



## Overview

---

This chapter contains the following sections:

- [Information About Quality of Service, on page 1](#)
- [Modular QoS CLI, on page 1](#)
- [QoS for Traffic Directed to the CPU , on page 2](#)

## Information About Quality of Service

The configurable Cisco NX-OS quality of service (QoS) features allow you to classify the network traffic, prioritize the traffic flow, and provide congestion avoidance.

The default QoS configuration on the device provides best-effort service for Ethernet traffic. QoS can be configured to provide additional classes of service for Ethernet traffic. Cisco NX-OS QoS features are configured using Cisco Modular QoS CLI (MQC).

In the event of congestion or collisions, Ethernet will drop packets. The higher level protocols detect the missing data and retransmit the dropped packets.

## Modular QoS CLI

The Cisco Modular QoS CLI (MQC) provides a standard set of commands for configuring QoS.

You can use MQC to define additional traffic classes and to configure QoS policies for the whole system and for individual interfaces. Configuring a QoS policy with MQC consists of the following steps:

1. Define traffic classes.
2. Associate policies and actions with each traffic class.
3. Attach policies to logical or physical interfaces as well as at the global system level.

MQC provides two 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.

The class map classifies incoming packets based on matching criteria, such as the IEEE 802.1p class of service (CoS) value. Unicast and multicast packets are classified.

**policy-map**

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

The policy map defines a set of actions to take on the associated traffic class, such as limiting the bandwidth or dropping packets.

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

**network-qos**

Defines MQC objects that you can use for system level related actions.

**qos**

Defines MQC objects that you can use for classification.

**queuing**

Defines MQC objects that you can use for queuing and scheduling.



---

**Note** The **qos** type is the default for the **class-map** and **policy-map** commands, but not for the **service-policy** which requires that you specify an explicit type.

---

You can attach policies to interfaces or EtherChannels as well as at the global system level by using the **service-policy** command.

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

An MQC target is an entity (such as an Ethernet interface) that represents a flow of packets. A service policy associates a policy map with an MQC target and specifies whether to apply the policy on incoming or outgoing packets. This mapping enables the configuration of QoS policies such as marking, bandwidth allocation, buffer allocation, and so on.

## QoS for Traffic Directed to the CPU

The device automatically applies QoS policies to traffic that is directed to the CPU to ensure that the CPU is not flooded with packets. Control traffic, such as bridge protocol data units (BPDU) frames, is given higher priority to ensure delivery.