# Configuration Commands

# Generic Commands

## description

**Syntax** **description** *description-string*
**no description**

**Context** config>qos>named-pool-policy>q1-pools>pool

**Description** The description command is used to define an informational ASCII string for the named pool policy. The string value may be defined or changed at anytime.

The **no** form of the command removes the explicit description string from the named pool policy.

**Parameters** *description-string* — The description-string parameter defines the ASCII description string for the named pool policy. If the string contains spaces, it must be placed within beginning and ending double quotation marks. Beginning and ending quotation marks are not considered part of the description string. Only printable ASCII characters are allowed in the string. The sting does not need to be unique within the system. If the command is executed without the description-sting present, any existing description string will be unaffected.

**Unit** ASCII String

**Length** Up to 80 characters

**Default** None

# Named Pool Policy Creation

## named-pool-policy

**Syntax** **named-pool-policy** *policy-name* **create**
**no named-pool-policy** *policy-name*

**Context** config>qos

**Description** This commands creates a template that may be applied at the MDA or port level to create named buffer pools. The policy may be applied at either the ingress or egress context for the port or MDA. Policies applied on the MDA will take effect only after the named pool mode is set on the IOM. Setting the IOM named pool on will reboot the card.

Within the policy, named pools may be defined in the q1-pools context indicating that the provisioned pools will be used on Q1 based hardware. When the policy is associated at the MDA level, named pools defined in the policy allow queues from any port to be associated. When the policy is associated at the port level, the named pools created are only available to queues associated with the port. Each pool defined allows the slope-policy, resv-cbs, access-allocation-weight and network-allocation-weight parameters to be configured for the pool. The policy also manages the port-allocation-weights used to divide the buffers managed by the port between named pools local to the port, named pools on the ports MDA and the default pools. The allocation weights for a given port are derived in the following way (lowest to highest preference):

1. Port default allocation weights

   **Default** default 50, mda 50, port 50

2. MDA named pool policy port allocation weights
3. Port named pool policy port allocation weights

A named-pool-policy that is currently applied to an MDA or port may not be deleted. All associations between the policy and MDAs must be removed prior to deleting the policy.

Pools in the policy may be added or removed at anytime. If the policy is currently associated with an MDA or port, the system will first check to ensure necessary resources exist on the port or MDA before allowing the pool creation within the policy to proceed. If the pool cannot be added, the pool pool-name command will fail. When a new pool is created, the system will scan all pool orphaned queues for queues associated with the new pool name wherever the policy is currently applied. (A queue with a defined pool name that does not exist is placed on its appropriate default pool until the pool comes into existence).

The **no** form of the command removes a specific named pool policy from the system. If the named pool policy is currently associated with an ingress or egress MDA or port, the command will fail. If the named pool policy does not exist, the command has no effect and does not return an error.

**Parameters** *named-pool-policy-name* — The named-pool-policy-name is required. Each named pool policy must be uniquely named within the system. Names of up to 32 ASCII characters are supported with the

normal character restrictions. A named pool policy must exist prior to applying the policy name to an MDA or port.

**Values** Up to 32 character ASCII string

**Default** None (A system default named pool policy does not exist)

**Limit** 1024 policies per system

**create —** The create keyword is required if creating a new named pool policy when the system is configured to require the explicit use of the keyword to prevent accidental object creation. Objects may be accidentally created when this protection is disabled and an object name is mistyped when attempting to edit the object. This keyword is not required when the protection is disabled. The keyword is ignored when the named pool policy already exists.

## description

**Syntax** **description** *description-string*
**no description**

**Context** config>qos>named-pool-policy

**Description** The description command is used to define an informational ASCII string for the named pool policy. The string value may be defined or changed at anytime.

The **no** form of the command removes an explicit description string from the named pool policy.

**Parameters** *description-string* — The description-string parameter defines the ASCII description string for the named pool policy. If the string contains spaces, it must be placed within beginning and ending double quotation marks. Beginning and ending quotation marks are not considered part of the description string. Only printable ASCII characters are allowed in the string. The sting does not need to be unique within the system. If the command is executed without the description-sting present, any existing description string will be unaffected.

**Values** Up to 80 character ASCII string

**Default** None

## q1-pools

**Syntax** **q1-pools**

**Context** config>qos>named-pool-policy

**Description** The q1-pools command is used to enter the configuration node for Q1 oriented named buffer pools. The named pool policy will support contexts for configuring pools of other types when other pool types exist.

## port-allocation-weights

**Syntax** **port-allocation-weights default** *weight* **mda** *weight* **port** *weight*
**no port-allocation-weights**

**Context** config>qos>named-pool-policy

**Description** The port-allocation-weights command is used to define the weights used to divide the buffers managed by a port into three categories: default, MDA and port. The default category is given to the default pools, the MDA category is given to the MDA named pools and the port category is used by the local port named pools. When the IOM is placed in named-pool-mode, each port has an inherent set of weights that places all port managed buffers into the default category (default = 100, MDA = 0, port = 0). The policy port-allocation-weights command is used to override this port inherent behavior. When the policy is applied to the MDA, the defined port-allocation-weights parameter values override the inherent values for all ports on the MDA. When the policy is applied to the port level, the defined port-allocation-weights override both the local ports inherent weights and the MDA level named pool policy weights (if existing).

The **no** form of the command resets all values to the default value.

**Parameters** *default weight* — The default keyword is used to identify the weight value for the port where the policy is applied used in the calculation of the amount of buffer space given to the default pools by the port. The following weight parameter is required and must be specified as an integer between 0 and 100. The specified weight only has meaning when compared to the mda and port weights. The sum of all three weights is divided into each weight to determine the amount of buffering given to the pools of each type.

**Values** Integers 0 to 100

**Default** 50

*mda weight* — The mda keyword is used to identify the weight value for the port where the policy is applied used in the calculation of the amount of buffer space given to the MDA level named pools by the port. The following weight parameter is required and must be specified as an integer between 0 and 100. The specified weight only has meaning when compared to the default and port weights. The sum of all three weights is divided into each weight to determine the amount of buffering given to the pools of each type.

**Values** Integers 0 to 100

**Default** 50

*port weight* — The port keyword is used to identify the weight value for the port where the policy is applied used in the calculation of the amount of buffer space given to the local port named pools by the port. The following weight parameter is required and must be specified as an integer between 0 and 100. The specified weight only has meaning when compared to the mda and port weights. The sum of all three weights is divided into each weight to determine the amount of buffering given to the pools of each type.

**Values** Integers 0 to 100

**Default** 50

## pool

**Syntax** **pool** *pool-name* **create**
**no pool** *pool-name*

**Context** config>qos>named-pool-policy>q1-pools

**Description** The pool command is used to create a new or edit an existing named pool within the policy. A CLI node is created for the named pool which contains the slope-policy and resv-cbs commands. A named pool created within the q1-pools context may be used by queues on any physical port or MDA where the policy is applied that has Q1 based buffer pools. Once the policy is applied on an MDA or port, creating a new pool will fail if a pool resource is not available for the port or MDA.

When creating a pool, the defined name must be unique within the policy. No other named pool may share the same name.

Once the pool is created, any queues currently on a default pool with a specified pool name the same as the new pool will be moved from the default pool to the new pool.

The **no** form of the command removes a specific named pool from the policy. If an instance of the named pool is currently associated with a created queue, the queue will be moved to the appropriate default pool. Once the pool is deleted, the pool is removed from both the policy and any instance of the pool on an MDA or port is removed. The pool buffers are freed and may be available other named pools.

**Parameters** *pool-name* — The pool-name parameter is required. Each named pool must be uniquely named within the policy. Names of up to 32 ASCII characters are supported with the normal character restrictions. A named pool must be defined prior to creating a queue associated with the named pool.

**Length** Up to 32 characters

**Default** None (All named pools must be explicitly created)

**Values** Up to 57 named pools may be created per policy

*create* — The create keyword is required if creating a new named pool within the policy when the system is configured to require the explicit use of the keyword to prevent accidental object creation. Objects may be accidentally created when this protection is disabled and an object name is mistyped when attempting to edit the object. This keyword is not required when the protection is disabled. The keyword is ignored when the named pool already exists within the policy.

## amber-alarm-threshold

**Syntax** **amber-alarm-threshold** *percentage*
**no amber-alarm-threshold**

**Context** config>qos>named-pool-policy>q1-pools>pool

**Description** This command configures the threshold for the amber alarm on the over-subscription allowed.

Users can selectively enable amber or red alarm thresholds. But if both are enabled (non-zero) then the red alarm threshold must be greater than the amber alarm threshold.

The **no** form of the command reverts to the default value.

**Default** 0

**Parameters** *percentage* — Specifies the amber alarm threshold.

**Values** 1 — 1000

## red-alarm-threshold

**Syntax** **red-alarm-threshold** *percentage*
**no red-alarm-threshold**

**Context** config>qos>named-pool-policy>q1-pools>pool

**Description** This command configures the threshold for the red alarm on the over-subscription allowed.

Users can selectively enable amber or red alarm thresholds. But if both are enabled (non-zero) then the red alarm threshold must be greater than the amber alarm threshold.

The **no** form of the command reverts to the default value.

**Default** 0

**Parameters** *percentage* — Specifies the amber alarm threshold.

**Values** 1 — 1000

## application-weights

**Syntax** **application-weights**

**Context** config>qos>named-pool-policy>q1-pools>pool

**Description** The application-weights CLI node context contains the network and access allocation weights. The network and access application weights are used to divide the network and access buffer space available to the pools between each named pool. When the policy is applied at the MDA level, the network and access application weights are applied to the network and access buffer space given to the MDA named pools by the ingress or egress ports. When the policy is applied at the port level, the network and access application weights are applied to the local port network and access buffer space.

## network-allocation-weight

**Syntax** **network-allocation-weight** *buffer-allocation-weight*
**no network-allocation-weight**

**Context** config>qos>named-pool-policy>q1-pools>pool>application-weights

**Description** The network-allocation-weight command is used to define the weight used when dividing network associated buffer space between the named pools. When the named pool is created on an MDA, the network associated buffer space is summed from all ports. The pool's network allocation weight is divided by the total network allocation weights from all named pools on the MDA. The resulting factor is multiplied by the summed port network associated buffer space to derive the amount of network

buffers applied to the pool. When the named pool is created on a port, the weight is applied against the local ports network associated buffer space to derive the network buffers applied to the pool. A similar process is done for with the access-allocation-weight. The total buffers applied to the pool are the sum of the access and network buffers given to the pool.

Changing the weight does not change the total buffers allocated to the pools, just the ratio of distribution between the pools.

A weight of '0' indicates that the pool will not receive any network associated buffers. If all pools on the port or MDA have a network-allocation-weight equal to 0, the network associated buffer will not be used at that level.

The **no** form of the command returns the pools network allocation weight to the default value of 50.

**Parameters** *buffer-allocation-weight* — The buffer-allocation-weight parameter is required when executing the network-allocation-weight command. A value of 0 to 100 is accepted. The default weight is 50. The weight value may be changed at anytime resulting in a redistribution of network associated buffers among the pools at the MDA or port level.

**Unit** Integer

**Length** 0 to 100

**Default** 50

## access-allocation-weight

**Syntax** **access-allocation-weight** *buffer-allocation-weight*
**no access-allocation-weight**

**Context** config>qos>named-pool-policy>q1-pools>pool>application-weights

**Description** The access-allocation-weight command is used to define the weight used when dividing access associated buffer space between the named pools. When the named pool is created on an MDA, the access associated buffer space is summed from all ports. The pool's access allocation weight is divided by the total access allocation weights from all named pools on the MDA. The resulting factor is multiplied by the summed port access associated buffer space to derive the amount of access buffers applied to the pool. When the named pool is created on a port, the weight is applied against the local ports access associated buffer space to derive the access buffers applied to the pool. A similar process is done for with the network-allocation-weight. The total buffers applied to the pool are the sum of the access and network buffers given to the pool.

Changing the weight does not change the total buffers allocated to the pools, just the ratio of distribution between the pools.

A weight of '0' indicates that the pool will not receive any access associated buffers. If all pools on the port or MDA have a access-allocation-weight equal to 0, the access associated buffer will not be used at that level.

**Parameters** *buffer-allocation-weight* — The buffer-allocation-weight parameter is required when executing the access-allocation-weight command. A value of 0 to 100 is accepted. The default weight is 50.

The weight value may be changed at anytime resulting in a redistribution of access associated buffers among the pools at the MDA or port level.

**Unit** Integer

**Values** 0 to 100

**Default** 50

The **no** access-allocation-weight command is used to return the pools access allocation weight to the default value of 50.

## slope-policy

**Syntax** **slope-policy** *slope-policy-name*
**no slope-policy**

**Context** config>qos>named-pool-policy>q1-pools>pool

**Description** The slope-policy command is used to override the default slope-policy configuration for the named buffer pool. The specified slope-policy-name must exist as a current slope policy name. If the slope policy does not exist, the slope-policy command will fail. If a slope policy is currently associated with a named pool within a named pool policy, the slope policy cannot be removed from the system.

The slope policy contains the High and Low WRED slope definitions that will be used by the pool on each MDA on which the pool is created. If the slope-policy command is not executed or the no slope-policy command is executed, the default slope policy will be associated with the pool.

The **no** form of the command restores the default slope policy to the named pool.

**Parameters** *slope-policy-name* — The slope-policy-name parameter is required and must specify an existing slope policy name. If slope-policy-name does not exist, the slope-policy command will fail.

**Unit** Slope Policy Name

**Default** Default Slope Policy

## resv-cbs

**Syntax** **resv-cbs** *percentage-of-pool-size* **amber-alarm-action step** *percent* **max** [1..100]
**resv-cbs** *percentage-of-pool-size*
**no resv-cbs**

**Context** config>qos>named-pool-policy>q1-pools>pool

**Description** The resv-cbs command is used to override the default reserved CBS size of the pool. The reserved CBS size defines the amount of buffer space within the pool that is not considered shared. When queues request buffers from the pool, they will be either ‘within-CBS’ or ‘above-CBS’. If the queue is ‘within-CBS’ based on the current queue depth and the configured CBS value for the queue, the requested buffer is taken from the reserved portion of the buffer pool. After the queues depth is beyond its configured CBS, the buffer will be taken from the pools shared space. Shared space buffers are subject to the WRED slope function within the buffer pool. If the WRED slopes are enabled, the buffer request may be denied based on WRED drop probability.

The default reserved CBS size of the pool is 30%.

The **no** form of the command restores the default reserved CBS size of 30%.

**Parameters** *percentage-of-pool-size* — The percentage-of-pool-size parameter is required and is an integer specifying a percentage from 0 to 100 percent. Specifying a value of 30 (the default) is equivalent to specifying no resv-cbs.

**Values** 0 — 100

**Default** 30

**amber-alarm-action step *percent* —** specifies the percentage step-size for the reserved CBS size of the pool. When using the default value, the adaptive CBS sizing is disabled. To enable adaptive CBS sizing, **step** *percent* must be set to non-default value along with the **max** parameter. When reserved CBS is default adaptive CBS sizing cannot be enabled. The reserved CBS (Committed Burst Size) defines the amount of buffer space within the pool that is not considered shared.

**Values** 1 — 100

**Default** 0

**max [1..100] —** Specifies the maximum percentage for the reserved CBS size of the pool. When using the default value, the adaptive CBS sizing is disabled. To enable adaptive CBS sizing, **max** value must be set to non-default value along with the **step** *percent*. When reserved CBS is default adaptive CBS sizing cannot be enabled. The reserved CBS (Committed Burst Size) defines the amount of buffer space within the pool that is not considered shared. Max reserved CBS must not be more than the reserved CBS.

**Values** 1 — 100

**Default** 0

# Service Ingress and Egress QoS Policy Commands

## sap-ingress

**Syntax** [**no**] **sap-ingress** *policy-id*

**Context** config>qos

**Description** This command is used to create or edit the ingress policy. The ingress policy defines the Service Level Agreement (SLA) enforcement service packets receive as they ingress a SAP. SLA enforcement is accomplished through the definition of queues that have Forwarding Class (FC), Committed Information Rate (CIR), Peak Information Rate (PIR) and Maximum Burst Size (MBS) characteristics. The simplest policy defines a single queue that all ingress traffic flows through. Complex policies have multiple queues combined with specific IP or MAC match criteria that indicate which queue a packet will flow though.

Policies in effect are templates that can be applied to multiple services as long as the **scope** of the policy is template. Queues defined in the policy are not instantiated until a policy is applied to a service SAP.

A SAP ingress policy is considered incomplete if it does not include definition of at least one queue and does not specify the default action. The router software does not allow incomplete SAP ingress policies to be applied to services.

SAP ingress policies can be defined with either IP headers as the match criteria or MAC headers as the match criteria. The IP and MAC criteria are mutually exclusive and cannot be part of the same SAP ingress policy.

It is possible that a SAP ingress policy will include the **dscp** map command, the **dot1p** map command and an IP or MAC match criteria. When multiple matches occur for the traffic, the order of precedence will be used to arrive at the final action. The order of precedence is as follows:

1. 802.1p bits
2. DSCP
3. IP Quintuple or MAC headers

The SAP ingress policy with *policy-id* 1 is a system-defined policy applied to services when no other policy is explicitly specified. The system SAP ingress policy can be modified but not deleted. The **no sap-ingress** command restores the factory default settings when used on *policy-id* 1. The default SAP ingress policy defines one queue associated with the best effort (`be`) forwarding class, with CIR of zero and PIR of line rate.

Any changes made to the existing policy, using any of the sub-commands are applied immediately to all services where this policy is applied. For this reason, when many changes are required on a policy, it is recommended that the policy be copied to a work area policy ID. That work-in-progress policy can be modified until complete and then written over the original policy-id. Use the **config qos copy** command to maintain policies in this manner.

The **no sap-ingress** *policy-id* command deletes the SAP ingress policy. A policy cannot be deleted until it is removed from all services where it is applied. The system default sap-ingress policy is a special case; the **no** command restores the factory defaults to policy-id 1.

The **no sap-ingress** *policy-id* **bw-reserved** command removes the bandwidth reservation attribute from the sap-ingress policy.

**Parameters** *policy-id* — The *policy-id* uniquely identifies the policy.

**Values** 1 — 65535

## queue

**Syntax** **queue** *queue-id* [**multipoint**] [*queue-type*] [*queue-mode*] **pool** *pool-name*
**queue** *queue-id* [**multipoint**] [*queue-type*] **pool** *pool-name*
**no queue** *queue-id*

**Context** config>qos>sap-ingress
config>qos>sap-egress

**Description** This command creates the context to configure an ingress service access point (SAP) QoS policy queue.

Explicit definition of an ingress queue's hardware scheduler status is supported. A single ingress queue allows support for multiple forwarding classes. The default behavior automatically chooses the expedited or non-expedited nature of the queue based on the forwarding classes mapped to it. As long as all forwarding classes mapped to the queue are expedited (`nc`, `ef`, `h1` or `h2`), the queue is treated as an expedited queue by the hardware schedulers. When any non-expedited forwarding classes are mapped to the queue (`be`, `af`, `l1` or `l2`), the queue is treated as best effort (`be`) by the hardware schedulers. The expedited hardware schedulers are used to enforce expedited access to internal switch fabric destinations. The hardware status of the queue must be defined at the time of queue creation within the policy.

The **queue** command allows the creation of multipoint queues. Only multipoint queues can receive ingress packets that need flooding to multiple destinations. By separating the unicast for multipoint traffic at service ingress and handling the traffic on separate multipoint queues special handling of the multipoint traffic is possible. Each queue acts as an accounting and (optionally) shaping device offering precise control over potentially expensive multicast, broadcast and unknown unicast traffic. Only the back-end support of multipoint traffic (between the forwarding class and the queue based on forwarding type) needs to be defined. The individual classification rules used to place traffic into forwarding classes are not affected. Queues must be defined as multipoint at the time of creation within the policy.

The multipoint queues are for multipoint-destined service traffic. Within non-multipoint services, such as Epipe services, all traffic is considered unicast due to the nature of the service type. Multicast and broadcast-destined traffic in an Epipe service will not be mapped to a multipoint service queue.

When an ingress SAP QoS policy with multipoint queues is applied to an Epipe SAP, the multipoint queues are not created. When an ingress SAP QoS policy with multipoint queues is applied to an IES SAP, a multipoint queue will be created when PIM is enabled on the IES interface.

Any billing or statistical queries about a multipoint queue on a non-multipoint service returns zero values. Any queue parameter information requested about a multipoint queue on a non-multipoint service returns the queue parameters in the policy. Buffers will not be allocated for multipoint queues on non-multipoint services. Buffer pool queries return zero values for actual buffers allocated and current buffer utilization.

The **no** form of this command removes the *queue-id* from the SAP ingress QoS policy and from any existing SAPs using the policy. If any forwarding class forwarding types are mapped to the queue, they revert to their default queues. When a queue is removed, any pending accounting information for each SAP queue created due to the definition of the queue in the policy is discarded.

The pool keyword is a create time parameter that allows the queue to receive buffers from an explicit buffer pool instead of the default buffer pool. The specified pool-name must have been explicitly created in a named-pool-policy and the policy must have been applied to the MDA or port on which the queue resides.

If the specified pool-name does not exist on the MDA, the queue will be treated as 'pool orphaned' and will be mapped to the appropriate default pool. Once the pool comes into existence on the MDA or port, the queue will be mapped to the new pool.

Once the queue is created within the policy, the pool command may be used to either remove the queue from the pool, or specify a new pool name association for the queue. The pool command does not appear in save or show command output. Instead, the current pool name for the queue will appear (or not appear) on the queue command output using the pool keyword.

**Parameters** *queue-id* — The *queue-id* for the queue, expressed as an integer. The *queue-id* uniquely identifies the queue within the policy. This is a required parameter each time the queue command is executed.

**Values** 1 — 32

*queue-type* — The **expedite**, **best-effort** and **auto-expedite** queue types are mutually exclusive to each other. Each defines the method that the system uses to service the queue from a hardware perspective. While parental virtual schedulers can be defined for the queue, they only enforce how the queue interacts for bandwidth with other queues associated with the same scheduler hierarchy. An internal mechanism that provides access rules when the queue is vying for bandwidth with queues in other virtual schedulers is also needed. A keyword must be specified at the time the queue is created in the SAP ingress policy. If an attempt to change the keyword after the queue is initially defined, an error is generated.

**expedite —** This keyword ensures that the queue is treated in an expedited manner independent of the forwarding classes mapped to the queue.

**best-effort —** This keyword ensures that the queue is treated in a non-expedited manner independent of the forwarding classes mapped to the queue.

**auto-expedite —** This keyword allows the system to auto-define the way the queue is serviced by the hardware. When **auto-expedite** is defined on the queue, the queue is treated in an expedited manner when all forwarding classes mapped to the queue are configured as expedited types `nc`, `ef`, `h1` or `h2`. When a single non-expedited forwarding class is mapped to the queue (`be`, `af`, `l1` and `l2`) the queue automatically falls back to non-expedited status.

**Values** expedite, best-effort, auto-expedite

**Default** auto-expedite

**multipoint —** This keyword specifies that this *queue-id* is for multipoint forwarded traffic only. This *queue-id* can only be explicitly mapped to the forwarding class multicast, broadcast, or unknown unicast ingress traffic. If you attempt to map forwarding class unicast traffic to a multipoint queue, an error is generated and no changes are made to the current unicast traffic queue mapping.

A queue must be created as multipoint. The **multipoint** designator cannot be defined after the queue is created. If an attempt is made to modify the command to include the **multipoint** keyword, an error is generated and the command will not execute.

The **multipoint** keyword can be entered in the command line on a pre-existing multipoint queue to edit *queue-id* parameters.

**Values** multipoint or not present

**Default** Present (the queue is created as non-multipoint)

*queue-mode* — Specifies the mode in which the queue is operating. This attribute is associated with the queue at the time of creation and cannot be modified thereafter.

**Values** **profile-mode**: When the queue is operating in the profile mode (or, the color aware mode), the queue tries to provide the appropriate bandwidth to the packets with different profiles. The profiles are assigned according to the configuration of the forwarding class or the sub-forwarding class.

**priority-mode**: The queue is capable of handling traffic differently with two distinct priorities. These priorities are assigned by the stages preceding the queueing framework in the system. In priority mode, the queue does not have the functionality to support the profiled traffic and in such cases the queue will have a degraded performance. However, the converse is not valid and a queue in profile mode should be capable of supporting the different priorities of traffic.

*pool-name* — The specified pool-name identifies a named pool where the policy will be applied. Each queue created within the system is tied to a physical port. When the policy is applied and the queue is created, the system will scan the named pools associated with the port to find the specified pool name. If the pool is not found on the port, the system will then look at named pools defined at the ports MDA level. If the pool name is not found on either the port or MDA, the queue will be marked as 'pool-orphaned' and will be mapped to the appropriate default pool. If the pool comes into existence, the queue will be moved from the default pool to the new named pool and the 'pool-orphaned' state will be cleared. The specified name must be an ASCII name string up to 16 characters long.

**Values** Any valid ASCII name string

**Default** None

The queue's pool association may only be removed by either re-executing the queue command without the pool keyword or by executing the no pool command within the queue's CLI context. When the pool name is removed, the queue will be placed on the appropriate default pool.

## pool

**Syntax** **pool** *pool-name* [**create**]
**no pool** *pool-name*

**Context** config>qos>sap-ingress>queue
config>qos>sap-egress>queue

**Description** This command is utilized once the queue is created within the policy. The pool command may be used to either remove the queue from the pool, or specify a new pool name association for the queue. The

pool command does not appear in save or show command output. Instead, the current pool name for the queue will appear (or not appear) on the queue command output using the pool keyword.

The **no** form of the command removes a named pool association for the queue. When the pool name is removed, the queue will be placed on the appropriate default pool.

**Parameters** *pool-name* — The specified pool-name identifies a named pool where the policy will be applied. Each queue created within the system is tied to a physical port. When the policy is applied and the queue is created, the system will scan the named pools associated with the port to find the specified pool name. If the pool is not found on the port, the system will then look at named pools defined at the ports MDA level. If the pool name is not found on either the port or MDA, the queue will be marked as 'pool-orphaned' and will be mapped to the appropriate default pool. If the pool comes into existence, the queue will be moved from the default pool to the new named pool and the 'pool-orphaned' state will be cleared. The specified name must be an ASCII name string up to 32 characters long.

**Default** **None**

## sap-egress

**Syntax** [**no**] **sap-egress** *policy-id*

**Context** config>qos

**Description** This command is used to create or edit a Service Egress QoS policy. The egress policy defines the Service Level Agreement (SLA) for service packets as they egress on the SAP.

Policies in effect are templates that can be applied to multiple services as long as the scope of the policy is template. The queues defined in the policy are not instantiated until a policy is applied to a service.

A sap-egress policy differs from sap-ingress policies in the complexity of the QoS parameters that can be defined. At ingress, policies determine queue mappings based on ingress DSCP, Dot1P and IP or MAC match criteria. Multiple queues can be created per forwarding class and each queue can have different CIR or PIR parameters.

At egress, the policies are much simpler, as the forwarding class and in or out of profile determination happened way back at the original service ingress SAP. Egress SAP QoS policies allow the definition of queues and the mapping of forwarding classes to those queues. Each queue needs to have a relative CIR for determining its allocation of QoS resources during periods of congestion. A PIR can also be defined that forces a hard limit on the packets transmitted through the queue. When the forwarding class is mapped to the queue, a Dot1p value can optionally be specified. If specified and the SAP has a Dot1q encapsulation type, the Dot1p value will be used for all packets that egress on that forwarding class. If the Dot1p value is not specified, a Dot1p value of zero will be used. If the SAP is null encapsulated, or on a SONET/SDH interface, the Dot1p value has no meaning.

Any unmapped traffic or FC will go to queue 1 (or 11 in case of B/U/M traffic).

The sap-egress policy with policy-id 1 is the default sap-egress QoS policy and is applied to service egress SAPs when an explicit policy is not specified or removed. The system sap-egress policy can be modified but not deleted. Using the **no sap-egress** command on **policy-id 1** causes it to revert to its factory default parameters.

The factory default settings for sap-egress policy-id 1 define a single queue with PIR set to the maximum value and a CIR set to 25. The single queue is the default queue and all forwarding classes will map to it. Packets being tagged according to the SAP encapsulation defined will have the Dot1p bits set to zero.

Any changes made to an existing policy, using any of the sub-commands, will be applied immediately to all egress SAPs where this policy is applied. For this reason, when many changes are required on a policy, it is highly recommended that the policy be copied to a work area policy-id. That work-in-progress policy can be modified until complete and then written over the original policy-id. Use the **config qos copy** command to maintain policies in this manner.

The **no** form of the command deletes the sap-egress policy. A policy cannot be deleted until it is removed from all service SAPs where it is applied. When a sap-egress policy is removed from a SAP, the SAP will revert to the default sap-egress policy-id 1.

The system default sap-egress policy is a special case. The **no** command restores the factory defaults to policy-id 1.

**Parameters** *policy-id* — The policy-id uniquely identifies the policy on the router.

**Default** none

**Values** 1 — 65535

# Shared Queue QoS Commands

## shared-queue

**Syntax** **shared-queue** *policy-name*

**Context** config>qos

**Description** This command enables the context to modify the QoS **default** shared-queue policy.

**Parameters** *policy-name* — The name of the **default** shared-queue policy.

**Values** **default**

## queue

**Syntax** **queue** *queue-id* [*queue-type*] [**profile-mode | priority-mode**] [**multipoint**] **pool** *pool-name*
**queue** *queue-id* [*queue-type*] [**multipoint**] **pool** *pool-name*
**no queue** *queue-id*

**Context** config>qos>shared-queue

**Description** This command creates the context to configure a shared queue QoS policy queue.
Explicit definition of an ingress queue's hardware scheduler status is supported. A single ingress queue allows support for multiple forwarding classes. The default behavior automatically chooses the expedited or non-expedited nature of the queue based on the forwarding classes mapped to it. As long as all forwarding classes mapped to the queue are expedited (`nc`, `ef`, `h1` or `h2`), the queue is treated as an expedited queue by the hardware schedulers. When any non-expedited forwarding classes are mapped to the queue (`be`, `af`, `l1` or `l2`), the queue is treated as best effort (`be`) by the hardware schedulers. The expedited hardware schedulers are used to enforce expedited access to internal switch fabric destinations. The hardware status of the queue must be defined at the time of queue creation within the policy.

The pool keyword is a create time parameter that allows the queue to receive buffers from an explicit buffer pool instead of the default buffer pool. The specified pool-name must have been explicitly created in a named-pool-policy and the policy must have been applied to the MDA or port on which the queue resides.

If the specified pool-name does not exist on the MDA, the queue will be treated as 'pool orphaned' and will be mapped to the appropriate default pool. Once the pool comes into existence on the MDA or port, the queue will be mapped to the new pool.

Once the queue is created within the policy, the pool command may be used to either remove the queue from the pool, or specify a new pool name association for the queue. The pool command does not appear in save or show command output. Instead, the current pool name for the queue will appear (or not appear) on the queue command output using the pool keyword.

**Parameters** *queue-id* — The *queue-id* for the queue, expressed as an integer. The *queue-id* uniquely identifies the queue within the policy. This is a required parameter each time the queue command is executed.

**Values** 1 — 32

*queue-type* — The **expedite**, **best-effort** and **auto-expedite** queue types are mutually exclusive to each other. Each defines the method that the system uses to service the queue from a hardware perspective. While parental virtual schedulers can be defined for the queue, they only enforce how the queue interacts for bandwidth with other queues associated with the same scheduler hierarchy. An internal mechanism that provides access rules when the queue is vying for bandwidth with queues in other virtual schedulers is also needed. A keyword must be specified at the time the queue is created in the SAP ingress policy. If an attempt to change the keyword after the queue is initially defined, an error is generated.

expedite — This keyword ensures that the queue is treated in an expedited manner independent of the forwarding classes mapped to the queue.

best-effort — This keyword ensures that the queue is treated in a non-expedited manner independent of the forwarding classes mapped to the queue.

auto-expedite — This keyword allows the system to auto-define the way the queue is serviced by the hardware. When **auto-expedite** is defined on the queue, the queue is treated in an expedited manner when all forwarding classes mapped to the queue are configured as expedited types `nc`, `ef`, `h1` or `h2`. When a single non-expedited forwarding class is mapped to the queue (`be`, `af`, `l1` and `l2`) the queue automatically falls back to non-expedited status.

**Default** auto-expedite

**multipoint —** This keyword specifies that this *queue-id* is for multipoint forwarded traffic only. This *queue-id* can only be explicitly mapped to the forwarding class multicast, broadcast, or unknown unicast ingress traffic. If you attempt to map forwarding class unicast traffic to a multipoint queue, an error is generated and no changes are made to the current unicast traffic queue mapping.

A queue must be created as multipoint. The **multipoint** designator cannot be defined after the queue is created. If an attempt is made to modify the command to include the **multipoint** keyword, an error is generated and the command will not execute.

The **multipoint** keyword can be entered in the command line on a pre-existing multipoint queue to edit *queue-id* parameters.

*pool-name* — The specified pool-name identifies a named pool where the policy will be applied. Each queue created within the system is tied to a physical port. When the policy is applied and the queue is created, the system will scan the named pools associated with the port to find the specified pool name. If the pool is not found on the port, the system will then look at named pools defined at the ports MDA level. If the pool name is not found on either the port or MDA, the queue will be marked as 'pool-orphaned' and will be mapped to the appropriate default pool. If the pool comes into existence, the queue will be moved from the default pool to the new named pool and the 'pool-orphaned' state will be cleared. The specified name must be an ASCII name string up to 16 characters long.

**Values** Any valid ASCII name string

**Default** None

The queue's pool association may only be removed by either re-executing the queue command without the pool keyword or by executing the no pool command within the queue's CLI context. When the pool name is removed, the queue will be placed on the appropriate default pool.

## pool

**Syntax** **pool** *pool-name* **[create]**
**no pool** *pool-name*

**Context** config>qos>shared-queue>queue

**Description** This command is utilized once the queue is created within the policy. The pool command may be used to either remove the queue from the pool, or specify a new pool name association for the queue. The pool command does not appear in save or show command output. Instead, the current pool name for the queue will appear (or not appear) on the queue command output using the pool keyword.

The **no** form of the command removes a named pool association for the queue. When the pool name is removed, the queue will be placed on the appropriate default pool.

**Parameters** *pool-name* — The specified pool-name identifies a named pool where the policy will be applied. Each queue created within the system is tied to a physical port. When the policy is applied and the queue is created, the system will scan the named pools associated with the port to find the specified pool name. If the pool is not found on the port, the system will then look at named pools defined at the ports MDA level. If the pool name is not found on either the port or MDA, the queue will be marked as 'pool-orphaned' and will be mapped to the appropriate default pool. If the pool comes into existence, the queue will be moved from the default pool to the new named pool and the 'pool-orphaned' state will be cleared. The specified name must be an ASCII name string up to 32 characters long.

**Default** **None**

# Network Queue QoS Policy Commands

## network-queue

**Syntax** [**no**] **network-queue** *policy-name*

**Context** config>qos

**Description** This command creates a context to configure a network queue policy. Network queue policies define the ingress network queuing at the MDA network node level and at the Ethernet port and SONET/SDH path level to define network egress queuing.

Network queue policies define ingress and egress network queues similar to a sap-ingress QoS policy.

**Default** default

**Parameters** *policy-name* — The name of the network queue policy.

**Values** Valid names consist of any string up to 32 characters long composed of printable, 7-bit ASCII characters. If the string contains special characters (#, $, spaces, etc.), the entire string must be enclosed within double quotes.

## queue

**Syntax** **queue** *queue-id* [**multipoint**] [*queue-type*] [*queue-mode*] **pool** *pool-name*
**queue** *queue-id* [**multipoint**] [*queue-type*] **pool** *pool-name*
**no queue** *queue-id*

**Context** config>qos>network-queue

**Description** This command creates the context to configure a QoS network-queue policy queue.

Explicit definition of an ingress queue's hardware scheduler status is supported. A single ingress queue allows support for multiple forwarding classes. The default behavior automatically chooses the expedited or non-expedited nature of the queue based on the forwarding classes mapped to it. As long as all forwarding classes mapped to the queue are expedited (`nc`, `ef`, `h1` or `h2`), the queue is treated as an expedited queue by the hardware schedulers. When any non-expedited forwarding classes are mapped to the queue (`be`, `af`, `l1` or `l2`), the queue is treated as best effort (`be`) by the hardware schedulers. The expedited hardware schedulers are used to enforce expedited access to internal switch fabric destinations. The hardware status of the queue must be defined at the time of queue creation within the policy.

The **queue** command allows the creation of multipoint queues. Only multipoint queues can receive ingress packets that need flooding to multiple destinations. By separating the unicast for multipoint traffic at service ingress and handling the traffic on separate multipoint queues, special handling of the multipoint traffic is possible. Each queue acts as an accounting and (optionally) shaping device offering precise control over potentially expensive multicast, broadcast and unknown unicast traffic. Only the back-end support of multipoint traffic (between the forwarding class and the queue based on forwarding type) needs to be defined. The individual classification rules used to place traffic into for-

warding classes are not affected. Queues must be defined as multipoint at the time of creation within the policy.

The multipoint queues are for multipoint-destined service traffic. Within non-multipoint services, such as Epipe services, all traffic is considered unicast due to the nature of the service type. Multicast and broadcast-destined traffic in an Epipe service will not be mapped to a multipoint service queue.

When a QoS policy with multipoint queues is applied to an Epipe or IES SAP, the multipoint queues are not created. Any billing or statistical queries about a multipoint queue on a non-multipoint service returns zero values. Any queue parameter information requested about a multipoint queue on a non-multipoint service returns the queue parameters in the policy. Buffers will not be allocated for multipoint queues on non-multipoint services. Buffer pool queries return zero values for actual buffers allocated and current buffer utilization.

The **no** form of this command removes the *queue-id* from the network-queue policy and from any existing SAPs using the policy. If any forwarding class forwarding types are mapped to the queue, they revert to their default queues. When a queue is removed, any pending accounting information for each SAP queue created due to the definition of the queue in the policy is discarded.

The pool keyword is a create time parameter that allows the queue to receive buffers from an explicit buffer pool instead of the default buffer pool. The specified pool-name must have been explicitly created in a named-pool-policy and the policy must have been applied to the MDA or port on which the queue resides.

If the specified pool-name does not exist on the MDA, the queue will be treated as 'pool orphaned' and will be mapped to the appropriate default pool. Once the pool comes into existence on the MDA or port, the queue will be mapped to the new pool.

Once the queue is created within the policy, the pool command may be used to either remove the queue from the pool, or specify a new pool name association for the queue. The pool command does not appear in save or show command output. Instead, the current pool name for the queue will appear (or not appear) on the queue command output using the pool keyword.

The queue's pool association may only be removed by either re-executing the **queue** command without the **pool** keyword or by executing the **no pool** command within the queue's CLI context. When the pool name is removed, the queue will be placed on the appropriate default pool.

**Parameters** *queue-id* — The *queue-id* for the queue, expressed as an integer. The *queue-id* uniquely identifies the queue within the policy. This is a required parameter each time the queue command is executed.

**Values** 1 — 32

*queue-type* — The **expedite**, **best-effort** and **auto-expedite** queue types are mutually exclusive to each other. Each defines the method that the system uses to service the queue from a hardware perspective. While parental virtual schedulers can be defined for the queue, they only enforce how the queue interacts for bandwidth with other queues associated with the same scheduler hierarchy. An internal mechanism that provides access rules when the queue is vying for bandwidth with queues in other virtual schedulers is also needed. A keyword must be specified at the time the queue is created in the network-queue policy. If an attempt is made to change the keyword after the queue is initially defined, an error is generated.

**expedite —** This keyword ensures that the queue is treated in an expedited manner independent of the forwarding classes mapped to the queue.

**best-effort —** This keyword ensures that the queue is treated in a non-expedited manner independent of the forwarding classes mapped to the queue.

**auto-expedite —** This keyword allows the system to auto-define the way the queue is serviced by the hardware. When **auto-expedite** is defined on the queue, the queue is treated in an expedited manner when all forwarding classes mapped to the queue are configured as expedited types `nc`, `ef`, `h1` or `h2`. When a single non-expedited forwarding class is mapped to the queue (`be`, `af`, `l1` and `l2`) the queue automatically falls back to non-expedited status.

**Values** expedite, best-effort, auto-expedite

**Default** auto-expedite

**multipoint —** This keyword specifies that this *queue-id* is for multipoint forwarded traffic only. This *queue-id* can only be explicitly mapped to the forwarding class multicast, broadcast, or unknown unicast ingress traffic. If you attempt to map forwarding class unicast traffic to a multipoint queue, an error is generated and no changes are made to the current unicast traffic queue mapping.

A queue must be created as multipoint. The **multipoint** designator cannot be defined after the queue is created. If an attempt is made to modify the command to include the **multipoint** keyword, an error is generated and the command will not execute.

The **multipoint** keyword can be entered in the command line on a pre-existing multipoint queue to edit *queue-id* parameters.

**Values** multipoint or not present

**Default** Not present (the queue is created as non-multipoint)

*queue-mode* — Specifies the mode in which the queue is operating. This attribute is associated with the queue at the time of creation and cannot be modified thereafter.

**Values** **profile-mode**: When the queue is operating in the profile mode (or, the color aware mode), the queue tries to provide the appropriate bandwidth to the packets with different profiles. The profiles are assigned according to the configuration of the forwarding class or the sub-forwarding class.

**priority-mode**: The queue is capable of handling traffic differently with two distinct priorities. These priorities are assigned by the stages preceding the queueing framework in the system. In priority mode, the queue does not have the functionality to support the profiled traffic and in such cases the queue will have a degraded performance. However, the converse is not valid and a queue in profile mode should be capable of supporting the different priorities of traffic.

*pool-name* — The specified pool-name identifies a named pool where the policy will be applied. Each queue created within the system is tied to a physical port. When the policy is applied and the queue is created, the system will scan the named pools associated with the port to find the specified pool name. If the pool is not found on the port, the system will then look at named pools defined at the ports MDA level. If the pool name is not found on either the port or MDA, the queue will be marked as 'pool-orphaned' and will be mapped to the appropriate default pool. If the pool comes into existence, the queue will be moved from the default pool to the new named pool and the 'pool-orphaned' state will be cleared. The specified name must be an ASCII name string up to 16 characters long.

**Values** Any valid ASCII name string

**Default** None

## pool

**Syntax** **pool** *pool-name* **[create]**
**no pool** *pool-name*

**Context** config>qos>network-queue>queue

**Description** This command is utilized once the queue is created within the policy. The pool command may be used to either remove the queue from the pool, or specify a new pool name association for the queue. The pool command does not appear in save or show command output. Instead, the current pool name for the queue will appear (or not appear) on the queue command output using the **pool** keyword.

The **no** form of the command removes a named pool association for the queue. When the pool name is removed, the queue will be placed on the appropriate default pool.

**Parameters** *pool-name* — The specified pool-name identifies a named pool where the policy will be applied. Each queue created within the system is tied to a physical port. When the policy is applied and the queue is created, the system will scan the named pools associated with the port to find the specified pool name. If the pool is not found on the port, the system will then look at named pools defined at the ports MDA level. If the pool name is not found on either the port or MDA, the queue will be marked as 'pool-orphaned' and will be mapped to the appropriate default pool. If the pool comes into existence, the queue will be moved from the default pool to the new named pool and the 'pool-orphaned' state will be cleared. The specified name must be an ASCII name string up to 32 characters long.

**Default** None

# Show Commands

# Named Pool Show Commands

## named-pool-policy

**Syntax** **named-pool-policy**
**named-pool-policy** *policy-name* [**detail**]
**named-pool-policy** *policy-name* **association**

**Context** show>qos

**Description** This command displays information on named pool policies.

**Parameters** *policy-name* — Each named pool policy must be uniquely named within the system. Names of up to 32 ASCII characters are supported with the normal character restrictions. A named pool policy must exist prior to applying the policy name to an MDA or port.

**Values** Up to 32 character ASCII string

**detail —** Displays detailed information on the given policy name.

**association —** Displays the associations connected to the given policy name.

**Sample Output**

```
show qos named-pool-policy
===============================================================================
Named-Pool Policies
===============================================================================
Policy Name                     Description
-------------------------------------------------------------------------------
test                            (not-specified)
test57                          (not-specified)
```

## sap-ingress

**Syntax** **sap-ingress** *policy-id* **detail**

**Context** show>qos

**Description** This command displays pools associated/configured to a queue.

**Sample Output**

```
B:SR7-10# show qos sap-ingress 2 detail
===============================================================================
QoS Sap Ingress
```

```
-------------------------------------------------------------------------------
Sap Ingress Policy (2)
-------------------------------------------------------------------------------
Policy-id      : 2                              Scope        : Template
Default FC     : be                             Priority     : Low
Criteria-type  : None
Description    : for ingress traffic
-------------------------------------------------------------------------------
Queue Mode    CIR Admin PIR Admin CBS     HiPrio  PIR Lvl/Wt    Parent
              CIR Rule  PIR Rule  MBS             CIR Lvl/Wt
              Named-Buffer Pool
-------------------------------------------------------------------------------
1     Prio      0         max     def    def           1/1            None
                closest   closest def                  0/1
                port_1
2     Prio      0         max     def    def           1/1            None
                closest   closest def                  0/1
                port_2
3     Prio      0         max     def    def           1/1            None
                closest   closest def                  0/1
                pool_50
```

## sap-egress

**Syntax** **sap-egress** *policy-id* **detail**

**Context** show>qos

**Description** This command displays pools associated/configured to a queue.

## network-queue

**Syntax** **network-queue** *policy-id* **detail**

**Context** show>qos

**Description** This command displays pools associated/configured to a queue.

## shared-queue

**Syntax** **shared-queue default detail**

**Context** show>qos

**Description** This command displays pool name details pertaining to a shared-queue.

**Sample Output**

```
A:ALA-A>show>qos# shared-queue default detail
===============================================================================
QoS Shared Queue Policy
```

```
-------------------------------------------------------------------------------
Shared Queue Policy (default)
-------------------------------------------------------------------------------
Policy         : default
Description    : Default Shared Queue Policy

-------------------------------------------------------------------------------
Queue CIR       PIR       CBS     MBS     HiPrio  Multipoint Pool-Name
-------------------------------------------------------------------------------
1     0         100       1       50      10      FALSE         pool1
...
-------------------------------------------------------------------------------
A:ALA-A>show>qos#
```

## slope-policy

**Syntax** **slope-policy** *slope-name* **detail**

**Context** show>qos

**Description** This command displays configuration details of a slope policy for a named pool.

**Sample Output**

```
A:ALA-A>show>qos#
..
-------------------------------------------------------------------------------
Named-Pool Associations
-------------------------------------------------------------------------------
Policy-Name                     Pool-Name
-------------------------------------------------------------------------------
test                            p1
testOrig                        p1
===============================================================================
..
A:ALA-A>show>qos#
```

## card

**Syntax** **card** *card-slot* **detail | match “Named Pool Mode”**

**Context** show

**Description** This command checks the card specified named pool mode.

**Sample Output**

```
show card 1 detail | match "Named Pool Mode"
=====================================================
Named Pool Mode               : Configured (Enabled)
=====================================================
```

## mda

**Syntax** **mda** *slot* **detail**
**mda** *slot* **qos** [*ingress* | *egress*] **buffer-allocation** [**detail**]
**mda** *slot* **qos** [*ingress* | *egress*] **orphaned-queues**

**Context** show

**Description** This command displays named pool policies configured for an MDA.

### Sample Output

```
show mda 1/2 detail
-------------------------------------------------------------------------------
QOS Settings
-------------------------------------------------------------------------------
Ing. Named Pool Policy            : test
Egr. Named Pool Policy            : test
===============================================================================

A:SR7-10# show mda 1/2 qos ingress buffer-allocation

Total buffer space:  122605 kBytes
Total default buffer space (kBytes): Access:         8171 Network:        12256
Total mda buffer space (kBytes):     Access:        16347 Network:        61306
===============================================================================
Port Allocation Weights
===============================================================================
Port     Percent Total Bw     Total Buffer Default %    Mda %         Port %
-------------------------------------------------------------------------------
1/2/1    100     1000000      12260        16           33            50
1/2/2    100     1000000      12260        16           33            50
1/2/3    100     1000000      12260        16           33            50
1/2/4    100     1000000      12260        16           33            50
1/2/5    100     1000000      12260        16           33            50
1/2/6    100     1000000      12260        16           33            50
1/2/7    100     1000000      12260        16           33            50
1/2/8    100     1000000      12260        16           33            50
1/2/9    100     1000000      12260        16           33            50
1/2/10   100     1000000      12260        16           33            50

===============================================================================
Named Pool Information
===============================================================================
Mda/Port Pool Name                                                  Pool Size
-------------------------------------------------------------------------------
1/2      p2                                                         11093
1/2      p3                                                         11093
1/2      port_1                                                     11093
1/2      port_2                                                     11093

A:SR7-10# show mda 1/2 qos ingress buffer-allocation detail

Total buffer space:  122605 kBytes
Total default buffer space (kBytes): Access:         8171 Network:        12256
Total mda buffer space (kBytes):     Access:        16347 Network:        61306
===============================================================================
Port Allocation Weights
```

```
===============================================================================
Port     Percent Total Bw     Total Buffer Default %    Mda %        Port %
         Acc/Net Acc/Net      Acc/Net             Buffer Space
-------------------------------------------------------------------------------
1/2/1    100     1000000      12260        16           33           50
         Access  0            0            0            0            0
         Network 1000000      12260        2042         4086         6130
1/2/2    100     1000000      12260        16           33           50
         Access  1000000      12260        2042         4086         6130
         Network 0            0            0            0            0
1/2/3    100     1000000      12260        16           33           50
         Access  0            0            0            0            0
         Network 1000000      12260        2042         4086         6130
1/2/4    100     1000000      12260        16           33           50
         Access  0            0            0            0            0
         Network 1000000      12260        2042         4086         6130
1/2/5    100     1000000      12260        16           33           50
         Access  1000000      12260        2042         4086         6130
         Network 0            0            0            0            0
1/2/6    100     1000000      12260        16           33           50
         Access  1000000      12260        2042         4086         6130
         Network 0            0            0            0            0
1/2/7    100     1000000      12260        16           33           50
         Access  1000000      12260        2042         4086         6130
         Network 0            0            0            0            0
1/2/8    100     1000000      12260        16           33           50
         Access  0            0            0            0            0
         Network 1000000      12260        2042         4086         6130
1/2/9    100     1000000      12260        16           33           50
         Access  0            0            0            0            0
         Network 1000000      12260        2042         4086         6130
1/2/10   100     1000000      12260        16           33           50
         Access  0            0            0            0            0
         Network 1000000      12260        2042         4086         6130

===============================================================================
Named Pool Information
===============================================================================
Mda/Port Pool Name                                                  Pool Size
               Access                                     Network
         Weight Total  Buffer Space                 Weight Total  Buffer Space
-------------------------------------------------------------------------------
1/2      p2                                                         11093
         50     350    2334                        50     350    8754
1/2      p3                                                         11093
         50     350    2334                        50     350    8754
1/2      port_1                                                     11093
         50     350    2334                        50     350    8754
1/2      port_2                                                     11093
         50     350    2334                        50     350    8754
1/2      port_3                                                     11093
         50     350    2334                        50     350    8754
```

## port

**Syntax** **port** *port-id* **detail**

**Context** show

**Description** This command displays named pool policies configured for a given port.

### Sample Output

```
show port 1/2/10 detail
============================
Ing. Pool Policy   : n/a
Egr. Pool Policy   : test
============================

show port 1/2/10 detail
================================================================================
Ing. Pool % Rate   : 100                        Egr. Pool % Rate : 100
================================================================================
```

## pools

**Syntax** **pools** *port-id*

**Context** show

**Description** This command displays MDA or port pools. If the pool size is zero, there are no queues associated with the pool and the pool is not in use (configured but not instantiated). To display details about an ingress/egress named pool, use the command **show pools 1/2 ingress | egress p2**. The output of the command shows which queues are using the named pool specified.

### Sample Output

```
A:SR7-10# show pools 1/2
================================================================================
Type    Id        App.    Pool Name                   Actual ResvCBS  PoolSize
                                                      Admin ResvCBS
--------------------------------------------------------------------------------
MDA     1/2       Acc-Ing default                     4096            8192
                                                      50%
MDA     1/2       Acc-Ing MC Path Mgnt                10240           20480
                                                      50%
MDA     1/2       Acc-Egr default                     7168            14336
                                                      50%
MDA     1/2       Net-Ing default                     5120            12288
                                                      40%
MDA     1/2       Net-Egr default                     12288           24576
                                                      50%
MDA     1/2       Ingress p1                          0               0
                              Policy: test            30%
```

```
MDA     1/2       Ingress p2                          4096            12288
                                Policy: test          30%
MDA     1/2       Ingress p3                          4096            12288
                                Policy: test          30%
===============================================================================
```