QoS–Hierarchical Queueing for Ethernet DSLAMs

First Published: November, 2006
Last Updated: October 3, 2008

This feature module describes how to configure quality of service (QoS) hierarchical queueing policy maps on sessions and subinterfaces in Ethernet Digital Subscriber Line Access Multiplexer (E-DSLAM) applications on a Cisco 10000 series router. The QoS–Hierarchical Queueing for Ethernet-DSLAMs feature supports IEEE 802.1 QinQ VLAN tag termination to configure inner VLAN identifiers on E-DSLAMs.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the "Feature Information for QoS–Hierarchical Queueing for Ethernet DSLAMs" section on page 20.

Use Cisco Feature Navigator to find information about platform support and Cisco IOS and Catalyst OS software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Contents

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Prerequisites for QoS–Hierarchical Queueing for Ethernet DSLAMs

- You must configure traffic classes using the **class-map** command.
- This feature requires a Performance Routing Engine 3 (PRE3).

Restrictions for QoS–Hierarchical Queueing for Ethernet DSLAMs

- Cisco IOS Release 12.2(31)SB2 do not include Modular QoS Command-Line Interface (CLI) (MQC) support for the following session-based queueing policies:
  - IP sessions
  - Inner VLAN sessions
- This feature is not supported in combination with load balancing when a session service policy is routed to a Layer 2 Tunnel Protocol (L2TP) tunnel. Do not configure load balancing on an L2TP tunnel if per-session queueing is enabled.

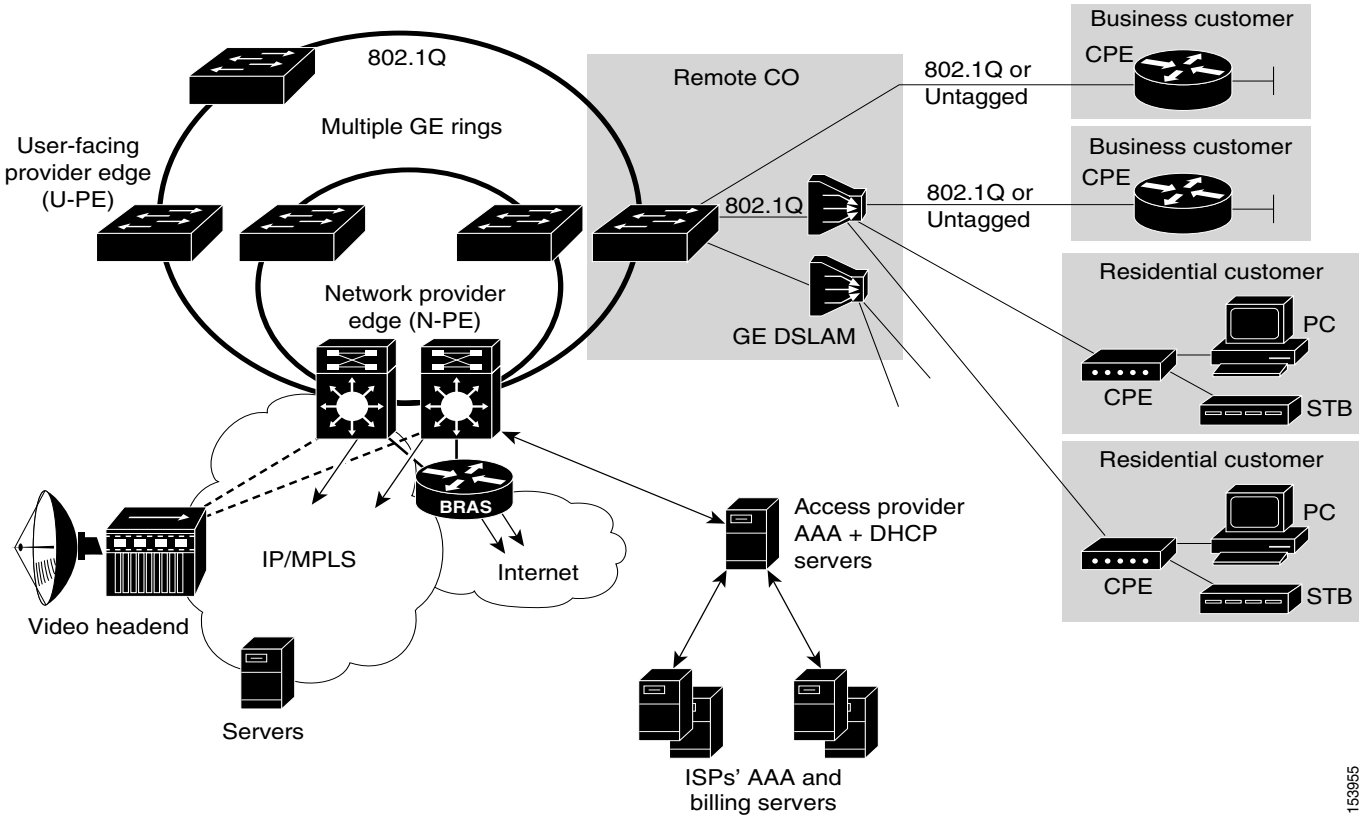
Information About QoS–Hierarchical Queueing for Ethernet DSLAMs

To configure QoS–Hierarchical Queueing for Ethernet DSLAMs, you should understand the following concepts:

Different Levels of QoS Provisioning

Traffic downstream from a Broadband Router Access Server (BRAS) requires different levels of QoS provisioning (for example, traffic shaping) depending on the network architecture between the BRAS and the subscriber. Figure 1 on page 3 illustrates an Ethernet DSL access network. The sample network includes multiple entities where QoS provisioning is required for different reasons.

Figure 1 Ethernet DSL Access Network



The following entities may require different traffic shaping:

- A VLAN shaped to a certain aggregate traffic rate to limit the traffic to a group of subscribers (different 802.1Q interfaces in Figure 1).
- Individual sessions shaped with certain QoS services for different classes of traffic (individual PCs in Figure 1).

Integrated Queueing Hierarchy

Different traffic shaping requirements result in QoS provisioning at multiple levels at the same time. The QoS–Hierarchical Queueing for Ethernet DSLAMs feature provides the ability to form one integrated queueing hierarchy that provides QoS provisioning at multiple levels with support for features such as bandwidth distribution at any of these levels.

The integrated queueing hierarchy is formed on the physical interface. When a service policy is instantiated on a session, the Subscriber Service Switch (SSS) infrastructure invokes the MQC and a common queueing control plane sets up and enables the queueing features.

Session-to-interface associations are resolved to determine the physical interface on which to form the integrated queueing hierarchy for all levels of QoS provisioning. As subinterface session-based policies are added, the respective queues are created and integrated into the queueing hierarchy.

When a subinterface is provisioned followed by session-based policy provisioning, the integrated queueing hierarchy is formed on top of the physical interface as a result of queueing policies provisioned at two different levels. When a session is provisioned before subinterface-based policy provisioning, the

queueing hierarchy has a placeholder logical level between the physical queue and the session queue. The placeholder queue becomes the default queue at that level, and all other sessions are parented to that queue.

A PRE3 supports three hierarchical levels of scheduling: physical port, session, and class queues. When sessions are established within a subinterface that is configured with a shaping policy, the subinterface level is lowered to the physical layer.

Configuration Guidelines for Hierarchical Queueing on Ethernet DSLAMs

When configuring QoS–Hierarchical Queueing for Ethernet DSLAMs feature, note the following guidelines:

- An individual subscriber is always identified by a PPP or IP session. A group of subscribers is identified by a particular VLAN by means of the outer tag ISP, E-DSLAM, or user-facing provider edge (U-PE).
- When a subinterface is used to aggregate a number of sessions with queueing policies, a queueing policy at a subinterface level must be a one-level policy map that is configured as class-default with only the shape feature enabled.
- Do not oversubscribe sessions to ensure distributed bandwidth for sessions with configured shape rates.

Configuring QoS–Hierarchical Queueing for Ethernet DSLAMs

This section contains the procedures for configuring the QoS–Hierarchical Queueing for Ethernet DSLAMs feature. While all three procedures are listed as optional, you must choose one of the first two.

- Configuring and Applying QoS–Hierarchical Queueing Policy Maps to Sessions, page 4 (optional)
- Configuring and Applying QoS–Hierarchical Queueing Policy Maps to Subinterfaces, page 9 (optional)
- Displaying Policy-Map Information for Hierarchical Queueing, page 11 (optional)

Configuring and Applying QoS–Hierarchical Queueing Policy Maps to Sessions

To configure and apply a QoS hierarchical queueing policy map to PPP/IP sessions through a virtual template, perform the following steps.

Note To configure and apply a QoS hierarchical queueing policy map to subinterfaces, skip this procedure and complete the steps in the "Configuring and Applying QoS–Hierarchical Queueing Policy Maps to Subinterfaces" section on page 9.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **policy-map** *policy-map-name*

4. **class** *class-map-name*
5. **bandwidth** {*bandwidth-kbps* | **percent** *percentage* | **remaining percent** *percentage*}
6. **precedence** *precedence min-threshold max-threshold mark-probability-denominator*
7. **set cos** *cos-value*
8. **exit**
9. **exit**
10. **policy-map** *policy-map-name*
11. **class class-default**
12. **shape average** *cir*
13. **bandwidth remaining ratio** *ratio*
14. **service-policy** *policy-map-name*
15. **exit**
16. **exit**
17. **interface virtual-template** *number*
18. **service-policy output** *policy-map-name*
19. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**policy-map** *policy-map-name* **Example:** `Router(config)# policy-map session_a_child`	Creates a child policy and enters policy-map configuration mode. • Enter the policy-map name. Names can be a maximum of 40 alphanumeric characters.
Step 4	**class** *class-map-name* **Example:** `Router(config-pmap)# class voip`	Configures the traffic class that you specify and enters policy-map class configuration mode. • Enter the name of a previously configured class map.

	Command or Action	Purpose
Step 5	**bandwidth** {*bandwidth-kbps* | **percent** *percentage* | **remaining percent** *percentage*} **Example:** `Router(config-pmap-c)# bandwidth 10000`	(Optional) Enables class-based weighted fair queueing based on the keywords and arguments specified, as described below. • *bandwidth-kbps*—Specifies the minimum bandwidth allocated for a class belonging to a policy map. Valid values are from 8 to 2,488,320, which represents from 1 to 99 percent of the link bandwidth. • **percent** *percentage*—Specifies the minimum percentage of the link bandwidth allocated for a class belonging to a policy map. Valid values are from 1 to 99. • **remaining percent** *percentage*—Specifies the minimum percentage of unused link bandwidth allocated for a class belonging to a policy map. Valid values are from 1 to 99.
Step 6	**precedence** *precedence min-threshold max-threshold mark-probability-denominator* **Example:** `Router(config-pmap-c)# precedence 0 32 256 100`	(Optional) Configures a precedence level for the traffic class based on the arguments specified, as described below. • *precedence*—Specifies the IP precedence number. Valid values are from 0 to 7. • *min-threshold*—Specifies the minimum threshold in number of packets. Valid values are from 1 to 4096. • *max-threshold*—Specifies the maximum threshold in number of packets. Valid values are from the minimum threshold to 4096. • *mark-probability-denominator*—Specifies the denominator for the fraction of packets dropped when the average queue depth is equal to the maximum threshold. For example, if the denominator is 512, 1 out of every 512 packets is dropped when the average queue is at the maximum threshold. Valid values are from 1 to 65536. The default value is 10 (1 out of every 10 packets is dropped at the maximum threshold).
Step 7	**set cos** *cos-value* **Example:** `Router(config-pmap-c)# set cos 1`	(Optional) Sets the Layer 2 class of service (CoS) value of an outgoing packet. • Enter the IEEE 802.1Q CoS value from 0 to 7. **Note** Use the **set cos** command only in service policies that are attached in the output direction of an interface; packets that enter an interface cannot be set with a CoS value. You can configure a CoS value on an Ethernet interface that is configured for 802.1Q or on a virtual access interface that is using an 802.1Q interface.
Step 8	**exit** **Example:** `Router(config-pmap-c)# exit`	Exits policy-map class configuration mode.

	Command or Action	Purpose
Step 9	**exit** **Example:** `Router(config-pmap)# exit`	Exits policy-map configuration mode.
Step 10	**policy-map** *policy-map-name* **Example:** `Router(config)# policy-map session_a_parent`	Creates a parent policy and enters policy-map configuration mode. • Enter the policy-map name. Names can be a maximum of 40 alphanumeric characters.
Step 11	**class class-default** **Example:** `Router(config-pmap)# class class-default`	Configures the traffic class as **class-default** and enters policy-map class configuration mode. **Note** Do not configure any other traffic class.
Step 12	**shape average** *cir* **Example:** `Router(config-pmap-c)# shape average 10000000`	Specifies average-rate traffic shaping for all traffic that does not match any other traffic class. • Enter the **average** keyword followed by the committed information rate (CIR), in bits per second (bps).
Step 13	**bandwidth remaining ratio** *ratio* **Example:** `Router(config-pmap-c)# bandwidth remaining ratio 10`	Specifies the weight (ratio) for the subinterface. • Enter the relative weight of this subinterface (or class queue). This number (ratio) indicates the proportional relationship between the other subinterfaces or class queues.
Step 14	**service-policy** *policy-map-name* **Example:** `Router(config-pmap-c)# service-policy session_a_child`	Applies the child policy map to the parent class-default class. • Enter the name of a previously configured child policy map.
Step 15	**exit** **Example:** `Router(config-pmap-c)# exit`	Exits policy-map class configuration mode.
Step 16	**exit** **Example:** `Router(config-pmap)# exit`	Exits policy-map configuration mode.
Step 17	**interface virtual-template** *number* **Example:** `Router(config)# interface virtual-template 1`	Creates a virtual template and enters interface configuration mode. • Enter the virtual template number. Valid range is from 1 to 200.

	Command or Action	Purpose
Step 18	**service-policy output** *policy-map-name* **Example:** `Router(config-if)# service-policy output session_a_parent`	Applies the service policy to the virtual interface. • Enter the name of the previously configured parent policy map. **Note** You must specify the **output** keyword to apply the service policy to outbound traffic on the interface.
Step 19	**end** **Example:** `Router(config-if)# end`	(Optional) Returns to privileged EXEC mode.

Examples

The following is an example of how to configure and apply a QoS hierarchical queueing policy map to PPP/IP sessions by using a virtual template:

```
Router> enable
Router# configure terminal
Router(config)# policy-map session_a_child
Router(config-pmap)# class voip
Router(config-pmap-c)# police 1000000
Router(config-pmap-c)# priority level 1
Router(config-pmap-c)# exit
Router(config-pmap)# class video
Router(config-pmap-c)# police 100000
Router(config-pmap-c)# priority level 2
Router(config-pmap-c)# exit
Router(config-pmap)# class precedence_0
Router(config-pmap-c)# bandwidth remaining ratio 10
Router(config-pmap-c)# exit
Router(config-pmap)# class precedence_1
Router(config-pmap-c)# bandwidth remaining ratio 20
Router(config-pmap-c)# exit
Router(config-pmap)# exit
Router(config)# policy-map session_a_parent
Router(config-pmap-c)# exit
Router(config-pmap)# class class-default
Router(config-pmap-c)# shape average 10000000
Router(config-pmap-c)# bandwidth remaining ratio 10
Router(config-pmap-c)# service-policy session_a_child
Router(config-pmap-c)# exit
Router(config-pmap)# exit
Router(config)# interface virtual-template 20
Router(config-if)# service-policy output session_a_parent
Router(config-if)# end
```

Configuring and Applying QoS–Hierarchical Queueing Policy Maps to Subinterfaces

To configure and apply a QoS hierarchical queueing policy map to a subinterface (and provide aggregate shaping for a large number of subscribers), perform the following steps.

Note When a subinterface is used to aggregate a number of sessions with queueing policies, a queueing policy at a subinterface level must be a one-level policy map configured as class-default with only the shape feature enabled.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **policy-map** *policy-map-name*
4. **class class-default**
5. **shape average** *cir*
6. **exit**
7. **exit**
8. **interface** *type slot/subslot/port.subinterface*
9. **encapsulation dot1q** *outer-vlan-id* [**second-dot1q** *inner-vlan-id*]
10. **service-policy output** *policy-map-name*
11. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**policy-map** *policy-map-name* **Example:** `Router(config)# policy-map subint_1`	Creates a policy map. and enters policy-map configuration mode. • *policy-map-name*—The name of the policy map, which can be a maximum of 40 alphanumeric characters.
Step 4	**class class-default** **Example:** `Router(config-pmap)# class class-default`	Configures the traffic class as **class-default** and enters policy-map class configuration mode. Do not configure any other traffic class.

	Command or Action	Purpose
Step 5	**shape average** *cir* **Example:** `Router(config-pmap-c)# shape average 10000000`	Specifies average-rate traffic shaping for all traffic that does not match any other traffic class. • Enter the **average** keyword followed by the CIR, in bps.
Step 6	**exit** **Example:** `Router(config-pmap-c)# exit`	Exits policy-map class configuration mode.
Step 7	**exit** **Example:** `Router(config-pmap)# exit`	Exits policy-map configuration mode.
Step 8	**interface** *type slot/subslot/port.subinterface* **Example:** `Router(config)# interface GigabitEthernet3/1/1.1`	Specifies the subinterface on which you are attaching the policy map and enters subinterface configuration mode. • Enter the interface type and slot number, subslot number, port number, and subinterface number.
Step 9	**encapsulation dot1q** *outer-vlan-id* [**second-dot1q** *inner-vlan-id*] **Example:** `Router(config-subif)# encapsulation dot1q 100`	Enables IEEE 802.1Q encapsulation of traffic on the subinterface. The **second-dot1q** keyword supports the IEEE 802.1 QinQ VLAN Tag Termination feature to configure an inner VLAN ID. • *outer-vlan-id*—The outer VLAN identifier. The range is from 1 to 4095. • *inner-vlan-id*—The inner VLAN identifier. The range is from 1 to 4095.
Step 10	**service-policy output** *policy-map-name* **Example:** `Router(config-subif)# service-policy output subint_1`	Attaches the service policy to the subinterface. • *policy-map-name*—The name of the previously configured policy map. **Note** You must specify the **output** keyword to apply the service policy to outbound traffic on the subinterface.
Step 11	**end** **Example:** `Router(config-subif)# end`	(Optional) Returns to privileged EXEC mode.

Examples

The following is an example of how to configure and apply a QoS hierarchical queueing policy map to a subinterface (and provide aggregate shaping for a large number of subscribers):

```
Router> enable
Router# configure terminal
Router(config)# policy-map subint_1
Router(config-pmap)# class class-default
```

```
Router(config-pmap-c)# shape average 10000000
Router(config-pmap-c)# exit
Router(config-pmap)# exit
Router(config)# interface GigabitEthernet3/1/1.1
Router(config-subif)# encapsulation dot1q 100
Router(config-subif)# service-policy output subint_1
Router(config-subif)# end
```

Displaying Policy-Map Information for Hierarchical Queueing

To display policy-map information, perform the following steps.

SUMMARY STEPS

1. **enable**
2. **show policy-map**
3. **show policy-map interface** *interface-name*
4. **show policy-map session**
5. **exit**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**show policy-map** **Example:** `Router# show policy-map`	(Optional) Displays all information for all class maps.
Step 3	**show policy-map interface** *interface-name* **Example:** `Router# show policy-map interface serial4/0`	(Optional) Displays the packet statistics of all classes that are configured for all service policies either on the specified interface or subinterface or on a specific PVC on the interface. • Enter the interface name.
Step 4	**show policy-map session** **Example:** `Router# show policy-map session`	(Optional) Displays the QoS policy map in effect for the SSS session.
Step 5	**exit** **Example:** `Router# exit`	(Optional) Exits privileged EXEC mode.

Configuration Examples for QoS–Hierarchical Queueing for Ethernet DSLAMs

This section provides the following configuration examples:

Configuring QoS–Hierarchical Queueing Policy Maps on VLANs or QinQ Subinterfaces: Example

The following example shows how to configure and apply QoS hierarchical queueing policy maps on VLANs or QinQ subinterfaces. A child queueing policy is applied to each parent subscriber line level policy. In this example, the policy maps are applied to create subscriber groups on subinterfaces.

```
Router> enable
Router# configure terminal
Router(config)# policy-map service_a_out
Router(config-pmap)# class voip
Router(config-pmap-c)# priority
Router(config-pmap-c)# police percent 20 bc 300 ms pir 40
Router(config-pmap-c)# set cos 1
Router(config-pmap-c)# exit
Router(config-pmap)# class video
Router(config-pmap-c)# police percent 20 bc 300 ms pir 40
Router(config-pmap-c)# set cos 2
Router(config-pmap-c)# exit
Router(config-pmap)# class gaming
Router(config-pmap-c)# bandwidth remaining percent 80
Router(config-pmap-c)# set cos 3
Router(config-pmap-c)# exit
Router(config-pmap)# class class-default
Router(config-pmap-c)# bandwidth remaining percent 20
Router(config-pmap-c)# set cos 4
Router(config-pmap-c)# exit
Router(config-pmap)# exit
!
Router(config)# policy-map service_z_out
Router(config-pmap)# exit
!
Router(config)# policy-map rate_1_service_a_in
Router(config-pmap)# class voip
Router(config-pmap-c)# police percent 25 4 ms 1 ms
Router(config-pmap-c)# exit
Router(config-pmap)# class gaming
Router(config-pmap-c)# police percent 50 2 ms 1 ms
Router(config-pmap-c)# exit
Router(config-pmap)# class class-default
Router(config-pmap-c)# police percent 20 bc 300 ms pir 40
Router(config-pmap-c)# exit
Router(config-pmap)# exit
```

```
!
Router(config)# policy-map rate_x_service_z_in
Router(config-pmap)# exit
!
Router(config)# policy-map rate_1_service_a_out
Router(config-pmap)# class class-default
Router(config-pmap-c)# bandwidth remaining ratio 10
Router(config-pmap-c)# shape average 100000
Router(config-pmap-c)# service policy service_a_out
Router(config-pmap-c)# exit
Router(config-pmap)# exit
!
Router(config)# policy-map rate_x_serviice_z_out
Router(config-pmap)# class class-default
Router(config-pmap-c)# bandwidth remaining ratio 10
Router(config-pmap-c)# shape average 100000
Router(config-pmap-c)# service policy service_z_out
Router(config-pmap-c)# exit
Router(config-pmap)# exit
Router(config)# interface GigabitEthernet1/0/0.1
Router(config-subif)# encapsulation dot1q 5 second dot1q 20
Router(config-subif)# service-policy output rate_1_service_a_out
Router(config-subif)# service-policy input rate_1_service_a_in
Router(config-subif)# exit
Router(config)# interface GigabitEthernet1/0/0.2
Router(config-subif)# encapsulation dot1q 5 second dot1q 25
Router(config-subif)# service-policy output rate_x_service_z_out
Router(config-subif)# service-policy input rate_x_service_z_in
Router(config-subif)# end
```

Configuring QoS–Hierarchical Queueing Policy Maps on VLANs with Arbitrary QinQ: Example

The following example shows how to configure and apply QoS hierarchical queueing policy maps on VLANs with subscriber lines grouped by arbitrary QinQ. A child queueing policy is applied to each parent subscriber line level policy. This example includes the configuration of multiple class maps.

```
Router> enable
Router# configure terminal
Router(config)# class-map match-all user_1
Router(config-cmap)# match vlan 10
Router(config-cmap)# exit
Router(config)# class-map match-all user_2
Router(config-cmap)# match vlan 11
Router(config-cmap)# exit
Router(config)# class-map match-all user_3
Router(config-cmap)# match vlan 10
Router(config-cmap)# exit
Router(config)# class-map match-any user_4
Router(config-cmap)# match vlan 11
Router(config-cmap)# exit
Router(config)# class-map match-all user_n
Router(config-cmap)# exit
Router(config)# class-map match-any isp_A
Router(config-cmap)# match class user_1
Router(config-cmap)# match class user_2
Router(config-cmap)# exit
Router(config)# class-map match-any isp_Z
Router(config-cmap)# match class user_3
Router(config-cmap)# match class user_4
Router(config-cmap)# exit
```

```
!
Router(config)# policy-map service_a_out
Router(config-pmap)# class voip
Router(config-pmap-c)# priority
Router(config-pmap-c)# police cir percent 20 bc 300 ms pir 40
Router(config-pmap-c)# set cos 1
Router(config-pmap-c)# exit
Router(config-pmap)# class video
Router(config-pmap-c)# police cir percent 20 bc 300 ms pir 40
Router(config-pmap-c)# set cos 2
Router(config-pmap-c)# exit
Router(config-pmap)# class gaming
Router(config-pmap-c)# bandwidth remaining percent 80
Router(config-pmap-c)# set cos 3
Router(config-pmap-c)# exit
Router(config-pmap)# class class-default
Router(config-pmap-c)# bandwidth remaining percent 20
Router(config-pmap-c)# set cos 4
Router(config-pmap-c)# exit
Router(config-pmap)# exit
!
Router(config)# policy-map service_z_out
Router(config)# policy-map service_a_in
Router(config-pmap)# class voip
Router(config-pmap-c)# police percent 25 4 ms 1 ms
Router(config-pmap-c)# exit
Router(config-pmap)# class gaming
Router(config-pmap-c)# police percent 50 2 ms 1 ms
Router(config-pmap-c)# exit
Router(config-pmap)# class class-default
Router(config-pmap-c)# police cir percent 20 bc 300 ms pir 40
Router(config-pmap-c)# exit
Router(config-pmap)# exit
!
Router(config)# policy-map service_z_in
Router(config-pmap)# exit
!
Router(config)# policy-map isp_A_out
Router(config-pmap)# class user_1
Router(config-pmap-c)# bandwidth remaining ratio 10
Router(config-pmap-c)# shape average 100000
Router(config-pmap-c)# service policy service_a_out
Router(config-pmap-c)# exit
Router(config-pmap)# class user_n
Router(config-pmap-c)# bandwidth remaining ratio 20
Router(config-pmap-c)# shape average 100000
Router(config-pmap-c)# service policy service_z_out
Router(config-pmap-c)# exit
Router(config-pmap)# exit
!
Router(config)# policy-map isp_Z_out
Router(config-pmap)# exit
!
Router(config)# policy-map isp_A_in
Router(config-pmap)# class user_1
Router(config-pmap-c)# service policy service_a_in
Router(config-pmap-c)# class user_n
Router(config-pmap-c)# service policy service_z_in
Router(config-pmap-c)# exit
Router(config-pmap)# exit
!
Router(config)# policy-map isp_Z_in
Router(config-pmap)# exit
!
```

```
Router(config)# policy-map interface_policy_out
Router(config-pmap)# class isp_A
Router(config-pmap-c)# shape average 100000
Router(config-pmap-c)# service policy isp_A_out
Router(config-pmap-c)# exit
Router(config-pmap)# class isp_Z
Router(config-pmap-c)# shape average 100000
Router(config-pmap-c)# service policy isp_Z_out
Router(config-pmap-c)# exit
Router(config-pmap)# exit
!
Router(config)# policy-map interface_policy_in
Router(config-pmap)# class isp_A
Router(config-pmap-c)# service policy isp_A_in
Router(config-pmap-c)# exit
Router(config-pmap)# class isp_Z
Router(config-pmap-c)# service policy isp_Z_in
Router(config-pmap-c)# exit
Router(config-pmap)# exit
!
Router(config)# interface GigabitEthernet1/0/0.1
Router(config-subif)# encapsulation dot1q 5 second dot1q any
Router(config-subif)# service-policy output interface_policy_out
Router(config-subif)# service-policy input interface_policy_in
Router(config-subif)# end
```

Configuring QoS–Hierarchical Queueing Policy Maps on Sessions: Example

The following example shows how to configure and apply QoS hierarchical queueing policy maps on sessions. A child queueing policy is applied to each parent subscriber line level policy.

```
Router> enable
Router# configure terminal
Router(config)# policy-map service_a_out
Router(config-pmap)# class voip
Router(config-pmap-c)# priority
Router(config-pmap-c)# set cos 1
Router(config-pmap-c)# exit
Router(config-pmap)# class video
Router(config-pmap-c)# set cos 2
Router(config-pmap-c)# exit
Router(config-pmap)# class gaming
Router(config-pmap-c)# bandwidth remaining percent 80
Router(config-pmap-c)# set cos 3
Router(config-pmap-c)# exit
Router(config-pmap)# class class-default
Router(config-pmap-c)# bandwidth remaining percent 20
Router(config-pmap-c)# set cos 4
Router(config-pmap-c)# exit
Router(config-pmap)# exit
!
Router(config)# policy-map service_z_out
Router(config-pmap)# exit
!
Router(config)# policy-map rate_1_service_a_out
Router(config-pmap)# class class-default
Router(config-pmap-c)# bandwidth remaining ratio 10
Router(config-pmap-c)# shape average 100000
Router(config-pmap-c)# service-policy service_a_out
Router(config-pmap-c)# exit
Router(config-pmap)# exit
!
```

```
Router(config)# policy-map rate_x_service_z_out
Router(config-pmap)# class class-default
Router(config-pmap-c)# bandwidth remaining ratio 10
Router(config-pmap-c)# shape average 100000
Router(config-pmap-c)# service-policy service_z_out
Router(config-pmap-c)# exit
Router(config-pmap)# exit
!
Router(config)# policy-map rate_1_service_a_in
Router(config-pmap)# class voip
Router(config-pmap-c)# police percent 25 4 ms 1 ms
Router(config-pmap-c)# exit
Router(config-pmap)# class gaming
Router(config-pmap-c)# police percent 50 2 ms 1 ms
Router(config-pmap-c)# exit
Router(config-pmap)# class class-default
Router(config-pmap-c)# police cir percent 20 bc 300 ms pir 40
Router(config-pmap-c)# exit
Router(config-pmap)# exit
!
Router(config)# policy-map rate_x_service_z_in
Router(config-pmap)# exit
!
Router(config)# policy-map isp_A_out
Router(config-pmap)# class class-default
Router(config-pmap-c)# shape average 100000
Router(config-pmap-c)# exit
Router(config-pmap)# exit
Router(config)# exit
Router(config)# policy-map isp_Z_out
Router(config-pmap-c)# exit
Router(config-pmap)# class class-default
Router(config-pmap-c)# shape average 100000
Router(config-pmap-c)# exit
Router(config-pmap)# exit
Router(config)# interface GigabitEthernet1/0/0.1
Router(config-subif)# encapsulation dot1q 1
Router(config-subif)# service-policy output isp_A_out
Router(config-subif)# exit
Router(config)# interface GigabitEthernet2/0/0.2
Router(config-subif)# encapsulation dot1q 2
Router(config-subif)# service-policy output isp_Z_out
Router(config-subif)# end
```

Configuring QoS–Hierarchical Queueing Policy Maps on Sessions with Aggregate Shaping: Example

The following example shows how to configure and apply QoS hierarchical queueing policy maps on sessions with multiple PPP/IP sessions per subscriber line. In this example, the same policies are applied to all sessions using the same virtual interface.

```
Router> enable
Router# configure terminal
Router(config)# policy-map service_a_out
Router(config-pmap)# class voip
Router(config-pmap-c) priority
Router(config-pmap-c)# police percent 25 4 ms 1 ms
Router(config-pmap-c)# set cos 1
Router(config-pmap-c)# exit
Router(config-pmap)# class video
```

```
Router(config-pmap-c)# police percent 30 5 ms 1 ms
Router(config-pmap-c)# set cos 2
Router(config-pmap-c)# exit
Router(config-pmap)# class class-default
Router(config-pmap-c)# bandwidth remaining percent 20
Router(config-pmap-c)# set cos 3
Router(config-pmap-c)# exit
Router(config-pmap)# exit
!
Router(config)# policy-map service_z_out
Router(config-pmap)# exit
!
Router(config)# policy-map rate_1_service_a_in
Router(config-pmap)# class voip
Router(config-pmap-c)# police percent 25 4 ms 1 ms
Router(config-pmap-c)# exit
Router(config-pmap)# class video
Router(config-pmap-c)# police percent 30 2 ms 1 ms
Router(config-pmap-c)# exit
Router(config-pmap)# class class-default
Router(config-pmap-c)# police percent 40 2 ms 1 ms
Router(config-pmap-c)# exit
Router(config-pmap)# exit
!
Router(config)# policy-map rate_x_service_z_in
Router(config-pmap)# exit
!
Router(config)# policy-map rate_1_service_a_out
Router(config-pmap)# class class-default
Router(config-pmap-c)# bandwidth remaining ratio 10
Router(config-pmap-c)# shape average 100000
Router(config-pmap-c)# service policy service_a_out
Router(config-pmap-c)# exit
Router(config-pmap)# exit
!
Router(config)# policy-map rate_x_service_z_out
Router(config-pmap)# class class-default
Router(config-pmap-c)# bandwidth remaining ratio 10
Router(config-pmap-c)# shape average 100000
Router(config-pmap-c)# service policy service_z_out
Router(config-pmap-c)# exit
Router(config-pmap)# exit
Router(config)# interface GigabitEthernet1/0/0
Router(config-if)# encapsulation dot1q 1
Router(config-if)# service-policy output isp_A_out
Router(config-if)# exit
Router(config)# interface GigabitEthernet2/0/0
Router(config-if)# encapsulation dot1q 2
Router(config-if)# service-policy output isp_Z_out
Router(config-if)# end
```

Additional References

The following sections provide references related to the QoS–Hierarchical Queueing for Ethernet DSLAMs feature.

Related Documents

Related Topic	Document Title
QoS commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples	*Cisco IOS Quality of Service Solutions Command Reference*
Traffic shaping	"Regulating Traffic Flow Using Traffic Shaping" module
MQC	"Applying QoS Features Using the MQC" module

Standards

Standard	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIB	MIBs Link
No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/techsupport

Command Reference

This feature uses no new or modified commands.

Feature Information for QoS–Hierarchical Queueing for Ethernet DSLAMs

Table 1 lists the release history for this feature.

Not all commands may be available in your Cisco IOS software release. For release information about a specific command, see the command reference documentation.

Use Cisco Feature Navigator to find information about platform support and software image support. Cisco Feature Navigator enables you to determine which Cisco IOS, Catalyst OS, and Cisco IOS XE software images support a specific software release, feature set, or platform. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn. An account on Cisco.com is not required.

Note Table 1 lists only the Cisco IOS software release that introduced support for a given feature in a given Cisco IOS software release train. Unless noted otherwise, subsequent releases of that Cisco IOS software release train also support that feature.

Table 1 Feature Information for QoS–Hierarchical Queueing for Ethernet DSLAMS

Feature Name	Releases	Feature Information
QoS–Hierarchical Queueing for Ethernet DSLAMs	12.2(31)SB2	This feature module describes how to configure QoS hierarchical queueing policy maps on sessions and subinterfaces in Ethernet Digital Subscriber Line Access Multiplexer (E-DSLAM) applications. For Release 12.3(31)SB2, this feature was introduced and implemented on the Cisco 10000 series router for the PRE3.