEPS

	Command	Purpose
Step 1	config t Example: switch# config t switch(config)#	Enters configuration mode.
Step 2	table-map table-map-name Example: switch(config)# table-map table1 switch(config-tmap)#	Creates or accesses the table map and then enters table-map mode. Table map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.
Step 3	exit Example: switch(config-tmap)# exit switch(config)#	Exits table-map mode and enters configuration mode.
Step 4	table-map {cir-markdown-map pir-markdown-map} Example: switch(config)# table-map cir-markdown-map switch(config-mrkdn-map)#	Accesses one of the system-defined markdown table maps, and then enters markdown-map mode.
Step 5	exit Example: switch(config-mrkdn-map)# exit switch(config)#	Exits table-map mode and enters configuration mode.
Step 6	show table-map [table-map-name cir-markdown-map pir-markdown-map cos-discard-class-map] Example: switch(config)# show table-map	(Optional) Displays information about all configured table maps or a selected table map.
Step 7	copy running-config startup-config Example: switch(config)# copy running-config startup-config	(Optional) Saves the running configuration to the startup configuration.

Configuring or Modifying a Policy Map

You can create or modify a policy map that you can use to define actions to perform on class maps.

SUMMARY STEPS

1. **config t**
2. **policy-map [type qos] [match-first] {qos-policy-map-name | qos-dynamic}**
3. **exit**
4. **policy-map type queuing [match-first] {queuing-policy-map-name | qos-dynamic}**
5. **exit**

Send document comments to nexus7k-docfeedback@cisco.com.

6. **show policy-map** [**type qos**] [*policy-map-name* | **qos-dynamic**]
7. **show policy-map type queuing** [*policy-map-name* | **qos-dynamic**]
8. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose
Step 1	config t Example: switch# config t switch(config)#	Enters configuration mode.
Step 2	policy-map [type qos] [match-first] [<i>policy-map-name</i> qos-dynamic] Example: switch(config)# policy-map policy1 switch(config-pmap-qos)#	Creates or accesses the policy map of type qos and then enters policy-map mode. Policy-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.
Step 3	exit Example: switch(config-tmap)# exit switch(config)#	Exits policy-map mode and enters configuration mode.
Step 4	policy-map type queuing [match-first] [<i>policy-map-name</i> qos-dynamic] Example: switch(config)# policy-map type queuing policy_queue1 switch(config-pmap-que)#	Creates or accesses the policy map of type queuing and then enters policy-map mode. You can specify a policy-map name. Policy-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.
Step 5	exit Example: switch(config-tmap)# exit switch(config)#	Exits policy-map mode and enters configuration mode.
Step 6	show policy-map [type qos] [<i>policy-map-name</i> qos-dynamic] Example: switch(config)# show policy-map	(Optional) Displays information about all configured policy maps or a selected policy map of type qos.
Step 7	show policy-map type queuing [<i>policy-map-name</i> qos-dynamic] Example: switch(config)# show policy-map type queuing	(Optional) Displays information about all configured policy maps or a selected policy map of type queuing.
Step 8	copy running-config startup-config Example: switch(config)# copy running-config startup-config	(Optional) Saves the running configuration to the startup configuration.

Send document comments to nexus7k-docfeedback@cisco.com.

Applying Descriptions to MQC Objects

You can apply the **description** command to any MQC object.

SUMMARY STEPS

1. **config t**
2. **class-map** [type qos] [match-any | match-all] *class-map-name*
or
table-map *table-map-name*
or
policy-map [type qos] [match-first] [*policy-map-name* | qos-dynamic]
3. **description** *string*
4. **exit**
5. **copy running-config startup-config**

DETAILED STEPS

	Command	Purpose
Step 1	config t Example: switch# config t switch(config)#	Enters configuration mode.

Send document comments to nexus7k-docfeedback@cisco.com.

	Command	Purpose
Step 2	class-map [type qos] [match-any match-all] <i>class-map-name</i> Example: switch(config-cmap)# class-map class1 switch(config-cmap)#	Creates or accesses the class map, and then enters class-map mode. The class-map name can contain alphabetic, hyphen, or underscore characters, is case sensitive, and can be up to 40 alphanumeric characters.
	table-map <i>table-map-name</i> Example: switch(config-tmap)# table-map table1 switch(config-tmap)#	Creates or accesses the table map, and then enters table-map mode. The table-map name can contain alphabetic, hyphen, or underscore characters, is case sensitive, and can be up to 40 characters.
	policy-map [type qos] [match-first] [<i>policy-map-name</i> qos-dynamic] Example: switch(config)# policy-map policy1 switch(config-pmap)#	Creates or accesses the policy map, and then enters policy-map mode. The policy-map name can contain alphabetic, hyphen, or underscore characters, is case sensitive, and can be up to 40 characters.
Step 3	description <i>string</i> Example: switch(config-cmap)# description my traffic class switch(config-cmap)#	Adds a description string to the MQC object. The description can be up to 200 alphanumeric characters. Note You cannot modify the description of system-defined queuing class maps.
Step 4	exit Example: switch(config-cmap)# exit switch(config)#	Exits table-map mode and enters configuration mode.
Step 5	copy running-config startup-config Example: switch(config)# copy running-config startup-config	(Optional) Saves the running configuration to the startup configuration.

Verifying an MQC Object

To display MQC object configuration information, perform one of the following tasks:

Command	Purpose
show class-map [type qos] [<i>class-map-name</i> conform-color-in conform-color-out exceed-color-in exceed-color-out]	Displays information about all configured class maps or a selected class map of type qos.
show class-map type queuing [<i>class-queuing-name</i>]	Displays information about all configured class maps or a selected class map of type queuing. Class queuing names are listed in Table 2-3 .
show table-map [<i>table-map-name</i> cir-markdown-map pir-markdown-map]	Displays information about all configured table maps or a selected table map.
show policy-map [type qos] [<i>policy-map-name</i> qos-dynamic]	Displays information about all configured policy maps or a selected policy map of type qos.
show policy-map type queuing [<i>policy-map-name</i> qos-dynamic]	Displays information about all configured policy maps or a selected policy map of type queuing.

Send document comments to nexus7k-docfeedback@cisco.com.

For detailed information about the fields in the output from these commands, see the *Cisco Nexus 7000 Series NX-OS Quality of Service Command Reference, Release 4.2*.

Attaching and Detaching a QoS Policy Action from an Interface

The software does not allow you to enable or disable QoS features with a configuration command. To enable or disable QoS features, you must attach or detach QoS policies to or from interfaces, VLANs, or tunnels as described in this section.

**Note**

You must enable the **tunnel feature** by entering the feature tunnel command and configure the tunnel before you attach policies.

The system-defined type queuing class maps (see [Table 2-3](#)) are attached to each interface unless you specifically attach a different class map.

**Note**

The device restricts QoS policies to one per interface per direction (ingress or egress) for each of the policy types qos and queuing.

Policies that are defined at multiple interfaces have the following restrictions:

- A QoS policy attached to the physical port will take effect when the port is not a member of a port channel.
- A QoS policy attached to a port channel will take effect even when policies are attached to member ports.
- A QoS policy attached to a VLAN is applied to all ports in that VLAN that do not have other policies specifically applied.
- One ingress policy type queuing is supported for each Layer 2 port- and Layer 2 port-channel interface in both the ingress and egress direction. Egress type qos policies are not allowed on Layer 2 port or Layer 2 port-channel interfaces.
- One ingress and one egress QoS policy are supported for each Layer 3 and Layer 3 port-channel interface.
- One ingress and one egress QoS policy are supported for each VLAN.
- One ingress and one egress queuing policy are supported for each Layer 2 port-, Layer 2 port-channel, Layer 3 port-, and Layer 3 port-channel interface.
- When a VLAN or port channel, or both, touches multiple forwarding engines, all policies that enforce a rate are enforced per forwarding engine.

For example, a policer configured on a specific VLAN that limits the rate for the VLAN to 100 Mbps and has one switch port in the VLAN on one module and has another switch port in the VLAN on another module, each forwarding engine enforces the 100-Mbps rate. In this case, you could actually have up to 200 Mbps in the VLAN you configured to limit the rate to 100 Mbps.

**Note**

Default queuing policies are active, unless you configure and apply another policy. See [Table 2-5](#) for the default queuing policies.

Send document comments to nexus7k-docfeedback@cisco.com.

The interface where a QoS policy is applied is summarized in Table 2-6. Each row represents the interface levels. The entry descriptions are as follows:

- Applied—Interface where an attached policy is applied.
- Present—Interface where a policy is attached but not applied.
- Not present—Interface where no policy is attached.
- Present or not—Interface where a policy is either attached or not, but not applied.

Table 2-6 QoS Policy Interfaces

Port Policy	Port-Channel Policy	VLAN Policy
Applied	Not present	Present or not
Present or not	Applied	Present or not
Not present	Not present	Applied

To attach a policy map to an interface, use the **service-policy** interface command mode or the VLAN command mode. You specify whether the policies defined in the policy map are applied to the input or output stream of packets on the interface.

To detach a policy map from an interface or VLAN, use the **no** form of the **service-policy** interface command mode or the VLAN command mode.

SUMMARY STEPS

1. **config t**
2. **interface** {[ethernet slot/port] | [port-channel channel-number] | [tunnel number]}
or
vlan [vlan-id]
3. **service-policy** [type qos] {input | output} {policy-map-name | qos-dynamic} [no-stats]
4. **show policy-map** [vlan vlan_id] [input | output] [type qos | queuing] [class [type qos | queuing] class-map-name]
5. **copy running-config startup-config**

Send document comments to nexus7k-docfeedback@cisco.com.

DETAILED STEPS

	Command	Purpose
Step 1	config t Example: switch# config t switch(config)#	Enters configuration mode.
Step 2	interface {[ethernet <i>slot/port</i>] [port-channel <i>channel-number</i>] [tunnel <i>number</i>]} Example 1: switch(config)# interface ethernet 1/1 switch(config-if)# Example 2: switch(config)# vlan 2	Enters interface mode on the interface or a specified VLAN.
Step 3	service-policy [type qos] { input output } { <i>policy-map-name</i> qos-dynamic } [no-stats] Example: switch(config-if)# service-policy input policy1 switch(config-if)#	Adds the policy map to the input or output packets of an interface or VLAN. Only one input policy and one output policy can be attached to an interface or VLAN. This example adds policy1 to the input interface.
Step 4	show policy-map [vlan <i>vlan-id</i>] [input output] [type qos queuing] [class [type qos queuing] <i>class-map-name</i>] Example: switch(config)# show policy-map vlan 1 input type qos	(Optional) Displays information about policy maps applied to all interfaces or the specified interface. You can limit what the device displays to input or output policies, qos or queuing polices, and to a specific class. This example shows all policy maps on the Ethernet 1/1 interface.
Step 5	copy running-config startup-config Example: switch(config)# copy running-config startup-config	(Optional) Saves the running configuration to the startup configuration.

Session Manager Support for QoS

Beginning in Cisco NX-OS Release 4.2, Session Manger supports the configuration of QoS. This feature allows you to verify the QoS configuration and confirm that the resources required by the configuration are available prior to committing them to the running configuration. See the *Cisco Nexus 7000 Series NX-OS System Management Configuration Guide, Release 4.2* for information about Session Manager.

Once you start the configuration session, do not enter any configuration commands using the configure terminal configuration mode until the configuration session is aborted or committed. Entering parallel configurations (one using the configuration session and another using the configuration terminal configuration mode) may cause verification failures in the configuration session mode.

Send document comments to nexus7k-docfeedback@cisco.com.

Feature History for Using Modular QoS CLI

Table 2-7 lists the release history for this feature.

Table 2-7 ***Feature History for Modular QoS CLI***

Feature Name	Releases	Feature Information
No change	4.1(2)	-
Support for Session Manager	4.2(1)	Allows you to verify the configuration and required resources prior to committing them to the running configuration.