Hierarchical Queuing Framework

Last updated: July 2008

Summary

This document describes the new behavioral changes and Command Line Interface (CLI) modifications to the queuing infrastructure on non-distributed hardware (Cisco 7200 Series Router and other lower-end platforms). The changes being applied to Cisco IOS Software Release 12.4(20)T will provide a consistent queuing behavior using the same Modular Quality of Service CLI (MQC) across all Cisco IOS Software platforms. These changes are currently present in the latest versions of Cisco IOS Software Releases 12.0S and 12.2S.

Document Purpose

The purpose of this document is to highlight the new queuing changes for customers who are upgrading to the latest T release, starting with Cisco IOS Software Release 12.4(20)T. This document does not intend to discuss all of the new MQC feature capabilities that have been introduced to the Cisco 7200 Series Router and the lower-end platforms. For a complete list of the Quality of Service (QoS) feature capabilities, refer to the appropriate Cisco IOS Software release documentation on Cisco.com.

Benefits

With the deployment of Hierarchical Queuing Framework (HQF) in the Release 12.4(20)T train, we are bringing together the queuing and shaping mechanisms that are currently used in Cisco IOS Software Releases 12.0S and 12.2S. With this migration, our customers benefit in several different ways:

- Consistent queuing behavior applied with common MQC across all main Cisco IOS Software releases.
- Common functionality for both distributed and non-distributed implementations, providing consistency of QoS feature behavior across all software-forwarding hardware, making implementation of QoS easier and transparent regardless of Cisco IOS Software Release is being used.
- With HQF customers using any of the IOS releases will have:
  - The ability to provide multiple levels of packet scheduling
  - The ability to support integrated class-based shaping and queuing
  - The ability to apply fair queuing and drop policies on a per-class basis

New QoS Feature Functionality

The following HQF features are new in Cisco IOS Software Release 12.4(20)T:

Hierarchical Policy with Queuing Features at Every Level

With this feature you can apply class-based queuing to any traffic class in the parent or child level of a hierarchical policy and obtain different service levels for different sessions or subscribers. In the example shown below, the traffic belonging to class parent-c2 will have more scheduling time than class parent-c1.

```
policy-map child
       class child-c1
              bandwidth 400
       class child-c2
              bandwidth 400
policy-map parent
       class parent-c1
              bandwidth 1000
              service-policy child
       class parent-c2
              bandwidth 2000
              service-policy child
```

Fair Queue in an MQC Class

This feature provides a fair-share of the bandwidth among the flows within any traffic class that has fair queuing enabled. You can apply the **fair-queue** command to a user-defined class as shown in the following example:

```
policy-map p1
       class c1
              bandwidth 1000
              fair-queue
```

Shaping in an ATM PVC Policy

The shape feature is now supported at the ATM PVC level through a service policy. Prior to applying the service policy to the PVC, a service category such as vbr-nrt has to be enabled on the PVC. You can then apply a class-based shaping within an ATM PVC as shown in the following example:

```
policy-map p1
       class c1ass-default
              shape average 1000000
           service-policy p2
policy-map p2
       class ef
```

```
            priority 1000
      class prec3
          bandwidth percent 40
      class prec1
         bandwidth percent 25
interface atm1/0.1
      pvc 1/100
            vbr-nrt 2000 2000
            service-policy output p1
```

Strict Priority with No Policing Rate

Only one class is allowed strict priority configuration. Other classes cannot have priority or bandwidth configuration. If minimum bandwidth is required by one of the other classes, the **bandwidth remaining percent** command must be used, as shown in the following example:

```
policy-map p1
      class c1
            priority
      class c2
            bandwidth remaining percent 20
```

Priority with Explicit Policing Rate

When a priority class is configured with an explicit policer, traffic is limited to the policer rate regardless of congestion conditions. In other words, even if bandwidth is available the priority traffic will not be able to exceed the rate specified with the explicit policer.

```
policy-map p1
    class c1
          priority
          police cir 1000000 conform-action transmit exceed-action drop
```

Random-detect Support in Class-default

The **random-detect** command can be configured for class-default to calculate the probability of dropping a packet. An example of applying random-detect to class-default traffic based on precedence bits is shown below.

```
policy-map p1
      class class-default
            random-detect precedence-based
            random-detect precedence 0  40  80
```

Random-detect Options and Thresholds Support

The **random-detect** command supports the **atm-clp-based**, and **cos-based** options to calculate the probability of dropping a packet.

random-detect atm-clp-based Command

```
policy-map p1
       class c1
              bandwidth 1000
              random-detect atm-clp-based
              random-detect clp 0 <min> <max> <mark-probability>
```

random-detect cos-based Command

```
policy-map p1
       class c1
              bandwidth 1000
              random-detect cos-based
              random-detect cos 0 <min> <max> <mark-probability>
```

The threshold settings for the different random-detect options can be set in terms of bytes or milliseconds.

random-detect thresholds set in bytes

```
policy-map p1
       class c1
              bandwidth 1000
              random-detect precedence-based
              random-detect precedence 0 100 bytes 400 bytes 100
```

random-detect thresholds set in milliseconds

```
policy-map p1
       class c1
              bandwidth 1000
              random-detect precedence-based
              random-detect precedence 0 200 ms 800 ms 100
```

Queue-limit Support in Bytes or ms

The **queue-limit** command can also be set in units of bytes or ms, in addition to its default units of packets.

```
policy-map p1
       class c1
              bandwidth 1000
           queue-limit 1000 bytes
       class c2
           bandwidth 1000
           queue-limit 500 bytes
```

QoS Behavioral Changes

With the migration of HQF into Cisco IOS Software Release 12.4(20)T, the following behavioral changes occur for some of the QoS features currently available in the T train.

Changes Related to Class-Default

When you do not explicitly configure the class-default class in a policy map, its default queuing behavior is FIFO. You can configure the **bandwidth**, **fair-queue**, or **service-policy** commands in the class-default class to achieve different queuing behaviors.

When fair-queue is applied to class-default, the behavior is that of flow-based. This is a change from the Weighted Fair Queuing (WFQ) behavior in previous releases. With flow-based fair queuing, the flow queues in the class-default class are scheduled equally instead of by weight based on the IP Precedence bits.

The bandwidth assigned to the class-default class is the unused interface bandwidth not consumed by user-defined classes. By default, the class-default class receives a minimum of 1% of the interface or parent shape bandwidth. In the example below, when the following policy-map is attached to a 10Mbps interface,

```
policy-map foo
       class c1
              priority 2000
       class c2
              bandwidth 4000
```

class-default will get the remaining bandwidth guarantee of 4Mbps (10 – 2 – 4). In the event that less than 4Mbps of class-default traffic were present, class c1 and class c2 will evenly share the available bandwidth not used by class-default.

The bandwidth command may be configured in class-default to explicitly assign a different bandwidth ratio.

Default Queuing Implementation for the Shape Feature

When you configure the **shape** command in a class, the default queuing behavior for the shape queue is FIFO instead of WFQ. You can configure the **bandwidth**, **fair-queue**, or **service-policy** commands in shape class to achieve different queuing behaviors.

Policy Map and Interface Bandwidth

In HQF, a policy map can reserve up to 100% of the interface bandwidth. Up to a maximum of 99% of the interface bandwidth can be assigned to **user-defined** classes as by default 1% of the bandwidth is reserved for the class-default class.

Note that when migrating to Release 12.4(20)T , if the configured policy map allocates 100% of the bandwidth to the user-defined classes, an error message will appear in the console after booting the HQF image. The message will indicate that the allocated bandwidth exceeds the allowable amount and the service policy will be rejected. In HQF, the policy map has to be re-configured to account for the minimum 1% bandwidth guaranteed for the class-default. The service policy can then be applied to the interface.

Per-Flow Queue Limit in Fair Queue

In HQF, when you enable fair queuing, the default per-flow queue limit is ¼ of the class queue limit. If you do not enable the queue limit in a class, the default per-flow queue limit is 16 packets (1/4 of 64).

Over-Subscription Support for Multiple Policies on Logical Interfaces

When you attach a shaping policy to multiple logical interfaces including a subinterface, and the sum of shape rate exceeds the physical interface bandwidth, congestion at the physical interface results in backpressure to each logical interface policy. This backpressure causes each policy to reduce the output rate down to its fair share of the interface bandwidth.

Example: 10 subinterface policies each shaped to 2Mbps, physical interface has 10Mbps bandwidth (2:1 oversubscription), when all 10 subinterfaces are sending at 2Mbps, each subinterface gets a throughput of 1Mbps (10 Mbps / 10 subinterfaces).

Shaping Behavior on GRE Tunnel

In HQF, the shape feature can be applied to a GRE tunnel via a hierarchical service policy. Shaping on GRE tunnel will be applied after encapsulation. This means the shape rate is based on packets with tunnel encapsulation and L2 encapsulation.

Shape is the only feature allowed in the service policy applied to a tunnel interface. When configuring the shape feature in the parent policy is applied to the tunnel interface, only the class-default class is permitted. Configuring a user-defined class in the parent policy is not allowed. A typical hierarchical policy applied to a GRE tunnel interface is shown below.

```
Interface tunnel0
Service-policy output parent
policy-map parent
       class class-default
              shape average 10000000
              service-policy child
policy-map child
       class voice
              priority 512
       class video
              bandwidth 6000
       class data
              bandwidth 3000
```

Currently, certain QoS deployments include a service policy with queuing features applied at the tunnel or a virtual interface, and a service policy with queuing features applied at the physical interface. In Release 12.4(20)T, a service policy with queuing features can only be supported at one of these interfaces. When migrating to Release 12.4(20)T, a router configuration containing service policies at both interfaces will only keep the one applied to the physical interface.

Change in FRF.12 and FRF.9 Behavior

With HQF implementation, when you enable Frame Relay Fragmentation (FRF.12) on an FR PVC or FR main interface, priority class packets are no longer subject to fragmentation. Priority packets, regardless of the packet size, always interleave among data fragments.

When you enable Frame Relay payload compression (FRF.9) on an FR PVC or main interface, priority class packets are no longer compressed. When you enable both FRF.12 and FRF.9, priority class packets are neither fragmented nor compressed.

Changes in CLI

Some CLI commands are being changed with the integration of HQF.

random-detect prec-based Command

The **random-detect prec-based** command within a policy-map has been replaced by the **random-detect precedence-based** command. The function of the command does not change.

shape max-buffers Command

The **shape max-buffers** command currently configured under a class in a policy-map will no longer be supported in HQF. It will be replaced by the **queue-limit** command, which provides similar functionality. Upon migration to an IOS release which integrates HQF, a router configured with the shape max-buffer command will be automatically configured with the queue-limit command.

max-reserved-bandwidth Command

The **max-reserved-bandwidth** command no longer affects the amount of bandwidth available to a service policy. Any policy-map can allocate up to 100% of the bandwidth without the need of the max-reserved-bandwidth command. The max-reserved-bandwidth command was used in previous IOS releases in order to overcome the restriction of allocating 75% of the bandwidth to user-defined classes. In HQF, that restriction does not exist anymore.

Show Command Changes

show queuing and show queue Commands

The **show queuing** and **show queue** commands are no longer supported. Instead, you can use the **show policy-map** and **show policy-map interface** commands to gather QoS-related information and statistics.

The revised **show policy-map interface** output has additional fields that display the DSCP value, WRED statistics in bytes, new column for number of transmitted packets by WRED, and a counter displaying packets output/bytes output in each class. The **packets queued/bytes queued** counter is no longer supported. Table 1 compares the old and new output formats for both commands. The outputs have been lined up for easier comparison and to see what is missing between them.

Table 1. Comparison of Old and New Command Output Formats

Old Output Format	New Output Format

Old Output Format	New Output Format
1841#sh policy-map test1 Policy Map test1 Class class-default Traffic Shaping Average Rate Traffic Shaping CIR 1536000 (bps) **Max. Buffers Limit 200 (Packets)** service-policy test2 1841# 1841#sh policy-map test2 Policy Map test2 Class RT **Strict Priority** **Bandwidth 20 (%)** Class BH Bandwidth 40 (%) **Max Threshold 320 (packets)** Class BL Bandwidth 20 (%) exponential weight 9 class min-threshold max-threshold mark-probablity ---------- 0 - - 1/10 1 - - 1/10 2 - - 1/10 3 - - 1/10 4 - - 1/10 5 - - 1/10 6 - - 1/10 7 - - 1/10 rsvp - - 1/10 1841# 1841#sh policy-map inter f0/0 FastEthernet0/0 Service-policy output: test1 Class-map: class-default (match-any) 135 packets, 14272 bytes 30 second offered rate 0 bps, drop rate 0 bps Match: any Traffic Shaping Target/Average Byte Sustain Excess Interval Increment Rate Limit bits/int bits/int (ms) (bytes) 1536000/1536000 9600 38400 38400 25 4800 Adapt Queue Packets Bytes Packets Bytes Shaping Active Depth Delayed Delayed Active - 0 135 14272 0 0 no Service-policy : test2 Class map: RT (match all) 0 packets, 0 bytes 30 second offered rate 0 bps, drop rate 0 bps Match: ip dscp ef (46)	1841#sh policy-map test1 Policy Map test1 Class class-default Average Rate Traffic Shaping cir 1536000 (bps) **queue-limit 200 packets** service-policy test2 1841# 1841#sh policy-map test2 Policy Map test2 Class RT **priority 20 (%)** Class BH bandwidth 40 (%) **queue-limit 320 packets** Class BL bandwidth 20 (%) packet-based wred, exponential weight 9 class min-threshold max-threshold mark-probablity ---------- 0 - - 1/10 1 - - 1/10 2 - - 1/10 3 - - 1/10 4 - - 1/10 5 - - 1/10 6 - - 1/10 7 - - 1/10 1841# 1841#sh policy-map inter f0/0 FastEthernet0/0 Service-policy output: test1 Class-map: class-default (match-any) 48479 packets, 4737065 bytes 30 second offered rate 0 bps, drop rate 0 bps Match: any Queueing **queue limit 200 packets** (queue depth/total drops/no-buffer drops) 0/0/0 (pkts output/bytes output) 48479/4737065 shape (average) cir 1536000, bc 6144, be 6144 target shape rate 1536000 Service-policy : test2 queue stats for all priority classes: queue limit 64 packets (queue depth/total drops/no-buffer drops) 0/0/0 (pkts output/bytes output) 0/0 Class-map: RT (match-all) 0 packets, 0 bytes 30 second offered rate 0 bps, drop rate 0 bps Match: ip dscp ef (46)

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV
Amsterdam, The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at **www.cisco.com/go/offices.**

Printed in USA C11-481449-00 06/08