



# QoS for the Cisco 6500 (Revisited)

Peter J. Welcher and Carole Warner Reece

## Introduction

Interest in QoS has been picking up lately. This may reflect the increasing numbers of IP phone deployments. Also the fact that folks have gotten their networks upgraded or other projects completed, and are ready to attempt new things. In any case, it seems like it's time for another article on QoS!

Pete has written several prior articles and presentations about QoS. See his web page at <http://www.netcraftsmen.net/welcher/>. Some previous QoS articles:

QoS in the Campus	<a href="http://www.netcraftsmen.net/welcher/papers/qoscampus.html">http://www.netcraftsmen.net/welcher/papers/qoscampus.html</a>
Configuring Campus QoS	<a href="http://www.netcraftsmen.net/welcher/papers/qoscampuscfg.html">http://www.netcraftsmen.net/welcher/papers/qoscampuscfg.html</a>
New Quality of Service Features in Cisco IOS 12.1	<a href="http://www.netcraftsmen.net/welcher/papers/newqos121.html">http://www.netcraftsmen.net/welcher/papers/newqos121.html</a>

The first of these somewhat addresses the statement, "I have lots of bandwidth to spare in my campus, so I don't need QoS there." Our answer to that is "You might need it, and until you look for drops, you won't know. Or things might work fine until you have a busy network day, when your Voice over IP (VoIP) call quality degrades intermittently." In short, QoS in the campus is about insurance -- insuring you've done your best to avoid dropping VoIP UDP packets.

The second of these documents talks about what you can do in the 6500. Since then, things have changed. Among them, Pete's perspective on 6500 QoS has changed.

The last of the above articles covers the Modular QoS CLI (Command Line Interface), or MQC. That's a hierarchical acronym (one acronym nested within another)!

## QoS for 6500 -- The Big Picture

If you're used to the Modular QoS CLI (MQC) and Class-Based Weighted Fair Queuing (CBWFQ) for IOS, you'll probably happily set out to configure your MSFC1 or MSFC2 or Sup720. You may or may not even end up thinking you've got QoS working. There's a gotcha here.

The 6500 does QoS in three places:

- 1 Software-based in the MSFC
- 1 Hardware-based (multi-layer switching-based) in the PFC
- 1 Software-based on some line cards

This trichotomy is more obvious when you're working with hybrid IOS (CatOS + IOS for MSFC). The CatOS-brain / IOS-brain duality is pretty obvious when you've got two sets of configuration commands! But when you configure QoS under native IOS, for example with the newer Sup32 or Sup720 engines, you're further from the hardware and so you may not realize what's going on. The line card part is less conspicuous, because it is invisible to the user.

Why do you need to know this? This is important because most of the traffic is multi-layer switched ("hardware switched"), and therefore handled by the PFC logic. The MSFC never even sees that traffic. So if you don't set up PFC-based QoS, you're missing most of the traffic that matters.

There's another factor to take into account. What you can configure for QoS depends on the type of interface you'll be applying the QoS policy to.

## Quick QoS Review

See <http://www.netcraftsmen.net/welcher/seminars/qos-seminar.pdf> for an overview with slides.

QoS is about getting various kinds of traffic to share a congested link while receiving the sort of treatment they require. QoS is partly about insuring that applications (including Voice over IP, VoIP) receive what they need, even if congestion is only occasional and short-lived.

QoS uses markings in L2 or L3 headers to specify what sort of traffic the packet is, and, indirectly, how to handle it. These markings are usually set at the edge or "Trust Boundary". The process of doing this is known as classification, and may involve processing of access lists or other rules specifying what traffic gets which marking. Downstream devices can then use the markings to determine appropriate handling of the frame or packet.

Markings include Class of Service (CoS, Cisco's term for 3 bits in the L2 frame header), and DSCP or Differentiated Services Code Point, which is the high-order 6 bits in the TOS field of the IP header. IP Precedence ("IPP") is an older scheme, basically the lead 3 bits of DSCP. See the [above seminar](#) for diagrams of this.

The modular QoS CLI interface is the way to configure QoS on most Cisco devices. Lately, we have found that we describe and think about QoS in terms of the MQC operations. Unfortunately, only some of this carries over to the 6500. (There are actually three QoS command sets for the 6500: MQC which supports most QoS function on Cisco devices; MLS QoS which supports 6500 hardware specific QoS features; and port based CLI for some specific LAN interface queueing functions.) Nonetheless, we'll use MQC as our common starting point for our discussion.

MQC involves three sets of commands, based around the class-map, policy-map, and the service-policy.

You use class-map commands to tell the router or switch how to recognize (or "classify") various kinds of traffic. The class-map assigns a name to the class of traffic, for use in further configuring that one router or switch. Once you've set up classes, you use some of them to build a policy with the policy-map command. The policy lists the relevant classes, and tells the router or switch what to do with each class of traffic. The service policy then tells the router or switch which policy to use on which interface, and whether it is used for inbound or outbound traffic.

The rest of QoS to some extent boils down to what you can do to the various classes of traffic. The choices are:

- 1 marking or re-marking traffic: set CoS, IP Precedence, DSCP, or other header bits to simplify downstream classification
- 1 priority or LLQ (low latency queueing): provide priority to critical traffic, usually VoIP. The priority traffic is policed to protect against too much priority traffic causing the non-priority queues to be ignored ("queue starvation")
- 1 policing (inbound or outbound): drop or re-mark packets exceeding some bits per second rate
- 1 shaping (outbound only): queue packets to smooth traffic to a bits per second rate
- 1 bandwidth: provide guaranteed minimum bandwidth to a class of traffic

For examples of MQC, see the [above seminar](#).

Weighted Random Early Detection (WRED) is a queue-management technique that attempts to automatically avoid congestion by selective random drops. It works well on higher speed links with many flows. It is arguably rather preferable to the other alternative, tail drops. It should not be used with UDP traffic (which doesn't slow down in response to a drop) or with priority traffic (where the point is trying to avoid drops).

There are also additional operations you might use on slow WAN interfaces: Link Fragmentation and Interleaving (LFI), and header compression. There's a lot more to QoS, for example Call Admission Control, but the above should be enough to get you started. See also Pete's previous articles and seminars for more info. Our focus here needs to stay on the 6500, which is quite a broad topic in itself.

## PFC QoS

If you're trying to understand PFC QoS for the 6500, it may be helpful to start by reading the documentation for CatOS, even if you do native IOS. The advantage to doing this is when you read the native IOS documentation on "PFC QoS", you'll be able to see how the IOS commands map to the underlying PFC capabilities that are more directly visible via the CatOS CLI.

Since there's so much there, we'll have to focus on essentials. The Cisco configuration guides lately provide a lot of good information, but sometimes fail to distinguish key commands from "kerner knobs". For the PFC QoS, the following are some of the key things to consider:

- 1 Enabling MLS QoS
- 1 Inbound trust
- 1 Classification limitations
- 1 Internal DSCP value

First, you have to enable QoS, for example with the "**mls qos**" configuration command on Native IOS. When you do so, egress traffic is mapped to queues based on an internal DSCP value and any changes you have made to the default DSCP to queue assignments. Just doing this gives preferential treatment to voice traffic marked with IP precedence 5 or DSCP value EF (Expedited Forwarding). How many queues there are, and what types they are (strict priority or not) depends on the line card.

The internal DSCP value is based on your inbound trust settings and the line card type. We generally trust received DSCP values, which is true for OSM WAN ports and FlexWAN ports. 6500 LAN ports default to untrusted CoS, and untrusted DSCP. If the switch is an access switch at the trust boundary, we use inbound QoS policies to classify and mark traffic, which specifies the internal DSCP value. If you trust inbound CoS bits, the CoS bits are copied or mapped to the internal DSCP field associated with the frame or packet. Upon output, the internal DSCP settings are copied to the IP header. If the outbound frame is being given a trunking header (ISL or 802.1Q) then the appropriate CoS bits are also set.

If you trust CoS (only), you can get the benefit of some inbound queues and WRED-like behavior, which may help if the inbound port is experiencing traffic bursts.

When you are doing PFC QoS on LAN ports and on the OSM WAN ports, your classification ability is somewhat limited compared to full Cisco IOS. Filtering or MQC match criteria can only be:

- 1 Access lists
- 1 Match on IPP or DSCP value(s) or MPLS EXP

The outbound policy actions are also somewhat restricted. They depend on the interface or port. We cover this in the next section below. Other actions not familiar from MQC include trust. The inbound policy can specify whether CoS, IPP, or DSCP is trusted, or not. You can also specify inbound micro-policer (per-flow policing).

Inbound PFC QoS can be port-based or VLAN-based. If you mix the two, you'll be at risk of missing some inbound ports.

Mapping commands specify how to turn CoS or IPP into DSCP, or vice versa. The DSCP to CoS mapping is used, for example, to determine how to set outbound CoS bits based on the internal DSCP value.

## QoS Feature Support Table

The following table represents summary information gleaned from documentation and extensive lab testing by Carole. She put a lot of effort into this, and certainly has my thanks for doing so!

Disclaimer: this table reflects our best effort to provide you with accurate information. Bear in mind that some of this does not seem to be clearly documented by Cisco and so may be subject to change.

The Flex WAN capabilities are basically full IOS MQC capabilities for a high-speed WAN interface. The new Shared Interface Process (SIP) line cards have similar QoS capabilities as the FlexWAN, but we have not tested them.

QoS Function	LAN Ports	OSM WAN Ports (includes GE-WAN ports, but not GigabitEthernet ports)	Flex WAN
Classification	<p>Classify traffic based on CoS, or trusted values for IPP, DSCP or MPLS EXP, and by ACL.</p> <p>Starting w/ 12.2(18)SXE, can support re-marked versions with the command <b>"platform ip features sequential"</b></p>	<p>Classify IP traffic based on IPP, DSCP or MPLS EXP and by ACL. COS is not applicable.</p> <p>Starting w/ 12.2(18)SXE, can support re-marked versions with the command <b>"platform ip features sequential"</b></p>	<p>Classify traffic based on:</p> <ul style="list-style-type: none"> <li>1 ACL</li> <li>1 ATM Cell Loss Priority bit</li> <li>1 BGP index</li> <li>1 CoS</li> <li>1 DSCP</li> <li>1 Frame Relay DLCI value</li> <li>1 Frame-Relay DE bit</li> <li>1 IPP</li> <li>1 MAC-layer address</li> <li>1 MPLS EXP</li> <li>1 Packet length</li> <li>1 Protocols supported by NBAR</li> <li>1 VLAN ID</li> </ul>
Marking	<p>Can set DSCP, which includes IPP. Also EXP.</p> <p>CoS is set indirectly on output.</p> <p>Can also mark via policing:</p> <p><b>police zzz conform-action set-prec-transmit</b></p> <p><b>police zzz conform-action set-dscp-transmit</b></p> <p><b>police zzz conform-action set-mpls-exp-imposition-transmit</b></p>	<p>Can set DSCP, which includes IPP. Also EXP.</p> <p>CoS is not applicable.</p> <p>Can also mark via policing:</p> <p><b>police zzz conform-action set-prec-transmit</b></p> <p><b>police zzz conform-action set-dscp-transmit</b></p> <p><b>police zzz conform-action set-mpls-exp-imposition-transmit</b></p>	<p>Can set DSCP, which includes IPP. Also EXP.</p> <p>CoS is not applicable.</p> <p>Can also mark via policing:</p> <p><b>police zzz conform-action set-prec-transmit</b></p> <p><b>police zzz conform-action set-dscp-transmit</b></p> <p><b>police zzz conform-action set-mpls-exp-imposition-transmit</b></p>
Matching (or "Filtering")	<p>Supports single match command, either:</p> <p><b>match access-group</b></p> <p><b>match ip dscp</b></p> <p><b>match ip precedence</b></p> <p><b>match mpls experimental</b></p> <p>With 12.2(18)SXE and later, also supports</p>	<p>Supports:</p> <p><b>match access-group</b></p> <p><b>match ip dscp</b></p> <p><b>match ip precedence</b></p> <p><b>match mpls experimental</b></p> <p>With 12.2(18)SXE and later, also supports</p>	<p>Supports:</p> <p><b>match access-group</b></p> <p><b>match bgp-index</b></p> <p><b>match class-map</b></p> <p><b>match cos</b></p> <p><b>match dscp</b></p> <p><b>match destination-address</b></p>

	<p><b>match any</b></p> <p><b>match dscp</b></p> <p><b>match precedence</b></p> <p><b>match protocol ip</b></p> <p><b>match protocol ipv6</b></p> <p>Also supports multiple match commands in SXE release.</p>	<p><b>match any</b></p> <p><b>match vlan</b></p>	<p><b>match dscp</b></p> <p><b>match fr-dlci</b></p> <p><b>match ip dscp</b></p> <p><b>match ip precedence</b></p> <p><b>match mpls experimental</b></p> <p><b>match mpls</b></p> <p><b>match precedence</b></p> <p><b>match protocol</b></p> <p><b>match source-address</b></p>
Police	<p>Configure with the <b>police bps...</b> command</p> <p>Supported in inbound and outbound direction</p> <p>Supports all LAN port match types</p>	<p>Configure with the <b>police bps...</b> command</p> <p>Supported in inbound and outbound direction</p> <p>Supports all OSM WAN port match types</p>	<p>Configure with the <b>police bps...</b> command</p> <p>Supported in inbound and outbound direction</p> <p>Supports all FlexWAN port match types</p>
Trust	<p>Untrusted by default, CoS set to 0</p> <p>Can define the port CoS with the <b>mls qos cos port-cos</b> command</p>	<p>Trust DSCP by default</p> <p>Can configure to trust CoS, DSCP, or IPP (ingress CoS always set to 0)</p> <p>Can <u>NOT</u> define the port CoS with the <b>mls qos cos port-cos</b> command</p>	<p>Trust DSCP by default</p> <p>Can change DSCP by marking</p> <p>Can <u>NOT</u> define the port CoS with the <b>mls qos cos port-cos</b> command</p>
Shaping	<p>Not supported.</p> <p>Note: Can use “<b>shape</b>” command in an outbound policy, but this policy does not shape traffic!</p>	<p>Inbound and outbound shaping supported using “<b>shape average bps</b>” command.</p> <p>Does NOT support ACL for matching in class map, but supports other OSM WAN match types</p> <p>Note: Can NOT review data in “<b>show mls qos ip</b>” results.</p>	<p>Outbound shaping supported using the “<b>shape average bps</b>” and “<b>shape peak bps</b>” commands for outbound traffic.</p> <p>Cannot use “<b>shape</b>” command inbound</p>
Bandwidth or "CBWFQ"	<p>Not supported</p> <p>Note: can be applied to interface, no error message provided, but not shown in “<b>show mls qos ip</b>” listing.)</p>	<p>Configure with either</p> <p><b>bandwidth kbps</b></p> <p>or the</p>	<p>Configure with either</p> <p><b>bandwidth kbps</b></p> <p>or the</p>

		<b>bandwidth percent percentage</b>  command in the outbound direction.  Does NOT support ACL for matching in class map, but supports other OSM WAN match types.  Note: Can NOT review data in " <b>show mls qos ip</b> " results	<b>bandwidth percent percentage</b>  command, in either the inbound or outbound direction.  Supports all FlexWAN port match types  Supported on physical interface interface if it is in default queueing mode.
Hierarchical Shaping	Not supported	Supported for output policies only  Can only have class default in parent layer  Child policy does not support marking with <b>set</b> command  Does NOT support ACL for matching in class map, but supports other OSM WAN match types  Use <b>bandwidth</b> and <b>priority</b> in child policy	Supported for output policies only  Child policy does not support marking with <b>set</b> command  Use <b>bandwidth</b> and <b>priority</b> in child policy
WRED	Supported on main interface, configure with <b>wrr-queue random-detect min-threshold queue_id</b>	Supported on enhanced OSMs  Supported in output direction  Configure with <b>random-detect</b> under policy map class	Supported in output direction  Configure with <b>random-detect</b> under policy map class
Verification	Can use  <b>show policy interface</b>  <b>show mls qos ip</b>	Can use  <b>show policy interface</b>  <b>show mls qos ip</b> (for PFC QoS, not for shaping and CBWFQ)	Can use  <b>show policy interface</b>

## Sample 6500 QoS Configuration

Here's a sample configuration, straight from a test lab. This was from a 6500 running native IOS with a Sup720.

```
mls qos map cos-dscp 0 8 16 24 32 46 48 54
mls qos
!
class-map match-all policedclass
  match access-group name policetest
!
```

```

class-map match-all voice
  match ip dscp ef
!
policy-map policed1
  class policedclass
    police 1000000 31250 31250 conform-action transmit exceed-action drop
!
policy-map qospolicy
  class voice
    set dscp ef
!
interface GigabitEthernet1/1
ip address 170.22.20.22 255.255.255.248
  load-interval 30
  mls qos trust dscp
  service-policy output qospolicy
!
ip access-list extended policetest
deny ip any any dscp ef log
permit ip any any

```

We provide show command samples in case you are not familiar with these commands. The main verification command remains **"show policy interface"**.

```

6500#show mls qos
  QoS is enabled globally
  Microflow policing is enabled globally
  QoS ip packet dscp rewrite enabled globally

  QoS trust state is DSCP on the following interfaces:
    Gi1/1 Gi1/2 Gi1/3 Gi1/4 Gi1/5 Gi1/6 Gi1/7 Gi1/8 Gi1/9 Gi1/10
    Gi1/11 Gi1/12 Gi1/13 Gi1/14 Gi1/15 Gi1/16 Gi1/17 Gi1/18 Gi1/19 Gi1/20
    Gi1/21 Gi1/22 Gi1/23 Gi1/24 Gi5/1 Gi5/2
  Vlan or Portchannel (Multi-Earl) policies supported: Yes
  Egress policies supported: Yes

----- Module [5] -----
QoS global counters:
  Total packets: 1730211
  IP shortcut packets: 0
  Packets dropped by policing: 0
  IP packets with TOS changed by policing: 0
  IP packets with COS changed by policing: 1729894
  Non-IP packets with COS changed by policing: 0
  MPLS packets with EXP changed by policing: 0

```

More useful data comes from "show mls qos ip":

```

6500#show mls qos ip

  QoS Summary [IP]:          (* - shared aggregates, Mod - switch module)

Int Mod Dir  Class-map DSCP  Agg  Trust Fl  AgForward-By  AgPoliced-By
      Id      Id
-----
Gi1/9  5 Out    voice   46   7   --  0   1064027990    0
Gi5/2  5 Out    voice   46   3   --  0           0    0

All    5  -    Default  0   0*  No  0   313359032612  0

```

Note that the "show mls qos ..." commands only show information that applies to PFC QoS, and not to QoS services performed on line cards such as the FlexWAN.

## Summary

Your comments, questions, and suggestions for future articles are of course welcome! See below to decipher Pete's email address.

---

Dr. Peter J. Welcher (CCIE #1773, CCSI #94014, CCIP) is a Senior Consultant with Chesapeake NetCraftsmen. NetCraftsmen is a high-end consulting firm and Cisco Premier Partner dedicated to quality consulting and knowledge transfer. NetCraftsmen has ten CCIE's, with expertise including large network high-availability routing/switching and design, VoIP, QoS, MPLS, IPSec VPN, wireless LAN and bridging, network management, security, IP multicast, and other areas. See <http://www.netcraftsmen.net> for more information about NetCraftsmen. Pete's links start at <http://www.netcraftsmen.net/welcher>. New articles will be posted under the Articles link. Questions, suggestions for articles, etc. can be sent to **[pjw <at> netcraftsmen <dot> net](mailto:pjw@netcraftsmen.net)** (formatted this way to fool email harvesting software).

Carole Warner Reece (CCIE #5168) is also a Senior Consultant with Chesapeake NetCraftsmen. She has worked with a wide variety of products and technologies in complex environments, and has broad experience in network design, consulting, and project implementation, as well as in developing network training materials. She can be reached at **[cwr <at> netcraftsmen <dot> net](mailto:cwr@netcraftsmen.net)**.

11/4/2005

Copyright (C) 2005 Peter J. Welcher