

Ascertain Bandwidth and Priority Commands of a QoS Service Policy

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Components Used](#)

[Conventions](#)

[Background Information](#)

[Summary of Differences](#)

[Configure the Bandwidth Command](#)

[Configure the Priority Command](#)

[Which Traffic Classes Can Use Excess Bandwidth?](#)

[Unused Bandwidth Allocation](#)

[Use Police Command to Set a Maximum](#)

[Understand the Available Bandwidth Value](#)

[Related Information](#)

Introduction

This document describes how the `bandwidth` and `priority` commands are applied in a modular quality of service command-line interface policy-map.

Prerequisites

Requirements

There are no specific requirements for this document.

Components Used

This document is not restricted to specific software and hardware versions.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Conventions

For more information on document conventions, refer to the [Cisco Technical Tips Conventions](#).

Background Information

The **bandwidth** and **priority** commands both define actions that can be applied within a modular quality of service command-line interface (MQC) policy-map, which you then apply to an interface, subinterface, or virtual circuit (VC) via the `service-policy` command. Specifically, these commands provide a bandwidth guarantee to the packets which match the criteria of a traffic class. However, the two commands have important functional differences in those guarantees. This Tech Note explains those differences and explains how the unused bandwidth of a class is distributed to flows that match other classes.

Summary of Differences

This table lists the functional differences between the `bandwidth` and `priority` commands:

Function	<code>bandwidthCommand</code>	<code>priorityCommand</code>
Minimum bandwidth guarantee	Yes	Yes
Maximum bandwidth guarantee	No	Yes
Built-in policer	No	Yes
Provides low latency	No	Yes

In addition, the `bandwidth` and **priority** commands are designed to meet different quality of service (QoS) policy objectives. This table lists those different objectives:

Application	<code>bandwidthCommand</code>	<code>priorityCommand</code>
Bandwidth management for WAN links	Yes	Somewhat
Manage delay and variations in delay (jitter)	No	Yes
Improve application response time	No	Yes

Even with fast interfaces, most networks still need a strong QoS management model to deal effectively with the congestion points and bottlenecks that inevitably occur due to speed-mismatch or diverse traffic patterns. Real world networks have limited resources and resources bottlenecks and need QoS policies to ensure proper resource allocation.

Configure the Bandwidth Command

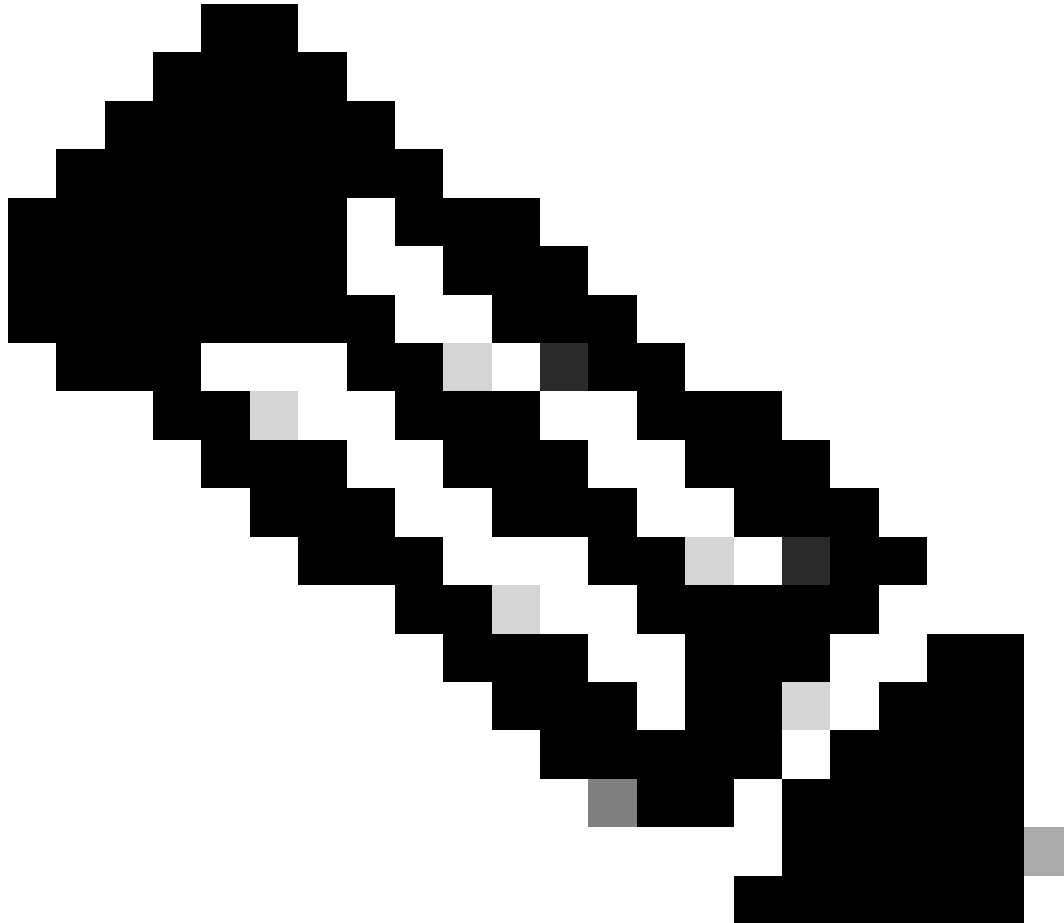
The Cisco IOS[®] Configuration Guides describe the `bandwidth` command as the "amount of bandwidth, in kbps, to be assigned to the class. . . .to specify or modify the bandwidth allocated for a class that belongs to a policy map."

Look at what these definitions mean.

The `bandwidth` command provides a *minimum* bandwidth guarantee during congestion. There are three forms of the command syntax, as illustrated in this table:

Command Syntax	Description
<pre><#root> bandwidth {kbps}</pre>	Specifies bandwidth allocation as a bit rate.
<pre><#root></pre>	Specifies bandwidth allocation as a percentage of the principal link rate.

bandwidth percent {value}	
<#root> bandwidth remaining percent {value}	Specifies bandwidth allocation as a percentage of the bandwidth that has not been allocated to other classes.



Note: The bandwidth command defines a behavior, which is a minimum bandwidth guarantee. Not all Cisco router platforms use `weighted-fair queueing` (WFQ) as the principal algorithm to implement this behavior. For more information, refer to [Why Use CBWFQ?](#)

Configure the Priority Command

The Cisco IOS Configuration Guides describe the priority command as a reserve for "a priority queue with a specified amount of available bandwidth for CBWFQ traffic...to give priority to a traffic class based on the amount of available bandwidth within a traffic policy." The next example explains what these definitions mean.

You create a priority queue with these sets of commands:

```
<#root>

Router(config)#
policy-map policy-name

Router(config-pmap)#
class class-name

Router(config-pmap-c)#
priority kpbs [bytes]
```

During congestion conditions, the traffic class is guaranteed bandwidth equal to the specified rate. (Recall that bandwidth guarantees are only an issue when an interface is congested.) In other words, the `priority` command provides a minimum bandwidth guarantee.

In addition, the `priority` command implements a maximum bandwidth guarantee. Internally, the priority queue uses a token bucket that measures the offered load and ensures that the traffic stream conforms to the configured rate. Only traffic that conforms to the token bucket is guaranteed low latency. Any excess traffic is sent if the link is not congested or is dropped if the link is congested. For more information, refer to [What Is a Token Bucket?](#)

The purpose of the built-in policer is to ensure that the other queues are serviced by the queueing scheduler. In the original Cisco priority queueing feature, which uses the `priority-group` and `priority-list` commands, the scheduler always serviced the highest priority queue first. In extreme cases, the lower priority queues rarely were serviced and effectively were starved of bandwidth.

The real benefit of the `priority` command—and its major difference from the `bandwidth` command—is how it provides a strict de-queueing priority to provide a bound on latency. Here is how the Cisco IOS Configuration Guide describes this benefit: "A strict priority queue (PQ) allows delay-sensitive data such as voice to be de-queued and sent before packets in other queues are de-queued." Look at what this means.

Every router interface maintains these two sets of queues:

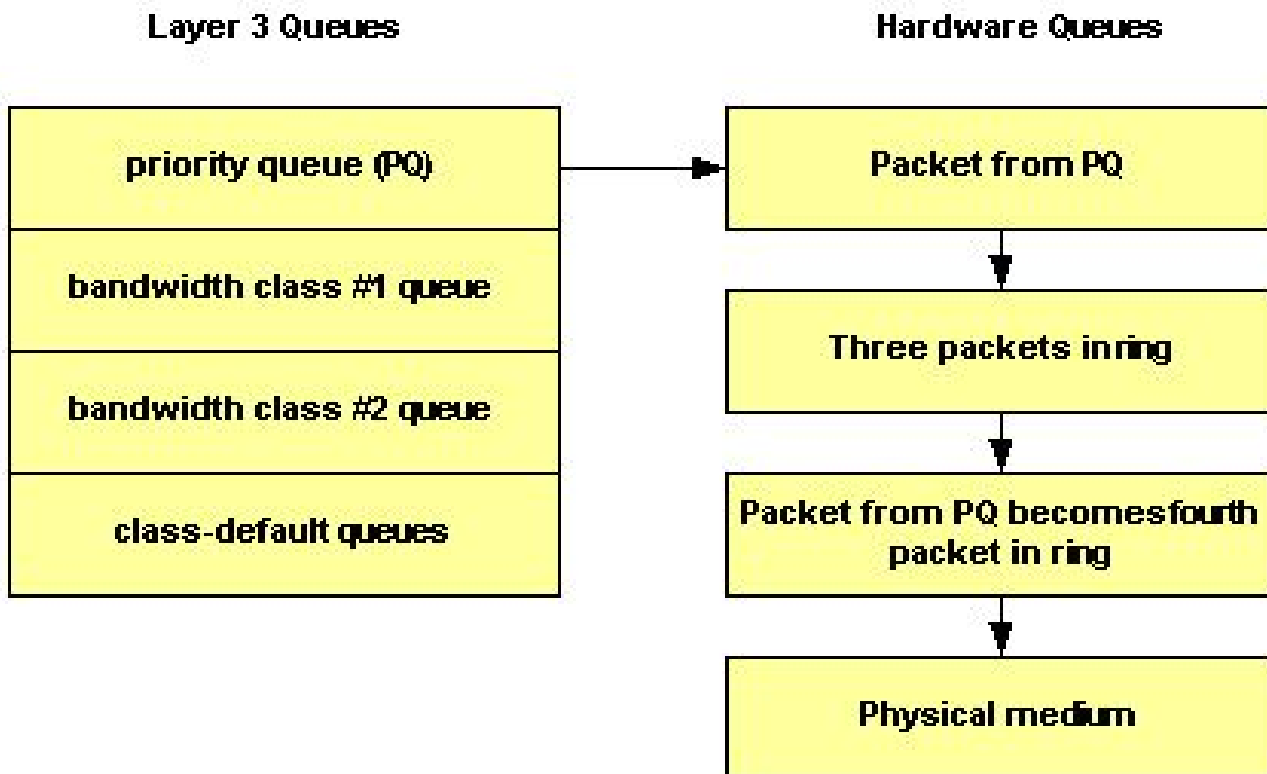
Queue	Location	Queueing Methods	Service Policies Apply	Command to Tune
Hardware Queue or transmit ring	Port adapter or network module	FIFO only	No	tx-ring-limit
Layer 3 Queue	Layer 3 processor system or interface buffers	Flow-based WFQ, CBWFQ, LLQ	Yes	Varies with queueing method. Use the queue-limit command with a bandwidth class.

From the previous table, we can see that a service-policy applies only to packets in the Layer 3 queue.

Strict de-queueing refers to the queueing scheduler that services the priority queue and forwards its packets to the transmit ring first. The transmit ring is the last stop before the physical media.

In the next illustration, the transmit ring has been configured to hold four packets. If three packets already are on the ring, then at best we can queue to the fourth position and then wait for the other three to empty. Thus, the Low Latency Queueing (LLQ) mechanism simply de-queues the packets to the tail end of the

driver-level first-in, first-out (FIFO) queue on the transmit ring.



Use the `tx-ring-limit` command to tune the size of the transmit ring to a non-default value. Cisco recommends that you tune the transmit ring when you transmit voice traffic.

Traffic prioritization is especially important for delay-sensitive, interactive transaction-based applications. To minimize delay and jitter, the network devices must be able to service voice packets as soon as they arrive, or in other words, in strict priority fashion. Only a strict priority works well for voice. Unless the voice packets are immediately de-queued, each hop can introduce more delay.

The International Telecommunications Union (ITU) recommends a maximum 150-millisecond one-way end-to-end delay. Without immediate de-queueing at the router interface, a single router hop can account for most of this delay budget. For more information, refer to the [Voice Quality Support](#).



Note: With both commands, the kbps value must take the Layer 2 overhead into account. In other words, if a guarantee is made to a class, that guarantee is with respect to Layer 2 throughput.

Which Traffic Classes Can Use Excess Bandwidth?

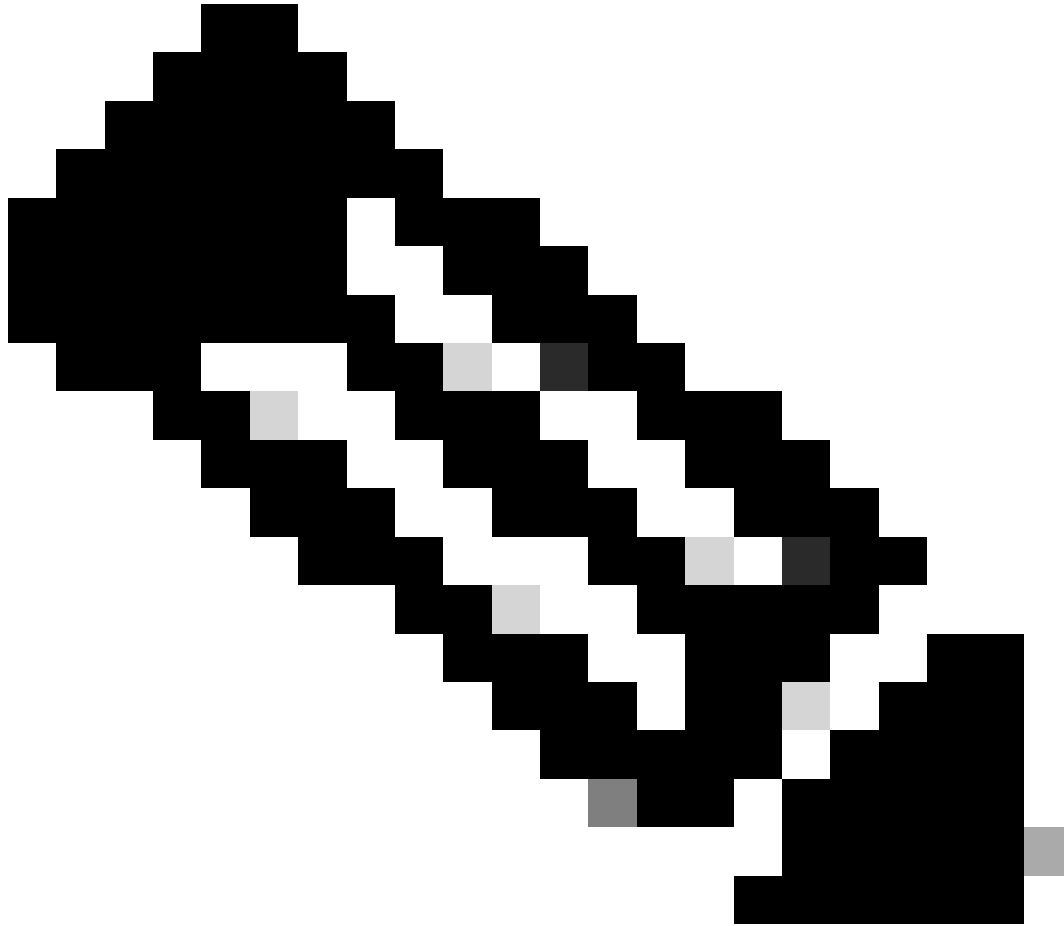
Although the bandwidth guarantees provided by the `bandwidth` and `priority` commands have been described with words like "reserved" and "bandwidth to be set aside", neither command implements a true reservation. In other words, if a traffic class does not use its configured bandwidth, any unused bandwidth is shared among the other classes.

The queuing system imposes an important exception to this rule with a priority class. As noted previously, the offered load of a priority class is metered by a traffic policer. *During congestion conditions, a priority class cannot use any excess bandwidth.*

This table describes when a bandwidth class and a priority class can use excess bandwidth:

Command	Congestion	Non-Congestion
<code>bandwidth</code> Command	Allowed to exceed the allocated rate.	Allowed to exceed the allocated rate.

priority Command	Cisco IOS meters the packets and applies a traffic measurement system via a token bucket. Packets that match are policed to the configured bps rate, and any excess packets are discarded.	The class can exceed its configured bandwidth.
-------------------------	--	--



Note: An exception to these guidelines for LLQ is Frame Relay on the Cisco 7200 router and other non-Route/Switch Processor (RSP) platforms. The original implementation of LLQ over Frame Relay on these platforms did not allow the priority classes to exceed the configured rate during periods of non-congestion. Cisco IOS Software Release 12.2 removes this exception and ensures that non-conforming packets are only dropped if there is congestion. In addition, packets smaller than an FRF.12 fragmentation size are no longer sent through the fragmenting process, and this reduces CPU utilization.

From the previous discussion, it is important to understand that since the priority classes are policed during congestion conditions, they are not allocated any remaining bandwidth from the bandwidth classes. Thus, *remaining bandwidth is shared by all bandwidth classes and class-default.*

Unused Bandwidth Allocation

This section explains how the queueing system distributes any remaining bandwidth. Here is how the [Class-Based Weighted Fair Queueing Feature Overview](#) describes the allocation mechanism: "If excess bandwidth is available, the excess bandwidth is divided amongst the traffic classes in proportion to their configured bandwidths. If not all of the bandwidth is allocated, the remaining bandwidth is proportionally allocated among the classes, based on their configured bandwidth." Look at two examples.

In the first example, `policy-map foo` guarantees 30 percent of the bandwidth to class bar and 60 percent of the bandwidth to class baz.

```
policy-map foo
  class bar
    bandwidth percent 30
  class baz
    bandwidth percent 60
```

If you apply this policy to a 1 Mbps link, it means that 300 kbps is guaranteed to class bar, and 600 kbps is guaranteed to class baz. Importantly, 100 kbps is leftover for class-default. If class-default does not need it, the unused 100 kbps is available for use by class bar and class baz. If both classes need the bandwidth, they share it in proportion to the configured rates. In this configuration, the ratio is shared 30:60 or 1:2.

The next sample configuration contains three policy maps—bar, baz, and poli. In the policy map called bar and the policy map called baz, the bandwidth is specified by percentage. However, in the policy map called poli, bandwidth is specified in kbps.

Remember that the class maps must already be created before you create the policy maps.

```
policy-map bar
  class voice
    priority percent 10
  class data
    bandwidth percent 30
  class video
    bandwidth percent 20
policy-map baz
  class voice
    priority percent 10
  class data
    bandwidth remaining percent 30
  class video
    bandwidth remaining percent 20
policy-map poli
  class voice
  class data
    bandwidth 30
  class video
    bandwidth 20
```



Note: The bandwidth remaining percent command was introduced in Cisco IOS version 12.2(T).

Use Police Command to Set a Maximum

If a bandwidth or priority class must not exceed its allocated bandwidth during periods of no congestion, you can combine the `priority` command with the `police` command. This configuration imposes a maximum rate that is always active on the class. The choice to configure a `police` statement in this configuration depends on the policy objective.

Understand the Available Bandwidth Value

This section explains how the queueing system derives the Available Bandwidth value, as displayed in the output of the `show interface` or `show queueing` commands.

We created this policy-map named Leslie:

```
<#root>
```

```
7200-16#
```

```
show policy-map leslie
```

```
Policy Map leslie
  Class voice
    Weighted Fair Queueing
      Strict Priority
      Bandwidth 1000 (kbps) Burst 25000 (Bytes)
  Class data
    Weighted Fair Queueing
      Bandwidth 2000 (kbps) Max Threshold 64 (packets)
```

We then created an ATM permanent virtual circuit (PVC), assigned it to the variable bit rate non-real-time ATM service category, and configured a sustained cell rate of 6 Mbps. We then applied the policy-map to the PVC with the `service-policy output leslie` command.

```
<#root>
```

```
7200-16(config)#
```

```
interface atm 4/0.10 point
```

```
7200-16(config-subif)#
```

```
pvc 0/101
```

```
7200-16(config-if-atm-vc)#
```

```
vbr-nrt 6000 6000
```

```
7200-16(config-if-atm-vc)#
```

```
service-policy output leslie
```

The `show queueing interface atm` command displays **Available Bandwidth 1500 kilobits/sec**.

```
<#root>
```

```
7200-16#
```

```
show queue interface atm 4/0.10
```

```
Interface ATM4/0.10 VC 0/101
queue strategy: weighted fair
Output queue: 0/512/64/0 (size/max total/threshold/drops)
  Conversations 0/0/128 (active/max active/max total)
  Reserved Conversations 1/1 (allocated/max allocated)
  Available Bandwidth 1500 kilobits/sec
```

Let us look at how this value is derived:

1. 6 Mbps is the sustained cell rate (SCR). By default, 75 percent of this is reservable:

$$0.75 * 6000000 = 4500000$$

2. 3000 kbps is already used by the voice and data classes:

$$4500000 - 3000000 = 1500000 \text{ bps}$$

3. Available bandwidth is 1500000 bps.

The default maximum reservable bandwidth value of 75 percent is designed to leave sufficient bandwidth for overhead traffic, such as routing protocol updates and Layer 2 keepalives. It also covers Layer 2 overhead for packets that match and are defined traffic classes or the class-default class. You now can increase the maximum reservable bandwidth value on ATM PVCs with the `max-reserved-bandwidth` command. For supported Cisco IOS releases and further background information, refer to [Understand the max-reserved-bandwidth Command on ATM PVC](#).

On Frame Relay PVCs, the `bandwidth` and `priority` commands calculate the total amount of available bandwidth in one of these ways:

- If a Minimum Acceptable Committed Information Rate (minCIR) is not configured, the CIR is divided by two.
- If a minCIR is configured, the minCIR setting is used in the calculation. The full bandwidth from the previous rate can be assigned to bandwidth and priority classes.

Thus, the `max-reserved-bandwidth` command is not supported on Frame Relay PVCs, although you must ensure that the amount of bandwidth configured is large enough to accommodate Layer 2 overhead. For more information, refer to [Configure CBWFQ on Frame Relay PVCs](#).

Related Information

- [Cisco Technical Support & Downloads](#)