7705 Service Aggregation Router Gen 2

Release 26.3.R1

Quality of Service Guide

3HE 29567 AAAA TQZZA 01
Edition: 01
March 2026

Nokia is committed to diversity and inclusion. We are continuously reviewing our customer documentation and consulting with standards bodies to ensure that terminology is inclusive and aligned with the industry. Our future customer documentation will be updated accordingly.

Table of contents

List of tables **12**

List of figures **13**

1 Getting started **15**

1.1 About this guide 15

1.2 Platforms and terminology 15

1.3 Conventions 16

1.3.1 Precautionary and information messages 16

1.3.2 Options or substeps in procedures and sequential workflows 16

2 QoS policies **18**

2.1 QoS overview 18

2.2 Preclassification 19

2.3 Forwarding classes 20

2.3.1 High-priority classes 21

2.3.2 Assured classes 21

2.3.3 Best-effort classes 22

2.4 Queue parameters 22

2.4.1 Queue ID 22

2.4.2 Unicast or multipoint queue 22

2.4.3 Queue type 22

2.4.4 Queue scheduling 23

2.4.5 Peak information rate 23

2.4.6 Committed information rate 24

2.4.7 Adaptation rule 25

2.4.8 Committed burst size 26

2.4.9 Maximum burst size 26

2.4.10 Queue drop tails 26

2.4.11 Packet markings 28

2.4.12 Queue counters 28

2.4.13 Color aware profiling 29

2.5 QoS policies overview 29

2.5.1 Service versus network QoS 32

2.5.2 QoS policy entities ... 32
2.5.3 Network QoS policies ... 33
2.5.4 Network queue QoS policies ... 36
2.5.5 Service ingress QoS policies ... 36
2.5.5.1 FC mapping based on EXP bits at VLL/VPLS SAP ... 38
2.5.5.2 Egress forwarding class override ... 41
2.5.6 Service egress QoS policies ... 42
2.5.7 Scheduler policies ... 43
2.5.7.1 Virtual hierarchical scheduling ... 45
2.5.7.2 Hierarchical scheduler policies ... 46
2.5.7.3 Tiers ... 47
2.5.7.4 Scheduler policies applied to applications ... 48
2.5.7.5 Scheduler policies applied to SAPs ... 49
2.5.7.6 Scheduler policies applied to customer SLAs ... 49
2.5.7.7 Scheduler policies applied to multiservice sites ... 50
2.5.8 Configuration notes ... 50

3 Match list for QoS policies ... 51

4 Network QoS policies ... 53
4.1 Network QoS policies overview ... 53
4.2 Network ingress ... 53
4.2.1 Network ingress tunnel QoS override ... 54
4.2.2 Network ingress IP match criteria ... 54
4.2.3 Network ingress IPv6 match criteria ... 55
4.3 Network egress ... 55
4.3.1 Egress packet reclassification based on IP precedence DSCP ... 56
4.3.2 Network egress IP match criteria ... 56
4.3.3 Network egress IPv6 match criteria ... 57
4.4 QoS for self-generated (CPU) traffic on network interfaces ... 57
4.4.1 Default DSCP mapping table ... 60
4.5 Basic configurations ... 61
4.5.1 Creating a network QoS policy ... 61
4.5.2 Applying network QoS policies ... 65
4.5.3 Default network QoS policy values ... 65
4.6 Service management tasks ... 70

4.6.1 Deleting QoS policies....70
4.6.2 Removing a policy from the QoS configuration....70
4.6.3 Editing QoS policies....71

5 Network queue QoS policies....72
5.1 Overview....72
5.2 Network queue parent scheduler....72
5.3 Basic configurations....73
5.3.1 Creating a network queue QoS policy....73
5.3.2 Applying network queue QoS policies....73
5.3.2.1 FPs....73
5.3.2.2 Ethernet ports....74
5.3.3 Default network queue policy values....74
5.4 Service management tasks....80
5.4.1 Deleting QoS policies....80
5.4.2 Removing a policy from the QoS configuration....80
5.4.3 Editing QoS policies....80

6 Service ingress and egress QoS policies....81
6.1 Overview....81
6.2 Basic configurations....81
6.3 Service ingress QoS policy....81
6.3.1 Service ingress QoS queue....82
6.3.2 Ingress percent-rate support....83
6.3.3 Ingress forwarding class (FC)....84
6.3.4 Ingress IP match criteria....85
6.3.5 Ingress IPv6 match criteria....86
6.3.6 Tagging of Ingress IP-criteria and IPv6-criteria....86
6.3.7 Ingress criteria classification directly to policer....89
6.3.8 Virtual network identifier classification....90
6.3.9 FC mapping based on EXP bits....91
6.3.10 Storing match criteria entries....92
6.4 Service egress QoS policy....92
6.4.1 Service egress QoS queue....92
6.4.2 Egress percent-rate support....93
6.4.3 Egress queue CBS and MBS as a function of delay....94

6.4.3.1 CBS, MBS, and burst limit as a function of queue delay ... 94
6.4.3.2 CBS and MBS as a function of SAP delay budget ... 94
6.4.4 Dynamic MBS for egress queue group queues ... 95
6.4.5 Egress SAP FC and FP overrides ... 97
6.4.6 Egress IP match criteria ... 98
6.4.7 Egress IPv6 match criteria ... 98
6.4.8 Storing match criteria entries ... 99
6.4.9 Dot1p egress remarking ... 99
6.4.9.1 DEI egress remarking ... 100
6.4.10 DSCP and IP precedence egress remarking ... 103
6.5 Service management tasks ... 103
6.5.1 Applying service ingress and egress policies ... 103
6.5.1.1 Epipe ... 104
6.5.1.2 IES ... 104
6.5.1.3 VPLS ... 105
6.5.1.4 VPRN ... 105
6.5.2 Editing QoS policies ... 106
6.5.3 Deleting QoS policies ... 106
6.5.4 Removing a policy from the QoS configuration ... 106

7 Queue sharing and redirection ... 107
7.1 Queue sharing and redirection ... 107
7.1.1 Supported platforms ... 107
7.2 Queue group applications ... 107
7.2.1 Access SAP queue group applications ... 107
7.2.1.1 Ingress per SAP statistics with ingress queue groups ... 108
7.2.2 Network port queue groups for IP interfaces ... 110
7.2.3 Pseudowire shaping for Layer 2 and Layer 3 services ... 110
7.2.4 Ingress pseudowire shaping ... 110
7.2.5 Egress pseudowire shaping ... 111
7.2.6 QoS on ingress bindings ... 111
7.3 Queue group templates ... 112
7.4 Port queue groups ... 112
7.4.1 Percent-rate support ... 113
7.5 Forwarding plane queue groups ... 113
7.6 Redirection models ... 113

7.7 Access SAP forwarding class-based redirection ... 114
7.7.1 Policy-based redirection ... 114
7.7.2 SAP-based redirection ... 115
7.7.3 Ingress and egress SAP forwarding class redirection association rules ... 116
7.7.3.1 Policy-based provisioning model ... 116
7.7.3.2 SAP-based provisioning model ... 117
7.7.4 Access queue group statistics ... 119
7.7.4.1 Port queue groups ... 119
7.7.4.2 Forwarding plane queue groups ... 119
7.8 Network IP interface forwarding class-based redirection ... 119
7.8.1 Egress network forwarding class redirection association rules ... 120
7.8.2 Egress network IP interface statistics ... 121
7.9 Queue group behavior on LAG ... 121
7.9.1 Queue group queue instantiation per link ... 121
7.9.2 Per-link queue group queue parameters ... 121
7.9.3 Adding a queue group to an existing LAG ... 121
7.9.4 Adding a port to a LAG ... 122
7.9.5 Removing a queue group from a LAG ... 122
7.10 Basic configurations ... 122
7.10.1 Configuring an ingress queue group template ... 122
7.10.2 Configuring an egress queue group template ... 122
7.10.3 Applying ingress queue group to SAP ingress policy ... 123
7.10.4 Applying egress queue group to SAP egress policy ... 123
7.10.5 Configuring SAP-based egress queue redirection ... 124
7.10.6 Configuring queue group on Ethernet access ingress port ... 125
7.10.7 Configuring overrides ... 126
7.10.8 Configuring queue group on Ethernet access egress port ... 127
7.10.9 Configuring queue group for network egress traffic on port ... 128
7.10.10 Configuring queue group for network ingress traffic on forwarding plane ... 128
7.10.11 Using queue groups to police ingress/egress traffic on network interface ... 129
7.10.12 Configuring ingress/egress PW shaping using spoke SDP forwarding class-based redirection ... 130
7.10.13 Specifying QoS policies on service SAPs ... 133

8 Scheduler QoS policies ... 134
8.1 Scheduler policies ... 134

8.1.1 Egress port-based schedulers ... 134
8.1.1.1 Service or multiservice site egress port bandwidth allocation ... 136
8.1.1.2 Service or multiservice site scheduler child to port scheduler parent ... 137
8.1.1.3 Frame and packet-based bandwidth allocation ... 140
8.1.1.4 Parental association scope ... 142
8.1.1.5 Service or subscriber or multiservice site-level scheduler parental association scope ... 142
8.1.1.6 Network queue parent scheduler ... 143
8.1.1.7 Foster parent behavior for orphaned queues and schedulers ... 143
8.1.2 Frame-based accounting ... 144
8.1.2.1 Operational modifications ... 144
8.1.2.2 Existing egress port-based virtual scheduling ... 144
8.1.2.3 Behavior modifications for frame-based accounting ... 145
8.1.2.4 Virtual scheduler rate and queue rate parameter interpretation ... 145
8.1.3 Configuring port scheduler policies ... 145
8.1.3.1 Port scheduler structure ... 145
8.1.3.2 Special orphan queue and scheduler behavior ... 146
8.1.3.3 Packet to frame bandwidth conversion ... 146
8.1.3.4 Aggregate rate limits for directly attached queues ... 146
8.1.3.5 SAP egress QoS policy queue parenting ... 147
8.1.3.6 Network queue QoS policy queue parenting ... 147
8.1.3.7 Egress port scheduler overrides ... 147
8.1.3.8 Applying a port scheduler policy to a virtual port ... 147
8.1.3.9 Weighted scheduler group in a port scheduler policy ... 149
8.2 Basic configurations ... 150
8.2.1 Creating a QoS scheduler policy ... 150
8.2.2 Applying scheduler policies ... 151
8.2.2.1 Customer ... 151
8.2.2.2 Epipe ... 152
8.2.2.3 IES ... 152
8.2.2.4 VPLS ... 153
8.2.2.5 VPRN ... 154
8.2.3 Creating a QoS port scheduler policy ... 154
8.2.4 Configuring port parent parameters ... 155
8.2.4.1 Within-CIR priority level parameters ... 155
8.2.4.2 Above-CIR priority level parameters ... 156

8.3 Service management tasks....156
8.3.1 Deleting QoS policies....157
8.3.1.1 Removing a QoS policy from a customer multiservice site....157
8.3.1.2 Removing a QoS policy from SAPs....157
8.3.1.3 Removing a policy from the QoS configuration....158
8.3.2 Copying and overwriting scheduler policies....158
8.3.3 Editing QoS policies....160

9 Class fair hierarchical policing (CFHP)....161
9.1 Overview....161
9.2 Parent policer priority and unfair sensitive discard thresholds....162
9.3 CFHP ingress and egress use cases....164
9.4 Post-CFHP queuing and scheduling....164
9.4.1 Ingress CFHP queuing....164
9.4.2 Egress CFHP queuing....164
9.4.2.1 Policer to local queue mapping....165
9.4.3 Egress subscriber CFHP queuing....166
9.4.4 SAP default destination string....166
9.5 CFHP policer control policy....166
9.5.1 Policer control policy root arbiter....167
9.5.2 Tier 1 and tier 2 explicit arbiters....167
9.5.3 Explicit arbiter rate limits....167
9.5.4 CFHP with child policer exceed PIR enabled....167
9.6 CFHP child policer definition and creation....168
9.7 Policer enabled SAP QoS policy applicability....168
9.8 Child policer parent association....169
9.9 Profile-capped policers....169
9.10 Policer interaction with profile, discard eligibility, and ingress priority....172
9.10.1 Ingress 'undefined' initial profile....173
9.10.2 Ingress explicitly 'in-profile' state packet handling without profile-capped mode....174
9.10.3 Ingress explicitly 'in-profile' state packet handling with profile-capped mode....174
9.10.4 Ingress explicit 'out-of-profile' state packet handling....174
9.10.5 Egress explicit profile reclassification....174
9.10.6 Preserving out of profile state at egress policer....175
9.10.7 Egress policer CIR packet handling without profile-capped mode....175
9.10.8 Egress policer CIR packet handling with profile-capped mode....175

9.10.9 Forwarding traffic exceeding PIR in egress policers ... 175
9.10.10 Post egress policer packet forwarding class and profile state remapping ... 176
9.10.11 Ingress child policer stat-mode ... 177
9.10.12 Egress child policer stat-mode ... 179
9.11 Profile-preferred mode root policers ... 180
9.12 Child policer hierarchical QoS parenting ... 181

10 Frequently used QoS terms ... 185
10.1 Overview ... 185
10.2 Above-CIR distribution ... 185
10.3 Available bandwidth ... 185
10.4 CBS ... 185
10.5 CIR ... 185
10.6 CIR level ... 186
10.7 CIR weight ... 186
10.8 Child ... 186
10.9 Level ... 186
10.10 MBS ... 187
10.11 Offered load ... 187
10.12 Orphan ... 187
10.13 Parent ... 187
10.14 Queue ... 187
10.15 Rate ... 188
10.16 Root scheduler ... 188
10.17 Scheduler policy ... 188
10.18 Tier ... 188
10.19 Virtual scheduler ... 189
10.20 Weight ... 189
10.21 Within-CIR distribution ... 189

11 Standards and protocol support ... 190
11.1 Bidirectional Forwarding Detection (BFD) ... 190
11.2 Border Gateway Protocol (BGP) ... 190
11.3 Bridging and management ... 191
11.4 Certificate management ... 192
11.5 Ethernet ... 192

11.6 Ethernet VPN (EVPN) 192
11.7 gRPC Remote Procedure Calls (gRPC) 193
11.8 Intermediate System to Intermediate System (IS-IS) 193
11.9 Internet Protocol (IP) general 194
11.10 Internet Protocol (IP) multicast 195
11.11 Internet Protocol (IP) version 4 196
11.12 Internet Protocol (IP) version 6 196
11.13 Internet Protocol Security (IPsec) 197
11.14 Label Distribution Protocol (LDP) 198
11.15 Multiprotocol Label Switching (MPLS) 199
11.16 Network Address Translation (NAT) 199
11.17 Network Configuration Protocol (NETCONF) 199
11.18 Media sanitization 199
11.19 Open Shortest Path First (OSPF) 200
11.20 Path Computation Element Protocol (PCEP) 200
11.21 Pseudowire (PW) 201
11.22 Quality of Service (QoS) 201
11.23 Remote Authentication Dial In User Service (RADIUS) 202
11.24 Resource Reservation Protocol - Traffic Engineering (RSVP-TE) 202
11.25 Routing Information Protocol (RIP) 202
11.26 Segment Routing (SR) 203
11.27 Simple Network Management Protocol (SNMP) 203
11.28 Timing 205
11.29 Two-Way Active Measurement Protocol (TWAMP) 205
11.30 Virtual Private LAN Service (VPLS) 205
11.31 Yet Another Next Generation (YANG) 206

List of tables

Table 1: Platforms and terminology....15
Table 2: Unsupported dot1p mapping combinations....19
Table 3: Forwarding classes....20
Table 4: QoS policy types and descriptions....31
Table 5: Service types and applicable service QoS policies....32
Table 6: Default network QoS policy egress marking....34
Table 7: Default network QoS policy DSCP to forwarding class mappings....35
Table 8: Forwarding class and enqueuing priority classification hierarchy based on rule type....37
Table 9: Forwarding class classification based on rule type....39
Table 10: MAC match Ethernet frame types....40
Table 11: MAC match criteria frame type dependencies....40
Table 12: Default service ingress policy ID 1 definition....41
Table 13: Default service egress policy ID 1 definition....43
Table 14: Tiers configured with CIR and PIR values....47
Table 15: Default QoS values for self-generated traffic....58
Table 16: Network policy defaults....65
Table 17: Effect of profile-capped mode on CIR output....170

List of figures

Figure 1: Ingress and egress queue drop tails....27
Figure 2: Service vs network traffic types....32
Figure 3: Example configuration — carrier application....38
Figure 4: Egress forwarding class override....42
Figure 5: Virtual scheduler internal bandwidth allocation....45
Figure 6: Hierarchical scheduler and queue association....47
Figure 7: Scheduler policy on SAP and scheduler hierarchy creation....48
Figure 8: Scheduler policy on customer site and scheduler hierarchy creation....49
Figure 9: Bandwidth distribution on network port with port-based scheduling....72
Figure 10: Ingress criteria classification directly to policer....90
Figure 11: DE Bit in the 802.1ad S-tag....101
Figure 12: DE aware 802.1ad access network....102
Figure 13: DEI processing ingress into the PE1 SAP....102
Figure 14: DE aware PBB topology....103
Figure 15: Ingress QoS control on VPRN bindings....111
Figure 16: Port-level virtual scheduler bandwidth allocation based on priority and CIR....137
Figure 17: Two-scheduler policy model for access ports....138
Figure 18: Schedulers on SAP or multiservice site receive bandwidth from port priority levels....139
Figure 19: Direct service or subscriber or multiservice site association to port scheduler model....140
Figure 20: Port bandwidth distribution for service and port scheduler hierarchies....141
Figure 21: Port bandwidth distribution for direct queue to port scheduler hierarchy....141

Figure 22: Bandwidth distribution on network port with port-based scheduling..... 143

Figure 23: Applying a port scheduler policy to a Vport..... 148

Figure 24: Policer bucket rate and packet flow interaction with bucket depth..... 163

Figure 25: Parent policer bucket and priority thresholds..... 163

1 Getting started

1.1 About this guide

This guide describes the Quality of Service (QoS) provided by the routers and presents examples to configure and implement various protocols and services.

This guide is organized into functional chapters and provides concepts and descriptions of the implementation flow, as well as Command Line Interface (CLI) syntax and command usage.

Unless otherwise indicated, the topics and commands described in this guide apply only to the 7705 SAR Gen 2 platforms listed in Platforms and terminology.

Command outputs shown in this guide are examples only; actual displays may differ depending on supported functionality and user configuration.

Note: Unless otherwise indicated, CLI commands, contexts, and configuration examples in this guide apply for both the classic CLI and the MD-CLI.

The SR OS CLI trees and command descriptions can be found in the following guides:

- *7705 SAR Gen 2 Classic CLI Command Reference Guide*
- *7705 SAR Gen 2 Clear, Monitor, Show, Tools CLI Command Reference Guide* (for both the MD-CLI and classic CLI)
- *7705 SAR Gen 2 MD-CLI Command Reference Guide*

Note: This guide generically covers Release 26.*x*.R*x* content and may contain some content that will be released in later maintenance loads. See the *SR OS R26.x.Rx Software Release Notes*, part number 3HE 29176 000*x* TQZZA, for information about features supported in each load of the Release 26.*x*.R*x* software. For a list of features and CLI commands that are present in SR OS but not supported on the 7705 SAR Gen 2 platforms, see "SR OS Features not Supported on SAR Gen 2" in the *SR OS R26.x.Rx Software Release Notes*.

1.2 Platforms and terminology

Note:
Unless explicitly noted otherwise, this guide uses the terminology defined in the following table to collectively designate the specified platforms.

Table 1: Platforms and terminology

Platform	Collective platform designation
7705 SAR-Hx	7705 SAR Gen 2
7705 SAR-Mx	

Platform	Collective platform designation
7705 SAR-1	

1.3 Conventions

This section describes the general conventions used in this guide.

1.3.1 Precautionary and information messages

The following information symbols are used in the documentation.

DANGER: Danger warns that the described activity or situation may result in serious personal injury or death. An electric shock hazard could exist. Before you begin work on this equipment, be aware of hazards involving electrical circuitry, be familiar with networking environments, and implement accident prevention procedures.

WARNING: Warning indicates that the described activity or situation may, or will, cause equipment damage, serious performance problems, or loss of data.

Caution: Caution indicates that the described activity or situation may reduce your component or system performance.

Note: Note provides additional operational information.

Tip: Tip provides suggestions for use or best practices.

1.3.2 Options or substeps in procedures and sequential workflows

Options in a procedure or a sequential workflow are indicated by a bulleted list. In the following example, at step 1, the user must perform the described action. At step 2, the user must perform one of the listed options to complete the step.

Example: Options in a procedure

1. User must perform this step.
2. This step offers three options. User must perform one option to complete this step.
   - This is one option.
   - This is another option.
   - This is yet another option.

Substeps in a procedure or a sequential workflow are indicated by letters. In the following example, at step 1, the user must perform the described action. At step 2, the user must perform two substeps (a. and b.) to complete the step.

Example: Substeps in a procedure

1. User must perform this step.
2. User must perform all substeps to complete this action.
   a. This is one substep.
   b. This is another substep.

Nested substeps within a procedure or a sequential workflow are indicated by roman numerals. In the following example, at step 1, the user must perform the described action. At step 2, the user must perform two substeps (a. and b.) to complete the step. At substep b, the user must perform two additional substeps (i. and ii.) to complete the step.

Example: Nested substeps in a procedure

1. User must perform this step.
2. User must perform all substeps to complete this action.
   a. This is one substep.
   b. User must perform all nested substeps to complete this action.
      i. This is a nested substep.
      ii. This is another nested substep.

2 QoS policies

2.1 QoS overview

Routers are designed with Quality of Service (QoS) mechanisms on both ingress and egress to support multiple customers and multiple services per physical interface. The routers can classify, police, shape, and mark traffic.

In the Nokia service router service model, a service is provisioned on the provider-edge (PE) equipment. Service data is encapsulated, then sent in a service tunnel to the far-end Nokia service router where the service data is delivered.

The operational theory of a service tunnel is that the encapsulation of the data between the two Nokia service routers appears like a Layer 2 path to the service data although it is really traversing an IP or IP/MPLS core. The tunnel from one edge device to the other edge device is provisioned with an encapsulation and the services are mapped to the tunnel that most appropriately supports the service needs.

The router supports eight forwarding classes internally named:

- Network-Control
- High-1
- Expedited
- High-2
- Low-1
- Assured
- Low-2
- Best-Effort

The forwarding classes are described in more detail in Forwarding classes chapter.

Router QoS policies control how QoS is handled at distinct points in the service delivery model within the device. There are different types of QoS policies that cater to the different QoS needs at each point in the service delivery model. QoS policies are defined in a global context in the router and only take effect when the policy is applied to a relevant entity.

QoS policies are uniquely identified with a policy ID number or name. Policy ID 1 or Policy ID "default" is reserved for the default policy that is used if no policy is explicitly applied.

The QoS policies within the router can be divided into three main types:

- QoS policies are used for classification, defining and queuing attributes, and marking.
- Slope policies define default buffer allocations and WRED slope definitions.
- Scheduler policies determine how queues are scheduled.

2.2 Preclassification

If the ingress traffic volume exceeds the processing capacity of the datapath, platforms may indiscriminately discard packets before classification or further action, such as policing. Ingress capacity exhaustion occurs when faceplate port capacity exceeds the datapath capacity. Preclassification performs a high level of discards to ensure that the datapath processing capabilities remain intact. Preclassification by the 7705 SAR Gen 2 ensures sanitized operation of the datapath at all times. The following usage guidelines apply:

- Preclassification is supported on all 7705 SAR Gen 2 platforms and is always enabled. All ports are assigned the default preclassifier policy. In the default preclassifer policy, dot1q encapsulated packets with dot1p marking of 7, 6, 5, and 4 are handled using the following policer configuration:

  ```
  configure qos pre-classifier-policy policer 1
  ```

  The **policer 1** rate and MBS are both set to the following option:

  ```
  configure qos pre-classifier-policy policer pir max
  configure qos pre-classifier-policy policer mbs max
  ```

  Null encapsulation traffic together with dot1q encapsulation packets with a dot1p marking of 3, 2, 1, and 0 are handled using the following policer configuration:

  ```
  configure qos pre-classifier-policy policer 2
  ```

  The **policer 2** rate is set to 75% of the port speed and the MBS is 75% of the policer buffer space.
- Preclassification applies to the following tagged packets:
  - dot1q packets
  - QinQ packets based on outer tags

  Note: Null encapsulation traffic is always handled using the default policer.

- Preclassification is based on dot1p markings of ingress traffic. DiffServe Code Point (DSCP). EXP markings are not used for preclassification.
  The 7705 SAR Gen 2 supports single-rate two color policing for preclassification.
- A custom preclassifier policy can be configured. Each preclassifier policy can support up to 4 different policers. Dot1p-to-policer mapping is flexible and user-configurable, except the combinations listed in the following table.

Table 2: Unsupported dot1p mapping combinations

Combination ID	Dot1p				
1	0	3	5	6	—
2	1	2	4	7	—
3	0	1	2	4	7

Combination ID	Dot1p				
4	0	1	3	5	6
5	0	2	3	5	6
6	0	3	4	5	6
7	0	3	5	6	7
8	1	2	3	4	7
9	1	2	4	5	7
10	1	2	4	6	7

For example, dot1p marking of 0, 3, 5, and 6 cannot be mapped to a single policer, as defined in combination ID 1.

2.3 Forwarding classes

Routers support multiple forwarding classes and class-based queuing, so the concept of forwarding classes is common to all QoS policies.

Each forwarding class, also called Class of Service (CoS), is important only in relation to the other forwarding classes. A forwarding class provides network elements a method to weigh the relative importance of one packet over another in a different forwarding class.

Queues are created for a specific forwarding class to determine how the queue output is scheduled into the switch fabric. The forwarding class of the packet, along with the profile state, determines how the packet is queued and handled (the Per Hop Behavior (PHB)) at each hop along its path to a destination egress point. Routers support eight forwarding classes.

Table 3: Forwarding classes lists the default definitions for the forwarding classes. The forwarding class behavior, in terms of ingress marking interpretation and egress marking, can be changed by Network QoS policies. All forwarding class queues support the concept of in-profile, out-of-profile and, at egress only, inplus-profile and exceed-profile.

Table 3: Forwarding classes

FC-ID	FC name	FC designation	DiffServ name	Class type	Notes
7	Network-Control	NC	NC2	High-Priority	Intended for network control traffic
6	High-1	H1	NC1		Intended for a second network control class or delay/jitter sensitive traffic
5	Expedited	EF	EF		Intended for delay/jitter sensitive traffic
4	High-2	H2	AF4		Intended for delay/jitter sensitive traffic

FC-ID	FC name	FC designation	DiffServ name	Class type	Notes
3	Low-1	L1	AF2	Assured	Intended for assured traffic. Also, is the default priority for network management traffic.
2	Assured	AF	AF1		Intended for assured traffic
1	Low-2	L2	CS1	Best Effort	Intended for BE traffic
0	Best-Effort	BE	BE		

The forwarding classes can be classified into three class types:

- High-priority or Premium
- Assured
- Best-Effort

2.3.1 High-priority classes

The high-priority forwarding classes are Network-Control (nc), Expedited (ef), High-1 (h1), and High-2 (h2). High-priority forwarding classes are always serviced at congestion points over other forwarding classes; this behavior is determined by the router queue scheduling algorithm. See Virtual hierarchical scheduling for more information.

With a strict PHB at each network hop, service latency is mainly affected by the amount of high-priority traffic at each hop. These classes are intended to be used for network control traffic or for delay- or jitter-sensitive services.

If the service core network is oversubscribed, a mechanism to engineer a path through the core network and to reserve bandwidth must be used to apply strict control over the delay and bandwidth requirements of high-priority traffic. In the router, RSVP-TE can be used to create a path defined by an MPLS LSP through the core. Premium services are then mapped to the LSP, with care to not oversubscribe the reserved bandwidth.

If the core network has sufficient bandwidth, it is possible to effectively support the delay and jitter characteristics of high-priority traffic without using traffic engineered paths, as long as the core treats high-priority traffic with the correct PHB.

2.3.2 Assured classes

The assured forwarding classes are Assured (af) and Low 1 (l1). Assured forwarding classes provide services with a committed rate and a peak rate, much like Frame Relay. Packets transmitted through the queue at or below the committed transmission rate are marked in-profile. If the core service network has sufficient bandwidth along the path for the assured traffic, all aggregate in-profile service packets reach the service destination.

Packets transmitted from the service queue that are above the committed rate are marked out-of-profile. When an assured out-of-profile service packet is received at a congestion point in the network, it is discarded before in-profile assured service packets.

Multiple assured classes are supported with relative weighting between them. In DiffServ, the code points for the various Assured classes are AF4, AF3, AF2, and AF1. Typically, AF4 has the highest weight of the four and AF1 the lowest. The Assured and Low-1 classes are differentiated based on the default DSCP mappings. All DSCP and EXP mappings can be modified by the user.

2.3.3 Best-effort classes

The best-effort classes are Low 2 (l2) and Best-Effort (be). The best-effort forwarding classes have no delivery guarantees. All packets within this class are treated by default as out-of-profile assured service packets.

2.4 Queue parameters

This section describes the queue parameters provisioned on access and queues for QoS.

2.4.1 Queue ID

The queue ID is used to uniquely identify the queue. The queue ID is only unique within the context of the QoS policy within which the queue is defined.

2.4.2 Unicast or multipoint queue

Unicast queues are used for all services and routing when the traffic is forwarded to a single destination.

Multipoint ingress queues are used by VPLS services for multicast, broadcast, and unknown traffic and by IES and VPRN services for multicast traffic when IGMP, MLD, or PIM is enabled on the service interface.

2.4.3 Queue type

The type of a queue dictates how it is scheduled relative to other queues at the hardware level. Being able to define the scheduling properties of a queue is important because a single queue allows support for multiple forwarding classes.

The queue type defines the relative priority of the queue and can be configured to **expedited** (higher priority) or **best-effort** (lower) priority. However, the instantaneous scheduling priority of a queue changes dynamically depending on its current scheduling rate compared to its operational Committed Information Rate (CIR) and Fair Information Rate (FIR) (see Queue scheduling). Parental virtual schedulers can be defined for the queue using scheduler policies which enforce how the queue interacts for bandwidth with other queues associated with the same scheduler hierarchy (see Scheduler policies).

The queue type of SAP ingress and egress queues, network queue policy queues, and ingress queue group template queues are defined at queue creation time. The queue type of egress queue group template queues and shared-queue policy queues can be modified after the queue has been created.

The default behavior for SAP ingress and egress queues, network queue policy queues, and shared queue policy queues is to automatically choose the expedited or best effort nature of the queue based on the forwarding classes mapped to it. This is achieved by configuring the queue type to **auto-expedited**. 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 best effort forwarding classes are mapped to the queue (be, af, l1, or l2), the queue is treated as best effort by the hardware schedulers.

The default queue type for ingress queue group template and egress queue group template queues is **best-effort**.

2.4.4 Queue scheduling

Packets are scheduled from queues by hardware schedulers based on the type of the queue (see Queue type) and the current scheduling rate of the queue compared to its operational CIR and FIR. This applies to unicast queues at both ingress and egress, and multipoint queues at ingress, but not to HSQ IOM queues.

The queue type should be chosen based on the kind of traffic in the forwarding classes mapped to the queue.

The hardware scheduler services queues to forward packets from them in a strict priority order, as follows:

- Ingress queues, in priority order:
  1. Expedited queues where the queue's current scheduling rate is below its operational FIR
  2. Best effort queues where the queue's current scheduling rate is below its operational FIR
  3. Expedited queues where the queue's current scheduling rate is below its operational CIR
  4. Best effort queues where the queue's current scheduling rate is below its operational CIR
  5. Expedited queues where the queue's current scheduling rate is above both its operational FIR and CIR
  6. Best effort queues where the queue's current scheduling rate is above both its operational FIR and CIR
- Egress queues, in priority order:
  1. Expedited queues where the queue's current scheduling rate is below its operational FIR
  2. Best effort queues where the queue's current scheduling rate is below its operational FIR
  3. Expedited queues where the queue's current scheduling rate is below its operational CIR
  4. Best effort queues where the queue's current scheduling rate is below its operational CIR
  5. Expedited queues where the queue's current scheduling rate is above both its operational FIR and CIR
  6. Best effort queues where the queue's current scheduling rate is above both its operational FIR and CIR

Note: Only network queues at egress support a non-zero FIR and can use priorities 1 and 2.

2.4.5 Peak information rate

The Peak Information Rate (PIR) defines the maximum rate at which packets are allowed to exit the queue. The PIR does not specify the maximum rate at which packets may enter the queue; this is governed by the queue's ability to absorb bursts and is defined by its maximum burst size (MBS).

The actual transmission rate of a service queue depends on more than just its PIR. Each queue is competing for transmission bandwidth with other queues. Each queue's PIR, CIR, FIR, and the queue

type (see Queue type) all combine to affect a queue's ability to transmit packets, as described in Queue scheduling.

The PIR is provisioned on ingress and egress queues within service ingress and egress QoS policies, network queue policies, ingress and egress queue group templates, and shared queue policies.

When defining the PIR for a queue, the value specified is the administrative PIR for the queue. The router has a number of native rates in hardware that it uses to determine the operational PIR for the queue. The user has some control over how the administrative PIR is converted to an operational PIR should the hardware not support the exact PIR value specified. The interpretation of the administrative PIR is discussed in Adaptation rule.

2.4.6 Committed information rate

The Committed Information Rate (CIR) for a queue performs two distinct functions:

- **profile marking by service ingress queues**

  Service ingress queues (configured in SAP ingress QoS policies or ingress queue group templates) mark packets in-profile or out-of-profile based on the CIR. For each packet in a service ingress queue, the CIR is compared to the current transmission rate of the queue. If the current rate is at or below the CIR threshold, the transmitted packet is internally marked in-profile. If the current rate is above the threshold, the transmitted packet is internally marked out-of-profile. This operation can be overridden by configuring **cir-non-profiling** under the queue. This allows the queue scheduling priority to continue to be based on the below CIR or above CIR of the queue, but packets are not re-profiled depending on the state of the queue when they are scheduled (below CIR or above CIR). Instead, their profile state remains as out-of-profile, unless they are explicitly classified as in-profile or out-of-profile in which case they remain in-profile or out-of-profile.

- **scheduler queue priority**

  The scheduler serving a group of service ingress or egress queues prioritizes individual queues based on their current FIR, CIR, and PIR states. See Queue scheduling for more information about queue scheduling.

All router queues support the concept of in-profile, out-of-profile, together with inplus-profile and exceed-profile at egress only. The network QoS policy applied at network egress determines how or if the profile state is marked in packets transmitted into the service core network. If the profile state is marked in the service core packets, the packets are dropped preferentially at congestion points in the core as follows:

- exceed-profile
- out-of-profile
- in-profile
- inplus-profile

When defining the CIR for a queue, the value specified is the administrative CIR for the queue. The router has a number of native rates in hardware that it uses to determine the operational CIR for the queue. The user has some control over how the administrative CIR is converted to an operational CIR if the hardware does not support the exact CIR specified. See Adaptation rule for more information about the interpretation of the administrative CIR.

Although the router is flexible in how the CIR can be configured, there are conventional ranges for the CIR based on the forwarding class of a queue. A service queue associated with the high-priority class normally has the CIR threshold equal to the PIR rate, although the router allows the CIR to be provisioned to any

rate below the PIR if this behavior is required. If the service queue is associated with a best-effort class, the CIR threshold is normally set to zero; however, this is flexible.

The CIR for a service queue is provisioned in ingress and egress service queues within service ingress QoS policies and service egress QoS policies respectively. CIRs for network queues are defined within network queue policies. CIRs for queue group instance queues are defined within ingress and egress queue group templates.

2.4.7 Adaptation rule

The adaptation rule provides the QoS provisioning system with the ability to adapt specific FIR-, CIR-, and PIR-defined administrative rates to the underlying capabilities of the hardware that the queue is created on to derive the operational rates. The administrative FIR, CIR, and PIR rates are translated to operational rates enforced by the hardware queue. The adaptation rule provides a constraint used when the exact rate is not available because of hardware implementation trade-offs.

For the FIR, CIR, and PIR parameters individually, the system attempts to find the best operational rate depending on the defined constraint. The supported constraints are:

minimum

finds the hardware supported rate that is equal to or higher than the specified rate

maximum

finds the hardware supported rate that is equal to or lesser than the specified rate

closest

finds the hardware supported rate that is closest to the specified rate

Depending on the hardware on which the queue is provisioned, the operational FIR, CIR, and PIR settings used by the queue depends on the method that the hardware uses to implement and represent the mechanisms that enforce the FIR, CIR, and PIR rates.

As the hardware has a very granular set of rates, Nokia's recommended method to determine which hardware rate is used for a queue is to configure the queue rates with the associated adaptation rule and use the **show pools output** command to display the rate achieved.

To illustrate how the adaptation rule constraints (minimum, maximum, and closest) are evaluated in determining the operational CIR or PIR rates, assume there is a queue where the administrative CIR and PIR rates are 401 Mb/s and 403 Mb/s, respectively.

The following output shows the operating CIR and PIR rates achieved for the different adaptation rule settings:

```
*A:PE# # queue using default adaptation-rule=closest
*A:PE# show qos sap-egress 10 detail | match expression "Queue-Id|CIR Rule"
Queue-Id             : 1                Queue-Type           : auto-expedite
PIR Rule             : closest          CIR Rule             : closest
*A:PE#
*A:PE# show pools 1/1/1 access-egress service 1 | match expression "PIR|CIR"
Admin PIR        : 403000               Oper PIR         : 403200
Admin CIR        : 401000               Oper CIR         : 401600
*A:PE#
*A:PE# configure qos sap-egress 10 queue 1 adaptation-rule pir max cir max
*A:PE#
*A:PE# show qos sap-egress 10 detail | match expression "Queue-Id|CIR Rule"
Queue-Id             : 1                Queue-Type           : auto-expedite
PIR Rule             : max              CIR Rule             : max
*A:PE#
```

```
*A:PE# show pools 1/1/1 access-egress service 1 | match expression "PIR|CIR"
Admin PIR        : 403000              Oper PIR         : 401600
Admin CIR        : 401000              Oper CIR         : 400000
*A:PE#
*A:PE# configure qos sap-egress 10 queue 1 adaptation-rule pir min cir min
*A:PE#
*A:PE# show qos sap-egress 10 detail | match expression "Queue-Id|CIR Rule"
Queue-Id             : 1               Queue-Type           : auto-expedite
PIR Rule             : min             CIR Rule             : min
*A:PE#
*A:PE# show pools 1/1/1 access-egress service 1 | match expression "PIR|CIR"
Admin PIR        : 403000              Oper PIR         : 403200
Admin CIR        : 401000              Oper CIR         : 401600
*A:PE#
```

2.4.8 Committed burst size

The Committed Burst Size (CBS) parameter specifies the size of buffer that can be drawn from the reserved buffer portion of the queue buffer pool. When the reserved buffers for a specific queue have been used, the queue competes with other queues for additional buffer resources up to the MBS.

The CBS is provisioned on ingress and egress service queues within service ingress QoS policies and service egress QoS policies, respectively. The CBS for a queue is specified in kbytes.

The CBS for a network queue is defined within network queue policies based on the forwarding class. The CBS for the queue for the forwarding class is defined as a percentage of buffer space for the pool.

2.4.9 Maximum burst size

The Maximum Burst Size (MBS) parameter specifies the maximum queue depth to which a queue can grow. This parameter ensures that a customer that is massively or continuously over-subscribing the PIR of a queue does not consume all the available buffer resources. For high-priority forwarding class service queues, the MBS can be relatively smaller than the other forwarding class queues because the high-priority service packets are scheduled with priority over other service forwarding classes.

The MBS is provisioned on ingress and egress service queues within service ingress QoS policies and service egress QoS policies, respectively. The MBS for a service queue is specified in bytes or kbytes.

The MBSs for network queues are defined within network queue policies based on the forwarding class. The MBSs for the queues for the forwarding class are defined as a percentage of buffer space for the pool.

2.4.10 Queue drop tails

The MBS determines the maximum queue depth after which no additional packets are accepted into the queue. Additional queue drop tails are available for the different packet profiles to allow preferential access to the queue's buffers which allows higher priority packets to be accepted into a queue when there is congestion for lower priority packets.

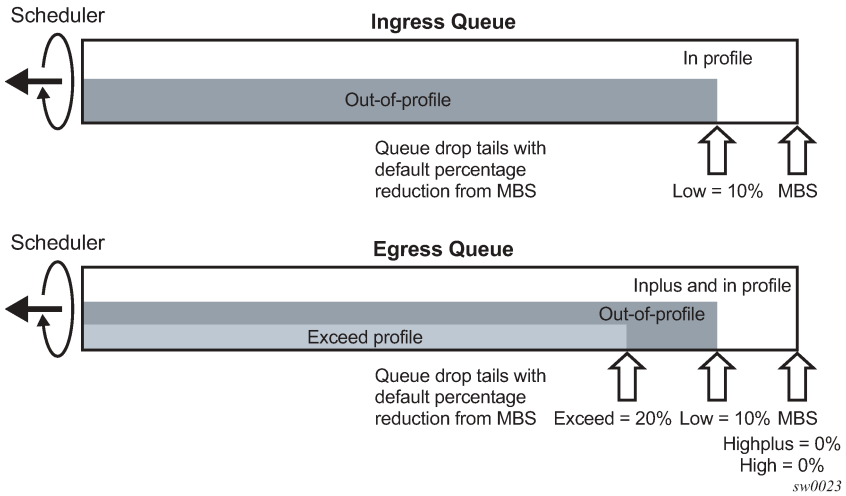
At ingress there is a low drop tail in addition to the MBS. High enqueuing priority packets (for ingress SAP priority mode queues) and in-profile packets (for ingress SAP profile mode queues, network, and shared queues) are allowed to fill up the queue up to the MBS, however, low enqueuing priority packets (for ingress SAP priority mode queues) and out-of-profile packets (for ingress SAP profile mode queues, network, and shared queues) can only fill the queue up to the queue's low drop tail setting.

At egress there are four drop tails in addition to the MBS, one for each profile state:

- an exceed drop tail for exceed-profile packets
- a low drop tail for out-of-profile packets
- a high drop tail for in-profile packets
- a highplus drop tail for highplus-profile packets

Each profile type can only fill the queue up to its corresponding drop tail. Figure 1: Ingress and egress queue drop tails shows the ingress and egress queue drop tails.

Figure 1: Ingress and egress queue drop tails



At both ingress and egress, the drop tails are configured as a percentage reduction from the MBS (specifying 10% places the drop tail at 90% of the MBS) and consequently all are limited by the queue's MBS.

The default percentage for the low drop tail for ingress SAP, queue group, and shared queues is a reduction from the MBS of 10% (low)

The default percentages for the drop tails for egress SAP, queue group, and network queues is a percentage reduction from the MBS of:

- exceed = 20%
- low = 10%
- high = 0%
- highplus = 0%

The exceed, high, and highplus drop tails are not configurable for network queues, however the exceed drop tail is set to a value of 10% in addition to low drop tail and capped by the MBS.

The four drop tails can be configured in any order within egress SAP and queue group queues, however it is logical to order them (from shortest to longest) as exceed, then low, then high, then highplus.

The low drop tail configuration can be overridden for ingress and egress SAP and queue group queues, and for network egress queues. It is also possible to override the low drop tail for subscriber queues within an SLA profile using the keyword **high-prio-only**.

When there is congestion the drop tail ordering gives preferential access to the queue's buffers. For example, if the drop tails on an egress SAP queue are configured as exceed = 20%, low = 10%, high = 5% and highplus = 0%, when the queue depth is below 80% all profile packet types are accepted into the queue. If the depth increases above 80%, then exceed profile packets are not accepted and are therefore dropped, while the out-of-profile, in-profile, and inplus profile packets are still accepted into the queue (giving them preference over the exceed profile packets). If the queue depth goes beyond 90% the out-of-profile packets are also dropped. Similarly, if the queue depth goes beyond 95% the in-profile packets are dropped. It is only when the MBS has been reached that the inplus profile packets are dropped. This example assumes that the pool in which the queue exists is not congested.

2.4.11 Packet markings

Typically, customer markings placed on packets are not treated as trusted from an in-profile or out-of-profile perspective. This allows the use of the ingress buffering to absorb bursts over PIR from a customer and only perform marking as packets are scheduled out of the queue (as opposed to using a hard-policing function that operates on the received rate from the customer). The resulting profile (in or out) based on ingress scheduling into the switch fabric is used by network egress for tunnel marking and egress congestion management.

The high/low priority feature allows a provider to offer a customer the ability to have some packets treated with a higher priority when buffered to the ingress queue. If the queue is configured with a non-zero low drop tail setting, a portion of the ingress queue's allowed buffers are reserved for high-priority traffic. An access ingress packet must hit an ingress QoS action in order for the ingress forwarding plane to treat the packet as high priority (the default is low priority).

If the ingress queue for the packet is above the low drop tail setting, the packet is discarded unless it has been classified as high priority. The priority of the packet is not retained after the packet is placed into the ingress queue. After the packet is scheduled out of the ingress queue, the packet is considered in-profile or out-of-profile based on the dynamic rate of the queue relative to the queue's CIR parameter.

At access ingress, the priority of a packet has no effect on which packets are scheduled first. Only the first buffering decision is affected.

At ingress and egress, the current dynamic rate of the queue relative to the queue's CIR and FIR (where supported) does affect the scheduling priority between queues going to the same destination (either the switch fabric tap or egress port). See Queue scheduling for information about the strict operating priority for queues.

For access ingress, the CIR controls both dynamic scheduling priority and marking threshold (unless **cir-non-profiling** is configured). At network ingress, the queue's CIR affects the scheduling priority but does not provide a profile marking function as the network ingress policy trusts the received marking of the packet, based on the network QoS policy.

At egress, the profile of a packet is only important for egress queue buffering decisions and egress marking decisions, not for scheduling priority. The egress queue's CIR determines the dynamic scheduling priority but does not affect the packet's ingress determined profile.

2.4.12 Queue counters

The router maintains counters for queues within the system for granular billing and accounting. Each queue maintains the following counters:

- counters for packets and octets accepted into the queue

- counters for packets and octets rejected at the queue
- counters for packets and octets transmitted in-profile
- counters for packets and octets transmitted out-of-profile

2.4.13 Color aware profiling

The normal handling of SAP ingress access packets applies an in-profile or out-of-profile state (associated with the colors green and yellow, respectively) to each packet relative to the dynamic rate of the queue while the packet is forwarded toward the egress side of the system. When the queue rate is within or equal to the configured CIR, the packet is considered in-profile. When the queue rate is above the CIR, the packet is considered out-of-profile (this applies when the packet is scheduled out of the queue, not when the packet is buffered into the queue).

Egress queues use the profile marking of packets to preferentially buffer in-profile packets during congestion events. When a packet has been marked in-profile or out-of-profile by the ingress access SLA enforcement, the packet is tagged with an in-profile or out-of-profile marking allowing congestion management in subsequent hops toward the packet's ultimate destination. Each hop to the destination must have an ingress table that determines the in-profile or out-of-profile nature of a packet, based on its QoS markings.

Color aware profiling adds the ability to selectively treat packets received on a SAP as in-profile or out-of-profile regardless of the queue forwarding rate. This allows a customer or access device to color a packet out-of-profile with the intention of preserving in-profile bandwidth for higher priority packets. The customer or access device may also color the packet in-profile, but this is rarely done as the original packets are usually already marked with the in-profile marking.

Each ingress access forwarding class may have one or multiple subclass associations for SAP ingress classification purposes. Each subclass retains the chassis-wide behavior defined to the parent class while providing expanded ingress QoS classification actions. Subclasses are created to provide a match association that enforces actions different than the parent forwarding class. These actions include explicit ingress remarking decisions and color aware functions.

All non-profiled and profiled packets are forwarded through the same ingress access queue to prevent out-of-sequence forwarding. Profiled packets in-profile are counted against the total packets flowing through the queue that are marked in-profile. This reduces the amount of CIR available to non-profiled packets causing fewer to be marked in-profile. Profiled packets out-of-profile are not counted against the total packets flowing through the queue that are marked in-profile. This ensures that the amount of non-profiled packets marked in-profile is not affected by the profiled out-of-profile packet rate.

2.5 QoS policies overview

Service ingress, service egress, and network QoS policies are defined with a scope of either template or exclusive. Template policies can be applied to multiple SAPs or IP interfaces; exclusive policies can only be applied to a single entity.

On most systems, the number of configurable SAP ingress and egress QoS policies per system is larger than the maximum number that can be applied per FP. The **tools>dump>resource-usage>card>fp** output displays the number of policies applied on an FP (the default SAP ingress policy is always applied once for internal use). The **tools>dump>resource-usage>system** output displays the usage of the policies at a system level. The **show>qos>sap-ingress** and **show>qos>sap-egress** commands can be used to show the number of policies configured.

One service ingress QoS policy and one service egress QoS policy can be applied to a specific SAP. One network QoS policy can be applied to a specific IP interface. A network QoS policy defines both ingress and egress behavior.

Router QoS policies are applied on service ingress, service egress, and network interfaces and define classification rules for how traffic is mapped to queues:

- the number of forwarding class queues
- the queue parameters used for policing, shaping, and buffer allocation
- QoS marking/interpretation

The router supports thousands of queues. The exact numbers depend on the hardware being deployed.

There are several types of QoS policies:

- service ingress
- service egress
- network (for ingress and egress)
- network queue (for ingress and egress)
- scheduler
- shared queue
- slope

Service ingress QoS policies are applied to the customer-facing SAPs and map traffic to forwarding class queues on ingress. The mapping of traffic to queues can be based on combinations of customer QoS marking (IEEE 802.1p bits, DSCP, and ToS precedence), IP criteria, and MAC criteria.

The characteristics of the forwarding class queues are defined within the policy as to the number of forwarding class queues for unicast traffic and the queue characteristics. There can be up to eight unicast forwarding class queues in the policy; one for each forwarding class. A service ingress QoS policy also defines up to three queues per forwarding class to be used for multipoint traffic for multipoint services.

In the case of VPLS, four types of forwarding are supported (that is not to be confused with forwarding classes): unicast, multicast, broadcast, and unknown. Multicast, broadcast, and unknown types are flooded to all destinations within the service while the unicast forwarding type is handled in a point-to-point manner within the service.

Service egress QoS policies are applied to SAPs and map forwarding classes to service egress queues for a service. Up to eight queues per service can be defined for the eight forwarding classes. A service egress QoS policy also defines how to remark the forwarding class to IEEE 802.1p bits in the customer traffic.

Network QoS policies are applied to IP interfaces. On ingress, the policy applied to an IP interface maps incoming DSCP and EXP values to forwarding class and profile state for the traffic received from the core network. On egress, the policy maps forwarding class and profile state to DSCP and EXP values for traffic to be transmitted into the core network.

Network queue policies are applied on egress to network ports and channels and on ingress to FPs. The policies define the forwarding class queue characteristics for these entities.

Service ingress, service egress, and network QoS policies are defined with a scope of either template or exclusive. Template policies can be applied to multiple SAPs or IP interfaces whereas exclusive policies can only be applied to a single entity.

One service ingress QoS policy and one service egress QoS policy can be applied to a specific SAP. One network QoS policy can be applied to a specific IP interface. A network QoS policy defines both ingress and egress behavior.

If no QoS policy is explicitly applied to a SAP or IP interface, a default QoS policy is applied.

A summary of the major functions performed by the QoS policies is listed in Table 4: QoS policy types and descriptions.

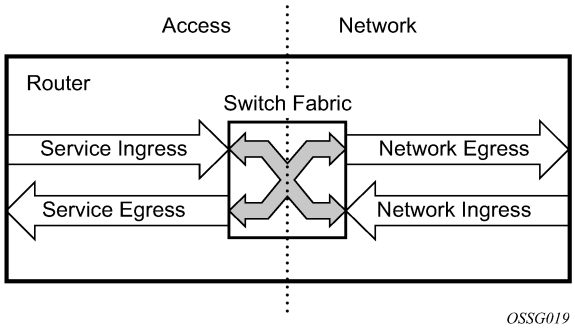
Table 4: QoS policy types and descriptions

Policy type	Applied at…	Description	See
Service Ingress	SAP ingress	Defines up to 32 forwarding class queues and queue parameters for traffic classification Defines up to 31 multipoint service queues for broadcast, multicast, and destination unknown traffic in multipoint services Defines match criteria to map flows to the queues based on combinations of customer QoS (IEEE 802.1p/DE bits, DSCP, TOS precedence), IP criteria, or MAC criteria	Service ingress QoS policies
Service Egress	SAP egress	Defines up to eight forwarding class queues and queue parameters for traffic classification Maps one or more forwarding classes to the queues	Service egress QoS policies
Network	Router interface	Used for classification/marking of IP and MPLS packets At ingress, defines DSCP, dot1p MPLS LSP-EXP, and IP criteria classification to FC mapping At ingress, defines FC to policer/queue-group queue mapping At egress, defines DSCP or precedence FC mapping At egress, defines FC to policer/queue-group queue mapping At egress, defines DSCP, MPLS LSP-EXP, and dot1p/DE marking	Network QoS policies
Network Queue	Network ingress FP and egress port	Defines forwarding class mappings to network queues and queue characteristics for the queues	Network queue QoS policies
Scheduler	Customer multiservice site Service SAP	Defines the hierarchy and parameters for each scheduler Defined in the context of a tier that is used to place the scheduler within the hierarchy Three tiers of virtual schedulers are supported	Scheduler policies

2.5.1 Service versus network QoS

The QoS mechanisms within the routers are specialized for the type of traffic on the interface. For customer interfaces, there is service ingress and egress traffic, and for network core interfaces, there is network ingress and network egress traffic, as shown in Figure 2: Service vs network traffic types.

Figure 2: Service vs network traffic types



The router uses QoS policies applied to a SAP for a service or to a network FP/port to define the queuing, queue attributes, and QoS marking/interpretation.

The router supports four types of service and network QoS policies:

- service ingress QoS policies
- service egress QoS policies
- network QoS policies
- network queue QoS policies

2.5.2 QoS policy entities

Services are configured with default QoS policies. Additional policies must be explicitly created and associated. There is one default service ingress QoS policy, one default service egress QoS policy, and one default network QoS policy. Only one ingress QoS policy and one egress QoS policy can be applied to a SAP or network entity.

When a new QoS policy is created, default values are provided for most parameters with the exception of the policy ID and queue ID values, descriptions, and the default action queue assignment. Each policy has a scope, default action, description, and at least one queue. The queue is associated with a forwarding class.

Table 5: Service types and applicable service QoS policies lists which service QoS policies are supported for each service type.

Table 5: Service types and applicable service QoS policies

Service type	QoS policies
Epipe	Both ingress and egress policies are supported on an Epipe SAP.

Service type	QoS policies
VPLS	Both ingress and egress policies are supported on a VPLS SAP.
IES	Both ingress and egress policies are supported on an IES SAP.
VPRN	Both ingress and egress policies are supported on a VPRN SAP.

Network QoS policies can be applied to the following entities:

- network interfaces
- Cpipe/Epipe/Ipipe and VPLS spoke SDPs
- VPLS spoke and mesh SDPs
- IES and VPRN interface spoke SDPs
- VPRN network ingress
- VXLAN network ingress

Network queue policies can be applied to:

- ingress FPs
- egress ports

Default QoS policies map all traffic with equal priority and allow an equal chance of transmission (Best Effort (be) forwarding class) and an equal chance of being dropped during periods of congestion. QoS prioritizes traffic according to the forwarding class and uses congestion management to control access ingress, access egress, and network traffic with queuing according to priority.

2.5.3 Network QoS policies

Network QoS policies define egress QoS marking and ingress QoS interpretation for traffic on core network IP interfaces. The router automatically creates egress queues for each of the forwarding classes on network IP interfaces.

A network QoS policy defines both the ingress and egress handling of QoS on the IP interface. The following functions are defined:

- **ingress**
  - defines DSCP, Dot1p, and IP criteria mappings to forwarding classes
  - defines LSP EXP value mappings to forwarding classes
- **egress**
  - defines DSCP and IP precedence mappings to forwarding classes
  - defines forwarding class to DSCP value markings
  - defines forwarding class to Dot1p/DE value markings
  - defines forwarding class to LSP EXP value markings
  - enables/disables remarking of QoS

– defines FC to policer/queue-group queue mapping

The required elements to be defined in a network QoS policy are:

- a unique network QoS policy ID
- egress forwarding class to DSCP value mappings for each forwarding class
- egress forwarding class to Dot1p value mappings for each forwarding class
- egress forwarding class to LSP EXP value mappings for each forwarding class
- enabling/disabling of egress QoS remarking
- a default ingress forwarding class and in-profile/out-of-profile state

Optional network QoS policy elements include:

- DSCP name-to-forwarding class and profile state mappings for all DSCP values received
- LSP EXP value-to-forwarding class and profile state mappings for all EXP values received
- ingress FC fp-redirect-group policer mapping
- egress FC port-redirect-group queue/policer mapping

Network policy ID 1 is reserved as the default network QoS policy. The default policy cannot be deleted or changed.

The default network QoS policy is applied to all network interfaces that do not have another network QoS policy explicitly assigned. Table 6: Default network QoS policy egress marking describes the default network QoS policy egress marking.

Table 6: Default network QoS policy egress marking

FC-ID	FC name	FC label	DiffServ name	Egress DSCP marking		Egress LSP EXP marking	
				In-profile name	Out-of-profile name	In-profile	Out-of-profile
7	Network Control	nc	NC2	nc2 111000 - 56	nc2 111000 - 56	111 - 7	111 - 7
6	High-1	h1	NC1	nc1 110000 - 48	nc1 110000 - 48	110 - 6	110 - 6
5	Expedited	ef	EF	ef 101110 - 46	ef 101110 - 46	101 - 5	101 - 5
4	High-2	h2	AF4	af41 100010 - 34	af42 100100 - 36	100 - 4	100 - 4
3	Low-1	l1	AF2	af21 010010 - 18	af22 010100 - 20	011 - 3	010 - 2
2	Assured	af	AF1	af11 001010 - 10	af12 001100 - 12	011 - 3	010 - 2

FC-ID	FC name	FC label	DiffServ name	Egress DSCP marking		Egress LSP EXP marking	
				In-profile name	Out-of-profile name	In-profile	Out-of-profile
1	Low-2	l2	CS1	cs1 001000 - 8	cs1 001000 - 8	001 - 1	001 - 1
0	Best Effort	be	BE	be 000000 - 0	be 000000 - 0	000 - 0	000 - 0

For network ingress, Table 7: Default network QoS policy DSCP to forwarding class mappings and Table 8: Forwarding class and enqueuing priority classification hierarchy based on rule type list the default mapping of DSCP name and LSP EXP values to forwarding class and profile state for the default network QoS policy.

Table 7: Default network QoS policy DSCP to forwarding class mappings

Ingress DSCP		Forwarding class			
dscp-name	dscp-value (binary - decimal)	FC ID	Name	Label	Profile state
Default		0	Best-Effort	be	Out
ef	101110 - 46	5	Expedited	ef	In
nc1	110000 - 48	6	High-1	h1	In
nc2	111000 - 56	7	Network Control	nc	In
af11	001010 - 10	2	Assured	af	In
af12	001100 - 12	2	Assured	af	Out
af13	001110 - 14	2	Assured	af	Out
af21	010010 - 18	3	Low-1	l1	In
af22	010100 - 20	3	Low-1	l1	Out
af23	010110 - 22	3	Low-1	l1	Out
af31	011010 - 26	3	Low-1	l1	In
af32	011100 - 28	3	Low-1	l1	Out
af33	011110 - 30	3	Low-1	l1	Out
af41	100010 - 34	4	High-2	h2	In
af42	100100 - 36	4	High-2	h2	Out

Ingress DSCP		Forwarding class			
dscp-name	**dscp-value (binary - decimal)**	**FC ID**	**Name**	**Label**	**Profile state**
af43	100110 - 38	4	High-2	h2	Out

2.5.4 Network queue QoS policies

Network queue policies define the network forwarding class queue characteristics. Network queue policies are applied on egress on core network ports or channels and on ingress FPs. Network queue policies can be configured to use as many queues as needed. This means that the number of queues can vary. Not all policies use the same number of queues as the default network queue policy. The multicast queues are only used at ingress.

The queue characteristics that can be configured on a per-forwarding class basis are:

- CBS as a percentage of the buffer pool
- MBS as a percentage of the buffer pool
- low drop tail as a percentage reduction from MBS
- PIR as a percentage of the FP ingress capacity or egress port bandwidth
- CIR as a percentage of the FP ingress capacity or egress port bandwidth
- FIR as a percentage of the FP ingress capacity or egress port bandwidth

Network queue policies are identified with a unique policy name that conforms to the standard router alphanumeric naming conventions.

The system default network queue policy is named **default** and cannot be edited or deleted. For information about the default network queue policy, see Network queue QoS policies.

2.5.5 Service ingress QoS policies

Service ingress QoS policies define ingress service forwarding class queues and map flows to those queues. When a service ingress QoS policy is created by default, it always has two queues defined that cannot be deleted: one for the default unicast traffic and one for the default multipoint traffic. These queues exist within the definition of the policy. The queues only get instantiated in hardware when the policy is applied to a SAP. In the case where the service does not have multipoint traffic, the multipoint queues are not instantiated.

In the simplest service ingress QoS policy, all traffic is treated as a single flow and mapped to a single queue, and all flooded traffic is treated with a single multipoint queue. The required elements to define a service ingress QoS policy are:

- a unique service ingress QoS policy ID
- a QoS policy scope of template or exclusive
- at least one default unicast forwarding class queue. The parameters that can be configured for a queue are discussed in Queue parameters.
- at least one multipoint forwarding class queue

Optional service ingress QoS policy elements include:

- additional unicast queues up to a total of 31
- additional multipoint queues up to 31
- QoS policy match criteria to map packets to a forwarding class

To facilitate more forwarding classes, subclasses are supported. Each forwarding class can have one or multiple subclass associations for SAP ingress classification purposes. Each subclass retains the chassis-wide behavior defined to the parent class while providing expanded ingress QoS classification actions.

There can be up to 64 classes and subclasses combined in a sap-ingress policy. With the extra 56 values, the size of the forwarding class space is more than sufficient to handle the various combinations of actions.

Forwarding class expansion is accomplished through the explicit definition of sub-forwarding classes within the SAP ingress QoS policy. The CLI mechanism that creates forwarding class associations within the SAP ingress policy is also used to create subclasses. A portion of the subclass definition directly ties the subclass to a parent, chassis-wide forwarding class. The subclass is only used as a SAP ingress QoS classification tool; the subclass association is lost when ingress QoS processing is finished.

When configured with this option, the forwarding class and drop priority of incoming traffic are determined by the mapping result of the EXP bits in the top label. Table 8: Forwarding class and enqueuing priority classification hierarchy based on rule type lists the classification hierarchy based on rule type.

Table 8: Forwarding class and enqueuing priority classification hierarchy based on rule type

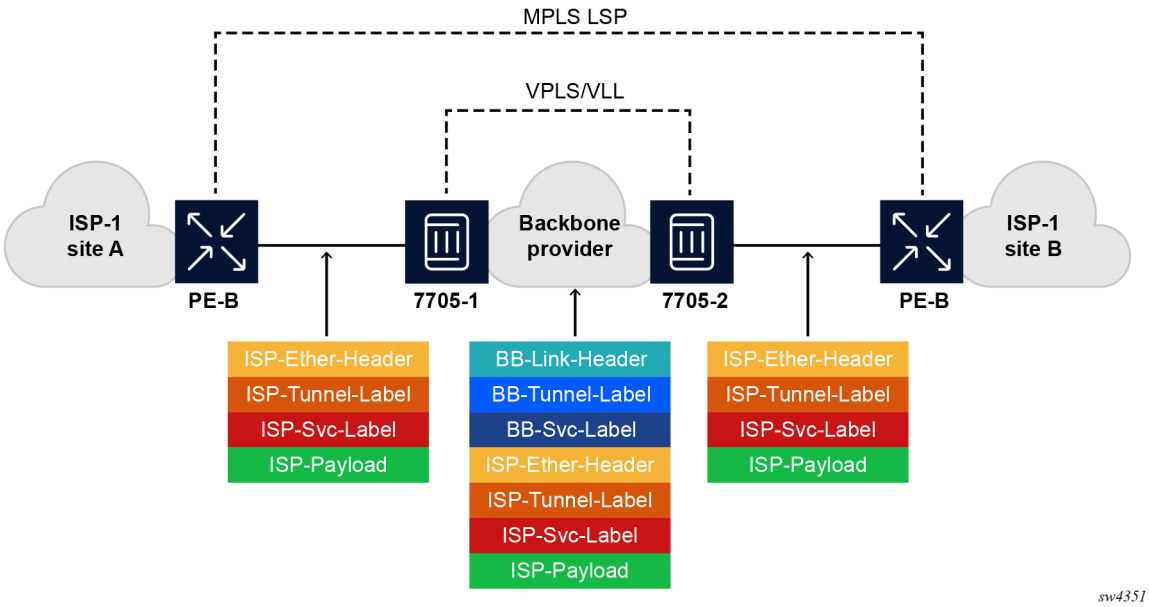
#	Rule	Forwarding class	Enqueuing priority	Comments
1	default-fc	Set the policy's default forwarding class.	Set to policy default	All packets match the default rule.
2	dot1p dot1p-value	Set when an fc-name exists in the policy. Otherwise, preserve from the previous match.	Set when the priority parameter is high or low. Otherwise, preserve from the previous match.	Each dot1p-value must be explicitly defined. Each packet can only match a single dot1p rule.
3	lsp-exp exp-value	Set when an fc-name exists in the policy. Otherwise, preserve from the previous match.	Set when the priority parameter is high or low. Otherwise, preserve from the previous match.	Each exp-value must be explicitly defined. Each packet can only match a single lsp-exp rule. This rule can only be applied on Ethernet L2 SAP.
4	prec ip-prec-value	Set when an fc-name exists in the policy. Otherwise, preserve from the previous match.	Set when the priority parameter is high or low. Otherwise, preserve from the previous match.	Each ip-prec-value must be explicitly defined. Each packet can only match a single prec rule.
5	dscp dscp-name	Set when an fc-name exists in the policy. Otherwise, preserve from the previous match.	Set when the priority parameter is high or low in the entry. Otherwise, preserve from the previous match.	Each dscp-name that defines the DSCP value must be explicitly defined. Each packet can only match a single DSCP rule.

#	Rule	Forwarding class	Enqueuing priority	Comments
6	IP criteria: multiple entries per policy Multiple criteria per entry	Set when an fc-name exists in the entry's action. Otherwise, preserve from the previous match.	Set when the priority parameter is high or low in the entry action. Otherwise, preserve from the previous match.	When IP criteria is specified, entries are matched based on ascending order until first match, then processing stops. A packet can only match a single IP criteria entry.
7	MAC criteria: multiple entries per policy Multiple criteria per entry	Set when an fc-name exists in the entry's action. Otherwise, preserve from the previous match.	Set when the priority parameter is specified as high or low in the entry action. Otherwise, preserve from the previous match.	When MAC criteria is specified, entries are matched based on ascending order until first match, then processing stops. A packet can only match a single MAC criteria entry.

2.5.5.1 FC mapping based on EXP bits at VLL/VPLS SAP

To accommodate backbone ISPs who want to provide VPLS/VLL to small ISPs as a site-to-site inter-connection service, small ISP routers can connect to Ethernet Layer 2 SAPs. The traffic is encapsulated in a VLL/VPLS SDP. These small ISP routers are typically PE routers. To provide appropriate QoS, the routers support a new classification option that based on received MPLS EXP bits. Figure 3: Example configuration — carrier application shows a sample configuration.

Figure 3: Example configuration — carrier application



The **lsp-exp** command is supported in sap-ingress qos policy. This option can only be applied on Ethernet Layer 2 SAPs.

Table 9: Forwarding class classification based on rule type lists forwarding class behavior by rule type.

Table 9: Forwarding class classification based on rule type

#	Rule	Forwarding class	Comments
1	default-fc	Set the policy's default forwarding class.	All packets match the default rule.
2	IP criteria: • multiple entries per policy • multiple criteria per entry	Set when an fc-name exists in the entry's action. Otherwise, preserve from the previous match.	When IP criteria is specified, entries are matched based on ascending order until first match, then processing stops. A packet can only match a single IP criteria entry.
3	MAC criteria: • multiple entries per policy • multiple criteria per entry	Set when an fc-name exists in the entry's action. Otherwise, preserve from the previous match.	When MAC criteria is specified, entries are matched based on ascending order until first match, then processing stops. A packet can only match a single MAC criteria entry.

The enqueuing priority is specified as part of the classification rule and is set to "high" or "low". The enqueuing priority relates to the forwarding class queue's low drop tail where only packets with a high enqueuing priority are accepted into the queue when the queue's depth reaches the defined threshold. See Queue drop tails.

The mapping of IEEE 802.1p bits, IP Precedence, and DSCP values to forwarding classes is optional as is specifying IP and MAC criteria.

The IP and MAC match criteria can be very basic or quite detailed. IP and MAC match criteria are constructed from policy entries. An entry is identified by a unique, numerical entry ID. A single entry cannot contain more than one match value for each match criteria. Each match entry has a queuing action that specifies:

- the forwarding class of packets that match the entry
- the enqueuing priority (high or low) for matching packets

The entries are evaluated in numerical order based on the entry ID from the lowest to highest ID value. The first entry that matches all match criteria has its action performed.

The supported service ingress QoS policy IP match criteria are:

- destination IP address/prefix
- destination port/range
- IP fragment
- protocol type (TCP, UDP, and so on)
- source port/range
- source IP address/prefix
- DSCP value

The supported service ingress QoS policy MAC match criteria are:

- IEEE 802.2 LLC SSAP value/mask
- IEEE 802.2 LLC DSAP value/mask
- IEEE 802.3 LLC SNAP OUI zero or non-zero value
- IEEE 802.3 LLC SNAP PID value
- IEEE 802.1p value/mask
- source MAC address/mask
- destination MAC address/mask
- EtherType value

Table 10: MAC match Ethernet frame types describes the frame format on which the MAC match criteria that can be used for an Ethernet frame depends on.

Table 10: MAC match Ethernet frame types

Frame format	Description
802dot3	IEEE 802.3 Ethernet frame. Only the source MAC, destination MAC, and IEEE 802.1p value are compared for match criteria.
802dot2-llc	IEEE 802.3 Ethernet frame with an 802.2 LLC header.
802dot2-snap	IEEE 802.2 Ethernet frame with 802.2 SNAP header.
ethernet-II	Ethernet type II frame where the 802.3 length field is used as an Ethernet type (Etype) value. Etype values are 2-byte values greater than 0x5FF (1535 decimal).

The 802dot3 frame format matches across all Ethernet frame formats where only the source MAC, destination MAC, and IEEE 802.1p value are compared. The other Ethernet frame types match those field values in addition to fields specific to the frame format. Table 11: MAC match criteria frame type dependencies lists the criteria that can be matched for the various MAC frame types.

Table 11: MAC match criteria frame type dependencies

Frame format	Source MAC	Dest MAC	IEEE 802.1p value	Etype value	LLC header SSAP/DSAP value/mask	SNAP-OUI zero/ non-zero value	SNAP- PID value
802dot3	Yes	Yes	Yes	No	No	No	No
802dot2-llc	Yes	Yes	Yes	No	Yes	No	No
802dot2-snap	Yes	Yes	Yes	No	No[1]	Yes	Yes
ethernet-II	Yes	Yes	Yes	Yes	No	No	No

Service ingress QoS policy ID 1 is reserved for the default service ingress policy. The default policy cannot be deleted or changed.

[1] When a SNAP header is present, the LLC header is always set to AA-AA.

The default service ingress policy is implicitly applied to all SAPs that do not explicitly have another service ingress policy assigned. Table 12: Default service ingress policy ID 1 definition lists the characteristics of the default policy.

Table 12: Default service ingress policy ID 1 definition

Characteristic	Item	Definition
Queues	Queue 1	One queue for all unicast traffic: • Forward Class: best-effort (be) • CIR = 0 • PIR = max (line rate) • FIR = 0 • MBS, CBS, and HP-only = default (values derived from applicable policy)
	Queue 11	One queue for all multipoint traffic: • CIR = 0 • PIR = max (line rate) • FIR = 0 • MBS, CBS, and HP-only = default (values derived from applicable policy)
Flows	Default Forwarding Class	One flow defined for all traffic; all traffic mapped to best-effort (be) with a low priority.

2.5.5.2 Egress forwarding class override

Egress forwarding class override provides additional QoS flexibility by allowing the use of a different forwarding class at egress than was used at ingress.

The ingress QoS processing classifies traffic into a forwarding class (or subclass) and by default the same forwarding class is used for this traffic at the access or network egress. The ingress forwarding class or subclass can be overridden so that the traffic uses a different forwarding class at the egress. This can be configured for the main forwarding classes and for subclasses, allowing each to use a different forwarding class at the egress.

The buffering, queuing, policing, and remarking operation at the ingress and egress remain unchanged. Egress reclassification is possible. The profile processing is completely unaffected by overriding the forwarding class.

When used in conjunction with QoS Policy Propagation Using BGP (QPPB), a QPPB assigned forwarding class takes precedence over both the normal ingress forwarding class classification rules and any egress forwarding class overrides.

Figure 4: Egress forwarding class override

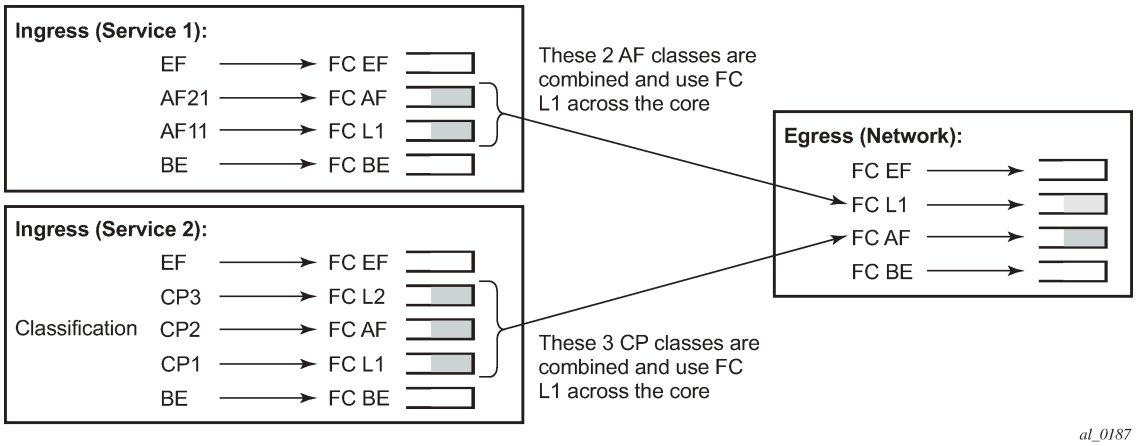


Figure 4: Egress forwarding class override shows the ingress service 1 using forwarding classes AF and L1 that are overridden to L1 for the network egress, while it also shows ingress service 2 using forwarding classes L1, AF, and L2 that are overridden to AF for the network egress.

2.5.6 Service egress QoS policies

Service egress queues are implemented at the transition from the service core network to the service access network. The advantages of per-service queuing before transmission into the access network are:

- per-service egress subrate capabilities especially for multipoint services
- more granular, fairer scheduling per-service into the access network
- per-service statistics for forwarded and discarded service packets

The subrate capabilities and per-service scheduling control are required to make multiple services per physical port possible. Without egress shaping, it is impossible to support more than one service per port. There is no way to prevent service traffic from bursting to the available port bandwidth and starving other services.

For accounting purposes, per-service statistics can be logged. When statistics from service ingress queues are compared with service egress queues, the ability to conform to per-service QoS requirements within the service core can be measured. The service core statistics are a major asset to core provisioning tools.

Service egress QoS policies define egress queues and map forwarding class flows to queues. In the simplest service egress QoS policy, all forwarding classes are treated like a single flow and mapped to a single queue. To define a basic egress QoS policy, the following are required:

- a unique service egress QoS policy ID
- a QoS policy scope of template or exclusive
- at least one defined default queue

Optional service egress QoS policy elements include:

- additional queues up to a total of eight separate queues (unicast)
- dot1p/DE, DSCP, and IP precedence remarking based on forwarding class

Each queue in a policy is associated with one of the forwarding classes. Each queue can have its individual queue parameters allowing individual rate shaping of the forwarding classes mapped to the queue.

More complex service queuing models are supported in the router where each forwarding class is associated with a dedicated queue.

The forwarding class determination per service egress packet is determined either at ingress or egress. If the packet ingressed the service on the same router, the service ingress classification rules determine the forwarding class of the packet. If the packet is received on a network interface, the forwarding class is marked in the tunnel transport encapsulation. In each case, the packet can be reclassified into a different forwarding class at service egress.

Service egress QoS policy ID 1 is reserved as the default service egress policy. The default policy cannot be deleted or changed. The default access egress policy is applied to all SAPs that do not have another service egress policy explicitly assigned. Table 13: Default service egress policy ID 1 definition lists the characteristics of the default policy.

Table 13: Default service egress policy ID 1 definition

Characteristic	Item	Definition
Queues	Queue 1	One queue defined for all traffic classes: • CIR = 0 • PIR = max (line rate) • MBS and CBS = default
Flows	Default Action	One flow defined for all traffic classes; all traffic mapped to queue 1 with no marking.

2.5.7 Scheduler policies

A scheduler policy defines the hierarchy and all operating parameters for the member schedulers. A scheduler policy must be defined in the QoS context before a group of virtual schedulers can be used. Although configured in a scheduler policy, the individual schedulers are created when the policy is applied to an object, such as a SAP or interface.

Scheduler objects define bandwidth controls that limit each child (other schedulers and policers or queues) associated with the scheduler. The scheduler object can also define a child association with a parent scheduler of its own.

A scheduler is used to define a bandwidth aggregation point within the hierarchy of virtual schedulers. The scheduler's rate defines the maximum bandwidth that the scheduler can consume. It is assumed that each scheduler created has policers, queues, or other schedulers defined as child associations. The scheduler can also be a child which takes bandwidth from a scheduler in a higher tier.

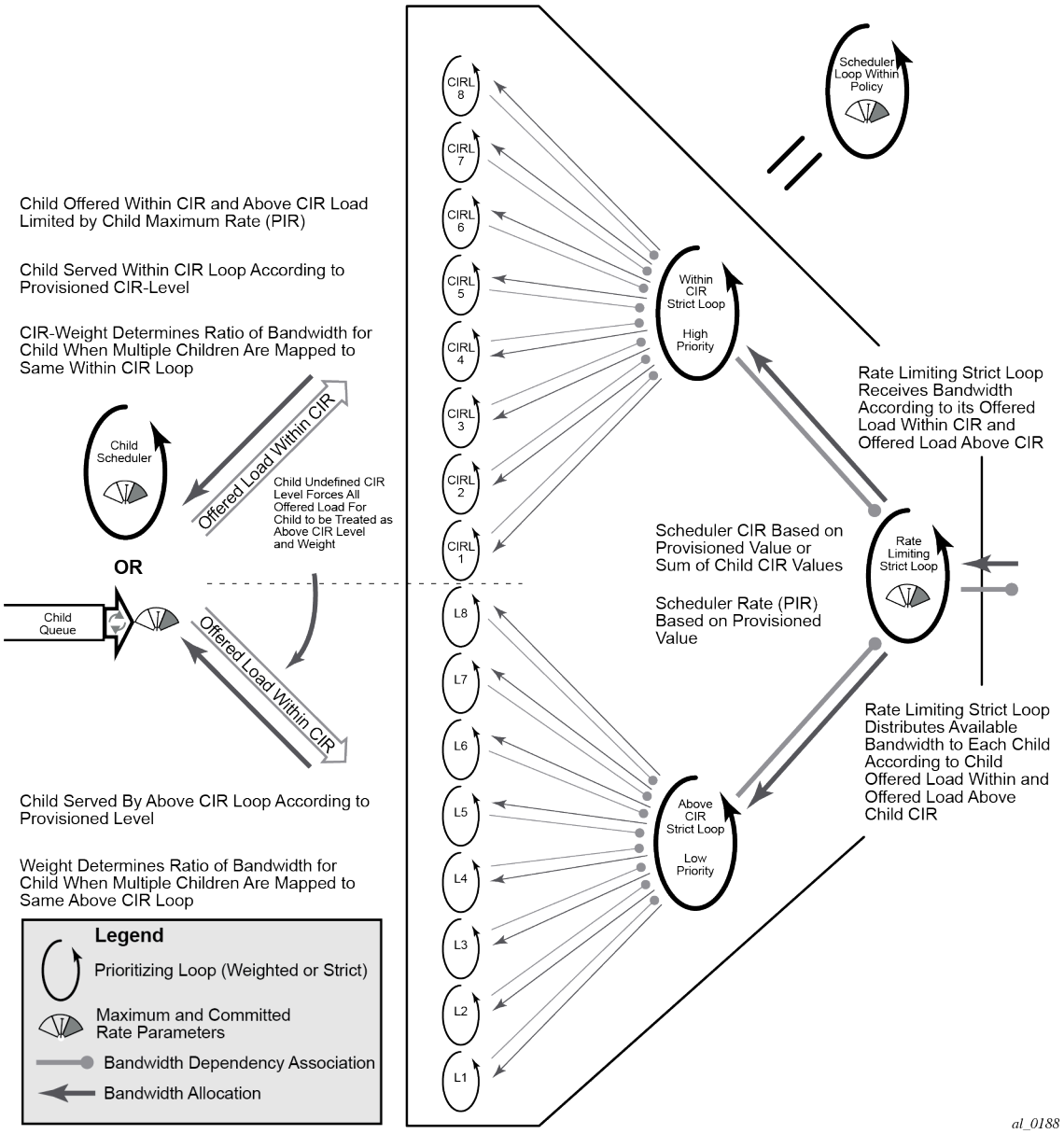
A parent parameter can be defined to specify a scheduler further up in the scheduler policy hierarchy. Only schedulers in Tiers 2 and 3 can have parental association. When multiple schedulers, policers, and queues share a child status with the scheduler on the parent, the weight or strict parameters define how this scheduler contends with the other children for the parent's bandwidth. The parent scheduler can be removed or changed at any time and is immediately reflected on the schedulers created by association of this scheduler policy.

When a parent scheduler is defined without specifying level, weight, or CIR parameters, the default bandwidth access method is weight with a value of 1.

If any orphaned policers or queues (where the specified scheduler name does not exist) exist on the ingress SAP and the policy application creates the required scheduler, the status on the queue becomes non-orphaned at this time.

Figure 5: Virtual scheduler internal bandwidth allocation shows how child queues, policers, and schedulers interact with their parent scheduler to receive bandwidth. The scheduler distributes bandwidth to the children by first using each child's CIR according to the CIR-level parameter (CIR L8 through CIR L1 weighted loops). The weighting at each CIR-level loop is defined by the CIR weight parameter for each child. The scheduler then distributes any remaining bandwidth to the children up to each child's rate parameter according to the Level parameter (L8 through L1 weighted loops). The weighting at each level loop is defined by the weight parameter for each child.

Figure 5: Virtual scheduler internal bandwidth allocation



2.5.7.1 Virtual hierarchical scheduling

Virtual hierarchical scheduling is a method that defines a bounded operation for a group of queues and, at egress only, policers. One or more policers and queues are mapped to a scheduler with strict and weighted metrics controlling access to the scheduler. The scheduler has an optional prescribed maximum operating rate that limits the aggregate rate of the child queues. This scheduler may then feed into another virtual scheduler in a higher tier. The creation of a hierarchy of schedulers and the association of queues to the hierarchy allows for a hierarchical SLA to be enforced.

The virtual hierarchical scheduling bandwidth distribution mechanism complements the packet scheduling provided by the queue scheduling (see Queue scheduling).

Scheduler policies in the routers determine how a policer or queue interacts with bandwidth with other children associated with the same scheduler hierarchy. Ingress queues and egress queues and policers can operate within the context of a scheduler. Multiple policers and queues can share the same scheduler. Schedulers control the data transfer between the following queues and destinations:

- service ingress queues to switch fabric destinations
- service egress queues to access egress ports
- network ingress queues to switch fabric destinations
- network egress queues to network egress interfaces

2.5.7.2 Hierarchical scheduler policies

Hierarchical scheduler policies are an alternate way of scheduling that can be used on service ingress queues and service egress queues and policers. Hierarchical scheduler policies allow the creation of a hierarchy of schedulers where policers, queues, or other schedulers are scheduled by superior schedulers.

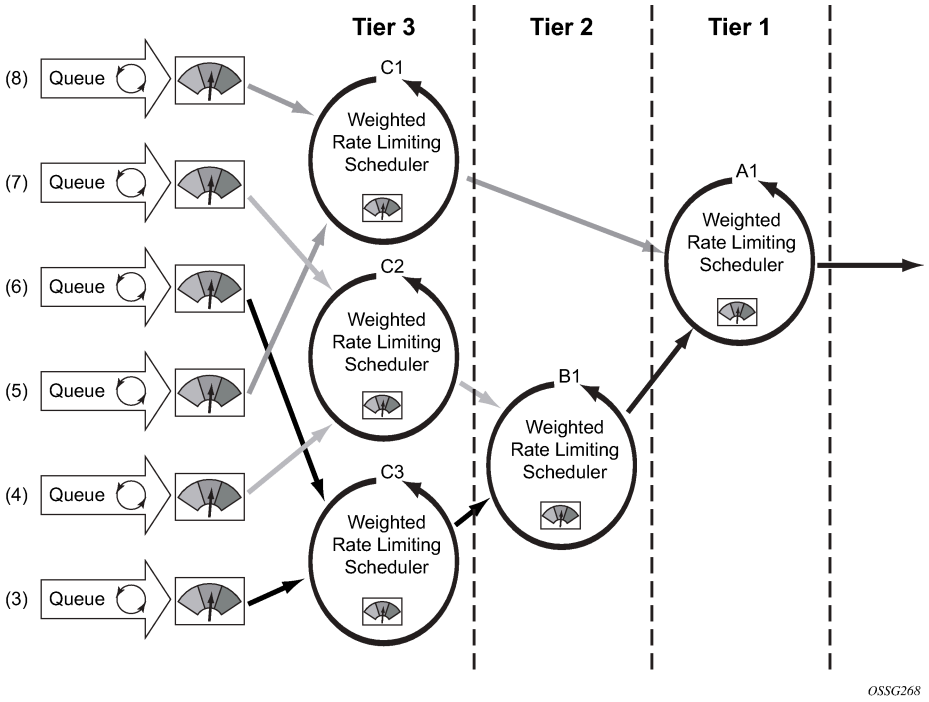
The use of the hierarchical scheduler policies is often referred to as hierarchical QoS or HQoS on the SR OS.

2.5.7.2.1 Hierarchical virtual schedulers

Virtual schedulers are created within the context of a hierarchical scheduler policy. A hierarchical scheduler policy defines the hierarchy and parameters for each scheduler. A scheduler is defined in the context of a tier (Tier 1, Tier 2, Tier 3). The tier level determines the scheduler's position within the hierarchy. Three tiers of virtual schedulers are supported (see Figure 6: Hierarchical scheduler and queue association).

Tier 1 schedulers (also called root schedulers) can also have a parent scheduler. A scheduler can enforce a maximum rate of operation for all child policers, queues, and associated schedulers.

Figure 6: Hierarchical scheduler and queue association



2.5.7.3 Tiers

To illustrate the difference between single-tier scheduling and hierarchical scheduling policies, consider a simple case where, on service ingress, three queues are created for gold, silver, and bronze service, and are configured as shown in Table 14: Tiers configured with CIR and PIR values.

Table 14: Tiers configured with CIR and PIR values

Tier	CIR and PIR value
Gold	CIR = 10 Mb/s, PIR = 10 Mb/s
Silver	CIR = 20 Mb/s, PIR = 40 Mb/s
Bronze	CIR = 0 Mb/s, PIR = 100 Mb/s

In the router, the CIR is used for profiling of traffic (in-profile or out-of-profile), and the PIR is the rate at which traffic is shaped out of the queue. In single-tier scheduling, each queue can burst up to its defined PIR, which means up to 150 Mb/s (10 Mb/s + 40 Mb/s + 100 Mb/s) can enter the service.

In a simple example of a hierarchical scheduling policy, a superior (or parent) scheduler can be created for the gold, silver, and bronze queues that limits the overall rate for all queues to 100 Mb/s. In this hierarchical scheduling policy, the customer can send in any combination of gold, silver, and bronze traffic conforming to the defined PIR values and not to exceed 100 Mb/s.

2.5.7.4 Scheduler policies applied to applications

A scheduler policy can be applied either on a SAP (see Figure 7: Scheduler policy on SAP and scheduler hierarchy creation) or on a multiservice customer site (a group of SAPs with common origination/termination point) (see Figure 8: Scheduler policy on customer site and scheduler hierarchy creation). Whenever a scheduler policy is applied, the individual schedulers comprising the policy are created on the object. When the object is an individual SAP, only policers and queues created on that SAP can use the schedulers created by the policy association. When the object is a multiservice customer site, the schedulers are available to any SAPs associated with the site (also see Scheduler policies applied to SAPs).

Figure 7: Scheduler policy on SAP and scheduler hierarchy creation

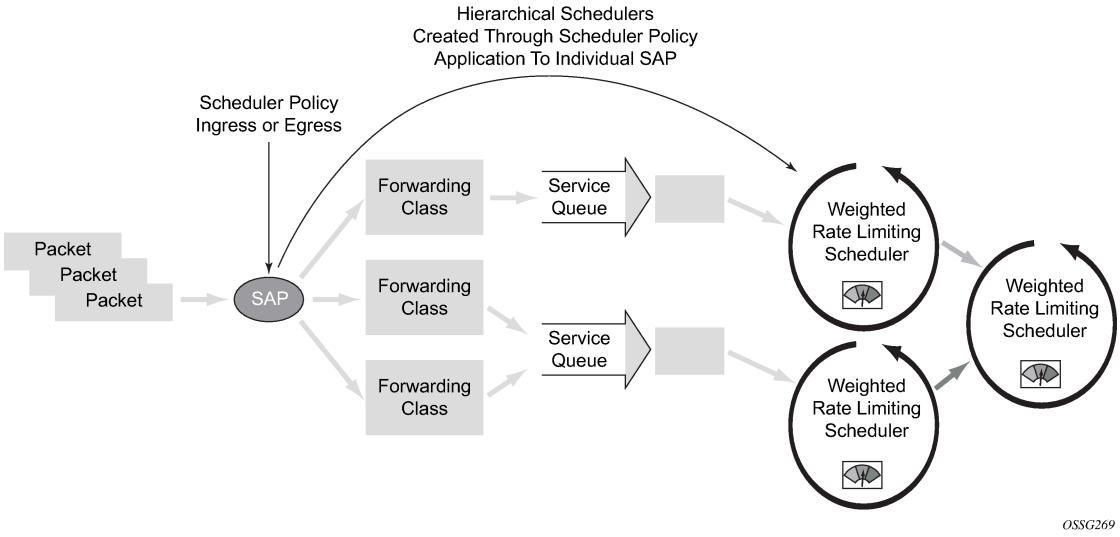
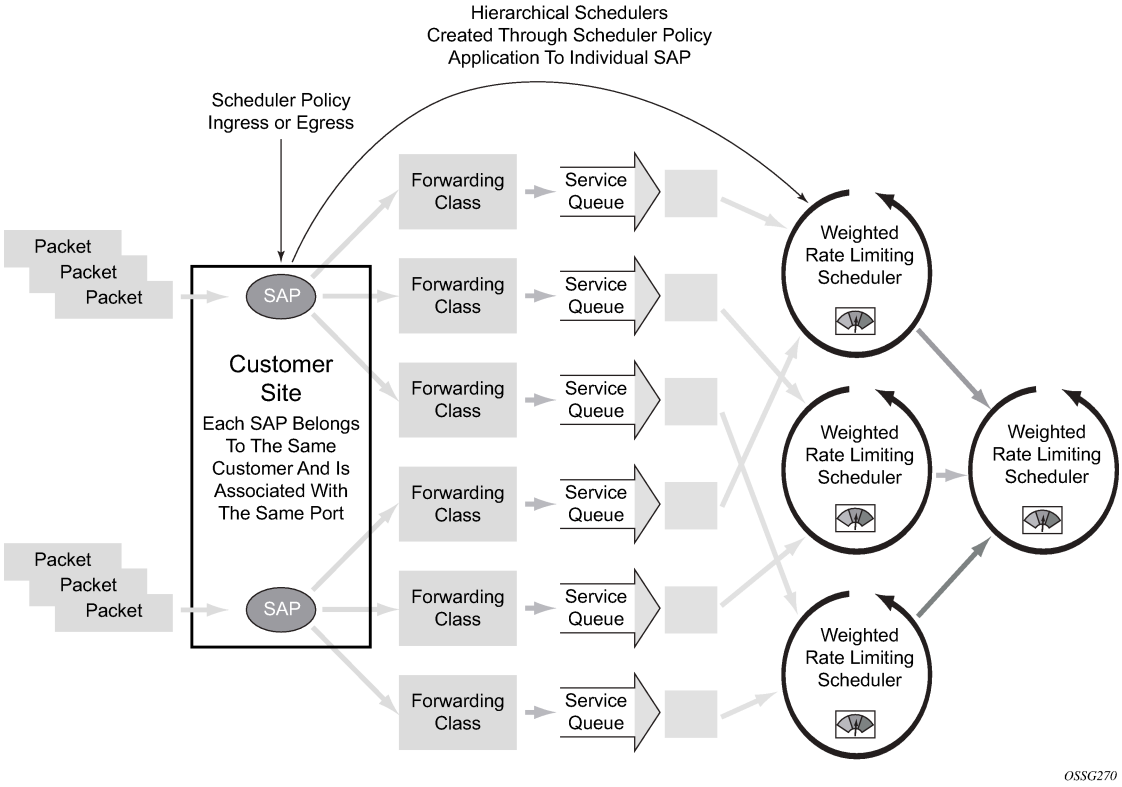


Figure 8: Scheduler policy on customer site and scheduler hierarchy creation



Queues and policers become associated with schedulers when the parent scheduler name is defined within the policer or queue definition in the SAP QoS policy. The scheduler is used to provide bandwidth to the queue relative to the operating constraints imposed by the scheduler hierarchy.

2.5.7.5 Scheduler policies applied to SAPs

A scheduler policy can be applied to create egress schedulers used by SAP policers and queues. The schedulers comprising the policy are created at the time the scheduler policy is applied to the SAP. If any orphaned policers or queues exist on the egress SAP (when the specified scheduler name does not exist), and the policy application creates the required scheduler, the status on the policer or queue becomes non-orphaned.

Queues and policers are associated with the configured schedulers by specifying the parent scheduler defined within the policer queue definition from the SAP QoS policy. The scheduler is used to provide bandwidth to the queue relative to the operating constraints imposed by the scheduler hierarchy.

2.5.7.6 Scheduler policies applied to customer SLAs

The router implementation of hierarchical QoS allows a common set of virtual schedulers to govern bandwidth over a set of customer services that is considered to be from the same site. Different service types purchased from a single customer can be accounted and billed as an aggregate, based on a single SLA.

By configuring multiservice sites within a customer context, the customer site can be used as an anchor point to create an ingress and egress virtual scheduler hierarchy.

When a site is created, it must be assigned to the chassis slot or a port. This allows the system to allocate the resources necessary to create the virtual schedulers defined in the ingress and egress scheduler policies. This also acts as verification that each SAP assigned to the site exists within the context of the customer ID and that the SAP was created on the correct slot, port, or channel. The specified slot or port must already be pre-provisioned (configured) on the system.

When scheduler policies are defined for ingress and egress, the scheduler names contained in each policy are created according to the parameters defined in the policy. Multiservice customer sites are configured only to create a virtual scheduler hierarchy and make it available to queues on multiple SAPs.

2.5.7.7 Scheduler policies applied to multiservice sites

Only existing scheduler policies can be applied to create the ingress or egress schedulers used by SAP policers or queues associated with a customer's multiservice site. The schedulers defined in the scheduler policy can only be created after the customer site has been appropriately assigned to a chassis port, channel, or slot. When a multiservice customer site is created, SAPs owned by the customer must be explicitly included in the site. The SAP must be owned by the customer within which the site was created, and the site assignment parameter must include the physical locale of the SAP.

2.5.8 Configuration notes

The following information describes QoS implementation restrictions:

- Creating additional QoS policies is optional.
- Default policies are created for service ingress, service egress, network, network queue, and slope policies. Scheduler policies must be explicitly created and applied to a port.
- Associating a service or access ports with a QoS policy other than the default policy is optional.
- A network queue, service egress, and service ingress QoS policy must consist of at least one queue. Queues define the forwarding class, CIR, and PIR associated with the queue.

3 Match list for QoS policies

Match lists provide a mechanism to simplify the configuration of IP and IPv6 criteria matching statements within QoS policies. Instead of defining multiple match statements in an **ip-criteria** or **ipv6-criteria** statement, a user can group the same types of matching criteria into a single match list and use that list as a match criterion value, thereby requiring only a single policy entry per each unique action. The same match list can be used in one or more QoS policies.

The match lists further simplify management and deployment of the policy changes. A change in a **match-list** content is automatically propagated across all policies employing that list in their match criteria, therefore, only a single configuration change is required to trigger policy changes when a list is used by entries in one or more QoS policies.

The hardware resource usage does not change when QoS match lists are used compared to when the user creates multiple entries (one for each element in the list). However, consideration must be given to how the lists are used to ensure only needed match permutations are created in a QoS policy entry (especially when other match criteria that are also lists or ranges are specified in the same entry). The system verifies whether a new list element, for example, an IP address prefix, can be added to a specific list, or a list can be used by a new QoS policy, by checking whether the resources exist in hardware to implement the required changes for all QoS policies that reference the updated list. If sufficient resources do not exist, the addition of a new element to the list or use of the list by another policy fails.

QoS match lists are created within **config**>**qos**>**match-list**. The following types of match lists are supported:

- **IPv4 prefix lists**

  These are applicable to **src-ip** and **dst-ip** matching in SAP ingress and SAP egress QoS policies used by both SAPs and subscribers, and in the ingress section of a network QoS policy.

- **IPv6 prefix lists**

  These are applicable to **src-ip** and **dst-ip** matching in SAP ingress and SAP egress QoS policies used by both SAPs and subscribers, and in the ingress section of a network QoS policy

- **port lists**

  These are applicable to **src-port** and **dst-port** matching in network QoS policies

A prefix list can be configured in criteria statements within SAP QoS policies or within network QoS policies, but not in both types simultaneously.

The following restrictions apply to the use of prefix lists in network QoS policies:

- A single IP prefix list (IPv4/IPv6) cannot be used by network QoS policy entries more than 128 times.
- A single entry in a network QoS policy can only refer to either a source or destination prefix list. It is not permitted to refer simultaneously to both a source and a destination prefix IPv4/IPv6 list.
- A single entry in a network QoS policy can only refer to either a source or destination port list. It is not permitted to refer simultaneously to both a source and a destination port list. Port lists can only be applied to network entries.
- Prefix lists and port lists are mutually exclusive within a single entry.

The following shows a created IPv4 prefix list which is configured within a SAP ingress QoS policy to rate limit the traffic from those prefixes.

```
configure
#------------------------------------------------
echo "QoS Policy Configuration"
#------------------------------------------------
    qos
        match-list
            ip-prefix-list "ip-prefix-list-1" create
                description "IPv4 prefix list"
                prefix 10.0.0.0/8
                prefix 192.168.0.0/16
            exit
        exit
    exit
#------------------------------------------------
echo "QoS Policy Configuration"
#------------------------------------------------
    qos
        sap-egress 10 create
            queue 1 create
            exit
            queue 2 create
            exit
            fc af create
                queue 2
            exit
            ip-criteria
                entry 10 create
                    match
                        dst-ip ip-prefix-list "ip-prefix-list-1"
                    exit
                    action fc "af"
                exit
            exit
        exit
    exit
```

The IPv4 prefix list can be shown as follows:

```
*A:PE# show qos match-list ip-prefix-list "ip-prefix-list-1"

===============================================================================
QoS Match IP Prefix List
===============================================================================
Prefix Name        : ip-prefix-list-1
Description        : IPv4 prefix list
-------------------------------------------------------------------------------
IP Prefixes
-------------------------------------------------------------------------------
10.0.0.0/8
192.168.0.0/16
-------------------------------------------------------------------------------
No. of Prefixes : 2
-------------------------------------------------------------------------------
===============================================================================
```

4 Network QoS policies

Network QoS policies can be applied to network interfaces, CSC network interfaces in a VPRN, pseudowires in VLL and VPLS services, and IES or VPRN spoke interfaces to provide ingress and egress QoS control for the traffic on those objects. Network QoS policies can also be applied to provide aggregate ingress QoS on VPRN and VXLAN services.

4.1 Network QoS policies overview

The ingress component of a network QoS policy defines the packet classification to internal forwarding class and profile state. The forwarding class and profile state define the Per Hop Behavior (PHB) or the QoS treatment through the router. The mapping defaults to that defined in the default network QoS policy until an explicit policy is defined for the network interface. Packets can also be redirected to an ingress FP queue group.

The egress component of a network QoS policy allows further egress classification to internal forwarding class and profile state. The egress of the network QoS policy also defines the marking based on the forwarding class and the profile state. Packets can also be redirected to an egress port queue group.

Network policy 1 is applied to all network interfaces by default. It cannot be modified or deleted. It defines the default DSCP-to-FC mapping and MPLS EXP-to-FC mapping for the ingress. For the egress, it defines eight forwarding classes and the packet marking criteria.

New (non-default) network policy parameters can be modified. A new network policy must include the definition of at least one queue and specify the default action. Incomplete network policies cannot be applied to network interfaces.

4.2 Network ingress

The following types of QoS mapping decisions are applicable to the ingress of a network QoS policy:

- default QoS mapping
- Ethernet dot1p value mapping (if defined)
- IP DSCP mapping

  The inner DSCP marking can be used at the ingress of an LER (see Network ingress tunnel QoS override).
- IP criteria mapping
- IPv6 criteria mapping
- MPLS LSP EXP mapping

The default QoS mapping always exists and every received packet is mapped to this default if no explicitly defined matching entry exists.

The packets for a specified forwarding class can be redirected to an ingress FP queue group.

4.2.1 Network ingress tunnel QoS override

By default, a tunnel that terminates on the ingress IP interface (the node is the last hop for the tunnel) is evaluated based on the type of tunnel: IP GRE or MPLS LSP. An IP tunneled packet may match a dot1p entry, IP ToS precedence entry, or IP ToS DSCP entry when defined in the applied policy. An MPLS LSP may match a dot1p entry or MPLS EXP entry when defined.

Tunnel termination QoS override only applies to IP routing decisions when the tunnel encapsulation is removed. Non-IP routed packets within a terminating tunnel are ignored by the override and are forwarded as described in Network ingress.

Any tunnel received on the ingress IP interface that traverses the node (where the node is not the ultimate hop for the tunnel) is not affected by the QoS override mechanism and is forwarded as described in Network ingress.

Tunnel termination QoS override, provides the ability to ignore the network ingress QoS mapping of a terminated tunnel containing an IP packet that is to be routed to a base router or VPRN destination. This is useful when the mapping for the tunnel QoS marking does not accurately or completely reflect the required QoS handling for the IP routed packet. When the mechanism is enabled on an ingress network IP interface using the ingress **ler-use-dscp** parameter, the IP interface ignores the tunnel's QoS mapping and derive the internal forwarding class and profile based on the precedence or DSCP values within the routed IP header ToS field compared to the network QoS policy defined on the IP interface.

4.2.2 Network ingress IP match criteria

IP match criteria classification is supported in the ingress section of a network QoS policy.

The classification only applies to the outer IPv4 header of non-tunneled traffic. Consequently, the use of an IP criteria statement in a network QoS policy is ignored for received traffic when the network QoS policy is applied on the ingress network IP interface in the following cases:

- mesh SDPs in VPLS services
- spoke SDPs in VPLS and *X*pipe services
- spoke SDPs under an IP interface in an IES or VPRN service
- spoke SDPs in a VPRN service
- bindings created automatically by the **auto-bind-tunnel** command in a VPRN service
- IPv6 over IPv4 tunnels
- VXLAN bindings (egress VTEP, VNI)

The only exception is for traffic received on a Draft Rosen tunnel, for which classification on the outer IP header only is supported.

Attempting to apply a network QoS policy containing an IP criteria statement to any object except a network IP interface results in an error.

The following is an example configuration:

```
configure
   qos
      network 10 name "10" create
          ingress
              ip-criteria
                  entry 10 create
```

```
                    match
                        dst-ip 10.0.0.1/32
                    exit
                    action fc "h2" profile in
                exit
            exit
        exit
    exit
```

4.2.3 Network ingress IPv6 match criteria

IPv6 match criteria classification is supported in the ingress section of a network QoS policy.

The classification only applies to the outer IPv6 header of non-tunneled traffic; consequently, the use of an ipv6-criteria statement in a network QoS policy is ignored for received traffic when the network QoS policy is applied on the ingress network IP interface in the following cases:

- mesh SDPs in VPLS services
- spoke SDPs in VPLS and *X*pipe services
- spoke SDPs under an IP interface in an IES or VPRN service
- spoke SDPs in a VPRN service
- bindings created automatically by the **auto-bind-tunnel** command in a VPRN service
- IPv6 over IPv4 tunnels
- VXLAN bindings (egress VTEP, VNI)

Attempting to apply a network QoS policy containing an IPv6 criteria statement to any object except a network IP interface results in an error.

The following is an example configuration:

```
configure
   qos
      network 10 name “10” create
          ingress
              ipv6-criteria
                  entry 10 create
                      match
                          dst-ip 2001:db8:1000::1/128
                      exit
                      action fc "ef" profile in
                  exit
              exit
          exit
      exit
   exit
```

4.3 Network egress

The following types of QoS mapping decisions are applicable to the egress of a network QoS policy:

- IP DSCP mapping

- IP ToS precedence mapping
- IP criteria mapping
- IPv6 criteria mapping

Default dot1p/DE, DSCP, and EXP packet marking are defined for each forwarding class for in and out of profile packets, together with the remarking capability based on the trusted state of the packet's ingress interface.

The packets of a specified forwarding class can be redirected to an egress port queue group.

4.3.1 Egress packet reclassification based on IP precedence DSCP

The user enables IP precedence or DSCP based egress reclassification by applying the following command in the context of the network QoS policy applied to the egress context of a spoke SDP.

config>qos>network>egress>prec *ip-prec-value* [**fc** *fc-name*] [**profile** {**exceed** | **out** | **in** | **inplus**}]

config>qos>network>egress>dscp *dscp-name* [**fc** *fc-name*] [**profile** {**exceed** | **out** | **in** | **inplus**}]

The IP precedence bits used to match against DSCP reclassification rules come from the Type of Service (ToS) field within the IPv4 header or the Traffic Class field from the IPv6 header.

The IP DSCP bits used to match against DSCP reclassification rules come from the Type of Service (ToS) field within the IPv4 header or the Traffic Class field from the IPv6 header.

If the packet does not have an IP header, DSCP or IP-precedence based matching is not performed.

The IP precedence and DSCP based re-classification are supported on a network interface, on a CSC network interface in a VPRN, and on a PW used in an IES or VPRN spoke-interface. The CLI blocks the application of a network QoS policy with the egress re-classification commands to a network IP interface or to a spoke SDP part of L2 service. Conversely, the CLI does not allow the user to add the egress re-classification commands to a network QoS policy if it is being used by an L2 spoke SDP.

In addition, the egress re-classification commands only take effect if the redirection of the spoke SDP or CSC interface to use an egress port queue-group succeeds. For example, the following CLI commands succeed:

config>service>vprn>if>spoke-sdp>egress>qos *network-policy-id* **port-redirect-group** *queue-group-name* **instance** *instance-id*

config>service>ies>if>spoke-sdp>egress>qos *network-policy-id* **port-redirect-group** *queue-group-name* **instance** *instance-id*

config>service>vprn>nw-if>qos *network-policy-id* **port-redirect-group** *queue-group-name* **instance** *instance-id*

When the redirection command fails in CLI, the PW uses the network QoS policy assigned to the network IP interface, however any reclassification in the network QoS policy applied to the network interface is ignored.

4.3.2 Network egress IP match criteria

IP match criteria classification is supported in the egress section of a network QoS policy.

The configuration of egress prec/DSCP classification and the configuration of an egress IP criteria entry statement within a network QoS policy are mutually exclusive.

Network QoS policies containing egress IP criteria entry statements are only applicable to network interfaces.

The following is an example configuration:

```
configure
   qos
      network 10 name “10” create
          egress
              ip-criteria
                  entry 10 create
                      match
                          dst-ip 192.168.1.0/24
                      exit
                      action fc "af" profile out
                  exit
              exit
          exit
      exit
```

4.3.3 Network egress IPv6 match criteria

IPv6 match criteria classification is supported in the egress section of a network QoS policy.

The configuration of egress prec/DSCP classification and the configuration of an egress IPv6 criteria entry statement within a network QoS policy are mutually exclusive.

Network QoS policies containing egress IPv6 criteria entry statements are only applicable to network interfaces.

The following is an example configuration:

```
configure
   qos
      network 10 name “10” create
          egress
              ipv6-criteria
                  entry 10 create
                      match
                          dst-ip 2001:db8:2000::1/128
                      exit
                      action fc "ef" profile in
                  exit
              exit
          exit
      exit
```

4.4 QoS for self-generated (CPU) traffic on network interfaces

A user can specify a Differentiated Devices Code Point (DSCP), forwarding class (FC), and IEEE 802.1p values to be used by protocol packets generated by the node. This enables prioritization or deprioritization of every protocol (as required). The markings effect a change in behavior on ingress when queuing.

DSCP marking for internally generated control and management traffic should be used for the specific application. This can be configured per routing instance. For example, OSPF packets can carry a different

DSCP marking for the base instance different than for a VPRN service. ARP, IS-IS, and PPPoE are not IP protocols, so only 802.1p values can be configured.

The DSCP value can be set per application. When an application is configured to use a specified DSCP value, the 802.1p and MPLS EXP bits are marked in accordance with the network (default 802.1p value of 7) or access (default 802.1p value of 0) egress policy as it applies to the logical interface that the packet is egressing.

The configuration of self-generated QoS is supported in the base router, VPRN, and management contexts.

The default values for self-generated traffic on network interfaces are:

- **routing protocols (OSPF, BGP, and so on)**
  - Forwarding class is Network Control (NC).
  - DSCP value is NC1 (not applicable for ARP, IS-IS, and PPPoE).
  - 802.1p value is dependent on the egress QoS policy (7 by default).
- **management protocols (SSH, SNMP, and so on)**
  - Forwarding class is Network Control (NC).
  - DSCP value is AF41.
  - 802.1p value is dependent on the egress QoS policy (7 by default).

The default QoS values for self-generated traffic on network interfaces are listed in the following table.

Table 15: Default QoS values for self-generated traffic

Protocol	DSCP
ANCP	NC1
APS	NC1
ARP	N/A
BFD	NC1
BGP	NC1
BMP	AF41
Call Trace	AF41
Cflowd	NC1
DHCP	NC1, AF41, NC2
Diameter	AF41
DNS	AF41
FTP	AF41
gRPC	AF41

Protocol	DSCP
GTP	NC1, NC2
HTTP	AF41
ICMP	BE, NC1
IGMP	NC1
IGMP Reporter	NC1
IS-IS	N/A
L2TP	NC1
LDP	NC1
MLD	NC1
MPLS UDP Return	NC1
MCS (Multichassis Support)	NC1
MSDP	NC1
Mtrace2	NC1
NETCONF	AF41
ND (NDIS)	NC1, NC2
NTP/SNTP	NC1
OpenFlow	NC1
OSPF	NC1
PCEP	NC1
PIM	NC1
PPPoE	N/A
PTP	NC1
RADIUS	NC1
RIP	NC1
RSVP	NC1
sFlow	NC1
SNMP Gets/Sets	AF41
SNMP Traps	AF41

Protocol	DSCP
SRRP	NC1
SSH, SCP, SFTP	AF41
Syslog	AF41
TACACS+	AF41
Telnet	AF41
TFTP	AF41
Traceroute	BE
TWAMP, TWAMP Light	N/A
VRRP	NC1
WSC	NC1
XMPP	NC1

Note: The following usage guidelines apply to self-generated traffic:

- ICMP echo requests (type 8) initiated from the router use the DSCP value set by the **sgt-qos** command. For both ICMP echo requests (type 8) and ICMPv6 echo requests (type 128), the FC value is NC by default, or the value specified in the **ping** command parameter **fc** *fc-name*.
- The DSCP values for TWAMP and TWAMP Light test packets are not configured with **sgt-qos** commands. The DSCP value for TWAMP test packets reflected by the TWAMP server is specified in the TWAMP control process. The DSCP value for TWAMP Light test packets is set by the test configuration. The TWAMP Light reflector uses the arriving TWAMP test packet to determine the return DSCP value.
- Some applications have multiple DSCP default values depending on the context or service.
- Values configured with the **sgt-qos** command take precedence over the egress QoS policy configuration.
- Configurable values for ANCP, APS, LLDP, LMP, MCS, NETCONF (not affected by the configurable SSH value), OpenFlow, WSC, and XMPP are not supported.
- The **sgt-qos application dhcp** command, includes the marking of following DHCPv6 packets:
  - downstream DHCPv6 packets egressing on a subscriber group-interface
  - upstream DHCPv6 relayed packets from subscriber group-interfaces to a DHCPv6 server
  - upstream DHCPv6 relayed packages from IES or VPRN service interfaces to a DHCPv6 server
  - spoofed DHCPv6 release packets on behalf on a DHCPv6 host

4.4.1 Default DSCP mapping table

```
DSCP Name  DSCP Value  DSCP Value    DSCP Value    Label
```

```
            Decimal    Hexadecimal    Binary
============================================================
Default       0            0x00       0b000000    be
nc1           48           0x30       0b110000    h1
nc2           56           0x38       0b111000    nc
ef            46           0x2e       0b101110    ef
af11          10           0x0a       0b001010    assured
af12          12           0x0c       0b001100    assured
af13          14           0x0e       0b001110    assured
af21          18           0x12       0b010010    l1
af22          20           0x14       0b010100    l1
af23          22           0x16       0b010110    l1
af31          26           0x1a       0b011010    l1
af32          28           0x1c       0b011100    l1
af33          30           0x1d       0b011110    l1
af41          34           0x22       0b100010    h2
af42          36           0x24       0b100100    h2
af43          38           0x26       0b100110    h2

default*    0
```

*The default forwarding class mapping is used for all DSCP names/values for which there is no explicit forwarding class mapping.

4.5 Basic configurations

A basic network QoS policy must conform to the following:

- Each network QoS policy must have a unique policy ID.
- Include the definition of at least one queue.
- Specify the default-action.

4.5.1 Creating a network QoS policy

Configuring and applying QoS policies other than the default policy is optional. A default network policy of the appropriate type is applied to each router interface.

To create a network QoS policy when operating, define the following:

- A network policy ID value. The system does not dynamically assign a value.
- Include a description. The description provides a brief overview of policy features.
- Egress criteria can be modified to customize the forwarding class queues to be instantiated. Otherwise, the default values are applied.
  - **remarking**

    When enabled, this command remarks all packets that egress on the specified network port. The remarking is based on the forwarding class to DSCP and LSP EXP bit mapping defined under the egress node of the network QoS policy.
  - **forwarding class criteria**

    The forwarding class name represents an egress queue. The forwarding class criteria define the egress characteristics of the queue and the marking criteria of packets flowing through it.

- **DE marking**

  This specifies that the DE bit should be marked based on whether the packet profile is in-profile, inplus-profile, out-of-profile, or exceed-profile.

- **dot1p**

  The dot1p value is used for all VLAN-tagged packets requiring marking that egress on this forwarding class queue, with the option of specifying a different value for packets that are in-profile or out-of-profile. Inplus-profile traffic is marked with the same values as in-profile traffic. Exceed-profile traffic is marked with the same values as out-of-profile traffic.

- **DSCP**

  The DSCP value is used for all IP packets requiring marking that egress on this forwarding class queue that are in-profile or out-of-profile. Inplus-profile traffic is marked with the same values as in-profile traffic. Exceed-profile traffic is marked with the same values as out-of-profile traffic.

- **LSP EXP**

  The EXP value is used for all MPLS-labeled packets requiring marking that egress on this forwarding class queue that are in-profile or out-of-profile. Inplus-profile traffic is marked with the same values as in-profile traffic. Exceed-profile traffic is marked with the same values as out-of-profile traffic.

- **port redirection**

  This specifies that the traffic should be redirected to a network egress queue group policer or queue.

– **DSCP**

  Creates a mapping between the DSCP of the network egress traffic and the forwarding class and profile. Egress traffic that matches the specified DSCP is assigned to the corresponding forwarding class with the specified profile.

– **IP criteria**

  Creates a mapping between the possible match criteria of the network egress traffic and the forwarding class and profile. Egress traffic that matches the IPv4 criteria is assigned to the corresponding forwarding class and profile.

– **IPv6 criteria**

  Creates a mapping between the possible match criteria of the network egress traffic and the forwarding class and profile. Egress traffic that matches the IPv6 criteria is assigned to the corresponding forwarding class and profile.

– **prec**

  Creates a mapping between the IP precedence of the network egress traffic and the forwarding class and profile. Egress traffic that matches the specified IP precedence is assigned to the corresponding forwarding class with the specified profile.

- Ingress criteria specifies the DSCP- or dot1p-to-forwarding class mapping for all IP packets and defines the MPLS EXP bits-to-forwarding class mapping for all labeled packets.

  – **default action**

    Defines the default action to be taken for packets that have an undefined configured classification. The default action specifies the forwarding class and profile to which such packets are assigned.

  – **dot1p**

Creates a mapping between the dot1p of the network ingress traffic and the forwarding class and profile. Ingress traffic that matches the specified dot1p is assigned to the corresponding forwarding class and profile.

– **DSCP**

Creates a mapping between the DSCP of the network ingress traffic and the forwarding class and profile. Ingress traffic that matches the specified DSCP is assigned to the corresponding forwarding class.

– **forwarding class**

The forwarding class name represents an ingress queue.

– **IP criteria**

Creates a mapping between the possible match criteria of the network ingress traffic and the forwarding class and profile. Ingress traffic that matches the IPv4 criteria is assigned to the corresponding forwarding class and profile.

– **IPv6 criteria**

Creates a mapping between the possible match criteria of the network ingress traffic and the forwarding class and profile. Ingress traffic that matches the IPv6 criteria is assigned to the corresponding forwarding class and profile.

– **LER use DSCP**

Specifies that DSCP matching based on the tunneled IP packet should be used on an LER instead of matching on the outer encapsulated header.

– **LSP EXP**

Creates a mapping between the LSP EXP bits of the network ingress traffic and the forwarding class and profile. Ingress traffic that matches the specified LSP EXP bits is assigned to the corresponding forwarding class and profile.

Use the following CLI syntax to create a network QoS policy:

```
config>qos# network network-policy-id [create]
  description description-string
  egress
    dscp dscp-name fc fc-name profile {in | out | exceed
      | inplus}
    fc {be | l2 | af | l1 | h2 | ef | h1 | nc}
      de-mark [force de-value]
      dot1p dot1p-priority
      dot1p-in-profile dot1p-priority
      dot1p-out-profile dot1p-priority
      dscp-in-profile dscp-name
      dscp-out-profile dscp-name
      lsp-exp-in-profile lsp-exp-value
      lsp-exp-out-profile lsp-exp-value
      port-redirect-group {queue queue-id | policer
        plcr-id [queue queue-id]}
prec dscp-name fc fc-name profile {in | out | exceed
  | inplus}
  remarking
ip-criteria
  entry entry-id [create]
    action [fc fc-name profile {in | out | exceed
      | inplus}]
      [port-redirect-group {queue queue-id |
```

```
        policer policer-id [queue queue-id]}]
    description description-string
    match [protocol protocol-id]
      dscp dscp-name
      dst-ip {ip-address/mask | ip-address ipv4-
        address-mask}
      dst-port [{lt | gt | eq} {dst-port-number |
        range start end}]
      fragment {true | false}
      icmp-type icmp-type
      src-ip {ip-address/mask | ip-address ipv4-
        address-mask}
      src-port [{lt | gt | eq} {src-port-number |
        range start end}]
   renum old-entry-id new-entry-id
ipv6-criteria
  entry entry-id [create]
    action fc fc-name profile {in | out |exceed |
      inplus}]
      [port-redirect-group {queue queue-id |
        policer policer-id [queue queue-id]}]
    description description-string
    match [next-header next-header]
      dscp dscp-name
      dst-ip {ipv6-address/mask | ipv6-address
ipv6-address-mask}
      dst-port [{lt | gt | eq} {dst-port-number |
        range start end}]
      fragment {true | false | first-only | non-
        first-only}
      src-ip {ipv6-address/mask | ipv6-address
ipv6-address-mask}
      src-port [{lt | gt | eq} src-port-number
      src-port range start end
   renum old-entry-id new-entry-id
 ler-use-dscp
 lsp-exp lsp-exp-value fc fc-name profile {in | out}
scope {exclusive | template}
```

```
A:ALA-10:A:ALA-12>config>qos# info
#------------------------------------------
echo "QoS Policy Configuration"
#------------------------------------------
...
     network 600 create
            description "Network Egress Policy"
            ingress
                default-action fc ef profile in
            exit
            egress
                remarking
            exit
        exit
...
#------------------------------------------
A:ALA-12>config>qos#
```

4.5.2 Applying network QoS policies

Use the following CLI syntax to apply network policies to the router access uplink port's IP interfaces:

```
config>router
interface interface-name
  qos network-policy-id
```

The following output displays the configuration for router interface ALA-1-2 with network policy 600 applied to the interface.

```
A:ALA-7>config>router# info
#------------------------------------------
echo "IP Configuration"
#------------------------------------------
...
        interface "ALA-1-2"
            address 10.10.4.3/24
            qos 600
        exit
...
----------------------------------------------
A:ALA-7>config>router#
```

4.5.3 Default network QoS policy values

The default network policy for IP interfaces is identified as policy ID 1. Default policies cannot be modified or deleted. Table 16: Network policy defaults lists default network policy parameters.

Table 16: Network policy defaults

Field				Default	
description				Default network QoS policy.	
scope				template	
ingress					
	default-action			fc be profile out	
		dscp			
			be	fc be	profile out
			ef	fc ef	profile in
			cs1	fc l2	profile in
			nc1	fc h1	profile in
			nc2	fc nc	profile in
			af11	fc af	profile in

Field				Default	
			af12	fc af	profile out
			af13	fc af	profile out
			af21	fc l1	profile in
			af22	fc l1	profile out
			af23	fc l1	profile out
			af31	fc l1	profile in
			af32	fc l1	profile out
			af33	fc l1	profile out
			af41	fc h2	profile in
			af42	fc h2	profile out
			af43	fc h2	profile out
		lsp-exp			
			0	fc be	profile out
			1	fc l2	profile in
			2	fc af	profile out
			3	fc af	profile in
			4	fc h2	profile in
			5	fc ef	profile in
			6	fc h1	profile in
			7	fc nc	profile in
egress					
	remarking			no	
	fc af				
		dscp-in-profile		af11	
		dscp-out-profile		af12	
		lsp-exp-in-profile		3	
		lsp-exp-out-profile		2	
	fc be				

Field			Default	
		dscp-in-profile	be	
		dscp-out-profile	be	
		lsp-exp-in-profile	0	
		lsp-exp-out-profile	0	
	fc ef			
		dscp-in-profile	ef	
		dscp-out-profile	ef	
		lsp-exp-in-profile	5	
		lsp-exp-out-profile	5	
	fc h1			
		dscp-in-profile	nc1	
		dscp-out-profile	nc1	
		lsp-exp-in-profile	6	
		lsp-exp-out-profile	6	
	fc h2			
		dscp-in-profile	af41	
		dscp-out-profile	af42	
		lsp-exp-in-profile	4	
		lsp-exp-out-profile	4	
	fc l			
		dscp-in-profile	af21	
		dscp-out-profile	af22	
		lsp-exp-in-profile	3	
		lsp-exp-out-profile	2	
	fc l2			
		dscp-in-profile	cs1	
		dscp-out-profile	cs1	
		lsp-exp-in-profile	1	

Field			Default	
		lsp-exp-out-profile	1	
	fc nc			
		dscp-in-profile	nc2	
		dscp-out-profile	nc2	
		lsp-exp-in-profile	7	
		lsp-exp-out-profile	7	

The following output displays the default configuration:

```
A:ALA-49>config>qos>network# info detail
----------------------------------------------
            description "Default network QoS policy."
            scope template
            ingress
                default-action fc be profile out
                no ler-use-dscp
                dscp be fc be profile out
                dscp ef fc ef profile in
                dscp cs1 fc l2 profile in
                dscp nc1 fc h1 profile in
                dscp nc2 fc nc profile in
                dscp af11 fc af profile in
                dscp af12 fc af profile out
                dscp af13 fc af profile out
                dscp af21 fc l1 profile in
                dscp af22 fc l1 profile out
                dscp af23 fc l1 profile out
                dscp af31 fc l1 profile in
                dscp af32 fc l1 profile out
                dscp af33 fc l1 profile out
                dscp af41 fc h2 profile in
                dscp af42 fc h2 profile out
                dscp af43 fc h2 profile out
                lsp-exp 0 fc be profile out
                lsp-exp 1 fc l2 profile in
                lsp-exp 2 fc af profile out
                lsp-exp 3 fc af profile in
                lsp-exp 4 fc h2 profile in
                lsp-exp 5 fc ef profile in
                lsp-exp 6 fc h1 profile in
                lsp-exp 7 fc nc profile in
            exit
            egress
                no remarking
                fc af
                    dscp-in-profile af11
                    dscp-out-profile af12
                    lsp-exp-in-profile 3
                    lsp-exp-out-profile 2
                    dot1p-in-profile 2
                    dot1p-out-profile 2
                exit
                fc be
                    dscp-in-profile be
```

```
                    dscp-out-profile be
                    lsp-exp-in-profile 0
                    lsp-exp-out-profile 0
                    dot1p-in-profile 0
                    dot1p-out-profile 0
                exit
                fc ef
                    dscp-in-profile ef
                    dscp-out-profile ef
                    lsp-exp-in-profile 5
                    lsp-exp-out-profile 5
                    dot1p-in-profile 5
                    dot1p-out-profile 5
                exit
                fc h1
                    dscp-in-profile nc1
                    dscp-out-profile nc1
                    lsp-exp-in-profile 6
                    lsp-exp-out-profile 6
                    dot1p-in-profile 6
                    dot1p-out-profile 6
                exit
                fc h2
                    dscp-in-profile af41
                    dscp-out-profile af42
                    lsp-exp-in-profile 4
                    lsp-exp-out-profile 4
                    dot1p-in-profile 4
                    dot1p-out-profile 4
                exit
                fc l1
                    dscp-in-profile af21
                    dscp-out-profile af22
                    lsp-exp-in-profile 3
                    lsp-exp-out-profile 2
                    dot1p-in-profile 3
                    dot1p-out-profile 3
                exit
                fc l2
                    dscp-in-profile cs1
                    dscp-out-profile cs1
                    lsp-exp-in-profile 1
                    lsp-exp-out-profile 1
                    dot1p-in-profile 1
                    dot1p-out-profile 1
                exit
                fc nc
                    dscp-in-profile nc2
                    dscp-out-profile nc2
                    lsp-exp-in-profile 7
                    lsp-exp-out-profile 7
                    dot1p-in-profile 7
                    dot1p-out-profile 7
                exit
            exit
-------------------------------------------
A:ALA-49>config>qos>network#
```

4.6 Service management tasks

4.6.1 Deleting QoS policies

A network policy is associated by default with router interfaces.

The default policy can be replaced with a non-default policy but cannot be removed from the configuration. When a non-default policy is removed, the policy association reverts to the appropriate default network policy.

```
config>router
interface interface-name
  qos network-policy-id
```

The following output displays an example configuration.

```
A:ALA-7>config>router# info
#------------------------------------------
echo "IP Configuration"
#------------------------------------------
...
        interface "ALA-1-2"
            address 10.10.4.3/24 broadcast host-ones
            no port
            no arp-timeout
            no allow-directed-broadcasts
            icmp
                mask-reply
                redirects 100 10
                unreachables 100 10
                ttl-expired 100 10
            exit
            qos 1
            ingress
                no filter
            exit
            egress
                no filter
            exit
            no mac
            no cflowd
            no shutdown
        exit
        interface "ALA-1-3"
...
#------------------------------------------
A:ALA-7>config>router#
```

4.6.2 Removing a policy from the QoS configuration

To delete a network policy, enter the following command:

config>qos# no network *network-policy-id*

4.6.3 Editing QoS policies

Existing policies, except the default policies and entries in the CLI, can be changed. The changes are applied immediately to all interfaces where the policy is applied. To prevent configuration errors, use the copy command to make a duplicate of the original policy in a work area, make the edits, then overwrite the original policy.

5 Network queue QoS policies

5.1 Overview

Network queue policies define the ingress network queuing at the FP network node level. Network queue policies are also used at the Ethernet port and SONET/SDH path level to define network egress queuing.

There is one default network queue policy. Each policy can have up to 16 queues (unicast and multicast). The default policies can be copied but they cannot be deleted or modified. The default policy is identified as **network-queue default**. Default network queue policies are applied to the card FP ingress network and port Ethernet network. Other network queue QoS policies must be explicitly created and associated.

For information about the tasks and commands necessary to access the CLI and to configure and maintain routers, see the "Entering CLI Commands" chapter in the *7705 SAR Gen 2 Classic CLI Command Reference Guide*.

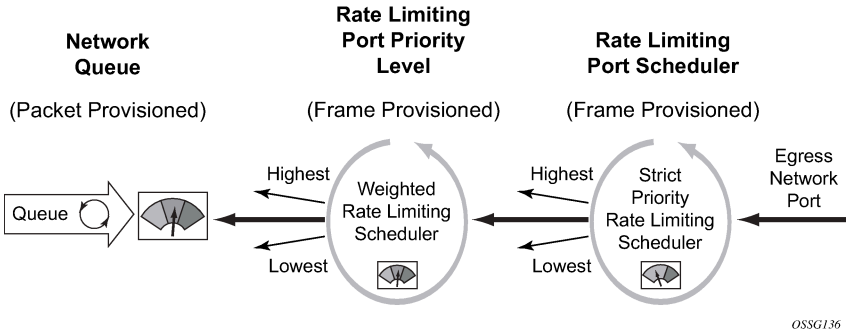
5.2 Network queue parent scheduler

Network queues support port scheduler parent priority-level associations. Using a port scheduler policy definition and mapping network queues to a port parent priority level, H-QoS functionality is supported providing eight levels of strict priority and weights within the same priority. A network queue's bandwidth is allocated using the within-CIR and above-CIR scheme normal for port schedulers.

Queue CIR and PIR percentages when port-based schedulers are in effect are based on frame-offered-load calculations. Figure 9: Bandwidth distribution on network port with port-based scheduling shows port-based virtual scheduling bandwidth distribution.

A network queue with a port parent association that exists on a port without a scheduler policy defined is considered to be orphaned.

Figure 9: Bandwidth distribution on network port with port-based scheduling



5.3 Basic configurations

A basic network queue QoS policy must conform to the following:

- Each network queue QoS policy must have a unique policy name.
- Queue parameters can be modified but cannot be deleted.

5.3.1 Creating a network queue QoS policy

Configuring and applying QoS policies other than the default policy is optional. A default network queue policy is applied to the card FP ingress network and port Ethernet network.

To create a network queue policy, enter the **configure qos network-queue** command with an unused policy name followed by the **create** keyword. Once created, the parameters in the policy, such as a description, the mapping of forwarding classes to queues, and the queue properties can be configured.

5.3.2 Applying network queue QoS policies

Apply network queue policies to the following entities:

- FPs
- Ethernet Ports
- SONET/SDH Ports

5.3.2.1 FPs

Use the following CLI syntax to apply a network queue policy to an FP ingress network:

```
config>card
fp fp-number
  ingress
    network
      queue-policy name
```

The following output displays FP ingress network queue policy using the default policy.

```
*A:PE>config>card>fp# info detail
----------------------------------------------
            ingress
                network
                    queue-policy "default"
                exit
            exit
----------------------------------------------
*A:PE>config>card>fp#
```

5.3.2.2 Ethernet ports

Use the following CLI syntax to apply a network queue policy to an Ethernet port.

```
config>port#
ethernet
  network
    queue-policy name
```

The following output displays the port configuration.

```
A:ALA-49>config>port# info
----------------------------------------------
        ethernet
            network
                queue-policy "nq1"
            exit
        exit
        no shutdown
----------------------------------------------
A:ALA-49>config>port#
```

5.3.3 Default network queue policy values

The default network queue policy is named **default** and cannot be modified or deleted. The default policy parameters are shown below.

```
*A:PE>config>qos# network-queue "default"
*A:PE>config>qos>network-queue# info detail
----------------------------------------------
            description "Default network queue QoS policy."
            hs-attachment-policy "default"
            queue 1 auto-expedite create
                no port-parent
                no avg-frame-overhead
                rate 100 cir 0 fir 0
                adaptation-rule pir closest cir closest fir closest
                mbs 50
                cbs 1
                hs-mbs 100
                hs-wrr-weight 1
                hs-class-weight 1
                hs-wred-queue policy "_tmnx_hs_default"
                no hs-alt-port-class-pool
                drop-tail
                    low
                        percent-reduction-from-mbs default
                    exit
                exit
            exit
            queue 2 auto-expedite create
                no port-parent
                no avg-frame-overhead
                rate 100 cir 25 fir 0
                adaptation-rule pir closest cir closest fir closest
                mbs 50
                cbs 3
                hs-mbs 100
```

```
                hs-wrr-weight 1
                hs-class-weight 1
                hs-wred-queue policy "_tmnx_hs_default"
                no hs-alt-port-class-pool
                drop-tail
                    low
                        percent-reduction-from-mbs default
                    exit
                exit
            exit
            queue 3 auto-expedite create
                no port-parent
                no avg-frame-overhead
                rate 100 cir 25 fir 0
                adaptation-rule pir closest cir closest fir closest
                mbs 50
                cbs 10
                hs-mbs 100
                hs-wrr-weight 1
                hs-class-weight 1
                hs-wred-queue policy "_tmnx_hs_default"
                no hs-alt-port-class-pool
                drop-tail
                    low
                        percent-reduction-from-mbs default
                    exit
                exit
            exit
            queue 4 auto-expedite create
                no port-parent
                no avg-frame-overhead
                rate 100 cir 25 fir 0
                adaptation-rule pir closest cir closest fir closest
                mbs 25
                cbs 3
                hs-mbs 100
                hs-wrr-weight 1
                hs-class-weight 1
                hs-wred-queue policy "_tmnx_hs_default"
                no hs-alt-port-class-pool
                drop-tail
                    low
                        percent-reduction-from-mbs default
                    exit
                exit
            exit
            queue 5 auto-expedite create
                no port-parent
                no avg-frame-overhead
                rate 100 cir 100 fir 0
                adaptation-rule pir closest cir closest fir closest
                mbs 50
                cbs 10
                hs-mbs 100
                hs-wrr-weight 1
                hs-class-weight 1
                hs-wred-queue policy "_tmnx_hs_default"
                no hs-alt-port-class-pool
                drop-tail
                    low
                        percent-reduction-from-mbs default
                    exit
                exit
            exit
```

```
            queue 6 auto-expedite create
                no port-parent
                no avg-frame-overhead
                rate 100 cir 100 fir 0
                adaptation-rule pir closest cir closest fir closest
                mbs 50
                cbs 10
                hs-mbs 100
                hs-wrr-weight 1
                hs-class-weight 1
                hs-wred-queue policy "_tmnx_hs_default"
                no hs-alt-port-class-pool
                drop-tail
                    low
                        percent-reduction-from-mbs default
                    exit
                exit
            exit
            queue 7 auto-expedite create
                no port-parent
                no avg-frame-overhead
                rate 100 cir 10 fir 0
                adaptation-rule pir closest cir closest fir closest
                mbs 25
                cbs 3
                hs-mbs 100
                hs-wrr-weight 1
                hs-class-weight 1
                hs-wred-queue policy "_tmnx_hs_default"
                no hs-alt-port-class-pool
                drop-tail
                    low
                        percent-reduction-from-mbs default
                    exit
                exit
            exit
            queue 8 auto-expedite create
                no port-parent
                no avg-frame-overhead
                rate 100 cir 10 fir 0
                adaptation-rule pir closest cir closest fir closest
                mbs 25
                cbs 3
                hs-mbs 100
                hs-wrr-weight 1
                hs-class-weight 1
                hs-wred-queue policy "_tmnx_hs_default"
                no hs-alt-port-class-pool
                drop-tail
                    low
                        percent-reduction-from-mbs default
                    exit
                exit
            exit
            queue 9 multipoint auto-expedite create
                no port-parent
                no avg-frame-overhead
                rate 100 cir 0 fir 0
                adaptation-rule pir closest cir closest fir closest
                mbs 50
                cbs 1
                hs-mbs 100
                hs-wrr-weight 1
                hs-class-weight 1
```

```
                hs-wred-queue policy "_tmnx_hs_default"
                no hs-alt-port-class-pool
                drop-tail
                    low
                        percent-reduction-from-mbs default
                    exit
                exit
            exit
            queue 10 multipoint auto-expedite create
                no port-parent
                no avg-frame-overhead
                rate 100 cir 5 fir 0
                adaptation-rule pir closest cir closest fir closest
                mbs 50
                cbs 1
                hs-mbs 100
                hs-wrr-weight 1
                hs-class-weight 1
                hs-wred-queue policy "_tmnx_hs_default"
                no hs-alt-port-class-pool
                drop-tail
                    low
                        percent-reduction-from-mbs default
                    exit
                exit
            exit
            queue 11 multipoint auto-expedite create
                no port-parent
                no avg-frame-overhead
                rate 100 cir 5 fir 0
                adaptation-rule pir closest cir closest fir closest
                mbs 50
                cbs 1
                hs-mbs 100
                hs-wrr-weight 1
                hs-class-weight 1
                hs-wred-queue policy "_tmnx_hs_default"
                no hs-alt-port-class-pool
                drop-tail
                    low
                        percent-reduction-from-mbs default
                    exit
                exit
            exit
            queue 12 multipoint auto-expedite create
                no port-parent
                no avg-frame-overhead
                rate 100 cir 5 fir 0
                adaptation-rule pir closest cir closest fir closest
                mbs 25
                cbs 1
                hs-mbs 100
                hs-wrr-weight 1
                hs-class-weight 1
                hs-wred-queue policy "_tmnx_hs_default"
                no hs-alt-port-class-pool
                drop-tail
                    low
                        percent-reduction-from-mbs default
                    exit
                exit
            exit
            queue 13 multipoint auto-expedite create
                no port-parent
```

```
                no avg-frame-overhead
                rate 100 cir 100 fir 0
                adaptation-rule pir closest cir closest fir closest
                mbs 50
                cbs 1
                hs-mbs 100
                hs-wrr-weight 1
                hs-class-weight 1
                hs-wred-queue policy "_tmnx_hs_default"
                no hs-alt-port-class-pool
                drop-tail
                    low
                        percent-reduction-from-mbs default
                    exit
                exit
            exit
            queue 14 multipoint auto-expedite create
                no port-parent
                no avg-frame-overhead
                rate 100 cir 100 fir 0
                adaptation-rule pir closest cir closest fir closest
                mbs 50
                cbs 1
                hs-mbs 100
                hs-wrr-weight 1
                hs-class-weight 1
                hs-wred-queue policy "_tmnx_hs_default"
                no hs-alt-port-class-pool
                drop-tail
                    low
                        percent-reduction-from-mbs default
                    exit
                exit
            exit
            queue 15 multipoint auto-expedite create
                no port-parent
                no avg-frame-overhead
                rate 100 cir 10 fir 0
                adaptation-rule pir closest cir closest fir closest
                mbs 25
                cbs 1
                hs-mbs 100
                hs-wrr-weight 1
                hs-class-weight 1
                hs-wred-queue policy "_tmnx_hs_default"
                no hs-alt-port-class-pool
                drop-tail
                    low
                        percent-reduction-from-mbs default
                    exit
                exit
            exit
            queue 16 multipoint auto-expedite create
                no port-parent
                no avg-frame-overhead
                rate 100 cir 10 fir 0
                adaptation-rule pir closest cir closest fir closest
                mbs 25
                cbs 1
                hs-mbs 100
                hs-wrr-weight 1
                hs-class-weight 1
                hs-wred-queue policy "_tmnx_hs_default"
                no hs-alt-port-class-pool
```

```
                drop-tail
                    low
                        percent-reduction-from-mbs default
                    exit
                exit
            exit
            hs-wrr-group 1
                rate 100
                adaptation-rule pir closest
                hs-class-weight 1
            exit
            hs-wrr-group 2
                rate 100
                adaptation-rule pir closest
                hs-class-weight 1
            exit
            fc af create
                multicast-queue 11
                queue 3
                exit
            exit
            fc be create
                multicast-queue 9
                queue 1
                exit
            exit
            fc ef create
                multicast-queue 14
                queue 6
                exit
            exit
            fc h1 create
                multicast-queue 15
                queue 7
                exit
            exit
            fc h2 create
                multicast-queue 13
                queue 5
                exit
            exit
            fc l1 create
                multicast-queue 12
                queue 4
                exit
            exit
            fc l2 create
                multicast-queue 10
                queue 2
                exit
            exit
            fc nc create
                multicast-queue 16
                queue 8
                exit
            exit
-------------------------------------------
*A:PE>config>qos>network-queue#
```

5.4 Service management tasks

This section discusses network queue QoS policy service management tasks.

5.4.1 Deleting QoS policies

A network queue policy is associated by default with the card FP ingress network and port Ethernet network. The default policy can be replaced with a customer-configured policy but cannot entirely be removed. When a QoS policy is removed, the policy association reverts to the default network-queue policy.

To delete a user-created network queue policy, enter the following command:

config>qos# no network-queue *policy-name*

Example:

config>qos# no network-queue nq1

5.4.2 Removing a policy from the QoS configuration

To delete a network policy, enter the following command:

config>qos# no network-queue *policy-name*

Example:

config>qos# no network-queue test

5.4.3 Editing QoS policies

Existing policies, except the default policies, and entries in the CLI can be modified. The changes are applied immediately to all interfaces where the policy is applied. To prevent configuration errors, use the copy command to make a duplicate of the original policy to a work area, make the edits, then overwrite the original policy.

6 Service ingress and egress QoS policies

6.1 Overview

There is one default service ingress policy and one default service egress policy. Each policy can have up to 32 ingress queues and 8 egress queues per service.

The default policies can be copied and modified but they cannot be deleted. The default policies are identified as policy ID 1.

The default policies are applied to the appropriate interface, by default. For example, the default SAP ingress policy is applied to access ingress SAPs. The default SAP egress policy is applied to access egress SAPs. Other QoS policies must be explicitly associated.

For information about the tasks and commands necessary to access the CLI and to configure and maintain routers, see the "Entering CLI Commands" chapter in the *7705 SAR Gen 2 Classic CLI Command Reference Guide*.

6.2 Basic configurations

A basic service egress QoS policy must have the following:

- a unique service egress QoS policy ID
- a QoS policy scope of template or exclusive
- at least one defined default queue

A basic service ingress QoS policy must have the following:

- a unique service ingress QoS policy ID
- a QoS policy scope of template or exclusive
- at least one default unicast forwarding class queue
- at least one multipoint forwarding class queue

Configuring and applying QoS policies is optional. If no QoS policy is explicitly applied to a SAP or IP interface, a default QoS policy is applied.

6.3 Service ingress QoS policy

To create a service ingress policy, define the following:

- a policy ID value. The system does not dynamically assign a value.
- a description. The description provides a brief overview of policy features.
- a default forwarding class for the policy. All packets received on an ingress SAP using this ingress QoS policy are classified to the default forwarding class.

- a default priority for all packets received on an ingress SAP using this policy
- mappings from incoming packet contents to a forwarding class, then separately, from the forwarding class to queue
- forwarding class parameters:
  - Modify the **multicast-queue** default value to override the default multicast forwarding type queues mapping for **fc** *fc-name*.
  - Modify the **unknown-queue** default value to override the default unknown unicast forwarding type queues mapping for **fc** *fc-name*.
  - Modify the **broadcast-queue** default value to override the default broadcast forwarding type queues mapping for **fc** *fc-name*.
- a precedence value for the forwarding class or enqueuing priority when a packet is marked with an IP precedence value
- IP, IPv6, and MAC-based SAP ingress policies to select the appropriate ingress queue and corresponding forwarding class for matched traffic
- a SAP ingress policy created with a template scope. The scope can be modified to exclusive for a special one-time use policy. Otherwise, the **template** scope enables the policy to be applied to multiple SAPs.

The following displays an service ingress policy configuration:

```
A:ALA-7>config>qos>sap-ingress# info
----------------------------------------------
...
        sap-ingress 100 create
            description "Used on VPN sap"
...
----------------------------------------------
A:ALA-7>config>qos>sap-ingress#
```

6.3.1 Service ingress QoS queue

To create a service ingress queue, define the following:

- a new queue ID value. The system does not dynamically assign a value
- queue parameters. Ingress queues support multipoint queues, explicit and auto-expedite hardware queue scheduling, and parent virtual scheduler definition.

The following displays an ingress queue configuration:

```
A:ALA-7>config>qos# info
#------------------------------------------
echo "QoS Policy Configuration"
#------------------------------------------
...
        sap-ingress 100 create
            description "Used on VPN sap"
            queue 1 create
            exit
            queue 2 multipoint create
            exit
            queue 10 create
```

```
                parent VPN_be
                rate 11000
            exit
            queue 12 create
                parent VPN_priority
                rate 11000
            exit
            queue 13 create
                parent VPN_reserved
                rate 1
            exit
            queue 15 create
                parent VPN_video
                rate 1500 cir 1500
            exit
            queue 16 create
                parent VPN_voice
                rate 2500 cir 2500
            exit
            queue 17 create
                parent VPN_nc
                rate 100 cir 36
            exit
            queue 20 multipoint create
                parent VPN_be
                rate 11000
            exit
            queue 22 multipoint create
                parent VPN_priority
                rate 11000
            exit
            queue 23 multipoint create
                parent VPN_reserved
                rate 1
            exit
            queue 25 multipoint create
                parent VPN_video
                rate 1500 cir 1500
            exit
            queue 26 multipoint create
                parent VPN_voice
                rate 2500 cir 2500
            exit
            queue 27 multipoint create
                parent VPN_nc
                rate 100 cir 36
            exit
...
#------------------------------------------
A:ALA-7>config>qos#
```

6.3.2 Ingress percent-rate support

The **percent-rate** command is supported in a SAP ingress QoS policy for **pir** and **cir** parameters for both queues and policers, with the **fir** parameter supported only for queues on FP4 or later hardware that is ignored when the related policy is applied to FP3-based hardware. For **pir**, the range is 0.01 to 100.00, and for **cir** and **fir**, the range is 0.00 to 100.00.

For queues, when the queue rate is **percent-rate**, either a **local-limit** or a **port-limit** can be applied.

When the **local-limit** is used, the **percent-rate** is relative to the queue's parent scheduler rate. If there is no parent scheduler rate, or its rate is **max**, the **port-limit** is used. When the **port-limit** is used, the **percent-rate** is relative to the rate of the port (including the ingress-rate setting) to which the queue is attached. The **port-limit** is the default.

For policers, the percent-rate rate is always relative to the immediate parent root policer/arbiter rate or the FP capacity.

The following shows a SAP ingress QoS policy configuration:

```
*A:PE>config>qos>sap-ingress# queue 1 percent-rate
  - no percent-rate
  - percent-rate <pir-percent> [cir <cir-percent>] [fir <fir-percent>] [port-limit | local-
limit] [fir <fir-percent>]
  - percent-rate <pir-percent> police [port-limit | local-limit]

 <pir-percent>        : [0.01..100.00]
 <cir-percent>        : [0.00..100.00]
 <fir-percent>        : [0.00..100.00]
 <police>             : keyword
 <port-limit | local-*> : keyword


*A:PE>config>qos>sap-ingress# policer 1 percent-rate
  - no percent-rate
  - percent-rate <pir-percent> [cir <cir-percent>]

 <pir-percent>        : [0.01..100.00]
 <cir-percent>        : [0.00..100.00]


*A:PE>config>qos>sap-ingress#
```

6.3.3 Ingress forwarding class (FC)

The following displays a forwarding class and precedence configurations:

```
A:ALA-7>config>qos# info
#------------------------------------------
...
            fc af create
                queue 12
                broadcast-queue 22
                multicast-queue 22
                unknown-queue 22
            exit
            fc be create
                queue 10
                broadcast-queue 20
                multicast-queue 20
                unknown-queue 20
            exit
            fc ef create
                queue 13
                broadcast-queue 23
                multicast-queue 23
                unknown-queue 23
            exit
            fc h1 create
                queue 15
```

```
                broadcast-queue 25
                multicast-queue 25
                unknown-queue 25
            exit
            fc h2 create
                queue 16
                broadcast-queue 26
                multicast-queue 26
                unknown-queue 26
            exit
            fc nc create
                queue 17
                broadcast-queue 27
                multicast-queue 27
                unknown-queue 27
            exit
            prec 0 fc be
            prec 2 fc af
            prec 3 fc ef
            prec 5 fc h1
            prec 6 fc h2
            prec 7 fc nc
...
#----------------------------------------
A:ALA-7>config>qos#
```

6.3.4 Ingress IP match criteria

When specifying SAP ingress match criteria, only one match criteria type (IP/IPv6 or MAC) can be configured in the SAP ingress QoS policy.

The following displays an ingress IP criteria configuration:

```
A:ALA-7>config>qos# info
...
#----------------------------------------
echo "QoS Policy Configuration"
#----------------------------------------
...
        sap-ingress 100 create
...
            ip-criteria
                entry 10 create
                    description "Entry 10-FC-AF"
                    match protocol 6
                        src-ip 10.10.10.103/24
                    exit
                    action fc af priority high
                exit
                entry 20 create
                    description "Entry 20-FC-BE"
                    match protocol 17
                        dst-port eq 255
                    exit
                    no action
                exit
            exit
        exit
..
#----------------------------------------
```

```
A:ALA-7>config>qos#
```

6.3.5 Ingress IPv6 match criteria

When specifying SAP ingress match criteria, only one match criteria type (IP/IPv6 or MAC) can be configured in the SAP ingress QoS policy.

The following displays an ingress IPv6 criteria configuration:

```
A:ALA-48>config>qos>sap-ingress# info
----------------------------------------------
            queue 1 create
            exit
            queue 11 multipoint create
            exit
            ip-criteria
            exit
            ipv6-criteria
                entry 10 create
                    description "IPv6 SAP-ingress policy"
                    match
                        src-ip 2001:db8:1000::/64
                        dst-ip 2001:db8:2000::/64
                    exit
                    action fc be priority low
                exit
                entry 20 create
                    description "Entry 20-FC-AF"
                    match next-header tcp
                        src-port eq 500
                    exit
                    action fc af priority high
                exit
            exit
----------------------------------------------
A:ALA-48>config>qos>sap-ingress#
```

6.3.6 Tagging of Ingress IP-criteria and IPv6-criteria

The SAP ingress QoS policy allows the assignment of a tag to each IPv4/IPv6 criteria statement entry. This is useful when a single SAP ingress QoS policy needs to be used for a different service context and it is still needed to apply service specific entries at individual SAP level.

In this concept, the SAP ingress policy can contain entries with different tag values as well as untagged entries. The user configures a tag entry override per-SAP to select which tagged entries are included for the related SAP (untagged entries are always included). Using this tagging concept and matching on destination-port are mutually exclusive.

In the following example, a base configuration IPv4 prefix list is created together with two other lists (list1 and list2) that are used for different VPRN IP interfaces. Also, a base configuration IPv6 prefix list is created together with two other lists (list1 and list2) that are to be used for different VPRN IP interfaces.

The following entries are used for both the IPv4 and IPv6 criteria statement:

- VPRN 1/int: 10, 20, 30

- VPRN 2/int: 10, 20, 30, 1000
- VPRN 3/int: 10, 20, 30, 2000

Example:

```
configure
    qos
        match-list
            ip-prefix-list "ipv4-base-config" create
                prefix 10.0.0.0/16
                prefix 10.1.0.0/16
            exit
            ip-prefix-list "list1" create
                prefix 172.16.1.0/24
                prefix 172.16.2.0/24
            exit
            ip-prefix-list "list2" create
                prefix 192.168.1.0/24
                prefix 192.168.2.0/24
            exit
            ipv6-prefix-list "ipv6-base-config" create
                prefix 2001:db8:1000::/64
                prefix 2001:db8:2000::/64
            exit
            ipv6-prefix-list "list1" create
                prefix 2001:db8:3000::/64
                prefix 2001:db8:4000::/64
            exit
            ipv6-prefix-list "list2" create
                prefix 2001:db8:5000::/64
                prefix 2001:db8:6000::/64
            exit
        exit
        sap-ingress 10 name "10" create
            queue 1 create
            exit
            queue 11 multipoint create
            exit
            policer 1 create
            exit
            policer 2 create
            exit
            ip-criteria
                type tagged-entries
                entry 10 create
                    match
                        src-ip ip-prefix-list "ipv4-base-config"
                    exit
                    action fc "l2" policer 1
                exit
                entry 20 create
                    match
                        src-ip 10.2.0.0/16
                    exit
                    action fc "af" policer 2
                exit
                entry 30 create
                    match
                        src-ip 10.3.0.0/16
                    exit
                    action fc "af"
                exit
                entry 1000 create
```

```
tag 1
                    match
                        src-ip ip-prefix-list "list1"
                    exit
                    action fc "l1"
                exit
                entry 2000 create
tag 2
                    match
                        src-ip ip-prefix-list "list2"
                    exit
                    action fc "ef"
                exit
            exit
        exit
            ipv6-criteria
                type tagged-entries
                entry 10 create
                    match
                        src-ip ipv6-prefix-list "ipv6-base-config"
                    exit
                    action fc "l2" policer 1
                exit
                entry 20 create
                    match
                        src-ip 10.2.0.0/16
                    exit
                    action fc "af" policer 2
                exit
                entry 30 create
                    match
                        src-ip 10.3.0.0/16
                    exit
                    action fc "af"
                exit
                entry 1000 create
tag 1
                    match
                        src-ip ip-prefix-list "list1"
                    exit
                    action fc "l1"
                exit
                entry 2000 create
tag 2
                    match
                        src-ip ip-prefix-list "list2"
                    exit
                    action fc "ef"
                exit
            exit
        exit
    exit
    service
        vprn 1 name "1" customer 1 create
            interface "int" create
                address 10.10.10.1/24
                sap 1/1/1:1 create
ingress
    sap-ingress "10"
                exit
            exit
        exit
        vprn 2 name "2" customer 1 create
            interface "int" create
```

```
                address 10.20.20.1/24
                sap 1/1/1:2 create
                    ingress
    sap-ingress "10"
                        criteria-overrides
                            ip-criteria
                                activate-entry-tag 1
                            ipv6-criteria
                                activate-entry-tag 1
                        exit
                    exit
                exit
            exit
        exit
        vprn 3 name "3" customer 1 create
            interface "int" create
                address 10/30/30.1/24
                sap 1/1/1:3 create
                    ingress
    sap-ingress "10"
                        criteria-overrides
                            ip-criteria activate-entry-tag 2
                            ipv6-criteria activate-entry-tag 2
                        exit
                    exit
                exit
            exit
        exit
    exit
```

6.3.7 Ingress criteria classification directly to policer

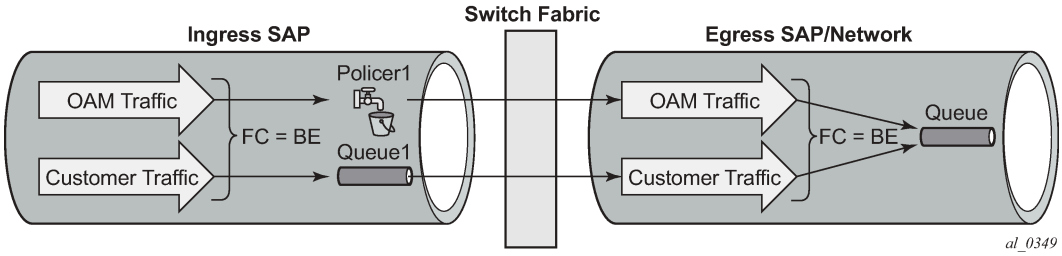
It is possible to classify traffic directly to a policer, independent of the policer/queue assigned to the traffic's forwarding class. This is supported at SAP ingress by configuring a policer in the action statement: ip-criteria, ipv6-criteria, or mac-criteria.

The standard mechanisms are still used to assign a forwarding class to the related traffic, and this forwarding class continues to be used for QoS processing at egress.

The use of explicitly configured broadcast, unknown, or multicast policers is not supported. QPPB processing takes precedence over this feature.

This could be used, for example, when it is required that ingress OAM traffic is not subject to the same QoS control as other customer traffic on a specific SAP. The OAM traffic could be classified based on its source MAC address (for example, with an OUI of 00-xx-yy as shown in Figure 10: Ingress criteria classification directly to policer) and directed to policer 1 while the remainder of the customer's traffic is processed using ingress queue 1.

Figure 10: Ingress criteria classification directly to policer



The configuration would be as follows:

```
sap-ingress 10 create
    queue 1 create
    exit
    queue 11 multipoint create
    exit
    policer 1 create
    exit
    mac-criteria
        entry 10 create
            match
                src-mac 00-xx-yy-00-00-00 ff-ff-ff-00-00-00
            exit
            action policer 1
        exit
    exit
exit
```

6.3.8 Virtual network identifier classification

Virtual Network Identifier (VNI) classification is supported for VXLAN and VXLAN GPE traffic within a SAP ingress QoS policy. This classification is configured in the **ip-criteria** and **ipv6-criteria** contexts with **type vxlan-vni** (changed from the default **type normal**). The matching entry must be created with **match protocol udp** for IPv4 or **match next-header udp** for IPv6 and uses the **vxlan-vni** parameter within the **match** statement to match on a single VNI or a range of VNIs.

The type cannot be changed when **ip-criteria** or **ipv6-criteria** entries are configured. If there are no **ip-criteria** or **ipv6-criteria** entries configured, the type can be changed from **vxlan-vni** to **normal**. The type can only be changed from **normal** to **vxlan-vni** if there are no **ip-criteria** or **ipv6-criteria** entries configured and if the SAP ingress QoS policy has not been applied to any object.

The following is an example where traffic received with a VNI of 1 is sent to policer 1 and VNIs 2 to 10 are sent to policer 2:

```
sap-ingress 10 create
    queue 1 create
    exit
    queue 11 multipoint create
    exit
    policer 1 create
    exit
    policer 2 create
```

```
            exit
            ip-criteria
                type vxlan-vni
                entry 10 create
                    match protocol udp
                        vxlan-vni eq 1
                    exit
                    action policer 1
                exit
                entry 20 create
                    match protocol udp
                        vxlan-vni range 2 10
                    exit
                    action policer 2
                exit
            exit
        exit
```

Ingress VNI classification is applicable to all Ethernet SAPs, except for PW-SAPs, B-VPLS SAPs, and CCAG SAPs, in any applicable service.

The following restrictions also apply:

- Source and destination port matching on a SAP, on which a SAP ingress QoS policy is applied that has **ip-criteria** or **ipv6-criteria** statements with **type vxlan-vni**, is not available for:
  - IPv4 QoS classification for VXLAN or VXLAN GPE traffic
  - IPv6 QoS and filter classification for VXLAN or VXLAN GPE traffic

  If the criteria type is set to **vxlan-vni** and if source or destination port matching entries are configured in an IPv4 or IPv6 SAP ingress QoS policy or in an IPv6 filter policy, any VXLAN or VXLAN GPE ingress traffic do not match these entries on the SAP on which the SAP ingress QoS policy is applied.
- The simultaneous configuration on a SAP of a QoS policy containing an **ip-criteria** entry with **type vxlan-vni** and of a MAC filter with **type vid** is not supported, and the other way around.

  This is only applicable to Epipe and VPLS services.
- If a SAP ingress QoS policy that has **ip-criteria** or **ipv6-criteria** statements with **type vxlan-vni** is applied to a SAP, any **ip-criteria** or **ipv6-criteria** entry match **vxlan-vni** statements do not match:
  - IPv4 packets containing options
  - IPv6 packets containing extension headers
  - ingress 802.1ah PBB frames
  - IPv6 over PPPoE traffic received with more than one VLAN tag
  - non-first fragments of an IPv4 or IPv6 fragmented packet
- The configuration of a SAP ingress QoS policy containing **ip-criteria** or **ipv6-criteria** entry match **vxlan-vni** statements is not supported within an SLA profile.

6.3.9 FC mapping based on EXP bits

Use the **lsp-exp** command to set the sap-ingress qos policy on Ethernet L2 SAPs to perform FC mapping based on EXP bits.

The **lsp-exp** option causes the forwarding class and drop priority of incoming traffic to be determined by the mapping result of the EXP bits in the top label.

The following example displays FC mapping based on EXP bits:

```
*A:Dut-T>config>qos>sap-ingress# info
----------------------------------------------
            queue 1 create
            exit
            queue 2 create
            exit
            queue 3 create
            exit
            queue 11 multipoint create
            exit
            fc "af" create
                queue 2
            exit
            fc "be" create
                queue 1
            exit
            fc "ef" create
                queue 3
            exit
            lsp-exp 0 fc "be" priority low
            lsp-exp 1 fc "af" priority high
```

6.3.10 Storing match criteria entries

Cards store QoS policy match-criteria entries in dedicated memory banks in hardware also referred to as CAM tables:

- IP/MAC ingress
- IP/MAC egress
- IPv6 ingress
- IPv6 egress

6.4 Service egress QoS policy

To create a service egress policy, define the following:

- a new policy ID value. The system does not dynamically assign a value.
- the scope. A QoS policy must be defined as having either an *exclusive* scope for one-time use, or a *template* scope, which enables its use with multiple SAPs.
- a description. The description provides a brief overview of policy features.

6.4.1 Service egress QoS queue

To create a service egress QoS queue, define the following:

- the forwarding class name or names associated with the egress queue. The egress queue for the service traffic is selected based on the forwarding classes that are associated with the queue.
- a new queue ID value. The system does not dynamically assign a value.

- queue parameters. Egress queues support explicit and auto-expedite hardware queue scheduling, and parent virtual scheduler definition.

The following displays an egress QoS policy configuration:

```
A:ALA-7>config>qos# info
----------------------------------------------
...
        sap-egress 105 create
            description "SAP egress policy"
            queue 1 create
                parent "scheduler-tier1"
            exit
            queue 2 create
            exit
            queue 3 expedite create
                parent "test1"
            exit
            fc af create
                queue 1
            exit
            fc ef create
            exit
        exit
...
----------------------------------------------
A:ALA-7>config>qos#
```

6.4.2 Egress percent-rate support

The **percent-rate** command is supported in a SAP egress QoS policy for **pir** and **cir** parameters for both queues and policers. For **pir**, the range is 0.01 to 100.00, and for **cir**, the range is 0.00 to 100.00.

For queues, when the queue rate is **percent-rate**, either a **local-limit** or a **port-limit** can be applied.

When the **local-limit** is used, the **percent-rate** is relative to the queue's parent scheduler rate or the **agg-rate** at egress. If there is no parent scheduler rate or **agg-rate**, or those rates are **max**, the **port-limit** is used. When the **port-limit** is used, the **percent-rate** is relative to the rate of the port (including the egress-rate setting) to which the queue is attached. The **port-limit** is the default.

For policers, the **percent-rate** rate is always relative to the immediate parent root policer/arbiter rate or the FP capacity.

The following shows a SAP egress QoS policy configuration:

```
*B:PE>config>qos>sap-egress# queue 1 percent-rate
- no percent-rate
- percent-rate <pir-percent> [cir <cir-percent>] [port-limit | local-limit]

<pir-percent> : [0.01..100.00]
<cir-percent> : [0.00..100.00]
<port-limit | local-*> : keyword

*B:PE>config>qos>sap-egress# policer 1 percent-rate
- no percent-rate
- percent-rate <pir-percent> [cir <cir-percent>]

<pir-percent> : [0.01..100.00]
<cir-percent> : [0.00..100.00]
```

```
*B:PE>config>qos>sap-egress#
```

6.4.3 Egress queue CBS and MBS as a function of delay

SR OS supports egress queue length configuration (CBS and MBS) as a function of the expected delay. The system automatically translates this configuration into kilobytes based on the administrative rate of the queue parent (for example, the scheduler or aggregate shaper) for queues that have a parent. The actual operational values of CBS and MBS are shown in kilobytes, and not as a function of the delay, in the corresponding **show pools** commands.

The delay value for the specified queue can be configured in the following ways:

- CBS, MBS, and burst limit as a function of queue delay
- CBS and MBS as a function of SAP delay budget

Nokia recommends using the same method of defining CBS, MBS, and burst limit for all queues within the same SAP egress policy. CBS and MBS as a function of delay is not supported on HSQ. Policies that contain a function of delay applied to an HSQ will have default egress queue CBS and MBS value.

6.4.3.1 CBS, MBS, and burst limit as a function of queue delay

SR OS supports CBS, MBS, and burst limit configuration, in microseconds, as a function of the delay in SAP egress policy using the following commands.

- **MD-CLI**

```
configure qos sap-egress queue
                burst-limit-delay-time
                cbs-delay-time
                mbs-delay-time
```

- **classic CLI**

```
configure qos sap-egress queue
                burst-limit delay-time
                cbs delay-time
                mbs delay-time
```

The actual CBS, MBS, and burst limit values are calculated based on the queue parent administrative rate and may differ depending on the scheduling hierarchy. This calculation can be done only if the queue parent has an explicit rate defined (for example, a scheduler rate or aggregate rate). The system uses the default values if the queue parent is a port without an explicit rate defined.

Note: This feature is supported on LAGs in **link** or **port-fair adapt-qos** modes only. It is not supported on ESM.

6.4.3.2 CBS and MBS as a function of SAP delay budget

SR OS supports egress queue CBS and MBS configuration as a function of the total latency budget for the specific SAP using the following procedures.

The actual CBS and MBS values are calculated based on the queue parent administrative rate and may differ depending on the scheduling hierarchy. This calculation can be done only if the queue parent has an explicit rate defined (for example, a scheduler rate or aggregate rate). The system uses the default values if the queue parent is a port without an explicit rate defined.

Note: This feature is supported on LAGs in **link** or **port-fair adapt-qos** modes only. It is not supported on ESM.

- **MD-CLI**
  1. Configure the SAP latency budget, in microseconds, using one of the following commands:

     ```
     configure service epipe sap egress qos latency-budget
     configure service ies interface sap egress qos latency-budget
     configure service vpls sap egress qos latency-budget
     configure service vprn interface sap egress qos latency-budget
     ```

  2. In the SAP egress policy, configure the percentage of the SAP latency budget for an individual queue:

     ```
     configure qos sap-egress queue cbs-delay-percent
     configure qos sap-egress queue mbs-delay-percent
     ```

- **classic CLI**
  1. Configure the SAP latency budget, in microseconds, using one of the following commands:

     ```
     configure service epipe sap egress latency-budget
     configure service ies interface sap egress latency-budget
     configure service vpls sap egress latency-budget
     configure service vprn interface sap egress latency-budget
     ```

  2. In the SAP egress policy, configure the percentage of the SAP latency budget for an individual queue:

     ```
     configure qos sap-egress queue cbs delay-percent
     configure qos sap-egress queue mbs delay-percent
     ```

6.4.4 Dynamic MBS for egress queue group queues

Dynamic MBS is used to constrain the maximum delay experienced by the traffic forwarded through an egress queue group queue when the operational PIR of the queue is modified as part of the HQoS algorithm.

The approximate maximum delay of traffic through a queue because of the length of the queue, when the queue is not using HQoS, is relative to its administrative PIR and can be approximated as (MBS[kB] × 8) / PIR[kb/s]) in seconds. A queue's PIR is set to **max**, and its administrative PIR is set to the rate of the port to which the queue is attached.

When using HQoS, the PIR is modified by the HQoS algorithm to give an operational PIR that is equal to or lower than the administrative PIR. As the operational PIR changes, the delay through the queue can also change if the length of the queue is fixed. Reducing the operational PIR could increase the delay, while increasing the operational PIR could reduce the delay. Enabling dynamic MBS on a queue allows the system to change the administrative MBS of the queue in a ratio of operational PIR to administrative PIR,

giving an operational MBS, which aims to maintain the maximum queue delay. A queue's drop tails and WRED slope parameters are defined as percentages of the MBS and are, therefore, adjusted accordingly.

When any of the queue parameters are reduced, packets that are already in the queue are not affected and are forwarded. Reducing these parameters constrains the latency for newly arriving packets, but those packets already in the queue before the new parameter values were set are forwarded with the delay associated with the actual queue depth when the packet was enqueued (based on the previous parameter values).

The configured CBS is used as a minimum operational MBS. The maximum MBS is capped by the maximum administrative MBS (1 GB).

If the operational MBS changes such that its value is similar or equal to the configured CBS, the system increases the CBS to ensure that buffers can be requested from the correct portion of the buffer pool (shared or reserved). This operation is automatic, and the CBS reverts to its configured value if the MBS is increased sufficiently. The automatic increase in the CBS could, however, cause the **resv-cbs** red or amber alarms to be raised if the increase in the related queues' CBS results in the total CBS assigned (but not necessarily used) matching or exceeding the **resv-cbs** red and amber thresholds.

If a LAG is used together with **pool-per-queue**, the related hardware queues exist in their own pool in the egress WRED megapool on a specific FP and the operational MBS is used to size the shared part of the pool with the sum of the CBS defining the reserved part of the pool.

Dynamic MBS is supported for both native FP and **pool-per-queue** queues within an egress queue group template, which can be applied to access or network Ethernet ports and used for egress network interface traffic, egress SAP traffic, and subscriber egress policed traffic.

The configuration of dynamic MBS and queue depth monitoring are mutually exclusive.

Dynamic MBS is configured as follows:

```
configure
    qos
       queue-group-templates
          egress
              queue-group queue-group-name create
                  queue queue-id
```

The operational MBS can be shown using the **show pools** and show **qos scheduler-hierarchy** commands.

The following example shows the use of dynamic MBS. A queue group template is applied to port 5/1/1 configured with multiple queues using HQoS, one of which has the following parameters:

```
queue-group "qg1" create
  queue 1 best-effort create
       parent "s1"
       rate 50000
       mbs 1000 kilobytes
       dynamic-mbs
   exit
```

Without any traffic in the other queues constraining the operational PIR on this queue, the MBS used is the administrative MBS.

```
*B:PE# show pools access-egress 5/1/1 queue-group "qg1" instance 1
...
===============================================================================
Queue : accQGrp->qg1:1(5/1/1)->1
```

```
===============================================================================
FC Map               : not-applicable
Dest Tap             : not-applicable   Dest FP          : not-applicable
Admin PIR            : 50000            Oper PIR         : 50000
Admin CIR            : 0                Oper CIR         : 0
Admin MBS            : 1008 KB          Oper MBS         : 1008 KB
High-Plus Drop Tail  : 1008 KB          High Drop Tail   : 1008 KB
Low Drop Tail        : 888 KB           Exceed Drop Tail : 792 KB
CBS                  : 0 KB             Depth            : 0
Slope                : not-applicable
```

If traffic is sent to the other queues in the queue group such that the operational PIR of queue 1 is reduced to 25 Mb/s, the show output changes to:

```
*B:PE# show pools 5/1/1 access-egress queue-group "qg1" instance 1
...
===============================================================================
Queue : accQGrp->qg1:1(5/1/1)->1
===============================================================================
FC Map               : not-applicable
Dest Tap             : not-applicable   Dest FP          : not-applicable
Admin PIR            : 50000            Oper PIR         : 25000
Admin CIR            : 0                Oper CIR         : 0
Admin MBS            : 1008 KB          Oper MBS         : 504 KB
High-Plus Drop Tail  : 1008 KB          High Drop Tail   : 1008 KB
Low Drop Tail        : 948 KB           Exceed Drop Tail : 900 KB
CBS                  : 0 KB             Depth            : 0
Slope                : not-applicable
```

The output shows that the operational MBS is now 50% of the administrative MBS and that the queue's drop tails have changed accordingly.

6.4.5 Egress SAP FC and FP overrides

An access egress packet's forwarding class can be changed to redirect the packet to an alternate queue than the ingress forwarding class determination would have used. An access egress packet's profile can also be changed to modifying the congestion behavior within the egress queue. In both cases, egress marking decisions are based on the new forwarding class and profile as opposed to the egress forwarding class or profile. The exception is when ingress remarking is configured. An ingress remark decision is not affected by egress forwarding class or egress profile overrides.

The SAP egress QoS policy allows reclassification rules that are used to override the ingress forwarding class and profile of packets that egress a SAP where the QoS policy is applied.

Dot1p, IP precedence, DSCP, and IP quintuple entries can be defined, each with an explicit forwarding class or profile override parameters. The reclassification logic for each entry follows the same basic hierarchical behavior as the classification rules within the SAP ingress QoS policy. Dot1p, IP precedence, and DSCP have the lowest match priority while the IP criteria (quintuple) entries have the highest.

When an optional parameter (such as **profile**) for Dot1p, IP precedence, or DSCP entries is not specified, the value from the lower priority IP quintuple match for that parameter is preserved. If the IP precedence values overlap with DSCP values in that they match the same IP header TOS field, the DSCP entry parameters override or remove the IP precedence parameters. When none of the matched entries override a parameter, the ingress classification is preserved.

6.4.6 Egress IP match criteria

The following displays a SAP egress QoS policy IP criteria configuration:

```
*A:PE>config>qos# info
----------------------------------------------
#------------------------------------------------
echo "QoS Policy Configuration"
#------------------------------------------------
        sap-egress 10 name "10" create
            queue 1 create
            exit
            ip-criteria
                entry 10 create
                    match
                        src-ip 192.168.0.0/16
                    exit
                    action fc "af"
                exit
                entry 20 create
                    match
                        dst-ip 10.0.0.0/8
                    exit
                    action fc "be"
                exit
            exit
        exit
----------------------------------------------
*A:PE>config>qos#
```

6.4.7 Egress IPv6 match criteria

The following displays a SAP egress QoS policy IPv6 criteria configuration:

```
*A:PE>config>qos# info
----------------------------------------------
#------------------------------------------------
echo "QoS Policy Configuration"
#------------------------------------------------
        sap-egress 10 name "10" create
            queue 1 create
            exit
            ipv6-criteria
                entry 10 create
                    match
                        dst-ip 2001:db8:1000::/64
                    exit
                exit
            exit
        exit
----------------------------------------------
*A:PE>config>qos#
```

6.4.8 Storing match criteria entries

Cards store QoS policy match-criteria entries in dedicated memory banks in hardware, also referred to as CAM tables:

- IP/MAC ingress
- IP/MAC egress
- IPv6 ingress
- IPv6 egress

6.4.9 Dot1p egress remarking

Dot1p remarking can be performed on egress for all services and with respect to the profile of the packet and the VLAN tag.

The following commands can be used to remark the dot1p values at a SAP egress:

```
configure qos sap-egress policy-id create
    use-policer-result-marking-dot1p-inner
    fc fc-name create
    dot1p {dot1p-value | in-profile dot1p-value out-
     profile dot1p-value [exceed-profile dot1p-value]}
    dot1p-inner {dot1p-value | in-profile dot1p-value
     out-profile dot1p-value | exceed-profile dot1p-value}
    dot1p-outer {dot1p-value | in-profile dot1p-value
     out-profile dot1p-value [exceed-profile dot1p-
     value]}
    exit
```

All inplus-profile traffic is marked with the same value as in-profile traffic.

The precedence of these commands is summarized as follows, from highest to lowest precedence:

- dot1p-outer used for outer tag markings
- dot1p-inner used for inner tag markings
- existing dot1p used for marking both tags
- markings taken from packet received at ingress
- dot1p-inner marking is dependent on policer result if **use-policer-result-marking-dot1p-inner** is enabled

The configuration of QinQ-mark-top-only under the SAP egress takes precedence over the use of the dot1p-inner in the policy, that is, the inner VLAN tag is not remarked when QinQ-mark-top-only is configured. The marking used for the inner VLAN tag is based on the current default, which is governed by the marking of the packet received at the ingress to the system. If QinQ-mark-top-only is omitted, both the inner and outer VLAN tags are remarked.

The egress remarking occurs after any egress classification.

6.4.9.1 DEI egress remarking

It is often desirable to meter traffic from different users to ensure fairness or to meet bandwidth guarantees. Dropping all traffic in excess of a committed rate is likely to result in severe under-utilization of the networks, because most traffic sources are bursty in nature. It is burdensome to meter traffic at all points in the network where bandwidth contention occurs. One solution is to mark those frames in excess of the committed rate as drop eligible on admission to the network.

Previously, the discard eligibility was determined using existing QoS fields; for example, the three MPLS EXP and Ethernet dot1p bits. Using specific combinations of these bits to indicate both forwarding class (priority) and discard eligibility meant decreasing the number of forwarding classes that can be differentiated in the network.

IEEE 802.1ad-2005 and IEEE 802.1ah standards allow drop eligibility to be conveyed separately from priority, preserving all the eight forwarding classes (priorities) that could be indicated using the three 802.1p bits. All the previously introduced traffic types are marked as drop eligible. Customers can continue to use the dot1p markings with the enhancement of changing the dot1p value used, in access, based on the profile information.

The following commands can be used to remark the DE values at a SAP egress:

```
sap-egress policy-id create
     fc fc-name create
           de-mark [force de-value]
           de-mark-inner [force de-value]
           de-mark-outer [force de-value]
     exit
exit
```

By default, the DE bit is set to 0 for inplus-profile and in-profile traffic and 1 for out-of-profile and exceed-profile traffic, unless explicitly forced.

The precedence of these commands is summarized from highest to lowest precedence, as follows:

- de-mark-outer used for outer tag markings
- de-mark-inner used for inner tag markings
- existing de-mark used for marking both tags
- markings taken from packet received at ingress

The configuration of QinQ-mark-top-only under the SAP egress takes precedence over the use of the de-mark-inner in the policy, that is, the inner VLAN tag is not remarked when QinQ-mark-top-only is configured. The marking used for the inner VLAN tag is based on the current default, which is governed by the marking of the packet received at the ingress to the system. If QinQ-mark-top-only is omitted, both the inner and outer VLAN tags are remarked.

Remarking the inner DE bit is not supported based on the profile result of egress policing.

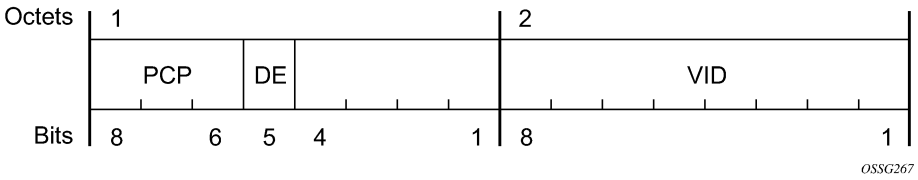
The egress remarking occurs after any egress classification.

6.4.9.1.1 DEI in IEEE 802.1ad

The *IEEE 802.1ad-2005 standard* allows drop eligibility to be conveyed separately from priority in-service VLAN TAGs (S-tags). The S-tag has a new format where the priority and discard eligibility parameters are

conveyed in the 3-bit priority code point (PCP) field and in the DE bit, respectively (see Figure 11: DE Bit in the 802.1ad S-tag).

Figure 11: DE Bit in the 802.1ad S-tag



The introduction of the DE bit allows the S-tag to convey eight forwarding classes/distinct priorities, each with a drop eligible indication.

When DE bit is set to 0 (DE = FALSE), the related packet is not discard eligible. This is the case for the packets that are within the CIR limits and must be given priority in case of congestion. If the DEI is not used or backwards compliance is required, the DE bit should be set to zero on transmission and ignored on reception.

When the DE bit is set to 1 (DE = TRUE), the related packet is discard eligible. This is the case for the packets that are sent above the CIR limit. In case of congestion, these packets are the first ones to be dropped.

6.4.9.1.2 DEI in IEEE 802.1ah

IEEE 802.1ah (PBB) standard provides a dedicated bit for DE indication in both the backbone VLAN ID (BVID) and the ITAG.

The BVID is a regular 802.1ad S-tag. Its DE bit may be used to convey the related tunnel QoS throughout an Ethernet backbone.

The ITAG header offers also an I-DEI bit that may be used to indicate the service drop eligibility associated with this frame.

These bits must follow the same rules as described in DEI in IEEE 802.1ad.

6.4.9.1.3 IEEE 802.1ad use case

Figure 12: DE aware 802.1ad access network shows an example of a topology where the new DE feature may be used: a DE aware, 802.1ad access network connected via a regular SAP to a router PE.

In the following example, PE1 can ensure coherent processing of the DE indication between the 802.1ad and the MPLS networks. For example, for packets ingressing the SAP connected to 802.1ad access, read the DE indication and perform classification, color aware metering/policing, marking of the related backbone QoS fields, and selective discarding of the frames throughout the queueing system, based on their discard eligibility. In addition, packets egressing the SAP toward the 802.1ad access provide correct DE indication by marking the new DE bit in the S-tag.

Figure 12: DE aware 802.1ad access network

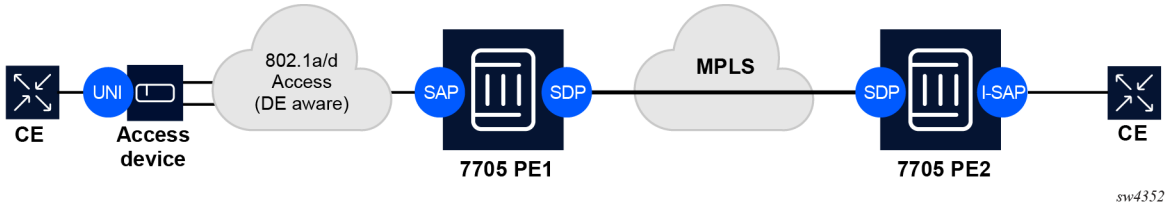
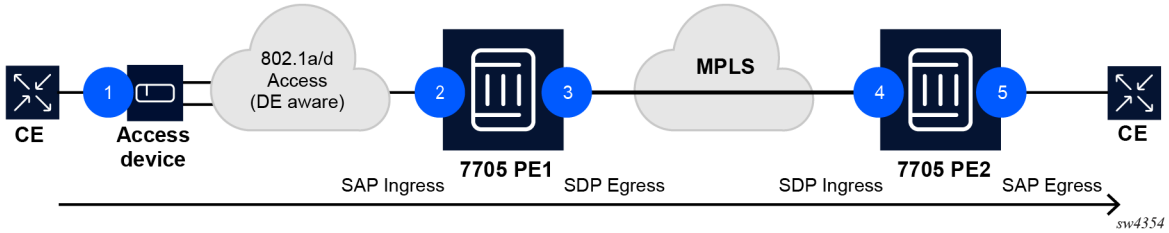


Figure 13: DEI processing ingress into the PE1 SAP shows an example of the QoS processing of the DEI processing steps for the IEEE 802.1ad use case for both ingress and egress directions (from a PE1 SAP perspective).

Figure 13: DEI processing ingress into the PE1 SAP



The following steps related to DEI are involved in the QoS processing as the packet moves from left to right:

1. The QinQ access device sets the DE bit from the S-tag based on the QoS classification or on the results of the metering/policing for the corresponding customer UNI.
2. The SAP on PE1 may use the DE bit from the customer S-tag to classify the frames as in/out-of-profile. Color aware policing/metering can generate additional out-of-profile packets as the result of packet flow surpassing the CIR.
3. When the packet leaves PE1 via SDP, the DE indication must be copied onto the appropriate tunnel QoS fields (outer VLAN ID or EXP bits, or both) using the internal PHB (per hop behavior) of the packet (for example, the FC and Profile).
4. As the packet arrives at PE2, on ingress into the related SDP, the DE indication is used to classify the packets into an internal PHB.
5. On egress from the PE2 SAP, the internal PHB may be used to perform marking of the DE bit.

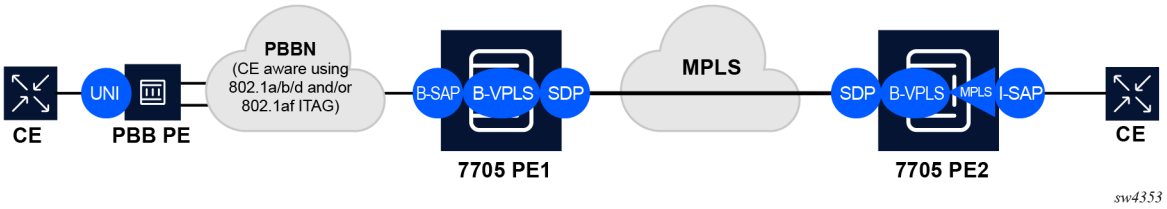
A combination of two access networks can be possible. If PBB encapsulation is used, the configuration used for DE in SAP and SDP policies applies to both BVID and ITAG DE bits. When both fields are used, the BVID takes precedence.

6.4.9.1.4 IEEE 802.1ah use case

Figure 14: DE aware PBB topology shows an example of a PBB topology where the DE feature can be used. The processing requirements highlighted in the 802.1ad use case apply to the 802.1ah BVID, format and etype, these being identical with the 802.1ad S-tag. In addition, the DE bit from the 802.1ah ITAG header may need to be processed following the same rules as for the related field in the BVID/S-tag; for

example, the DE bit from the BVID header represents the QoS associated with the "Ethernet Tunnel" while the DE bit from the ITAG represents the service QoS.

Figure 14: DE aware PBB topology



In this example, the BVID is not used for a part of the network, leaving the I-DEI bit from the ITAG as the only option for a dedicated DE field. If both are included, the QoS information from the BVID is to be used.

6.4.10 DSCP and IP precedence egress remarking

DSCP and IP precedence remarking can be performed on egress for layer 3 services only with respect to the profile of the packet.

Use the following CLI syntax to remark the DSCP and IP precedence values at a SAP egress:

```
configure qos sap-egress policy-id create
  fc fc-name create
    dscp {dscp dscp-name | in-profile dscp-name out-
      profile dscp-name [exceed-profile dscp-name]}
    prec {ip-prec-value | in-profile ip-prec-value out-
      profile ip-prec-value} [exceed-profile ip-prec-
      value]}
  exit
exit
```

All inplus-profile traffic is marked with the same value as in-profile traffic.

Remarking the DSCP and IP precedence based on the profile result of egress policing must be enabled under the related policer configuration, as follows:

```
sap-egress policy-id create
  policer policer-id create
    enable-dscp-prec-remarking
  exit
exit
```

6.5 Service management tasks

This section discusses service ingress and egress service management tasks.

6.5.1 Applying service ingress and egress policies

Apply SAP ingress and egress policies to the following service SAPs:

- Epipe
- IES
- VPLS
- VPRN

See the "Subscriber Services Overview" section of the *7705 SAR Gen 2 Services Overview Guide* for information about configuring service parameters on the 7705 SAR Gen 2.

6.5.1.1 Epipe

The following output displays an Epipe service configuration with SAP ingress policy 100 and SAP egress 105 applied to the SAP.

```
A:ALA-7>config>service# info
----------------------------------------------
        epipe 6 customer 6 vpn 6 create
            description "Distributed Epipe service to west coast"
            sap 1/1/10:010 create
                ingress
                    qos 100
                exit
                egress
                    qos 105
                exit
            exit
            spoke-sdp 2:6 create
                ingress
                    vc-label 6298
                exit
                egress
                    vc-label 6300
                exit
            exit
            no shutdown
        exit
----------------------------------------------
A:ALA-7>config>service#
```

6.5.1.2 IES

The following output displays an IES service configuration with SAP ingress policy 100 and SAP egress 105 applied to the SAP.

```
A:ALA-7>config>service# info
----------------------------------------------
        ies 88 customer 8 vpn 88 create
            interface "Sector A" create
                sap 1/1/1.2.2 create
                    ingress
                        qos 100
                    exit
                    egress
                        qos 105
                    exit
                exit
            exit
```

```
            no shutdown
        exit
-------------------------------------------
```

6.5.1.3 VPLS

The following output displays a VPLS service configuration with SAP ingress policy 100. The SAP egress policy 1 is applied to the SAP by default.

```
A:ALA-7>config>service# info
-------------------------------------------
        vpls 700 customer 7 vpn 700 create
            description "test"
            stp
                shutdown
            exit
            sap 1/1/9:010 create
                ingress
                    qos 100
                exit
            exit
            spoke-sdp 2:222 create
            exit
            mesh-sdp 2:700 create
            exit
            no shutdown
        exit
-------------------------------------------
A:ALA-7>config>service#
```

6.5.1.4 VPRN

The following output displays a VPRN service configuration for the 7705 SAR Gen 2.

```
A:ALA-7>config>service# info
-------------------------------------------
...
        vprn 1 customer 1 create
            ecmp 8
            autonomous-system 10000
            route-distinguisher 10001:1
            auto-bind-tunnel
                resolution-filter
                resolution-filter ldp
            vrf-target target:10001:1
            interface "to-ce1" create
                address 10.1.0.1/24
                sap 1/1/10:1 create
                    ingress
                        qos 100
                    exit
                    egress
                        qos 105
                    exit
                exit
            exit
            no shutdown
        exit
```

```
...
-------------------------------------------
A:ALA-7>config>service#
```

6.5.2 Editing QoS policies

QoS existing policies and entries can be edited. The changes are applied immediately to all services where this policy is applied. To prevent configuration errors, copy the policy to a work area, make the edits, then write over the original policy.

6.5.3 Deleting QoS policies

Every service SAP is associated, by default, with the appropriate egress or ingress policy (policy ID 1). The default policy can be replaced with a customer-configured policy but cannot entirely remove the policy from the SAP configuration. When a non-default service egress or ingress policy is removed, the association reverts to the default policy ID 1.

A QoS policy cannot be deleted until it is removed from all SAPs where it is applied.

```
A:ALA-7>config>qos# no sap-ingress 100
MINOR: CLI SAP ingress policy "100" cannot be removed because it is in use.
A:ALA-7>config>qos#
```

6.5.4 Removing a policy from the QoS configuration

Use the following syntax to remove a policy from the QoS configuration:

config>qos# no sap-ingress *policy-id*

Example:

```
config>qos# no sap-ingress 100
config>qos# no sap-egress 1010
```

7 Queue sharing and redirection

7.1 Queue sharing and redirection

Queue groups are objects created on the access or network Ethernet port or ingress forwarding plane of an IOM/IMM/XMA that allow SAP or IP interface forwarding classes to be redirected from the normal type of queue mapping to a shared queue. Queue groups may contain queues, policers, or a combination of the two depending on the type of queue group. The following types of queue groups are supported:

- Access ingress supports a single queue group instance per ingress port, or multiple queue groups created at the ingress forwarding plane level of the IOM/IMM/XMA. Access ingress port queue groups may only contain queues, whereas access ingress forwarding plane queue groups may only contain policers.
- Access egress supports the creation of multiple queue groups per egress port. These queue groups may only contain queues.
- Network ingress supports the creation of multiple queue groups at the ingress forwarding plane level of the IOM/IMM/XMA. These queue groups may only contain policers.
- Network egress supports the creation of multiple queue groups per egress port. These queue groups may contain queues only, or queues and policers.

7.1.1 Supported platforms

Queue sharing and redirection is supported on the 7705 SAR Gen 2, as follows:

- access SAP port and network port queue groups
- ingress access and network forwarding plane queue groups

Queue sharing and redirection are also supported in conjunction with the use of a C-XMA, XMA, Ethernet MDA, Ethernet CMA, and IOM-4-HS.

7.2 Queue group applications

7.2.1 Access SAP queue group applications

Normally, each SAP (Service Access Point) has dedicated ingress and egress queues that are only used by that particular SAP. The SAP queues are created based on the queue definitions within the SAP ingress and SAP egress QoS policy applied to the SAP. Each packet that enters or egresses the SAP has an associated forwarding class. The QoS policy is used to map the forwarding class to one of the SAP's local queue IDs. This per-SAP queuing has advantages over a shared queuing model in that it allows each SAP to have a unique scheduling context per queue. During congestion, SAPs operating within their conforming bandwidth experience little impact because they do not need to compete for queue buffer space with misbehaving or heavily loaded SAPs.

The situation is different for a shared or port-queuing model that is based on policing color packets that conform or exceed a static rate before the single queue and that use WRED or drop tail functions to essentially reserve room for the conforming packets.

In this model, there is no way for the conforming packets to go to the head of the line in the view of the port scheduler. Another advantage of per-SAP queuing is the ability for the SAP queues to perform shaping to control burst sizes and forwarding rates based on the SAPs defined SLA. This is especially beneficial when a provider is enforcing a sub-line rate bandwidth limit and the customer does not have the ability to shape at the CE.

However, there are cases where per-SAP queuing is not preferred. Per-SAP queuing requires a more complex provisioning model to properly configure the SAPs ingress and egress SLAs. This requires service awareness at some points in the network where an aggregation function is being performed. In this case, a shared queuing or per-port queuing model is sufficient. Creating ingress and egress access queue groups and mapping the SAPs forwarding classes to the queues within the queue group provides this capability.

A further use case is where a set of ingress SAPs, which may represent a subset of the total number of ingress SAPs, is to be shaped or policed on an aggregate per-forwarding class basis when those SAPs are spread across a LAG on multiple ingress ports, and where color aware treatment is required so that explicitly in-profile traffic is honored up to CIR, but above which it is marked as out-of-profile.

The preceding scenarios can be supported with access queue groups. A single ingress queue group is supported per access port, while multiple ingress queue group instances are supported per IOM/IMM/XMA forwarding plane. To provide more flexibility on the egress side of the access port, multiple egress access queue group queue-group instances are supported per egress access port.

Because queue redirection is defined per forwarding class, it is possible to redirect some forwarding classes to a queue group while having others on the SAP use the SAP local queues. This is helpful when shared queuing is only wanted for a few applications such as VoIP or VoD while other applications still require queuing at the SAP level.

7.2.1.1 Ingress per SAP statistics with ingress queue groups

A statistic displaying the number of valid ingress packets received on a SAP, or subscribers on that SAP, is shown in the *sap-stats* output. This is available for SAPs in all services. This is particularly useful to display SAP level traffic statistics when forwarding classes in a SAP ingress policy have been redirected to an ingress queue group.

In the following example, traffic is received on an ingress FP policer with a packet-byte-offset of subtract 10. It can be seen that the ingress queueing stats and offered forwarding engine stats are all zero as the traffic is using the FP ingress policer. The Received Valid statistic is non-zero and matches that seen on the ingress FP queue group, with the difference being that the packet-byte-offset is applied to the queue group policer octets but not the Received Valid octets.

The value in the Received Valid field may not instantaneously match the sum of the offered stats (even in the case where all traffic is using the SAP queues) when traffic is being forwarded; however, when the traffic has stopped, the Received Valid equals the sum of the offered stats.

```
*A:PE# show service id 1 sap 1/1/9:1 sap-stats

===============================================================================
Service Access Points(SAP)
===============================================================================
Service Id         : 1
SAP                : 1/1/9:1                 Encap            : q-tag
Description        : (Not Specified)
```

```
Admin State        : Up                      Oper State        : Up
Flags              : None
Multi Svc Site     : None
Last Status Change : 08/31/2018 11:09:25
Last Mgmt Change   : 08/31/2018 11:09:25
-------------------------------------------------------------------------------
Sap Aggregate Stats
-------------------------------------------------------------------------------
                        Packets                 Octets
Ingress
Aggregate Offered     : 0                       0
Aggregate Forwarded   : 0                       0
Aggregate Dropped     : 0                       0

Egress
Aggregate Forwarded   : 0                       0
Aggregate Dropped     : 0                       0
-------------------------------------------------------------------------------
Sap Statistics
-------------------------------------------------------------------------------
Last Cleared Time     : 08/31/2018 11:12:17

                        Packets                 Octets
CPM Ingress           : 0                       0

Forwarding Engine Stats
Dropped               : 0                       0
Received Valid        : 5                       530
Off. HiPrio           : 0                       0
Off. LowPrio          : 0                       0
Off. Uncolor          : 0                       0
Off. Managed          : 0                       0

Queueing Stats(Ingress QoS Policy 10)
Dro. HiPrio           : 0                       0
Dro. LowPrio          : 0                       0
For. InProf           : 0                       0
For. OutProf          : 0                       0

Queueing Stats(Egress QoS Policy 1)
Dro. In/InplusProf    : 0                       0
Dro. Out/ExcProf      : 0                       0
For. In/InplusProf    : 0                       0
For. Out/ExcProf      : 0                       0
===============================================================================
*A:PE#
*A:PE# show card 1 fp 1 ingress queue-group "qg1" instance 1 mode access statistics

===============================================================================
Card:1  Acc.QGrp: qg1  Instance: 1
===============================================================================
Group Name    : qg1
Description   : (Not Specified)
Pol Ctl Pol   : None                    Acct Pol      : None
Collect Stats : disabled
-------------------------------------------------------------------------------
Statistics
-------------------------------------------------------------------------------
                        Packets                 Octets

Ing. Policer:  1  Grp: qg1
(Stats mode: minimal)
Off. All              : 5                       530
Dro. All              : 0                       0
```

```
For. All              : 5                       530
===============================================================================
*A:PE#
```

7.2.2 Network port queue groups for IP interfaces

Queue groups may be created on egress network ports to provide network IP interface queue redirection. A single set of egress port-based forwarding class queues are available by default and all IP interfaces on the port share the queues. Creating a network queue group allows one or more IP interfaces to selectively redirect forwarding classes to the group to override the default behavior. Using network egress queue groups, it is possible to provide dedicated queues for each IP interface.

Non-IPv4/non-IPv6/non-MPLS packets remain on the regular network port queues. Therefore, when using an egress port-scheduler, it is important to parent the related regular network port queues to appropriate port-scheduler priority levels to ensure the needed operation under port congestion. This is particularly important for protocol traffic such as LACP, EFM-OAM, ETH-CFM, ARP, and IS-IS, which by default use the FC NC regular network port queue.

7.2.3 Pseudowire shaping for Layer 2 and Layer 3 services

This feature allows the user to perform ingress and egress datapath shaping of packets forwarded within a spoke SDP (PW). It applies to a VLL service, a VPLS/B-VPLS service, and an IES/VPRN spoke-interface.

7.2.4 Ingress pseudowire shaping

About this task

The ingress PW rate-limiting feature uses a policer in the queue-group provisioning model. This model allows the mapping of one or more PWs to the same instance of policers that are defined in a queue-group template.

Operationally, the provisioning model in the case of the ingress PW shaping feature consists of the following steps:

Procedure

Step 1. Create an ingress queue-group template and configure policers for each FC that needs to be redirected and optionally, for each traffic type (unicast, broadcast, unknown, or multicast).

Step 2. Apply the queue-group template to the network ingress forwarding plane where there exists a network IP interface that the PW packets can be received on. This creates one instance of the template on the ingress of the FP. One or more instances of the same template can be created.

Step 3. Configure FC-to-policer mappings together with the policer redirect to a queue-group in the ingress context of a network QoS policy. No queue-group name is specified in this step, which means the same network QoS policy can redirect different PWs to different queue-group templates.

Step 4. Apply this network QoS policy to the ingress context of a spoke SDP inside a service, or to the ingress context of a PW template and specify the redirect queue-group name.

7.2.5 Egress pseudowire shaping

About this task

One or more spoke SDPs can have their FCs redirected to use policers in the same policer queue-group instance.

The egress PW shaping provisioning model allows the mapping of one or more PWs to the same instance of queues, or policers and queues, that are defined in the queue-group template.

Operationally, the provisioning model consists of the following steps:

Procedure

Step 1. Create an egress queue-group template and configure queues only, or policers and queues, for each FC that needs to be redirected.

Step 2. Apply the queue-group template to the network egress context of all ports where there exists a network IP interface that the PW packets can be forwarded on. This creates one instance of the template on the egress of the port. One or more instances of the same template can be created.

Step 3. Configure FC-to-policer or FC-to-queue mappings together with the redirect to a queue-group in the egress context of a network QoS policy. No queue-group name is specified in this step, which means the same network QoS policy can redirect different PWs to different queue-group templates.

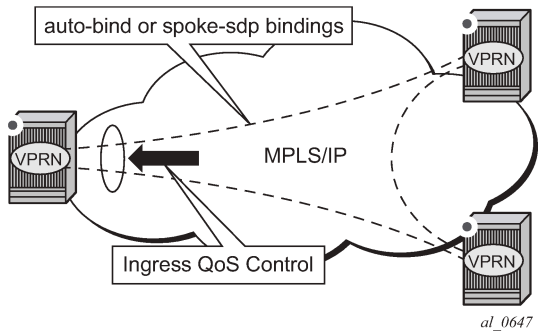
Step 4. Apply this network QoS policy to the egress context of a spoke SDP inside a service, or to the egress context of a PW template and specify the redirect queue-group name.

One or more spoke SDPs can have their FCs redirected to use queues only, or queues and policers, in the same queue-group instance.

7.2.6 QoS on ingress bindings

Traffic is tunneled between VPRN service instances on different PEs over service tunnels bound to MPLS LSPs or GRE tunnels. The binding of the service tunnels to the underlying transport is achieved either automatically (using the **auto-bind-tunnel** command) or statically (using the **spoke-sdp** command; not on the VPRN IP interface). QoS control can be applied to the service tunnels for traffic ingressing into a VPRN service; see Figure 15: Ingress QoS control on VPRN bindings.

Figure 15: Ingress QoS control on VPRN bindings



An ingress queue group must be configured and applied to the ingress network FP where the traffic is received for the VPRN. All traffic received on that FP for any binding in the VPRN (either automatically or statically configured) that is redirected to a policer in the FP queue group (using **fp-redirect-group** in the network QoS policy) is controlled by that policer. As a result, the traffic from all such bindings is treated as a single entity (per forwarding class) with regard to ingress QoS control. Any **fp-redirect-group**, **mcast-policer**, **broadcast-policer**, or **unknown-policer** commands in the network QoS policy are ignored for this traffic (IP multicast traffic would use the ingress network queues or queue group related to the network interface).

Ingress classification is based on the configuration of the ingress section of the specified network QoS policy, noting that the dot1p and exp classification is based on the outer Ethernet header and MPLS label whereas the DSCP applies to the outer IP header if the tunnel encapsulation is GRE, or the DSCP in the first IP header in the payload if **ler-use-dscp** is enabled in the ingress section of the referenced network QoS policy.

Ingress bandwidth control does not consider the outer Ethernet header, the MPLS labels/control word or GRE headers, or the FCS of the incoming frame.

The following command configures the association of the network QoS policy and the FP queue group and instance within the network ingress of a VPRN:

```
configure
    vprn
        network
            ingress
                qos <network-policy-id> fp-redirect-group <queue-group-name>
                                        instance <instance-id>
```

When this command is configured, it overrides the QoS applied to the related network interfaces for unicast traffic arriving on bindings in that VPRN. The IP and IPv6 criteria statements are not supported in the applied network QoS policy.

This is supported for all available transport tunnel types and is independent of the label mode (**vrf** or **next-hop**) used within the VPRN. It is also supported for Carrier-Supporting-Carrier VPRNs.

7.3 Queue group templates

Before a queue group with a specific name may be created on a port or an IOM/IMM/XMA ingress forwarding plane, a queue group template with the same name must first be created. The template is used to define each queue, scheduling attributes, and its default parameters. When a queue or policer is defined in a queue group template, that queue exists in every instance of a port or forwarding plane queue group with that template name. The default queue or policer parameters (such as rate or MBS values) may be overridden with a specific value in each queue group. This works in a similar manner to SAP ingress or SAP egress QoS policies.

7.4 Port queue groups

When an ingress or egress queue group template is defined, a port-based queue group with the same name may be created. Port queue groups are named objects that act as a container for a group of queues. The queues are created based on the defined queue IDs within the associated queue group template. Port

queue groups must be created individually on the ingress and egress sides of the port, but multiple port queue groups of the same template name may be created on egress ports if they have a different instance identifier. These are termed 'queue group instances'. Each instance of a named queue group created on a port is an independent set of queues structured as per the queue group template. Port queue groups are only supported on Ethernet ports and may be created on ports within a LAG.

Additional parameters can be configured under port queue groups, for example, an accounting policy, queue overrides, a scheduler policy, and scheduler overrides.

7.4.1 Percent-rate support

The **percent-rate** command is supported in an egress queue group template for **pir** and **cir** parameters. For **pir**, the range is 0.01 to 100.00, and for **cir**, the range is 0.00 to 100.00.

When the queue rate is configured with **percent-rate**, a **port-limit** is applied, specifically, the **percent-rate** is relative to the rate of the port to which the queue is attached.

```
*A:PE>config>qos>qgrps>egr>qgrp>queue# percent-rate
  - no percent-rate
  - percent-rate <pir-percent> [cir <cir-percent>]

 <pir-percent>        : [0.01..100.00]
 <cir-percent>        : [0.00..100.00]
```

7.5 Forwarding plane queue groups

Ingress forwarding plane queue groups allow groups of SAPs on one or more ports, or on a LAG on the IOM, IMM, or XMA, to be bundled together from a QoS enforcement perspective with an aggregate rate limit to be enforced across all SAPs of a bundle. Multiple queue groups are supported per IOM/IMM/XMA or port on access ingress. These are implemented at the forwarding plane level on the ingress IOM so that SAPs residing on different ingress ports or SAPs on a LAG spread across ports on a specific IOM can be redirected to the same queue group.

When an ingress queue group template is defined, a forwarding plane queue group with the same name may be created on an ingress forwarding plane of an IOM, IMM, or XMA. Forwarding plane queue groups are named objects that act as a container for a group of policers. Queues are not supported in forwarding plane queue groups. Only hierarchical policers are supported in the forwarding plane queue group, instead of queues. These policers may be configured to use profile-aware behavior. The policers are created based on the defined policer IDs within the associated queue group template. Multiple forwarding plane queue groups of the same template name may be created on ingress if they have a different instance identifier. These are termed *queue group instances*. Each instance of a named queue group created on a forwarding plane is an independent set of policers structured as per the queue group template. Forwarding plane queue groups are only supported with Ethernet ports and may be created on IOMs, IMMs, or XMAs with ports in a LAG.

7.6 Redirection models

Two models are supported for forwarding class redirection. In the first, the actual instance of a queue group to use for forwarding class redirection is named in the QoS policy. This is called policy-based redirection.

In the second model, the forwarding class queue or policers to apply redirection to are identified in the ingress or egress QoS policy. However, the specific named queue group instance is not identified until a QoS policy is applied to a SAP. This is called SAP-based redirection.

Policy-based redirection allows different forwarding classes in the same QoS policy to be redirected to different queue groups, but it requires at least one QoS policy to be configured per queue group instance.

SAP-based redirection can require less QoS policies to be configured because the policy does not have to name the queue group. However, if redirected, all forwarding classes of a SAP must use the same named queue group instance.

Policy-based redirection is applicable to port queue groups on access ingress and access and network egress, while SAP-based redirection is applicable to forwarding plane queue groups on access and network ingress, and port queue groups on access and network egress.

7.7 Access SAP forwarding class-based redirection

Forwarding class redirection is provisioned within the SAP ingress or SAP egress QoS policy. In each policy, the forwarding class to queue ID mapping may optionally specify a named queue group instance (policy-based redirection) or may simply tag the forwarding class for redirection (SAP-based redirection). When the name is specified, the defined queue ID must exist in the queue group template with the same name.

7.7.1 Policy-based redirection

About this task

Redirecting a SAP forwarding class to a queue within a port-based queue group using policy-based redirection requires four steps.

Procedure

Step 1. Create an ingress or egress queue group template. If the forwarding class redirection is in the ingress SAP path, an ingress queue group template must be created. Similarly, an egress queue group template must be created for egress forwarding class redirection. Optionally, the queues in a template can be created using default parameters. Individual queues must be created before they are associated with a forwarding class. The default queue parameters may be overridden on each port-based queue group.

Step 2. (This step and the following step can be done in the opposite order.) Create an ingress or egress queue group instance with the same name as the template on the port associated with the SAP. Examples are as follows:

On ingress ports:

config>port>ethernet>access>ingress>queue-group *queue-group-name*

On egress ports:

config>port>ethernet>access>egress>queue-group *queue-group-name* [**instance** *instance-id*]

Queue parameter overrides can also be applied at this time.

Step 3. Redirect the SAP ingress or SAP egress QoS policy forwarding class policer or queue to the queue group name and wanted queue ID. Examples are as follows:

On ingress:

config>qos>sap-ingress *policy-id*

fc *fc-name*

queue *queue-id* **group** *queue-group-name*

On egress:

config>qos>sap-egress *policy-id*

fc *fc-name*

queue *queue-id* **group** *queue-group-name* **instance** *instance-id*

config>qos>sap-egress *policy-id*

fc *fc-name*

policer *policer-id* **group** *queue-group-name* **instance** *instance-id*

Step 4. Finally, the SAP ingress or SAP egress QoS policy must be applied to the SAP.

7.7.2 SAP-based redirection

About this task

Redirecting a SAP forwarding class to a queue within an egress port-based or ingress forwarding plane queue group using SAP-based redirection requires four steps.

Procedure

Step 1. Create an ingress or egress queue group template. If the forwarding class redirection is in the ingress SAP path, an ingress queue group template must be created. Similarly, an egress queue group template must be created for egress forwarding class redirection. Optionally, the queues in a template can be created using default parameters. Individual queues must be created before they are associated with a forwarding class. The default queue parameters may be overridden on each port-based queue group.

Step 2. (This step and the following step can be done in the opposite order.) Create an ingress queue group instance on the forwarding plane of the IOM/IMM/XMA, or an egress port queue group with the same name as the template on the port associated with the SAP.

On ingress:

config>card>fp>ingress>access>queue-group *queue-group-name* **instance** *instance-id* [**create**]

On egress:

config>port>ethernet>access>egress>queue-group *queue-group-name* [**instance** *instance-id*]

Step 3. Redirect the SAP ingress forwarding class policer in the SAP-ingress QoS policy using the keyword **fp-redirect-group** keyword on the policer, or SAP egress forwarding class queue or policer using the **port-redirect-group** keyword. (Steps 2 and 3 may be done in opposite order.)

On ingress:

config>qos>sap-ingress *policy-id*

fc *fc-name*

queue *queue-id* **fp-redirect-group**

On egress:

config>qos>sap-egress *policy-id*

fc *fc-name*

queue *queue-id* **port-redirect-group-queue**

config>qos>sap-egress *policy-id*

fc *fc-name*

policer *policer-id* **port-redirect-group-queue**

Step 4. Finally, the SAP ingress or SAP egress QoS policy must be applied to the SAP. The named queue group instance that was created on the ingress forwarding plane or the egress port must be specified at this time.

On ingress:

config>service>epipe>sap *sap-id*

ingress

qos *sap-ingress-policy-id* **fp-redirect-group** *queue-group-name* **instance** *instance-id*

On egress:

config>service>epipe>sap *sap-id*

egress

qos *sap-egress-policy-id* **port-redirect-group** *queue-group-name* **instance** *instance-id*

7.7.3 Ingress and egress SAP forwarding class redirection association rules

7.7.3.1 Policy-based provisioning model

The association rules between SAP ingress and egress QoS policies and queue group templates are as follows: both the target queue group name and queue ID within the group are explicitly stated within the access QoS policies.

The following association rules apply when the policy-based provisioning model is applied with port queue groups.

When a SAP ingress QoS policy forwarding class is redirected to a queue group queue ID:

- If the queue group name does not exist as an ingress queue group template, the forwarding class redirection fails.
- If a redirection queue ID does not exist within the ingress queue group template, the forwarding class redirection fails.
- If the SAP ingress QoS policy is currently applied to a non-Ethernet port or an Ethernet port where the specified ingress queue group does not exist, the forwarding class redirection fails.

When a SAP ingress QoS policy forwarding class redirection is removed from a queue group queue ID:

- If the forwarding class is being moved to another queue group queue ID that does not exist within an ingress queue group template, the redirection removal from the current queue group queue ID fails.
- If the forwarding class is being moved to a local queue ID within the SAP ingress QoS policy and the local queue ID does not exist, the redirection removal from the current queue group queue ID fails.
- If the forwarding class is being moved to a local queue ID within the SAP ingress QoS policy and it is the first forwarding class to be mapped to the queue ID, the system attempts to instantiate the queue on

each ingress SAP where the SAP ingress QoS policy is applied. If the queue cannot be created on any of the SAPs, the redirection removal from the current queue group ID fails.

When a SAP egress QoS policy forwarding class is redirected to a queue group queue ID:

- If the queue group name does not exist as an egress queue group template, the forwarding class redirection fails.
- If a redirection queue ID does not exist within the egress queue group template, the forwarding class redirection fails.
- If the SAP egress QoS policy is currently applied to a non-Ethernet port or an Ethernet port where the specified egress queue group does not exist, the forwarding class redirection fails.

When a SAP egress QoS policy forwarding class redirection is removed from a queue group queue ID:

- If the forwarding class is being moved to another queue group queue ID that does not exist within an egress queue group template, the redirection removal from the current queue group queue ID fails.
- If the forwarding class is being moved to a local queue ID within the SAP egress QoS policy and the local queue ID does not exist, the redirection removal from the current queue group queue ID fails.
- If the forwarding class is being moved to a local queue ID within the SAP egress QoS policy and it is the first forwarding class to be mapped to the queue ID, the system attempts to instantiate the queue on each egress SAP where the SAP egress QoS policy is applied. If the queue cannot be created on any of the SAPs, the redirection removal from the current queue group ID fails.

If the preceding operation is successful:

- The system decrements the association counter for the egress queue group template with the same name as the queue group previously specified in the forwarding class redirection.
- The system decrements the queue ID association counter within the queue group template for the queue ID previously specified in the forwarding class redirection.
- The system decrements the port queue group association counter for each egress port queue group where the SAP egress QoS policy is applied to a SAP.

When a SAP ingress QoS policy with a forwarding class redirection to a queue group queue ID is applied to a SAP, the SAP ingress QoS policy application fails if the queue group specified in any forwarding class redirection does not exist as an ingress port queue group on the port associated with the SAP.

If the preceding operation is successful, the system increments the port queue group association counter for each ingress port queue group referenced in a forwarding class redirection on the port associated with the SAP. The ingress port queue group association counter is incremented for each forwarding class redirected to the queue group within the added policy.

When a SAP ingress QoS policy with a forwarding class redirection to a queue group queue ID is removed from a SAP, the SAP ingress QoS policy removal action fails.

If the preceding operation is successful, the system decrements the port queue group association counter for each egress port queue group referenced in a forwarding class redirection within the removed SAP egress QoS policy. The egress port queue group association counter is decremented for each forwarding class redirected to the queue group within the removed policy.

7.7.3.2 SAP-based provisioning model

When a redirection to a named forwarding plane queue group instance is applied to a SAP on ingress:

- If the queue group name does not exist as an ingress queue group template, the redirection fails.

- If a queue group name does exist as an ingress queue group template, but the specified instance-id has not been instantiated on the same forwarding plane as used by the SAP, the redirection fails.
- If a redirected policer ID in the SAP ingress QoS policy does not match a policer ID in the named ingress queue group template, the redirection fails.
- If the SAP ingress QoS policy is currently applied to a non-Ethernet port or an Ethernet port where the specified ingress queue group instance does not exist on the forwarding plane, the redirection fails.

If the preceding operation is successful:

- The system increments the association counter for the ingress queue group template with the same name as the queue group specified in the SAP redirection for each forwarding class redirected to the template.
- The system increments the policer ID association counter within the queue group template for each forwarding class redirected to a policer ID.
- The system increments the forwarding plane queue group instance association counter for each ingress queue group instance where a SAP ingress QoS policy specifying redirection is applied to a SAP.

When redirection to a named queue group is removed from an ingress SAP:

- If the forwarding class is being moved to another queue group policer ID that does not exist within the ingress FP queue group, the redirection removal from the current queue group policer ID fails.
- If the forwarding class is being moved to a local policer ID within the SAP ingress QoS policy and the local policer ID does not exist, the redirection removal from the current queue group policer ID fails.
- If the forwarding class is being moved to a local policer ID within the SAP ingress QoS policy and it is the first forwarding class to be mapped to the policer ID, the system attempts to instantiate the policer on each ingress SAP where the SAP ingress QoS policy is applied. If the policer cannot be created on any of the SAPs, the redirection removal from the current queue group policer ID fails.

If the preceding operation is successful:

- The system decrements the association counter for the ingress queue group template with the same name as the queue group previously specified in the forwarding class redirection.
- The system decrements the policer ID association counter within the queue group template for the policer ID previously specified in the forwarding class redirection.

The system decrements the forwarding plane queue group template association counter for each ingress queue group where redirection is applied to the ingress SAP.

For the SAP-based provisioning model, the rules for redirecting a forwarding class queue to an egress port queue group are similar to those on ingress.

- If an egress QoS policy containing one or more redirections is applied to a SAP, but either no queue group instance is specified at association time, or a named queue group instance is specified and either the queue group name or the instance identifier does not correspond to a queue group that has been created on the egress port, the association is rejected.
- If all of the redirections in an egress QoS policy are to queue ids that do not exist in the named queue group instance, then the association is rejected.
- If a policer local to a SAP feeds into a SAP-based queue group queue instance, and the queue ID to use is not explicitly specified in the egress QoS policy (through the command policer policer-id port-redirect-group-queue) and is instead inferred from the forwarding class of the policer, but that forwarding class does not exist in the queue group template, then no error is generated. Instead, the queue with the lowest queue ID is used in the queue group instance. If at a later time, a user attempts to add a queue with a specific queue-id to a policer redirect for a specific forwarding class in the egress

QoS template, then the system checks that the corresponding queue-id exists in any queue group instances associated with any SAPs using the QoS policy.

7.7.4 Access queue group statistics

7.7.4.1 Port queue groups

When an ingress or egress queue group template is defined, a port-based queue group with the same name may be created. Port queue groups are named objects that act as a container for a group of queues. The queues are created based on the defined queue IDs within the associated queue group template. Port queue groups must be created individually on the ingress and egress sides of the port, but multiple port queue groups of the same template name may be created on egress ports if they have a different instance identifier. These are termed 'queue group instances'. Each instance of a named queue group created on a port is an independent set of queues structured as per the queue group template. Port queue groups are only supported on Ethernet ports and may be created on ports within a LAG.

Additional parameters can be configured under port queue groups, for example, an accounting policy, queue overrides, a scheduler policy, and scheduler overrides.

7.7.4.2 Forwarding plane queue groups

When a forwarding class is redirected to a forwarding plane queue group queue or policer, the packets sent to the queue or policer are statistically tracked by a set of counters associated with the queue group queue/policer and not with any of the counters associated with the SAP.

This means that it is not possible to perform accounting within a queue group based on the source SAPs feeding packets to the queue. That is, the statistics associated with the SAP do not include packets redirected to a queue group queue.

If the user enables the **packet-byte-offset** {**add** *bytes* | **subtract** *bytes*} option under the ingress queue-group policer, the byte counters of that policer reflect the adjusted packet size.

The set of statistics per queue are eligible for collection in a similar manner to SAP queues. The **collect-stats** command enables or disables statistics collection into a billing file based on the accounting policy applied to the queue group.

7.8 Network IP interface forwarding class-based redirection

About this task

Forwarding class redirection for a network IP interface is defined in a four-step process:

Procedure

Step 1. Create an ingress or egress queue group template with the appropriate queues or policers.

Step 2. Apply an instance of an ingress queue-group template created in step 1 (containing only policers) to the FP ingress network configuration context of card X. In addition, or alternatively, apply an instance of an egress queue-group template created in step 1 to the network egress configuration context of port Y.

Step 3. Configure the network QoS policy used on the IP interface to redirect ingress traffic to a policer ID (defined in the ingress queue-group template created in step 1) on the basis of forwarding-class and forwarding-type (unicast vs. multicast). In addition, or alternatively, configure the network QoS policy to redirect egress traffic to a queue ID or a policer ID, or both, based on forwarding-class.

Step 4. Apply the network QoS policy to the network IP interface and at the same time specify the ingress or egress queue-group instances, or both, associated with the interface.

7.8.1 Egress network forwarding class redirection association rules

The association rules work differently for network egress IP interfaces than they do for access SAPs. Because the network QoS policy does not directly reference the queue group names, the system is unable to check for queue group template existence or queue ID existence when the forwarding class queue redirection is defined. Configuration verification can only be checked at the time the network QoS policy is applied to a network IP interface.

The system keeps an association counter for each queue group template and an association counter for each queue ID within the template. The system also keeps an association counter for each queue group created on a port.

When a network QoS policy is applied to an IP interface with the queue group parameter specified:

- If the queue group name does not exist as an egress queue group template, the QoS policy application fails.
- If a redirection queue ID within the policy does not exist within the egress queue group template, the QoS policy application fails.
- If the IP interface is bound to a port (or LAG) and the specified queue group name does not exist on the port, the QoS policy application fails.

If the preceding operation is successful:

- The system increments the association counter for the queue group template with the same name as the queue group specified when the QoS policy is applied.
- The system increments the queue ID association counter within the queue group template for each forwarding class redirected to the queue ID.
- If the IP interface is currently bound to a port (or LAG), the association counter for the queue group on the port is incremented.

When the queue group parameter is removed from an IP interface:

- The system decrements the association counter for the queue group template with the same queue group name that was removed from the IP interface.
- The system decrements the queue ID association counter within the queue group template for each forwarding class that had previously been redirected to the queue ID.
- If the IP interface is currently bound to a port (or LAG), the association counter for the removed queue group on the port is decremented.

When a network QoS policy egress forwarding class redirection to a queue ID is removed or added, the redirection fails if a redirection is being added to a forwarding class and the queue ID does not exist on the queue groups for IP interfaces where the QoS policy is applied.

If the preceding operation is successful:

- The system finds all IP interfaces where the policy is applied.
- The system finds all affected queue group templates based on the queue group associated with the QoS policy on each interface.
- If removing, the queue ID association counter is decremented within each queue group template based on the queue ID removed from the policy.
- If adding, the queue ID association counter is incremented within each queue group template based on the queue ID added to the policy.

When an IP interface associated with a queue group is bound to a port, the port binding fails if the specified egress queue group does not exist on the port.

If the preceding operation is successful, the system increments the association counter for the queue group on the port.

When an IP interface associated with a queue group is unbound from a port, the system decrements the association counter for the queue group on the unbound port.

7.8.2 Egress network IP interface statistics

The statistics for network interfaces work differently than statistics on SAPs. Counter sets are created for each egress IP interface and not per egress queue. When a forwarding class for an egress IP interface is redirected from the default egress port queue to a queue group queue, the system continues to use the same counter set.

7.9 Queue group behavior on LAG

7.9.1 Queue group queue instantiation per link

When a port queue group is created on a Link Aggregation Group (LAG) context, it is individually instantiated on each link in the LAG.

7.9.2 Per-link queue group queue parameters

The queue parameters for a queue within the queue group are used for each port queue and are not divided or split between the port queues representing the queue group queue. For instance, when a queue rate of 100 Mb/s is defined on a queue group queue, each instance of the queue group (on each LAG port) has a rate of 100 Mb/s.

7.9.3 Adding a queue group to an existing LAG

A queue group must be created on the primary (lowest port ID) port of the LAG. If an attempt is made to create a queue group on a port other than the primary, the attempt fails. When the group is defined on the primary port, the system attempts to create the queue group on each port of the LAG. If sufficient resources are not available on each port, the attempt to create the queue group fails.

Any queue group queue overrides defined on the primary port are automatically replicated on all other ports within the LAG.

7.9.4 Adding a port to a LAG

When adding a port to a LAG group, the port must have the same queue groups defined as the existing ports on the LAG before it is allowed as a member. This includes all queue group override parameters.

7.9.5 Removing a queue group from a LAG

A queue group must be removed from the primary port of the LAG. The queue group is deleted by the system from each of the port members of the LAG.

7.10 Basic configurations

7.10.1 Configuring an ingress queue group template

The following displays an ingress queue group template configuration example:

```
*A:Dut-T>cfg>qos>qgrps# info
----------------------------------------------
            ingress
                queue-group "QG_ingress_1" create
                    queue 1 best-effort create
                    exit
                    queue 2 best-effort create
                    exit
                    queue 3 best-effort create
                    exit
                    queue 4 best-effort create
                    exit
                exit
            exit
----------------------------------------------
*A:Dut-T>cfg>qos>qgrps#
```

Note: To fully use the queue group feature to save queues, explicitly map all forwarding classes to queue group queues. This rule is applicable to SAP ingress, SAP egress, and network QoS policies.

7.10.2 Configuring an egress queue group template

The following displays an egress queue group template configuration example:

```
*A:Dut-T>cfg>qos>qgrps# info
----------------------------------------------
...
            egress
```

```
                queue-group "QG_egress_1" create
                    description "Egress queue group"
                    queue 1 best-effort create
                        mbs 100
                    exit
                    queue 2 best-effort create
                        mbs 100
                    exit
                    queue 3 best-effort create
                        mbs 100
                    exit
                    queue 4 best-effort create
                        mbs 100
                    exit
                exit
            exit
-------------------------------------------
*A:Dut-T>cfg>qos>qgrps#
```

7.10.3 Applying ingress queue group to SAP ingress policy

The following displays a SAP ingress policy configuration with **group** *queue-group-name* specified:

```
*A:Dut-T>config>qos>sap-ingress# info
-------------------------------------------
            queue 1 create
            exit
            queue 11 multipoint create
            exit
            fc "af" create
                queue 2 group "QG_ingress_1"
            exit
            fc "be" create
                queue 1 group "QG_ingress_1"
            exit
            fc "ef" create
                queue 3 group "QG_ingress_1"
            exit
            fc "nc" create
                queue 4 group "QG_ingress_1"
            exit
            dot1p 0 fc "be"
            dot1p 2 fc "af"
            dot1p 4 fc "ef"
            dot1p 6 fc "nc"
-------------------------------------------
*A:Dut-T>config>qos>sap-ingress#
```

7.10.4 Applying egress queue group to SAP egress policy

The following displays a SAP egress policy configuration with **group** *queue-group-name* specified:

```
A:Dut-T>config>qos>sap-egress# info
-------------------------------------------
            queue 1 create
            exit
            fc af create
                queue 2 group "QG_egress_1"
```

```
            exit
            fc be create
                queue 1 group "QG_egress_1"
            exit
            fc ef create
                queue 3 group "QG_egress_1"
            exit
            fc nc create
                queue 4 group "QG_egress_1"
            exit
-------------------------------------------
A:Dut-T>config>qos>sap-egress#
```

7.10.5 Configuring SAP-based egress queue redirection

The following displays a SAP egress policy configuration with port-redirect-group-queue construct (shown for regular egress queues) and the actual queue-group-name is determined by the SAP egress QoS configuration:

```
*A:Dut-A# configure qos sap-egress 3
*A:Dut-A>config>qos>sap-egress# info
-------------------------------------------
            queue 1 create
            exit
            queue 2 create
            exit
            policer 8 create
                rate 50000
            exit
            fc af create
                queue 3 port-redirect-group-queue
                exit
            exit
            fc be create
                queue 3 port-redirect-group-queue
                exit
            exit
            fc ef create
                policer 8 port-redirect-group-queue
                exit
            exit
            fc h1 create
                queue 3 port-redirect-group-queue
                exit
            exit
            fc h2 create
                queue 3 port-redirect-group-queue
                exit
            exit
            fc l1 create
                queue 3 port-redirect-group-queue
                exit
            exit
            fc l2 create
                queue 3 port-redirect-group-queue
                exit
            exit
            fc nc create
                queue 3 port-redirect-group-queue
                exit
```

```
            exit
----------------------------------------------
```

This is to be configured in-conjunction with the following:

```
*A:Dut-A# configure service vpls 1
*A:Dut-A>config>service>vpls# info
----------------------------------------------
            stp
                shutdown
            exit
            sap 9/1/2:1 create
                egress
                    qos 3 port-redirect-group qg1 instance 101
                exit
            exit
```

7.10.6 Configuring queue group on Ethernet access ingress port

The provisioning steps involved in using a queue-group queue on an ingress port are:

1. Create the queue group template.
   a. Create the queue group template in the ingress context.
   b. Create the queue within the queue group template.
2. Create the queue group.
   a. Identify the ingress port (or ports) for which the queue group is needed (for LAG, use the primary port member).
   b. Create a queue group with the same name as the template on the port or ports.
3. Map a forwarding class to the queue-id within the queue group.
   a. Map forwarding classes to queue-group queues.
   b. Identify or create the SAP ingress QoS policy that is used on the ingress SAP where queue redirection is needed.
   c. Map the needed forwarding classes to the queue group name and the specific queue ID within the group.
4. Apply the SAP ingress QoS policy.
   a. Identify or create the ingress SAP requiring forwarding class redirection to the queue group.
   b. Assign the QoS policy to the SAP.

The following displays an Ethernet access ingress port queue-group configuration example:

```
*A:Dut-T>config>port# /configure port 9/2/1
*A:Dut-T>config>port# info
----------------------------------------------
        ethernet
            mode access
            access
                ingress
                    queue-group "QG_ingress_1" create
```

```
                    exit
                exit
                egress
                    queue-group "QG_egress_1" create
                    exit
                exit
            exit
        exit
        no shutdown
-----------------------------------------------
*A:Dut-T>config>port#

*A:Dut-T>config>port# /configure port 9/2/2
*A:Dut-T>config>port# info
-----------------------------------------------
        ethernet
            mode access
            access
                ingress
                    queue-group "QG_ingress_1" create
                    exit
                exit
                egress
                    queue-group "QG_egress_1" create
                    exit
                exit
            exit
        exit
        no shutdown
-----------------------------------------------
*A:Dut-T>config>port#
```

7.10.7 Configuring overrides

The following output displays a port queue group queue override example.

```
*A:Dut-T>config>port>ethernet>access# /configure port 9/2/1
*A:Dut-T>config>port# info
-----------------------------------------------
        ethernet
            mode access
            access
                ingress
                    queue-group "QG_ingress_1" create
                        queue-overrides
                            queue 2 create
                                rate 800000 cir 20000
                            exit
                        exit
                    exit
                exit
                egress
                    queue-group "QG_egress_1" create
                    exit
                exit
            exit
        exit
        no shutdown
-----------------------------------------------
*A:Dut-T>config>port# /configure port 9/2/2
*A:Dut-T>config>port# info
```

```
-----------------------------------------------
        ethernet
            mode access
            access
                ingress
                    queue-group "QG_ingress_1" create
                    exit
                exit
                egress
                    queue-group "QG_egress_1" create
                        queue-overrides
                            queue 3 create
                                rate 1500000 cir 2000
                            exit
                        exit
                    exit
                exit
            exit
        exit
        no shutdown
-----------------------------------------------
*A:Dut-T>config>port#
```

7.10.8 Configuring queue group on Ethernet access egress port

The provisioning steps involved in using a queue-group queue on an egress access port are:

1. Create the queue group template.
   a. Create the queue group template in the egress context.
   b. Create the queue within the queue group template.
2. Create the queue group.
   a. Identify the egress port (or ports) for which the queue group is needed (for LAG use the primary port member).
   b. Create a queue group instance with the same name as the template on the port or ports.

From this point, there are two methods for regular Ethernet-based SAPs to have port access egress redirection, policy-based redirection and SAP-based redirection. For policy-based redirection:

1. Map a forwarding class to the queue-id within the queue group.
   a. Identify or create the SAP egress QoS policy that is used on the egress SAP where policy-based queue redirection is needed.
   b. Map the needed forwarding classes to the queue group name and the specific queue ID within the group with the "group" keyword.
2. Apply the SAP egress QoS policy.
   a. Identify or create the egress SAP requiring forwarding class redirection to the queue group.
   b. Assign the QoS policy to the SAP.

For SAP-based redirection:

1. Map a forwarding class to the queue-id within the queue group.
   a. Identify or create the SAP egress QoS policy that is used on the egress SAP where SAP-based queue redirection is needed.

b. Map the needed forwarding classes to the queue group specific queue-id, and the keyword "port-redirect-group-queue". The actual queue-group template name is determined by the sap instance's configuration that associated the sap-egress qos policy in conjunction with the port-redirect-group's instance.

2. Apply the SAP egress QoS policy and the queue-group template's instance under the SAP.
   a. Identify or create the egress SAP requiring forwarding class redirection to the queue group.
   b. Assign the QoS policy and the egress queue-group template's instance to the SAP.

7.10.9 Configuring queue group for network egress traffic on port

The provisioning steps involved in using a queue-group queue on an egress network port are:

1. Create the queue group template.
   a. Create the egress queue group template.
   b. Create the queues or policers, or both, within the queue group template.
2. Create the queue group.
   a. Identify the egress port (or ports) on which the queue group is needed (for LAG, use the primary port member).
   b. Create a queue group with the same name as the template on the port or ports. The instance ID is optional.
3. Map a forwarding class to the queue-id within the queue group.
   a. Identify or create the network QoS policy that is used on the egress IP interface where queue redirection is needed.
   b. Map the needed egress forwarding classes within the network QoS policy to the specific queue IDs or policer IDs, or both, within the group (the group name is supplied when the QoS policy is applied to the IP interface).
4. Apply the network QoS policy.
   a. Identify or create the IP interface requiring forwarding class redirection to the queue group.
   b. Assign the QoS policy to the IP interface and specify the queue group name (and optionally, instance ID) for redirection of egress traffic.

When a queue within a template is mapped by a forwarding class on any object, the queue may be edited, but not deleted.

7.10.10 Configuring queue group for network ingress traffic on forwarding plane

The provisioning steps involved in using a queue-group for ingress traffic on a network interface are:

1. Create the queue group template.
   a. Create the ingress queue group template.
   b. Create the policers within the queue group template.
2. Create the queue group.
   a. Identify the ingress forwarding plane on which the queue group is needed.

b. Create a queue group with the same name as the template in the FP ingress network configuration context. An instance ID is mandatory.

3. Map a forwarding class to the policer-id within the queue group.
   a. Identify or create the network QoS policy that is used on the ingress IP interface where queue redirection is needed.
   b. Map the needed ingress forwarding classes within the network QoS policy to the specific policer IDs within the group (the group name is supplied when the QoS policy is applied to the IP interface).
4. Apply the network QoS policy.
   a. Identify or create the IP interface requiring forwarding class redirection to the queue group.
   b. Assign the QoS policy to the IP interface and specify the queue group name and instance ID for redirection of ingress traffic.

7.10.11 Using queue groups to police ingress/egress traffic on network interface

An example of the provisioning steps involved in using a queue-group to police ingress and egress traffic on a network interface is as follows:

```
config
    qos
        queue-group-templates
            ingress
                queue-group "Ingress_QG_1" create
                    policer 2 create
                        rate 9000
                    exit
                exit
            exit
            egress
                queue-group "Egress_QG_1" create
                    queue 1 best-effort create
                    exit
                    policer 2 create
                        rate 9000
                    exit
                exit
            exit
        exit

        network 2 create
            ingress
                fc be
                    fp-redirect-group policer 2
                exit
            exit
            egress
                fc be
                    port-redirect-group policer 2
                exit
            exit
        exit


    card 1
        card-type xcm-x20
            mda 1                mda-type cx20-10g-sfp                no shutdown
```

```
            exit
        fp 1
            ingress
                network
                    queue-group "Ingress_QG_1" instance 550 create
                    exit
                exit
            exit
        exit
        no shutdown

    port 1/1/3
        ethernet
                mtu 1514
                network
                    egress
                        queue-group "Egress_QG_1" instance 550 create
                        exit
                    exit
                exit
            exit
        no shutdown
    exit

    router
        interface “to-D”
        address 10.10.11.3/24
        port 1/1/3
        qos 2 egress-port-redirect-group "Egress_QG_1" egress-instance
        550 ingress-fp-redirect-group "Ingress_QG_1" ingress-instance
        550
        no shutdown
```

7.10.12 Configuring ingress/egress PW shaping using spoke SDP forwarding class-based redirection

An example of the provisioning steps involved in configuring PW shaping using spoke SDP forwarding class-based redirection is as follows:

```
configure
#--------------------------------------------------
echo "QoS Policy Configuration"
#--------------------------------------------------
    qos
        queue-group-templates
            ingress
                queue-group "QGIng1" create
                    policer 1 create
                    exit
                    policer 2 create
                    exit
                    policer 3 create
                    exit
                    policer 4 create
                    exit
                exit
            exit
```

```
            egress
                queue-group "QGEgr1" create
                    queue 1 best-effort create
                    exit
                    policer 1 create
                    exit
                    policer 2 create
                    exit
                    policer 3 create
                    exit
                    policer 4 create
                    exit
                exit
            exit
        exit
    exit
        network 10 create
            ingress
                lsp-exp 0 fc be profile out
                lsp-exp 1 fc be profile out
                lsp-exp 2 fc be profile out
                lsp-exp 3 fc be profile out
                lsp-exp 4 fc be profile out
                lsp-exp 5 fc be profile out
                lsp-exp 6 fc be profile out
                lsp-exp 7 fc be profile out
                fc af
                    fp-redirect-group policer 4
                exit
                fc be
                    fp-redirect-group policer 1
                exit
                fc l1
                    fp-redirect-group policer 2
                exit
                fc l2
                    fp-redirect-group policer 3
                exit
            exit
            egress
                fc af
                    port-redirect-group policer 4
                exit
                fc be
                    port-redirect-group policer 1
                exit
                fc l1
                    port-redirect-group policer 2
                exit
                fc l2
                    port-redirect-group policer 3
                exit
            exit
        exit
    exit
#--------------------------------------------------
echo "Card Configuration"
#--------------------------------------------------
    card 3
        fp 1
            ingress
                network
                    queue-group "QGIng1" instance 1 create
                    exit
```

```
                    queue-group "QGIng1" instance 2 create
                    exit
                exit
            exit
        exit
    exit
#--------------------------------------------------
echo "Port Configuration"
#--------------------------------------------------
    port 3/2/1
        ethernet
            encap-type dot1q
            network
                egress
                    queue-group "QGEgr1" instance 1 create
                    exit
                    queue-group "QGEgr1" instance 2 create
                    exit
                exit
            exit
        exit
        no shutdown


*A:Dut-T>config>service#
        customer 1 create
            description "Default customer"
        exit
        sdp 1 mpls create
            description "Default sdp description"
            far-end 198.51.100.0
            ldp
            path-mtu 9000
            keep-alive
                shutdown
            exit
            no shutdown
        exit
        vpls 1 customer 1 vpn 1 create
            description "Default tls description for service id 1"
            service-mtu 9000
            stp
                shutdown
            exit
            service-name "XYZ Vpls 1"
            sap 9/2/1:1.* create
                description "Default sap description for service id 1"
                static-mac 00:00:1e:00:01:02 create
                ingress
                    qos 10
                exit
            exit
            spoke-sdp 1:101 vc-type vlan create
                description "Description for Sdp Bind 1 for Svc ID 1"
                ingress
                    qos 10 fp-redirect-group "QGIng1" instance 1
                exit
                egress
                    qos 10 port-redirect-group "QGEgr1" instance 1
                exit
                static-mac 00:00:28:00:01:02 create
                no shutdown
            exit
            no shutdown
```

```
        exit


    router
        interface "ip-192.168.0.0"
            address 192.168.0.0/24
            port 3/2/1:1
        exit
        interface "system"
            address 192.168.0.1/32
        exit
#--------------------------------------------
```

7.10.13 Specifying QoS policies on service SAPs

The following output displays a VPLS service configuration example.

```
*A:Dut-T>config>service>vpls# info
----------------------------------------------
            stp
                shutdown
            exit
            sap 9/2/1 create
                ingress
                    qos 10
                exit
                egress
                    qos 10
                exit
            exit
            sap 9/2/2 create
                ingress
                    qos 10
                exit
                egress
                    qos 10
                exit
            exit
            no shutdown
----------------------------------------------
*A:Dut-T>config>service>vpls#
```

8 Scheduler QoS policies

8.1 Scheduler policies

Virtual schedulers are created within the context of a scheduler policy that is used to define the hierarchy and parameters for each scheduler. A scheduler is defined in the context of a tier that is used to place the scheduler within the hierarchy. Three tiers of virtual schedulers are supported. Root schedulers are often defined without a parent scheduler, meaning it is not subject to obtaining bandwidth from a higher tier scheduler. A scheduler has the option of enforcing a maximum rate of operation for all child queues, policers, and schedulers associated with it.

Because a scheduler is designed to arbitrate bandwidth between many inputs, a metric must be assigned to each child queue, policer, or scheduler vying for transmit bandwidth. This metric indicates whether the child is to be scheduled in a strict or weighted fashion and the level or weight the child has to other children.

8.1.1 Egress port-based schedulers

H-QoS root (top-tier) schedulers always assumed that the configured rate was available, regardless of egress port-level oversubscription and congestion. This resulted in the possibility that the aggregate bandwidth assigned to queues was not actually available at the port level. When the H-QoS algorithm configures policers and queues with more bandwidth than available on an egress port, the actual bandwidth distribution to the policers and queues on the port is solely based on the action of the hardware scheduler. This can result in a forwarding rate at each queue that is very different than the wanted rate.

The port-based scheduler feature was introduced to allow H-QoS bandwidth allocation based on available bandwidth at the egress port level. The port-based scheduler works at the egress line rate of the port to which it is attached. Port-based scheduling bandwidth allocation automatically includes the Inter-Frame Gap (IFG) and preamble for packets forwarded on policers and queues servicing egress Ethernet ports. However, on PoS and SDH based ports, the HDLC encapsulation overhead and other framing overhead per packet is not known by the system. Instead of automatically determining the encapsulation overhead for SDH or SONET queues, the system provides a configurable frame encapsulation efficiency parameter that allows the user to select the average encapsulation efficiency for all packets forwarded out the egress queue.

A special port scheduler policy can be configured to define the virtual scheduling behavior for an egress port. The port scheduler is a software-based state machine managing a bandwidth allocation algorithm that represents the scheduling hierarchy shown in Figure 16: Port-level virtual scheduler bandwidth allocation based on priority and CIR.

The first tier of the scheduling hierarchy manages the total frame-based bandwidth that the port scheduler allocates to the eight priority levels.

The second tier receives bandwidth from the first tier in two priorities: a within-CIR loop and an above-CIR loop. The second-tier within-CIR loop provides bandwidth to the third-tier within-CIR loops, one for each of the eight priority levels. The second tier above-CIR loop provides bandwidth to the third-tier above-CIR loops for each of the eight priority levels.

The within-CIR loop for each priority level on the third tier supports an optional rate limiter used to restrict the maximum amount of within-CIR bandwidth the priority level can receive. A maximum priority level rate limit is also supported that restricts the total amount of bandwidth the level can receive for both within-CIR and above-CIR. The amount of bandwidth consumed by each priority level for within-CIR and above-CIR is predicated on the rate limits described and the ability for each child queue, policer, or scheduler attached to the priority level to use the bandwidth.

The priority 1 above-CIR scheduling loop has a special two-tier strict-distribution function. The high-priority level 1 above-CIR distribution is weighted between all queues, policers, and schedulers attached to level 1 for above-CIR bandwidth. The low-priority distribution for level 1 above-CIR is reserved for all orphaned policers, queues, and schedulers on the egress port. Orphans are policers, queues, and schedulers that are not explicitly or indirectly attached to the port scheduler through normal parenting conventions. By default, all orphans receive bandwidth after all parented queues and schedulers and are allowed to consume whatever bandwidth is remaining. This default behavior for orphans can be overridden on each port scheduler policy by defining explicit orphan port parent association parameters.

Ultimately, any bandwidth allocated by the port scheduler is given to a child policer or queue. The bandwidth allocated to the policer or queue is converted to a value for the PIR (maximum rate) setting of the policer or queue. This way, the hardware schedulers operating at the egress port level only schedule bandwidth for all policers or queues on the port up to the limits prescribed by the virtual scheduling algorithm.

The following lists the bandwidth allocation sequence for the port virtual scheduler:

1. Priority level 8 offered load up to priority CIR
2. Priority level 7 offered load up to priority CIR
3. Priority level 6 offered load up to priority CIR
4. Priority level 5 offered load up to priority CIR
5. Priority level 4 offered load up to priority CIR
6. Priority level 3 offered load up to priority CIR
7. Priority level 2 offered load up to priority CIR
8. Priority level 1 offered load up to priority CIR
9. Priority level 8 remaining offered load up to remaining priority rate limit
10. Priority level 7 remaining offered load up to remaining priority rate limit
11. Priority level 6 remaining offered load up to remaining priority rate limit
12. Priority level 5 remaining offered load up to remaining priority rate limit
13. Priority level 4 remaining offered load up to remaining priority rate limit
14. Priority level 3 remaining offered load up to remaining priority rate limit
15. Priority level 2 remaining offered load up to remaining priority rate limit
16. Priority level 1 remaining offered load up to remaining priority rate limit
17. Priority level 1 remaining orphan offered load up to remaining priority rate limit (default orphan behavior unless orphan behavior has been overridden in the scheduler policy)

When a policer or queue is inactive or has a limited offered load that is below its fair share (fair share is based on the bandwidth allocation a policer or queue would receive if it was registering adequate activity), its operational PIR must be set to some value to handle what would happen if the queues offered load increased before the next iteration of the port virtual scheduling algorithm. If an inactive policer or queue PIR was set to zero (or near zero), the policer or queue would throttle its traffic until the next algorithm

iteration. If the operational PIR was set to its configured rate, the result could overrun the expected aggregate rate of the port scheduler.

To accommodate inactive policers and queues, the system calculates a Minimum Information Rate (MIR) for each policer and queue. To calculate each policer or queue MIR, the system determines what the Fair Information Rate (FIR) of the queue or policer would be if that policer or queue had actually been active during the latest iteration of the virtual scheduling algorithm. For example, if three queues are active (1, 2, and 3) and two queues are inactive (4 and 5), the system first calculates the FIR for each active queue. Then, it recalculates the FIR for queue 4 assuming queue 4 was active with queues 1, 2, and 3, and uses the result as the queue's MIR. The same is done for queue 5 using queues 1, 2, 3, and 5. The MIR for each inactive queue is used as the operational PIR for each queue.

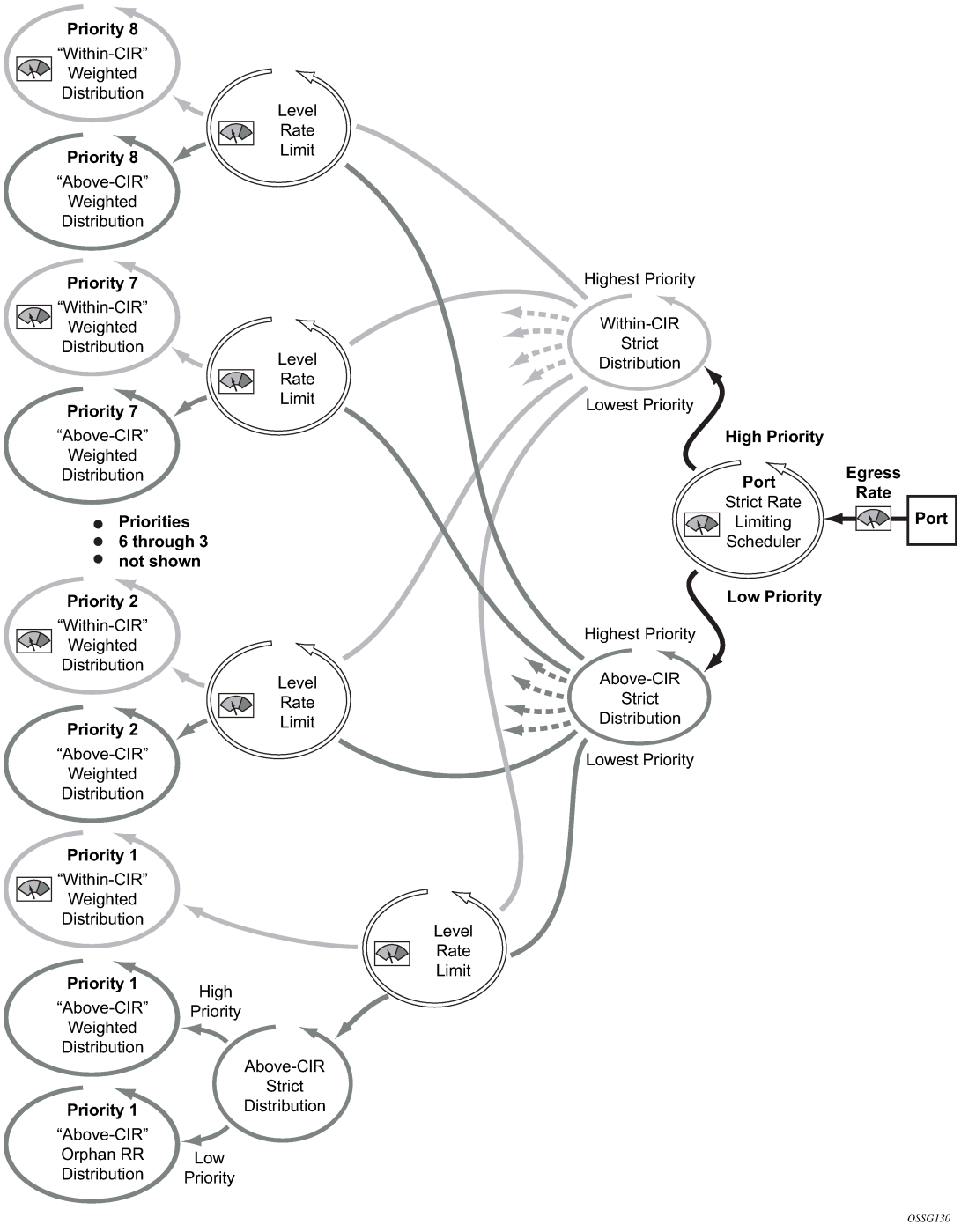
8.1.1.1 Service or multiservice site egress port bandwidth allocation

The port-based egress scheduler can be used to allocate bandwidth to each service or multiservice site associated with the port. While egress policers and queues on the service can have a child association with a scheduler policy on the SAP or multiservice site, all policers and queues must vie for bandwidth from an egress port. Two methods are supported to allocate bandwidth to each service or subscriber or multiservice site queue:

- service or multiservice site queue association with a scheduler on the SAP or multiservice site, either of which is associated with a port-level scheduler
- service or multiservice site queue association directly with a port-level scheduler

Note: Subscribers are not supported on the 7705 SAR Gen 2. Subscriber information is in this guide is included in this guide for reference only.

Figure 16: Port-level virtual scheduler bandwidth allocation based on priority and CIR

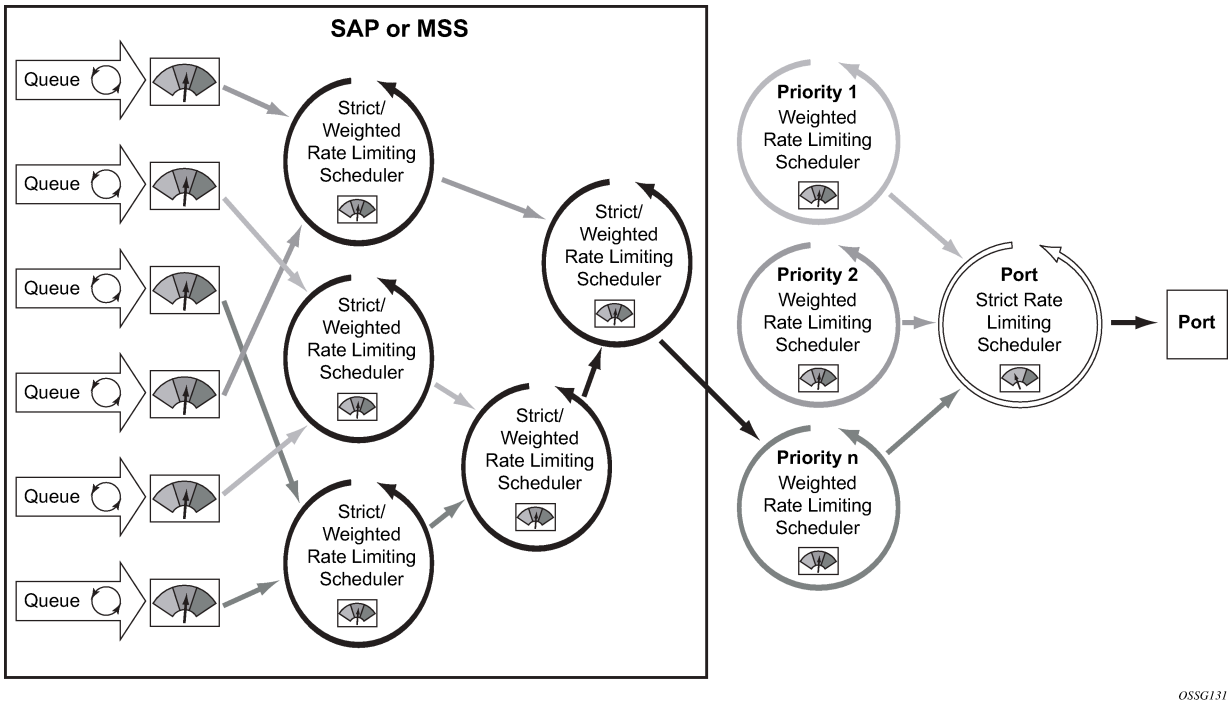


8.1.1.2 Service or multiservice site scheduler child to port scheduler parent

The service or multiservice site scheduler to port scheduler association model allows for multiple services or multiservice sites to have independent scheduler policy definitions while the independent schedulers

receive bandwidth from the scheduler at the port level. By using two-scheduler policies, available egress port bandwidth can be allocated fairly or unfairly depending on the needed behavior. Figure 17: Two-scheduler policy model for access ports shows this model.

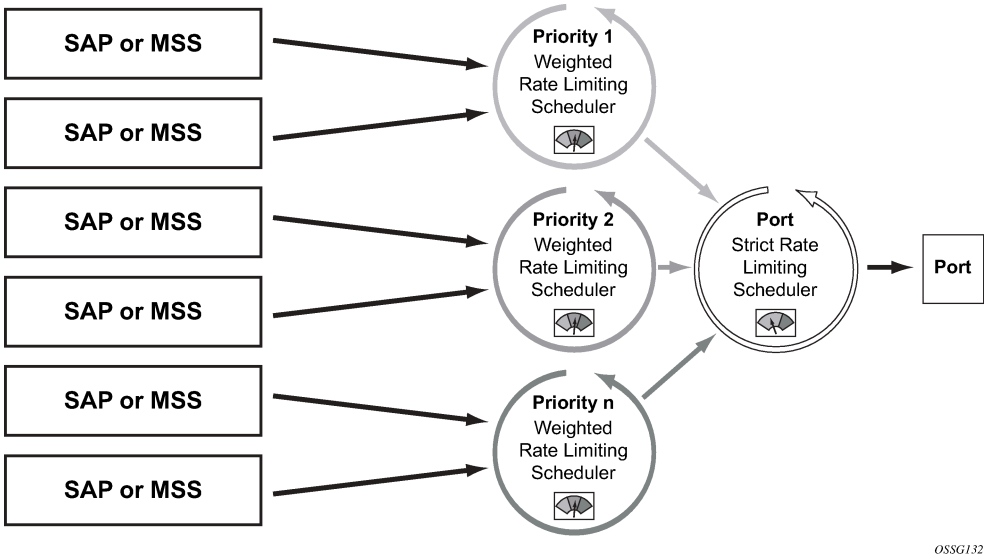
Figure 17: Two-scheduler policy model for access ports



When a two-scheduler policy model is defined, the bandwidth distribution hierarchy allocates the available port bandwidth to the port schedulers based on priority, weights, and rate limits. The service or multiservice site level schedulers and the policers and queues they service become an extension of this hierarchy.

Because of the nature of the two-scheduler policy, bandwidth is allocated on a per-service or multiservice site basis as opposed to a per-class basis. A common use of the two-policy model is for a carrier-of-carriers mode of business. with the goal of a carrier to provide segments of bandwidth to providers who purchase that bandwidth as services. While the carrier does not care about the interior services of the provider, it does care how congestion affects the bandwidth allocation to each provider's service. As an added benefit, the two-policy approach provides the carrier with the ability to preferentially allocate bandwidth within a service or subscriber or multiservice site context through the service or multiservice site level policy, without affecting the overall bandwidth allocation to each service or multiservice site. Figure 18: Schedulers on SAP or multiservice site receive bandwidth from port priority levels shows a per-service bandwidth allocation using the two-scheduler policy model. While the figure shows services grouped by scheduling priority, it is expected that many service models place the services in a common port priority and use weights to provide a weighted distribution between the service instances. Higher weights provide for relatively higher amounts of bandwidth.

Figure 18: Schedulers on SAP or multiservice site receive bandwidth from port priority levels



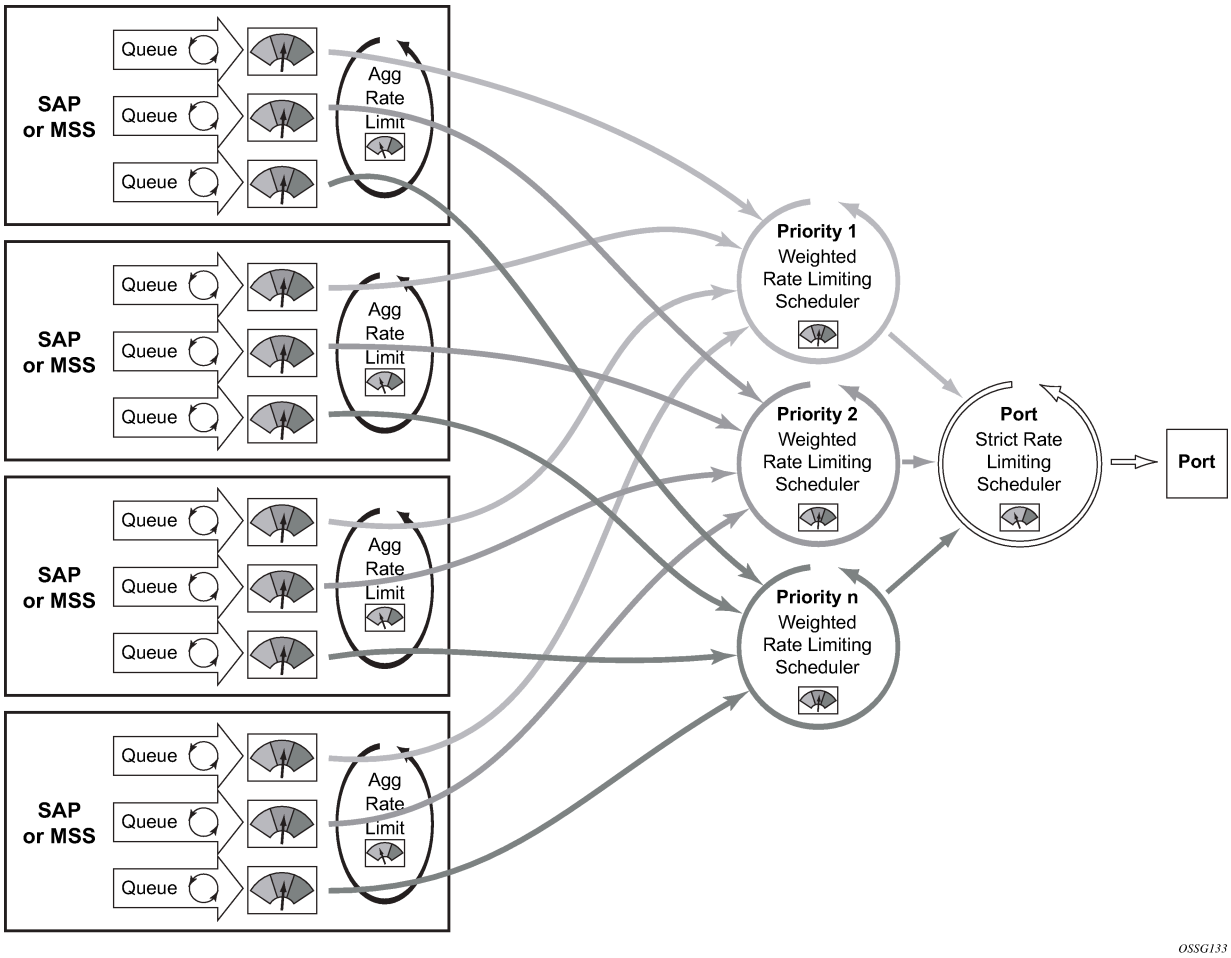
8.1.1.2.1 Direct service or multiservice site queue association to port scheduler parents

The second model of bandwidth allocation on an egress access port is to directly associate a service or subscriber or multiservice site policer or queue to a port-level scheduler. This model allows the port scheduler hierarchy to allocate bandwidth on a per class or priority basis to each service or subscriber or multiservice site policer or queue. This allows the provider to manage the available egress port bandwidth on a service tier basis ensuring that during egress port congestion, a deterministic behavior is possible from an aggregate perspective. While this provides an aggregate bandwidth allocation model, it does not inhibit per-service or per-subscriber or multiservice site queuing. Figure 19: Direct service or subscriber or multiservice site association to port scheduler model shows the single, port scheduler policy model.

Figure 19: Direct service or subscriber or multiservice site association to port scheduler model also shows the optional aggregate rate limiter at the SAP, multiservice site or subscriber or multiservice site level. The aggregate rate limiter is used to define a maximum aggregate bandwidth at which the child queues and policers, if used, can operate. While the port-level scheduler is allocating bandwidth to each child queue, the current sum of the bandwidth for the service or subscriber or multiservice site is monitored. When the aggregate rate limit is reached, no more bandwidth is allocated to the children associated with the SAP, multiservice site, or subscriber or multiservice site. Aggregate rate limiting is restricted to the single scheduler policy model and is mutually exclusive to defining SAP, multiservice site, or subscriber or multiservice site scheduling policies.

The benefit of the single scheduler policy model is that the bandwidth is allocated per priority for all queues associated with the egress port. This allows a provider to preferentially allocate bandwidth to higher priority classes of service independent of service or subscriber or multiservice site instance. In many cases, a subscriber can purchase multiple services from a single site (VoIP, HSI, Video) and each service can have a higher premium value relative to other service types. If a subscriber has purchased a premium service class, that service class should get bandwidth before another subscriber's best effort service class. When combined with the aggregate rate limit feature, the single port-level scheduler policy model provides a per-service instance or per-subscriber instance aggregate SLA and a class-based port bandwidth allocation function.

Figure 19: Direct service or subscriber or multiservice site association to port scheduler model



8.1.1.3 Frame and packet-based bandwidth allocation

A port-based bandwidth allocation mechanism must consider the effect that line encapsulation overhead plays relative to the bandwidth allocated per service or subscriber or multiservice site. The service or subscriber or multiservice site level bandwidth definition (at the queue level) operates on a packet accounting basis. For Ethernet, this includes the DLC header, the payload, and the trailing CRC. This does not include the IFG or the preamble. This means that an Ethernet packet consumes 20 bytes more bandwidth on the wire than what the policer or queue accounted for.

The port-based scheduler hierarchy must translate the frame-based accounting (on-the-wire bandwidth allocation) it performs to the packet-based accounting in the queues. When the port scheduler considers the maximum amount of bandwidth a queue should get, it must first determine how much bandwidth the policer or queue can use. This is based on the offered load the policer or queue is currently experiencing (how many octets are being offered). The offered load is compared to the configured CIR and PIR of the queue or policer. The CIR value determines how much of the offered load should be considered in the within-CIR bandwidth allocation pass. The PIR value determines how much of the remaining offered load (after within-CIR) should be considered for the above-CIR bandwidth allocation pass.

For Ethernet policers or queues (associated with an egress Ethernet port), the packet to frame conversion is relatively easy. The system multiplies the number of offered packets by 20 bytes and adds the result to the offered octets (offeredPackets x 20 + offeredOctets = frameOfferedLoad). This frame-offered-load value represents the amount of line rate bandwidth the policer or queue is requesting. The system computes the ratio of increase between the offered-load and frame-offered-load and calculates the current frame-based CIR and PIR. The frame-CIR and frame-PIR values are used as the limiting values in the within-CIR and above-CIR port bandwidth distribution passes.

From a provisioning perspective, queues, policers, and service level (and subscriber level) scheduler policies are always provisioned with packet-based parameters. The system converts these values to frame-based on-the-wire values for the purpose of port bandwidth allocation. However, port-based scheduler policy scheduler maximum rates and CIR values are always interpreted as on-the-wire values and must be provisioned accordingly. Figure 20: Port bandwidth distribution for service and port scheduler hierarchies and Figure 21: Port bandwidth distribution for direct queue to port scheduler hierarchy show a logical view of bandwidth distribution from the port to the queue level and shows the packet or frame-based provisioning at each step.

Figure 20: Port bandwidth distribution for service and port scheduler hierarchies

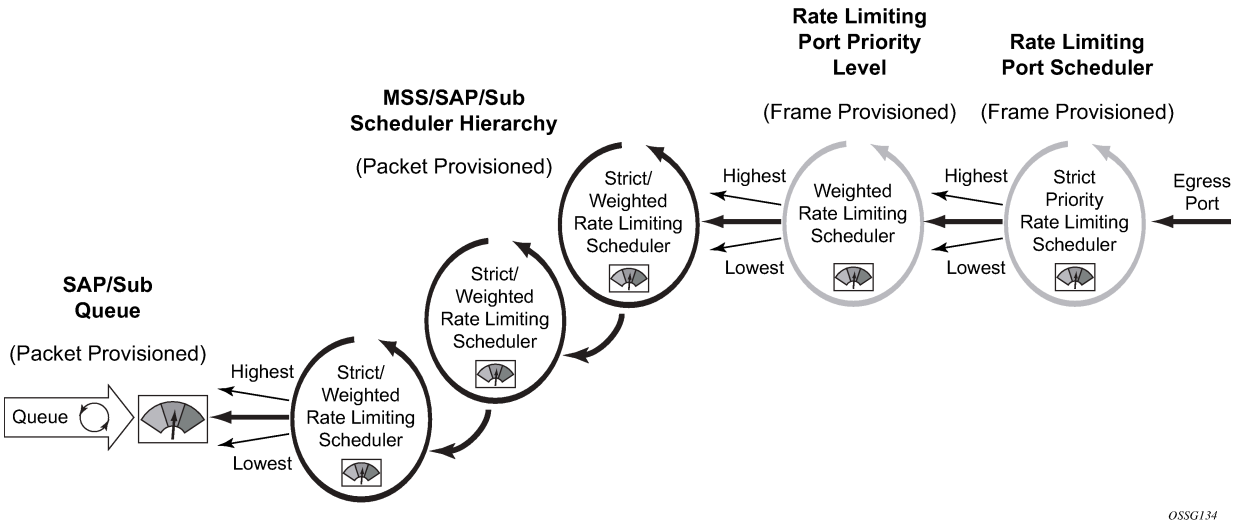
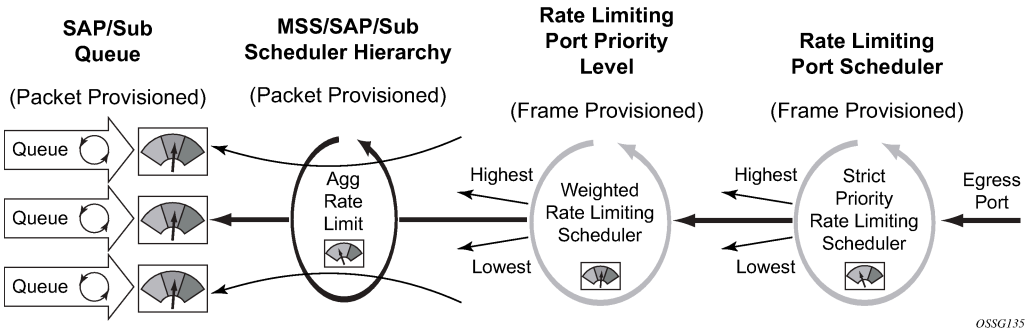


Figure 21: Port bandwidth distribution for direct queue to port scheduler hierarchy



8.1.1.4 Parental association scope

A **port-parent** command in the sap-egress and network-queue QoS policy policer or queue context defines the direct child/parent association between an egress policer or queue and a port scheduler priority level. The **port-parent** command is mutually exclusive to the already-existing **scheduler-parent** or **parent** command, which associates a policer or queue with a scheduler at the SAP, multiservice site, or subscriber or multiservice site profile level. It is possible to mix local parented (parent to service or subscriber or multiservice site level scheduler) and port parented policers and queues with schedulers on the same egress port.

The **port-parent** command only accepts a child/parent association to the eight priority levels on a port scheduler hierarchy. Similar to the local **parent** command, two associations are supported: one for within-CIR bandwidth (cir-level) and a second one for above-CIR bandwidth (level). The within-CIR association is optional and can be disabled by using the default within-CIR weight value of 0. If a policer or queue with a defined parent port is on a port without a port scheduler policy applied, that policer or queue is considered orphaned. If a policer or queue with a **scheduler-parent** or **parent** command is defined on a port and the named scheduler is not found due a missing scheduler policy or a missing scheduler of that name, the policer or queue is considered orphaned as well.

A queue or policer can be moved from a local parent (on the SAP, multiservice site, or subscriber or multiservice site profile) to a port parent priority level simply by executing the **port-parent** command. When the **port-parent** command is executed, any local parent information for the policer or queue is lost. The policer or queue can also be moved back to a local scheduler-parent or parent at any time by executing the **scheduler-parent** or **parent** command. Lastly, the local scheduler parent, parent, or port parent association can be removed at any time by using the **no** form of the appropriate parent command.

8.1.1.5 Service or subscriber or multiservice site-level scheduler parental association scope

The **port-parent** command in the scheduler-policy scheduler context (at all tier levels) allows a scheduler to be associated with a port scheduler priority level. The **port-parent** command, the **scheduler-parent** command, and the **parent** command for schedulers at tiers 2 and 3 within the scheduler policy are mutually exclusive. The **port-parent** command is the only parent command allowed for schedulers in tier 1.

The **port-parent** command only accepts a child/parent association to the eight priority levels on a port scheduler hierarchy. Similar to the normal local parent command, two associations are supported: one for within-CIR bandwidth (cir-level) and a second one for above-CIR bandwidth (level). The within-CIR association is optional and can be disabled by using the default within-CIR weight value of 0. If a scheduler with a port parent defined is on a port without a port scheduler policy applied, that scheduler is considered an orphaned scheduler.

A scheduler in tiers 2 and 3 can be moved from a local (within the policy) parent to a port parent priority level simply by executing the **port-parent** command. When the **port-parent** command is executed, any local scheduler-parent or parent information for the scheduler is lost. The schedulers at tiers 2 and 3 can also be moved back to a local scheduler-parent or parent at any time by executing the local **scheduler-parent** or **parent** command. Lastly, the local scheduler parent, parent, or port parent association can be removed at any time by using the **no** form of the appropriate parent command. A scheduler in tier 1 with a port parent definition can be added or removed at any time.

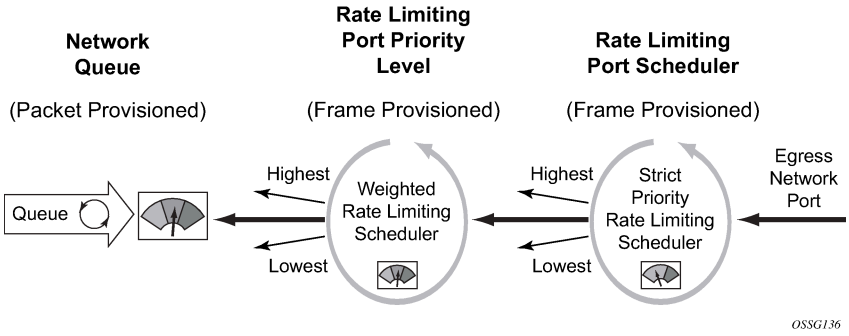
8.1.1.6 Network queue parent scheduler

Network queues support port scheduler parent priority-level associations. Using a port scheduler policy definition and mapping network queues to a port parent priority level, H-QoS functionality is supported providing eight levels of strict priority and weights within the same priority. A network queue's bandwidth is allocated using the within-CIR and above-CIR scheme normal for port schedulers.

Queue CIR and PIR percentages when port-based schedulers are in effect are based on frame-offered-load calculations. Figure 22: Bandwidth distribution on network port with port-based scheduling shows port-based virtual scheduling bandwidth distribution.

A network queue with a port parent association that exists on a port without a scheduler policy defined is considered to be orphaned.

Figure 22: Bandwidth distribution on network port with port-based scheduling



8.1.1.7 Foster parent behavior for orphaned queues and schedulers

All queues, policers, and schedulers on a port that has a port-based scheduler policy configured are subject to bandwidth allocation through the port-based schedulers. All queues, policers, and schedulers that are not configured with a scheduler parent are considered to be orphaned when port-based scheduling is in effect. This includes access and network queue schedulers at the SAP, multiservice site, subscriber, and port level.

By default, orphaned queues, policers, and schedulers are allocated bandwidth after all queues, policers, and schedulers in the parented hierarchy have had bandwidth allocated within-CIR and above-CIR. Therefore, an orphaned scheduler, policer, or queue can be considered as being foster parented by the port scheduler. Orphaned queues, policers, and schedulers have an inherent port scheduler association as shown below:

- Within-CIR priority = 1
- Within-CIR weight = 0
- Above-CIR priority = 1
- Above-CIR weight = 0

The above-CIR weight = 0 value is only used for orphaned policers, queues, and schedulers on port scheduler enabled egress ports. The system interprets weight = 0 as priority level 0 and only distributes bandwidth to level 0 when all other properly parented queues, policers, and schedulers have received bandwidth. Orphaned queues, policers, and schedulers all have equal priority to the remaining port bandwidth.

The default orphan behavior can be overridden for each port scheduler policy by using the orphan override command. The orphan override command accepts the same parameters as the port parent command. When the orphan override command is executed, all orphan queues, policers, and schedulers are treated in a similar fashion as other properly parented queues, policers, and schedulers based on the override parenting parameters.

It is expected that an orphan condition is not the wanted state for a queue, policer, or scheduler and is the result of a temporary configuration change or configuration error.

8.1.2 Frame-based accounting

The standard accounting mechanism uses 'packet-based' rules that account for the DLC header, any existing tags, Ethernet payload, and the 4-byte CRC. The Ethernet framing overhead that includes the Inter-Frame Gap (IFG) and preamble (20 bytes total) are not included in packet-based accounting. When frame-based accounting is enabled, the 20-byte framing overhead is included in the queue or policer CIR, PIR, and scheduling operations, allowing the operations to take into consideration on-wire bandwidth consumed by each Ethernet packet.

Because the native queue accounting functions (stats, CIR, and PIR) are based on packet sizes and do not include Ethernet frame encapsulation overhead, the system must manage the conversion between packet-based and frame-based accounting. To accomplish this, the system requires that a policer or queue operates in frame-based accounting mode and must be managed by a virtual scheduler policy or by a port virtual scheduler policy. Egress policers or queues can use either port or service schedulers to accomplish frame-based accounting, but ingress queues are limited to service-based scheduling policies.

Turning on frame-based accounting for a policer or queue is accomplished through a frame-based accounting command defined on the scheduling policy level associated with the policer or queue or through a policer or queue frame-based accounting parameter on the aggregate rate limit command associated with the queues SAP, multiservice site or subscriber or multiservice site context. Packet byte offset settings are not included in the applied rate when frame-based accounting is configured, however the offsets are applied to the statistics.

8.1.2.1 Operational modifications

To add frame overhead to the existing QoS Ethernet packet handling functions, the system uses the already existing virtual scheduling capability of the system. The system currently monitors each queue included in a virtual scheduler to determine its offered load. This offered load value is interpreted based on the defined CIR and PIR threshold rates of the policer or queue to determine bandwidth offerings from the policer or queue virtual scheduler. Frame-based usage on the wire allows the port bandwidth to be accurately allocated to each child policer and queue on the port.

8.1.2.2 Existing egress port-based virtual scheduling

The port-based virtual scheduling mechanism takes the native packet-based accounting results from the policer or queue and adds 20 bytes to each packet to derive the frame-based offered load of the policer or queue. The ratio between the frame-based offered load and the packet-based offered load is then used to determine the effective frame-based CIR and frame-based PIR thresholds for the policer or queue. When the port virtual scheduler computes the amount of bandwidth allowed to the policer or queue (in a frame-based fashion), the bandwidth is converted back to a packet-based value and used as the operational PIR

or the policer or queue. The native packet-based mechanisms of the policer or queue continue to function, but the maximum operational rate is governed by frame-based decisions.

8.1.2.3 Behavior modifications for frame-based accounting

The frame-based accounting feature extends this capability to allow the policer or queue CIR and PIR thresholds to be defined as frame-based values as opposed to packet-based values. The policer or queue continues to internally use its packet-based mechanisms, but the provisioned frame-based CIR and PIR values are continuously revalued based on the ratio between the calculated frame-based offered load and actual packet-based offered load. As a result, the operational packet-based CIR and PIR of the policer or queue are accurately modified during each iteration of the virtual scheduler to represent the provisioned frame-based CIR and PIR. Packet byte offset settings are not included in the applied rate when frame-based accounting is configured, however the offsets are applied to the statistics.

8.1.2.4 Virtual scheduler rate and queue rate parameter interpretation

Normally, a scheduler policy contains rates that indicate packet-based accounting values. When the children associated with the policy are operating in frame-based accounting mode, the parent schedulers must also be governed by frame-based rates. Because either port-based or service-based virtual scheduling is required for queue or policer frame-based operation, enabling frame-based operation is configured at either the scheduling policy or aggregate rate limit command level. All policers and queues associated with the policy or the aggregate rate limit command inherit the frame-based accounting setting from the scheduling context.

When frame-based accounting is enabled, the policer and queue CIR and PIR settings are automatically interpreted as frame-based values. If a SAP ingress QoS policy is applied with a queue PIR set to 100 Mb/s on two different SAPs, one associated with a policy with frame-based accounting enabled and the other without frame-based accounting enabled, the 100 Mb/s rate is interpreted differently for each queue. The frame-based accounting queue adds 20 bytes to each packet received by the queue and limits the rate based on the extra overhead. The packet-based accounting queue does not add the 20 bytes per packet and therefore allows more packets through per second. Packet byte offset settings are not included in the applied rate when frame-based accounting is configured; however, the offsets are applied to the statistics.

Similarly, the rates defined in the scheduling policy with frame-based accounting enabled are automatically interpreted as frame-based rates.

The port-based scheduler aggregate rate limit command always interprets its configured rate limit value as a frame-based rate. Setting the frame-based accounting parameter on the aggregate rate limit command only affects the policers and queues managed by the aggregate rate limit and converts them from packet-based to frame-based accounting mode.

8.1.3 Configuring port scheduler policies

8.1.3.1 Port scheduler structure

Every port scheduler supports eight strict priority levels with a two-pass bandwidth allocation mechanism for each priority level. Priority levels 8 through 1 (level 8 is the highest priority) are available for port-parent association for child queues, policers, and schedulers. Each priority level supports a maximum rate limit parameter that limits the amount of bandwidth that may be allocated to that level. A CIR parameter is also

supported that limits the amount of bandwidth allocated to the priority level for the child queue or policer offered load, within their defined CIR. An overall maximum rate parameter defines the total bandwidth that is allocated to all priority levels.

8.1.3.2 Special orphan queue and scheduler behavior

When a port scheduler is present on an egress port or channel, the system ensures that all policers, queues, and schedulers receive bandwidth from that scheduler to prevent free-running policers or queues that can cause the aggregate operational PIR of the port or channel to oversubscribe the bandwidth available. When the aggregate maximum rate for the policers and queues on a port or channel operates above the available line rate, the forwarding ratio between the policers and queues is affected by the hardware schedulers on the port and may not reflect the scheduling defined on the port or intermediate schedulers. Policers, queues, and schedulers that are either explicitly attached to the port scheduler using the port-parent command or are attached to an intermediate scheduler hierarchy that is ultimately attached to the port scheduler are managed through the normal eight priority levels. Queues, policers, and schedulers that are not attached directly to the port scheduler and are not attached to an intermediate scheduler that itself is attached to the port scheduler are considered orphaned and, by default, are tied to priority 1 with a weight of 0. All weight 0 policers, queues, and schedulers at priority level 1 are allocated bandwidth after all other children and each weight 0 child is given an equal share of the remaining bandwidth. This default orphan behavior may be overridden at the port scheduler policy by using the orphan-override command. The orphan-override command accepts the same parameters as the port-parent command. When the orphan-override command is executed, the parameters are used as the port parent parameters for all orphans associated with a port using the port scheduler policy.

8.1.3.3 Packet to frame bandwidth conversion

Another difference between the service-level scheduler-policy and the port-level port-scheduler-policy is in bandwidth allocation behavior. The port scheduler is designed to offer on-the-wire bandwidth. For Ethernet ports, this includes the IFG and the preamble for each frame and represents 20 bytes total per frame. The policers, queues, and intermediate service-level schedulers (a service-level scheduler is a scheduler instance at the SAP, multiservice site, or subscriber or multiservice site profile level) operate based on packet overhead that does not include the IFG or preamble on Ethernet packets. In order for the port-based virtual scheduling algorithm to function, it must convert the policer, queue, and service scheduler packet-based required bandwidth and bandwidth limiters (CIR and rate PIR) to frame-based values. This is accomplished by adding 20 bytes to each Ethernet frame offered at the queue or policer level to calculate a frame-based offered load. Then, the algorithm calculates the ratio increase between the packet-based offered load and the frame-based offered load and uses this ratio to adapt the CIR and rate PIR values for the policer or queue to frame-CIR and frame-PIR values. When a service-level scheduler hierarchy is between the policers, queues, and the port-based schedulers, the ratio between the average frame-offered-load and the average packet-offered-load is used to adapt the scheduler’s packet-based CIR and rate PIR to frame-based values. The frame-based values are then used to distribute the port-based bandwidth down to the policer and queue level.

8.1.3.4 Aggregate rate limits for directly attached queues

When all policers and queues for a SAP, multiservice site or subscriber or multiservice site instance are attached directly to the port scheduler (using the port-parent command), it is possible to configure an agg-rate limit for the queues. This is beneficial because the port scheduler does not provide a mechanism to enforce an aggregate SLA for a service or subscriber or multiservice site and the agg-rate limit provides

this ability. Queues and policers may be provisioned directly on the port scheduler when it is desirable to manage the congestion at the egress port-based on class priority instead of on a per service object basis.

The agg-rate limit is not supported when one or more policers or queues on the object are attached to an intermediate service scheduler. In this event, it is expected that the intermediate scheduler hierarchy is used to enforce the aggregate SLA. Attaching an agg-rate limit is mutually exclusive to attaching an egress scheduler policy at the SAP or multiservice site profile level. When an aggregate rate limit is in effect, a scheduler policy cannot be assigned. When a scheduler policy is assigned on the egress side of a SAP or multiservice site profile, an agg-rate limit cannot be assigned.

Because the sap-egress policy defines a policer or queue parent association before the policy is associated with a service SAP or multiservice site profile, it is possible for the policy to either not define a port-parent association or define an intermediate scheduler parenting that does not exist. Policers and queues in this state are considered to be orphaned and automatically attached to port scheduler priority 1. Orphaned policers and queues are included in the aggregate rate limiting behavior on the SAP or multiservice site instance they are created within.

8.1.3.5 SAP egress QoS policy queue parenting

A SAP-egress QoS policy policer or queue may be associated with either a port parent or an intermediate scheduler parent. The validity of the parent definition cannot be checked at the time that it is provisioned because the application of the QoS policy is not known until it is applied to an egress SAP, subscriber, or multiservice site profile. Port or intermediate parenting can be decided on a queue-by-queue or policer-by-policer basis, some policers and queues tied directly to the port scheduler priorities while others are attached to intermediate schedulers.

8.1.3.6 Network queue QoS policy queue parenting

A network-queue policy only supports direct port parent priority association. Intermediate schedulers are not supported on network ports or channels.

8.1.3.7 Egress port scheduler overrides

When a port scheduler has been associated with an egress port, it is possible to override the following parameters:

- the max-rate allowed for the scheduler
- the maximum rate for each priority level 8 through 1
- the CIR associated with each priority level 8 through 1

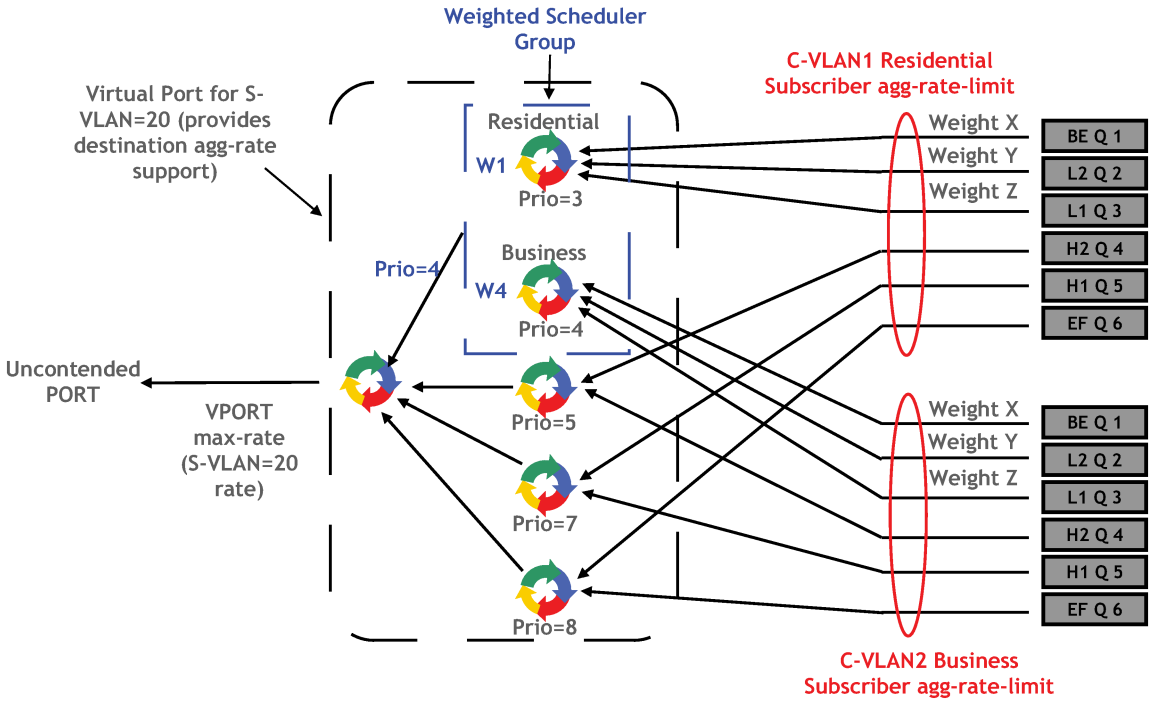
The orphan priority level (level 1) has no configuration parameters and cannot be overridden.

8.1.3.8 Applying a port scheduler policy to a virtual port

To represent a downstream network aggregation node in the local node scheduling hierarchy, a new scheduling node, referred to as virtual port (Vport in CLI) have been introduced. The Vport operates exactly like a port scheduler except multiple Vport objects can be configured on the egress context of an Ethernet port.

Figure 23: Applying a port scheduler policy to a Vport shows the use of the Vport on an Ethernet port of a Broadband Network Gateway (BNG). In this case, the Vport represents a specific downstream DSLAM.

Figure 23: Applying a port scheduler policy to a Vport



The user adds a Vport to an Ethernet port using the following command:

config>port>ethernet>access>egress>vport *vport-name* **create**

The Vport is always configured at the port level even when a port is a member of a LAG. The Vport name is local to the port it is applied to but must be the same for all member ports of a LAG. However, it does not need to be unique globally on a chassis.

The user applies a port scheduler policy to a Vport using the following command:

config>port>ethernet>access>egress>vport>port-scheduler-policy *port-scheduler-policy-name*

A Vport cannot be parented to the port scheduler when the Vport is using a port scheduler policy. It is important that the user ensures that the sum of the **max-rate** parameter value in the port scheduler policies of all Vport instances on a specific egress Ethernet port does not oversubscribe the port's rate. If it does, the scheduling behavior degenerates to that of the H/W scheduler on that port. A Vport that uses an **agg-rate**, or a scheduler-policy, can be parented to a port scheduler. The application of the **agg-rate** *rate*, **port-scheduler-policy**, and **scheduler-policy** commands under a Vport are mutually exclusive.

Each subscriber host or SAP policer or queue is port parented to the Vport that corresponds to the destination DSLAM using the existing **port-parent** command:

```
config>qos>sap-egress>queue>port-parent [weight weight]
 [level level] [cir-weight cir-weight] [cir-level cir-
 level]
```

```
config>qos>sap-egress>policer>port-parent [weight
```

```
weight] [level level] [cir-weight cir-weight] [cir-
level cir-level]
```

This command can parent the policer or queue to either a port or to a Vport. These operations are mutually exclusive in CLI. When parenting to a Vport, the parent Vport for a subscriber-host or SAP policer or queue is not explicitly indicated in the command; it is determined indirectly.

Associate a SAP with a Vport using the following commands.

1. Create a Vport in the port context:
   - **MD-CLI**

     ```
     configure port ethernet access egress virtual-port
     ```

   - **classic CLI**

     ```
     configure port ethernet access egress vport
     ```

2. Associate a SAP with the corresponding Vport:

   ```
   configure service service sap egress virtual-port
   ```

Subscriber host policers or queues, SLA profile schedulers, subscriber profile schedulers, PW SAPs (in IES or VPRN services), and regular SAPs can be parented to a Vport.

8.1.3.9 Weighted scheduler group in a port scheduler policy

The port scheduler policy defines a set of eight priority levels. To allow for the application of a scheduling weight to groups of queues competing at the same priority level of the port scheduler policy applied to the Vport, or to the Ethernet port, a group object is defined under the port scheduler policy, as follows:

config>qos>port-scheduler-policy>group group-name rate pir-rate [cir cir-rate]

Up to eight groups can be defined within each port scheduler policy. One or more levels can map to the same group. A group has a rate and, optionally, a cir-rate, and inherits the highest scheduling priority of its member levels. For example, the scheduler group for the 7705 SAR Gen 2 shown in the Vport in Figure 23: Applying a port scheduler policy to a Vport consists of level priority 3 and level priority 4. Therefore, it inherits priority 4 when competing for bandwidth with the standalone priority levels 8, 7, and 5.

A group receives bandwidth from the port or from the Vport and distributes it within the member levels of the group according to the weight of each level within the group. Each priority level competes for bandwidth within the group based on its weight under congestion situation. If there is no congestion, a priority level can achieve up to its rate (cir-rate) worth of bandwidth.

The mapping of a level to a group is performed as follows:

config>qos>port-scheduler-policy>level priority-level rate pir-rate [cir cir-rate] group group-name [weight weight-in-group]

The CLI enforces that mapping of levels to a group are contiguous. In other words, a user would not be able to add a priority level to group unless the resulting set of priority levels is contiguous.

When a level is not explicitly mapped to any group, it maps directly to the root of the port scheduler at its own priority like in existing behavior.

8.2 Basic configurations

A basic QoS scheduler policy must conform to the following:

- Each QoS scheduler policy must have a unique policy ID.
- A tier level 1 parent scheduler name cannot be configured.

A basic QoS port scheduler policy must conform to the following:

Each QoS port scheduler policy must have a unique policy name.

8.2.1 Creating a QoS scheduler policy

Configuring and applying QoS policies is optional. If no QoS policy is explicitly applied to a SAP or IP interface, a default QoS policy is applied.

To create a scheduler policy, define the following:

- Define a scheduler policy ID value. The system does not dynamically assign a value.
- Include a description. The description provides a brief overview of policy features.
- Specify the tier level. A tier identifies the level of hierarchy that a group of schedulers are associated with.
- Specify a scheduler name. A scheduler defines bandwidth controls that limit each child (other schedulers and queues) associated with the scheduler.
- Specify a parent scheduler name to be associated with a level 1, 2, or 3 tier.
- Optionally, modify the bandwidth that the scheduler can offer its child queues or schedulers. Otherwise, the scheduler is allowed to consume bandwidth without a scheduler-defined limit.

The following displays a scheduler policy configuration:

```
A:ALA-12>config>qos# info
#------------------------------------------
echo "QoS Policy Configuration"
#------------------------------------------
...
    scheduler-policy "SLA1" create
            description "NetworkControl(3), Voice(2) and NonVoice(1) have strict
priorities"
            tier 1
                scheduler "All_traffic" create
                    description "All traffic goes to this scheduler eventually"
                    rate 11000
                exit
            exit
            tier 2
                scheduler "NetworkControl" create
                    description "network control traffic within the VPN"
                    parent All_traffic level 3
                    rate 100
                exit
                scheduler "NonVoice" create
                    description "NonVoice of VPN and Internet traffic will be serviced
by this scheduler"
                    parent All_traffic
                    rate 11000
```

```
                exit
                scheduler "Voice" create
                    description "Any voice traffic from VPN and Internet use this
scheduler"
                    parent All_traffic level 2
                    rate 5500
                exit
            exit
            tier 3
                scheduler "Internet_be" create
                    parent NonVoice
                exit
                scheduler "Internet_priority" create
                    parent NonVoice level 2
                exit
                scheduler "Internet_voice" create
                    parent Voice
                exit
                scheduler "VPN_be" create
                    parent NonVoice
                exit
                scheduler "VPN_nc" create
                    parent NetworkControl
                    rate 100 cir 36
                exit
                scheduler "VPN_priority" create
                    parent NonVoice level 2
                exit
                scheduler "VPN_reserved" create
                    parent NonVoice level 3
                exit
                scheduler "VPN_video" create
                    parent NonVoice level 5
                    rate 1500 cir 1500
                exit
                scheduler "VPN_voice" create
                    parent Voice
                    rate 2500 cir 2500
                exit
            exit
        exit
        sap-ingress 100 create
            description "Used on VPN sap"
...
-------------------------------------------
A:ALA-12>config>qos#
```

8.2.2 Applying scheduler policies

Apply scheduler policies to the entities in subsequent sections.

8.2.2.1 Customer

Use the following CLI syntax to associate a scheduler policy to a customer’s multiservice site:

```
config>customer customer-id
multiservice-site customer-site-name
  egress
    scheduler-policy scheduler-policy-name
```

```
  ingress
    scheduler-policy scheduler-policy-name
```

8.2.2.2 Epipe

Use the following CLI syntax to apply QoS policies to ingress or egress, or both, Epipe SAPs:

```
config>service# epipe service-id [customer customer-id]
  sap sap-id
  egress
    scheduler-policy scheduler-policy-name
  ingress
    scheduler-policy scheduler-policy-name

config>service# epipe service-id [customer customer-id]
  sap sap-id
  egress
    qos sap-egress-policy-id
  ingress
    qos sap-ingress-policy-id
```

The following output displays an Epipe service configuration with SAP scheduler policy SLA2 applied to the SAP ingress and egress.

```
A:SR>config>service# info
----------------------------------------------
        epipe 6 customer 6 vpn 6 create
            description "Distributed Epipe service to west coast"
            sap 1/1/10:0 create
                ingress
                    scheduler-policy "SLA2"
                    qos 100
                exit
                egress
                    scheduler-policy "SLA2"
                    qos 1010
                exit
            exit
...
----------------------------------------------
A:SR>config>service#
```

8.2.2.3 IES

Use the following CLI syntax to apply scheduler policies to ingress or egress, or both, IES SAPs:

```
config>service# ies service-id [customer customer-id]
interface ip-int-name
  sap sap-id
    egress
      scheduler-policy scheduler-policy-name
    ingress
      scheduler-policy scheduler-policy-name
```

The following output displays an IES service configuration with scheduler policy SLA2 applied to the SAP ingress and egress.

```
A:SR>config>service# info
----------------------------------------------
        ies 88 customer 8 vpn 88 create
            interface "Sector A" create
                sap 1/1/1.2.2 create
                    ingress
                        scheduler-policy "SLA2"
                        qos 101
                    exit
                    egress
                        scheduler-policy "SLA2"
                        qos 1020
                    exit
                exit
            exit
            no shutdown
        exit
----------------------------------------------
A:SR>config>service#
```

8.2.2.4 VPLS

Use the following CLI syntax to apply scheduler policies to ingress or egress, or both, VPLS SAPs:

```
config>service# vpls service-id [customer customer-id]
  sap sap-id
   egress
     scheduler-policy scheduler-policy-name
   ingress
     scheduler-policy scheduler-policy-name
```

The following output displays an VPLS service configuration with scheduler policy SLA2 applied to the SAP ingress and egress.

```
A:SR>config>service# info
----------------------------------------------
...
        vpls 700 customer 7 vpn 700 create
            description "test"
            stp
                shutdown
            exit
            sap 1/1/9:0 create
                ingress
                    scheduler-policy "SLA2"
                    qos 100
                exit
                egress
                    scheduler-policy "SLA2"
                exit
            exit
            spoke-sdp 2:222 create
            exit
            mesh-sdp 2:700 create
            exit
            no shutdown
```

```
        exit
...
-------------------------------------------
A:SR>config>service#
```

8.2.2.5 VPRN

Use the following CLI syntax to apply scheduler policies to ingress or egress VPRN SAPs on the 7705 SAR Gen 2:

```
config>service# vprn service-id [customer customer-id]
interface ip-int-name
  sap sap-id
     egress
       scheduler-policy scheduler-policy-name
     ingress
       scheduler-policy scheduler-policy-name
```

The following output displays a VPRN service configuration with the scheduler policy SLA2 applied to the SAP ingress and egress.

```
A:SR7>config>service# info
----------------------------------------------
...
        vprn 1 customer 1 create
            ecmp 8
            autonomous-system 10000
            route-distinguisher 10001:1
            auto-bind-tunnel
                resolution-filter
                resolution-filter ldp
            vrf-target target:10001:1
            interface "to-ce1" create
                address 192.168.0.0/24
                sap 1/1/10:1 create
                    ingress
                        scheduler-policy "SLA2"
                    exit
                    egress
                        scheduler-policy "SLA2"
                    exit
                exit
            exit
            no shutdown
        exit
        epipe 6 customer 6 vpn 6 create
----------------------------------------------
A:SR7>config>service#
```

8.2.3 Creating a QoS port scheduler policy

Configuring and applying QoS port scheduler policies is optional. If no QoS port scheduler policy is explicitly applied to a SAP or IP interface, a default QoS policy is applied.

To create a port scheduler policy, define the following:

- a port scheduler policy name

- a description. The description provides a brief overview of policy features.

Use the following CLI syntax to create a QoS port scheduler policy.

The **create** keyword is included in the command syntax upon creation of a policy.

```
config>qos
port-scheduler-policy scheduler-policy-name [create]
  description description-string
  level priority-level rate pir-rate [cir cir-rate]
  max-rate rate
  orphan-override [level level]
```

The following displays a scheduler policy configuration example:

```
*A:ALA-48>config>qos>port-sched-plcy# info
----------------------------------------------
            description "Test Port Scheduler Policy"
            no orphan-override
----------------------------------------------
*A:ALA-48>config>qos>port-sched-plcy#
```

8.2.4 Configuring port parent parameters

The **port-parent** command defines a child/parent association between an egress queue and a port-based scheduler or between an intermediate service scheduler and a port-based scheduler. The command may be issued in the following contexts:

- **sap-egress>queue** *queue-id*
- **sap-egress>policer** *policer-id*
- **cfg>qos>qgrps>egr>qgrp>queue** *queue-id*
- **network-queue> queue** *queue-id*
- **scheduler-policy>scheduler** *scheduler-name*

The **port-parent** command allows for a set of within-CIR and above-CIR parameters that define the port priority levels and weights for the policer, queue, or scheduler. If the port-parent command is executed without any parameters, the default parameters are assumed.

8.2.4.1 Within-CIR priority level parameters

The within-CIR parameters define which port priority level the policer, queue, or scheduler should be associated with when receiving bandwidth for the policer, queue, or schedulers within-CIR offered load. The within-CIR offered load is the amount of bandwidth the policer, queue, or scheduler could use that is equal to or less than its defined or summed CIR value. The summed value is only valid on schedulers and is the sum of the within-CIR offered loads of the children attached to the scheduler. The parameters that control within-CIR bandwidth allocation are the **port-parent** command **cir-level** and **cir-weight** keywords. The **cir-level** keyword defines the port priority level that the scheduler, policer, or queue uses to receive bandwidth for its within-CIR offered load. The **cir-weight** is used when multiple queues, policers, or schedulers exist at the same port priority level for within-CIR bandwidth. The weight defines the relative ratio that is used to distribute bandwidth at the priority level when more within-CIR offered load exists than the port priority level has bandwidth.

A cir-weight equal to zero (the default value) has special meaning and informs the system that the queue, policer, or scheduler does not receive bandwidth from the within-CIR distribution. Instead, all bandwidth for the queue or scheduler must be allocated in the port scheduler's above-CIR pass.

8.2.4.2 Above-CIR priority level parameters

The above-CIR parameters define which port priority level the policer, queue, or scheduler should be associated with when receiving bandwidth for the above-CIR offered load of the policer, queue, or scheduler. The above-CIR offered load is the amount of bandwidth the queue, policer, or scheduler could use that is equal to or less than its defined PIR value (based on the queue, policer, or scheduler **rate** command) less any bandwidth that was given to the queue, policer, or scheduler during the above-CIR scheduler pass. The parameters that control above-CIR bandwidth allocation are the **port-parent** commands **level** and **weight** keywords. The **level** keyword defines the port priority level that the policer, scheduler, or queue uses to receive bandwidth for its above-CIR offered load. The weight is used when multiple policers, queues, or schedulers exist at the same port priority level for above-CIR bandwidth. The weight defines the relative ratio that is used to distribute bandwidth at the priority level when more above-CIR offered load exists than the port priority level has bandwidth.

```
config>qos# scheduler-policy scheduler-policy-name
tier {1 | 2 | 3}
  scheduler scheduler-name
  port-parent [level priority-level] [weight
    priority-weight] [cir-level cir-priority-level]
    [cir-weight cir-priority-weight]
```

```
config>qos#
sap-egress sap-egress-policy-id [create]
  queue queue-id [{auto-expedite | best-effort |
    expedite}] [priority-mode | profile-mode] [create]
    port-parent [level priority-level] [weight
      priority-weight] [cir-level cir-priority-level]
      [cir-weight cir-priority-weight]
 policer policer-id [create]
   port-parent [level priority-level] [weight
     priority-weight] [cir-level cir-priority level]
     [cir-weight cir-priority-weight]
```

```
config>qos#
network-queue network-queue-policy-name [create]
no network-queue network-queue-policy-name
  queue queue-id [multipoint] [{auto-expedite | best-
    effort | expedite}] [priority-mode | profile-mode]
    [create]
    port-parent [level priority-level] [weight
    priority-weight] [cir-level cir-priority-level]
      [cir-weight cir-priority-weight]
```

8.3 Service management tasks

This section discusses QoS scheduler policy service management tasks.

8.3.1 Deleting QoS policies

There are no scheduler or port-scheduler policies associated with customer or service entities. Removing a scheduler or port-scheduler policy from a multiservice customer site causes the created schedulers to be removed, which makes them unavailable for SAP policers or queues associated with the customer site. Queues or policers that lose their parent scheduler association are deemed to be orphaned and are no longer subject to a virtual scheduler.

A QoS scheduler policy cannot be deleted until it is removed from all customer multiservice sites or service SAPs where it is applied.

```
SR7>config>qos# no scheduler-policy SLA2
MINOR: QoS #1003 The policy has references
SR7>config>qos#
```

8.3.1.1 Removing a QoS policy from a customer multiservice site

Use the following syntax to remove a QoS policy from a customer multiservice site

```
config>service>customer customer-id
multi-service-site customer-site-name
  egress
    no scheduler-policy
  ingress
    no scheduler-policy
```

Example:

```
config>service>customer# multi-service-site “Test”
config>service>cust>multi-service-site# ingress
config>service>cust>multi-service-site>ingress# no
 scheduler-policy
```

8.3.1.2 Removing a QoS policy from SAPs

Use the following syntax to remove a QoS policy from SAPs

```
config>service# {epipe | vpls} service-id [customer
 customer-id]
sap sap-id
   egress
     no scheduler policy
   ingress
     no scheduler policy
```

```
config>service# {ies | vprn} service-id [customer
 customer-id]
interface ip-int-name
  sap sap-id
     egress
       no scheduler policy
     ingress
```

```
        no scheduler policy
```

Example:

```
config>service# epipe 6
config>service>epipe# sap sap 1/1/9:0
config>service>epipe>sap# egress
config>service>epipe>sap>egress# no scheduler-policy
config>service>epipe>sap>egress# exit
config>service>epipe>sap# ingress
config>service>epipe>sap>ingress#
config>service>epipe>sap>ingress# no scheduler-policy
```

```
config>service# vprn 1
config>service>vprn# interface "to-ce1"
config>service>vprn>if# sap 1/1/10:1
config>service>vprn>if>sap# ingress
config>service>vprn>if>sap>ingress# no scheduler-policy
config>service>vprn>if>sap>ingress# exit
config>service>vprn>if>sap# egress
config>service>vprn>if>sap>egress# no scheduler-policy
config>service>vprn>if>sap>egress# exit
config>service>vprn>if>sap#
```

8.3.1.3 Removing a policy from the QoS configuration

To delete a scheduler policy, enter the following command:

config>qos# no scheduler-policy *network-policy-id*

Example:

config>qos# no scheduler-policy *SLA1*

To delete a port scheduler policy, enter the following command:

config>qos# no port-scheduler-policy *network-policy-id*

Example:

config>qos# no port-scheduler-policy *test1*

8.3.2 Copying and overwriting scheduler policies

An existing QoS policy can be copied, renamed with a new QoS policy value, or used to overwrite an existing policy. The overwrite option must be specified or an error occurs if the destination policy exists.

CLI syntax:

config>qos>copy scheduler-policy *src-name dst-name* **[overwrite]**

Example:

config>qos# copy scheduler-policy *SLA1 SLA2*

```
A:SR>config>qos#
...
#------------------------------------------
echo "QoS Policy Configuration"
```

```
#------------------------------------------
        scheduler-policy "SLA1" create
            description "NetworkControl(3), Voice(2) and NonVoice(1) have strict
priorities"
            tier 1
                scheduler "All_traffic" create
                    description "All traffic goes to this scheduler eventually"
                    rate 11000
                exit
            exit
            tier 2
                scheduler "NetworkControl" create
                    description "network control traffic within the VPN"
                    parent "All_traffic" level 3 cir-level 3
                    rate 100
                exit
                scheduler "NonVoice" create
                    description "NonVoice of VPN and Internet traffic will be serviced
by this scheduler"
                    parent "All_traffic" cir-level 1
                    rate 11000
                exit
                scheduler "Voice" create
                    description "Any voice traffic from VPN and Internet use this
scheduler"
                    parent "All_traffic" level 2 cir-level 2
                    rate 5500
                exit
            exit
            tier 3
                scheduler "Internet_be" create
                    parent "NonVoice" cir-level 1
                exit
                scheduler "Internet_priority" create
                    parent "NonVoice" level 2 cir-level 2
                exit
...
        scheduler-policy "SLA2" create
            description "NetworkControl(3), Voice(2) and NonVoice(1) have strict
priorities"
            tier 1
                scheduler "All_traffic" create
                    description "All traffic goes to this scheduler eventually"
                    rate 11000
                exit
            exit
            tier 2
                scheduler "NetworkControl" create
                    description "network control traffic within the VPN"
                    parent "All_traffic" level 3 cir-level 3
                    rate 100
                exit
                scheduler "NonVoice" create
                    description "NonVoice of VPN and Internet traffic will be serviced
by this scheduler"
                    parent "All_traffic" cir-level 1
                    rate 11000
                exit
                scheduler "Voice" create
                    description "Any voice traffic from VPN and Internet use this
scheduler"
                    parent "All_traffic" level 2 cir-level 2
                    rate 5500
                exit
```

```
            exit
            tier 3
                scheduler "Internet_be" create
                    parent "NonVoice" cir-level 1
                exit
                scheduler "Internet_priority" create
                    parent "NonVoice" level 2 cir-level 2
                exit
...
#---------------------------------------
A:SR>config>qos#
```

8.3.3 Editing QoS policies

Existing policies and entries in the CLI can be edited. The changes are applied immediately to all customer multiservice sites and service SAPs where the policy is applied. To prevent configuration errors, use the copy command to make a duplicate of the original policy to a work area, make the edits, then overwrite the original policy.

9 Class fair hierarchical policing (CFHP)

9.1 Overview

CFHP merges the benefits of non-delay rate enforcement inherent to policers with the priority and fairness sensitivity of queuing and scheduling. CFHP is implemented as a group of child policers mapped to a parent policer where the rate enforced by the parent both obeys strict priority levels and is class fair within a priority level. At the parent policer, the output of a lower priority child policer cannot prevent forwarding of packets of a higher priority child policer and when multiple child policers share the same priority level, the system maintains a Fair Information Rate (FIR) for each child that is separate from a child's PIR and CIR rates. Policers can also be used standalone. The parent is optional.

Multiservice sites support policer-control-policy in the in the ingress and egress in addition to scheduler-policy.

Below are the capabilities and limitations for CFHP under a multiservice site:

- Support for SAP only (no subscriber support).
- Assignment is for port only (not for card).
- Supported both in ingress and egress.
- Policer overrides are not supported under a multiservice site.

```
*A:Dut-A>config>service>cust>multi-service-site# pwc
------------------------------------------------------------------------------
Present Working Context :
------------------------------------------------------------------------------
<root>
configure
service
customer 2
multi-service-site "mss1"
------------------------------------------------------------------------------
*A:Dut-A>config>service>cust>multi-service-site# info
----------------------------------------------
assignment port 9/1/4
ingress
policer-control-policy "pcp"
exit
egress
policer-control-policy "pcp"
exit
----------------------------------------------
```

Example of a service using mss is as follows. The sap-egress qos policy "3" has policers parented to arbiters that are configured in the policer-control-policy "pcp" as in the preceding example.

```
*A:Dut-A>config>service>vpls# pwc
------------------------------------------------------------------------------
Present Working Context :
------------------------------------------------------------------------------
```

```
<root>
configure
service
vpls "101"
------------------------------------------------------------------------------
*A:Dut-A>config>service>vpls# info
----------------------------------------------
shutdown
stp
shutdown
exit
sap 9/1/4 create
multi-service-site "mss1"
egress
qos 3
exit
exit
----------------------------------------------
```

9.2 Parent policer priority and unfair sensitive discard thresholds

Priority-level bandwidth control is managed on the parent policer through the use of progressively higher discard thresholds for each in-use priority level. Up to eight priority levels are supported and are individually enabled per parent policer instance based on child policer priority level association. When multiple child policers are associated with a parent policer priority level, two separate discard thresholds are maintained for that priority level. A lower "discard-unfair" threshold ensures that when a child policer has exceeded its FIR rate, its unfair packets are discarded first (assuming the parent policer's bucket depth has reached the priority level's "discard-unfair" threshold) protecting the priority level's fair traffic from the priority level's unfair traffic.

A second "discard-all" threshold is used to discard all remaining packets associated with the priority level in the case where higher priority traffic exists and the sum of both the priority level's traffic and the higher priority traffic exceeds the parent policer rate. This protects the higher priority traffic on the parent policer from being discarded because of lower priority traffic. The child and parent policers operate in an atomic fashion, any conformance effect on a child policer's bucket depth is canceled when the parent policer discards a packet. Policer bucket rate and packet flow interaction with bucket depth are shown in Figure 24: Policer bucket rate and packet flow interaction with bucket depth. Parent policer bucket and priority thresholds are shown in Figure 25: Parent policer bucket and priority thresholds.

Figure 24: Policer bucket rate and packet flow interaction with bucket depth

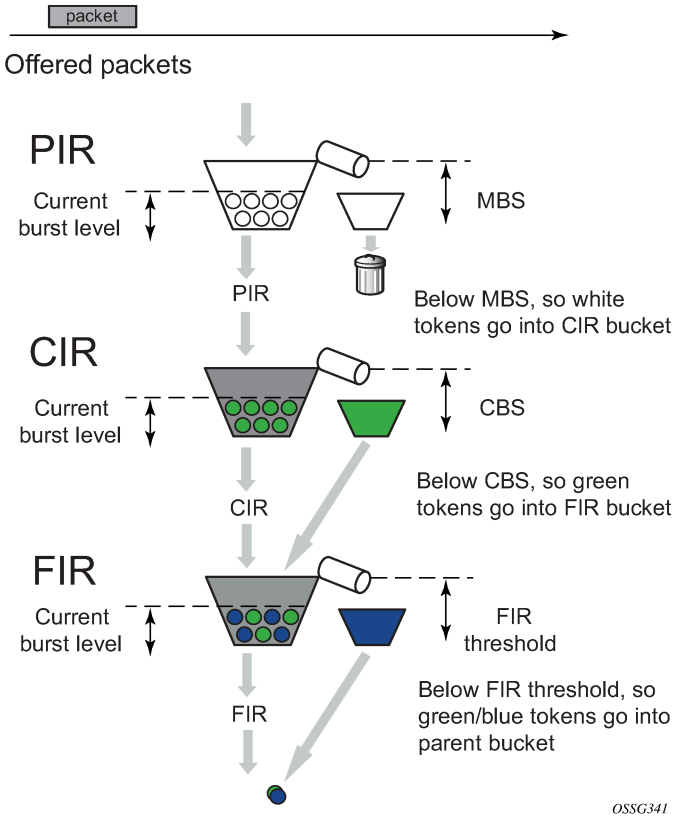
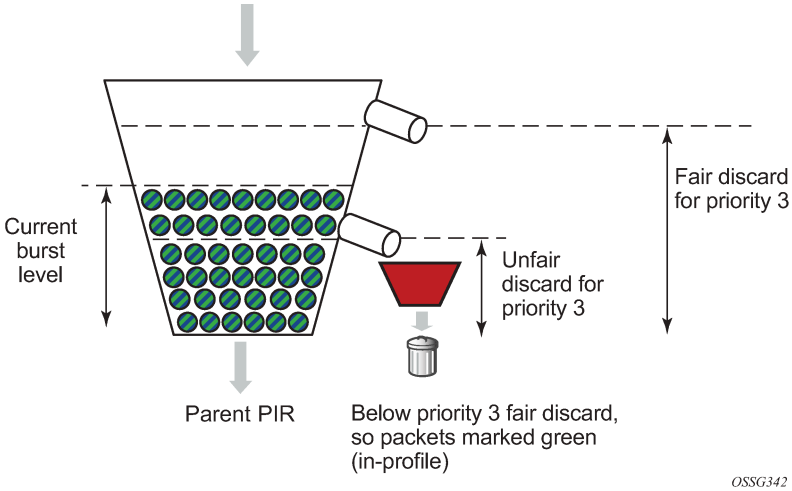


Figure 25: Parent policer bucket and priority thresholds



9.3 CFHP ingress and egress use cases

While ingress CFHP is more common, CFHP may also be used at egress. The reasons for utilizing egress CFHP may be to provide a non-jitter or latency inducing aggregate SLA for multiple ingress flows or just to provide higher scale in the number of egress aggregate SLAs supported.

9.4 Post-CFHP queuing and scheduling

Although CFHP enforces aggregate rate limiting while maintaining sensitivity to strict priority and fair access to bandwidth within a priority, CFHP output packets still require queuing and scheduling to provide access to the switch fabric or to an egress port.

9.4.1 Ingress CFHP queuing

At ingress, CFHP output traffic is automatically mapped to a unicast or multipoint queue in to reach the correct switch fabric destinations. To manage this automatic queuing function, a shared queue policy has been created or exists by default and is named policer-output-queues.

The unicast queues in the policy are automatically created on each destination switch fabric tap and ingress CFHP unicast packets automatically map to one of the queues based on forwarding class and destination tap. 16 multicast paths are supported by default. The multicast paths represent an available multicast switch fabric path; the number of each being controlled using the command:

```
configure mcast-management bandwidth-policy policy-name t2-paths secondary-path
    – number-paths number-of-paths [dual-sfm number-of-paths]
```

For ingress CFHP multicast packets (Broadcast, Unknown unicast, or Multicast—referred to as BUM traffic), the system maintains a conversation hash table per forwarding class and populates the table's forwarding class hash result entry with the one of the multicast paths. Best-effort traffic uses the secondary paths, and expedited traffic uses the primary paths. When a BUM packet is output by ingress CFHP, a conversation hash is performed and used along with the packet's forwarding class to select a hash table entry to derive the multicast path to be used. Each table entry maintains a bandwidth counter that is used to monitor the aggregate traffic per multicast path. The process can be optimized by enabling IMPM on any forwarding complex, which allows the system to redistribute this traffic across the IMPM paths on all forwarding complexes to achieve a more even capacity distribution. Be aware that enabling IMPM causes routed and VPLS (IGMP and PIM) snooped IPv4 multicast groups, and routed and PIM snooped (with **sg-based** forwarding) IPv6 multicast groups to be managed by IMPM.

Any discards performed in the ingress shared queues are reflected in the ingress child policer's discard counters and reported statistics, assuming a discard counter capable stat-mode is configured for the child policer.

9.4.2 Egress CFHP queuing

When CFHP is being performed at egress, queuing of the CFHP output packets is accomplished through egress queue group queues. The system maintains a special egress queue group template (policer-output-queues) that is automatically applied to all Ethernet access ports that are up. The number of queues,

queue types (expedite or best-effort), queue parameters, and the default forwarding class mappings to the queues are managed by the template. On each Ethernet port, the queue parameters may be overridden.

When a SAP egress QoS policy is applied to an Ethernet SAP and the policy contains a forwarding class mapping to a CFHP child policer, the default behavior for queuing the CFHP output is to use the egress Ethernet port's policer-output-queues queue group and the forwarding class mapping within the group to choose the egress queue. Optionally, the SAP egress QoS policy may also explicitly define which egress queue to use within the default queue group or even map the policer output to a different, explicitly created queue group on the port.

Any discards performed in the egress queue group queues are reflected in the egress child policer's discard counters and reported statistics assuming a discard counter capable stat-mode is configured for the child policer. Exceed-profile traffic from the policer is counted as out-of-profile traffic in the egress queue group queues.

9.4.2.1 Policer to local queue mapping

Egress policers can be optionally mapped to a local queue instead of a queue group queue where required.

The syntax for assigning one such egress policer mapped to local queue is as below:

```
*A:Dut-A>config>qos>sap-egress$ pwc
------------------------------------------------------------------------------
Present Working Context :
------------------------------------------------------------------------------
<root>
configure
qos
sap-egress 3 create
------------------------------------------------------------------------------
*A:Dut-A>config>qos>sap-egress$ info
----------------------------------------------
queue 1 create
exit
queue 2 create
exit
policer 2 create
exit
fc ef create
policer 2 queue 2
exit
----------------------------------------------
```

To a local queue, as in "queue 2" in the previous example, both a policer and also a forwarding class can be concurrently mapped as shown below:

```
*A:Dut-A>config>qos>sap-egress$ info
----------------------------------------------
queue 1 create
exit
queue 2 create
exit
policer 2 create
exit
fc af create
queue 2
exit
fc ef create
```

```
policer 2 queue 2
exit
-----------------------------------------------
```

A queue resource is allocated whenever there is either an fc or a policer referencing it. The local queue is freed when there are no references to it. The local queue cannot be deleted when it is being referenced. Exceed-profile traffic from the policer is counted as out-of-profile traffic in the egress local queues.

9.4.3 Egress subscriber CFHP queuing

When a subscriber packet is mapped to a child policer through the SAP egress QoS policy, the actual egress queue group is derived from the subscriber host identification process within the subscriber management module; otherwise, the default queue-group is used.

When egress policed packets are directed to a local SAP queue, and, when this is configured, the output of a **show service id** *id* **sap** *sap-id* **sap-stats** only counts these packets through the policer; they are not counted a second time through the queue to avoid double counting. Consequently, any packets sent directly (not through a policer) to a local SAP post-policer queue are not counted in the sap-stats output. The output of **show service id** *id* **sap** *sap-id* **stats** always counts these packets in both the related policer and queue. If it is required to count packets sent directly to the local SAP post-policer queue in the sap-stats output, the packets could be sent into a policer with the rate set to max, then into the local SAP queue.

9.4.4 SAP default destination string

To simplify subscriber destination string provisioning, a **def-inter-dest-id** command can be used in a sub-sla-mgmt node within a SAP, which allows the definition of a default destination string for all subscribers associated with the SAP. The command also accepts the use-top-q flag that automatically derives the string based on the topmost delineating Dot1Q tag from the SAP's encapsulation.

The command is also supported within the msap-policy allowing similar provisioning behavior for automatically created managed SAPs.

9.5 CFHP policer control policy

Provisioning CFHP requires creating policer control policies (policer-control-policy), and applying a policer control policy to the ingress or egress context of a SAP or to the ingress or egress context of a subscriber profile (sub-profile), much the same way scheduler policies (scheduler-policy) are applied.

Applying a policer control policy to a SAP creates an instance of the policy that is used to control the bandwidth associated with the child policers on the SAP. In a similar fashion, an instance of the policy is created when a subscriber profile associated with the policy is applied to a subscriber context. The subscriber policy instance is used to control the bandwidth of the child policers created by the SLA profile instances within the subscriber context.

Policer control policies can only be applied to SAPs created on Ethernet ports. When the policy instance is created, any policers created on the SAP that have an appropriate parent command defined are considered child policers.

9.5.1 Policer control policy root arbiter

Similar to a scheduler context within a scheduler-policy, the policer-control-policy contains objects called arbiters that control the amount of bandwidth that may be distributed to a set of child policers. Each policer control policy always contains a root arbiter that represents the parent policer. The max-rate defined for the arbiter specifies the decrement rate for the parent policer that governs the overall aggregate rate of every child policer associated with the policy instance. The root arbiter also contains the parent policers MBS configuration parameters that the system uses to individually configure the priority thresholds for each policer instance.

Child policers may parent directly to the root arbiter or to one of the tier 1 or tier 2 explicitly created arbiters.

Each arbiter provides bandwidth to its children using eight strict levels. Children parented at level 8 are first to receive bandwidth. The arbiter continues to distribute bandwidth until either all of its children's bandwidth requirements are met or until the bandwidth it is allowed to distribute is exhausted. The root arbiter is special in that its strict priority levels directly represent the priority thresholds within the parent policer.

9.5.2 Tier 1 and tier 2 explicit arbiters

Other arbiters may be explicitly created in the policy for the purpose of creating an arbitrary bandwidth distribution hierarchy. The explicitly created arbiters must be defined within tier 1 or tier 2 on the policy. Tier 1 arbiters must always be parented by the root arbiter and therefore become a child of the root arbiter. Any child policers directly parented by a tier 1 policer treat the root arbiter as its grandparent. Inversely, the root arbiter considers the child policers as grandchildren. All grandchild policers inherit the priority level of their parent arbiter (the level that the tier 1 arbiter attaches to the root arbiter) within the parent policer.

An arbiter created on tier 2 may be parented by either an arbiter in tier 1 or by the root arbiter. If the tier 2 arbiter is parented by the root arbiter, it is internally treated the same as a tier 1 arbiter and its child policers have a grandchild to grandparent association with the root arbiter.

When a tier 2 arbiter is parented by a tier 1 arbiter, the child policers parented by a tier 2 arbiter are in a great-grandchild to great-grandparent association with the root arbiter. A great-grandchild policer inherits its indirectly parented tier 1 arbiter’s level association with the root arbiter and therefore the parent policer.

A child policer’s priority level on the root arbiter (directly or indirectly) defines which priority level discards thresholds are associated with packets mapped to the child policer for use in the parent policer (assuming the packet is not discarded by its child policer).

9.5.3 Explicit arbiter rate limits

The bandwidth a tier 1 or tier 2 arbiter receives from its parent may be limited by the use of the rate command within the arbiter. When a rate limit is defined for a root arbiter, the system enforces the aggregate rate by calculating a per child policer PIR rate based on the distributed bandwidth per child. This calculated PIR is used to override the child's defined PIR and is represented as the child's operational PIR. The calculated rate is never greater than a child policer's provisioned rate.

9.5.4 CFHP with child policer exceed PIR enabled

A child policer parented to an arbiter can be enabled to forward traffic exceeding its PIR, in which case:

- Traffic exceeding the operational PIR of the child policer is reprofiled to be exceed-profile, where the operational PIR is determined by the H-pol algorithm from the configuration of the policer parent and the associated arbiters (root or intermediate, or both).
- Traffic exceeding the child policer's operational PIR and exceed-profile traffic entering the child policer does not consume capacity from the parent policer (meaning that it does not contribute to the parent policer bucket depth with respect to any of its thresholds).
- Traffic that did not exceed the child policer's operational PIR (when that child is configured with **enable-exceed-pir**) can exceed its parent rate (**max-rate** for the root arbiter) in which case the traffic is forwarded and reprofiled to be exceed-profile and its effect on the child policer is revoked (meaning that it does not contribute to any of the child policer bucket (PIR, CIR, FIR) depths with respect to any of its thresholds).

9.6 CFHP child policer definition and creation

Policers are created within the context of SAP ingress (sap-ingress) and SAP egress (sap-egress) QoS policies. Policer creation in a QoS policy is defined similar to SAP-based queues. A policer is identified using a policer ID. Queues and policers have different ID spaces (both a policer and queue may be defined with ID 1).

The only create time parameter currently available is the unique policer ID within the policy. Policers do not have a scheduling mode (expedite or best-effort), they also do not need to be placed in in-profile-mode to accept traffic from profile in or profile out forwarding classes or subclasses.

All policers within a SAP ingress or egress QoS policy must be explicitly created. No policers are created by default. After a policer is created, forwarding classes or subclasses may be mapped to the policer within the policy. For ingress, each of the individual forwarding types (unicast, multicast, broadcast, and unknown) may be selectively mapped to a policer, policy-created queue or to an ingress port queue group queue. At egress, forwarding classes are not divided into forwarding types, so all packets matched to the forwarding class may be mapped to either a policer, policy-created queue or egress port queue group queue.

Similar to queues, a policer is not created on the SAPs where the policy is applied until at least one forwarding class is mapped to the policer. When the last forwarding class is unmapped from the policer, all the instances of the policer on the SAPs to which the policy is applied are removed.

9.7 Policer enabled SAP QoS policy applicability

Policers are not created on a SAP or multiservice site context until at least one forwarding class has been mapped to the policer. Creating a policer within a QoS policy does not cause policers to be created on the SAPs multiservice sites where the policy is applied.

SAP QoS policy applicability and policy policer forwarding class mappings are dependent on policer resource availability. Attempting to map the first forwarding class to a policer causes the policer to be created on the SAPs multiservice site where the policy is applied. If the forwarding plane where the SAP or multiservice site exists either does not support policers or has insufficient resources to create the policer for the object, the forwarding class mapping fails.

When a forwarding class is successfully mapped to a policer within the policy, attempting to apply the policy to a SAP or a subscriber or multiservice site where the policer cannot be created either because of a lack of policer support or insufficient policer resources fail.

Policing is supported only on Ethernet SAPs or Ethernet-based subscribers. Policing is also only supported on FlexPath2-based systems or IOMs with the exception of CCAG SAPs or subscribers.

9.8 Child policer parent association

Each policer configured within a SAP ingress or SAP egress QoS policy may be configured to be child policer by defining a parent arbiter association using the parent command. If the command is not executed, the policer operates as a stand-alone policer wherever the policy is applied. If the parent command is executed, but the defined arbiter name does not exist within the SAP context or a subscriber or multiservice site context, the policer is treated as an orphan. The SAP or multiservice site context is placed into a degraded state. The system indicates the degraded state by the system setting the ingress-policer-mismatch or egress-policer-mismatch flag for the object. An orphaned policer functions in the same manner as a policer without a parent defined.

An arbiter exists on a SAP when a policer-control-policy containing the arbiter is applied to the appropriate direction (ingress or egress) of the SAP. An arbiter exists on a subscriber when a policer-control-policy containing the arbiter is applied to the subscriber's sub-profile in the appropriate direction as well.

9.9 Profile-capped policers

Profile-capped mode enforces an overall inplus-profile and in-profile burst limit to the CIR bucket for the following packet types:

- ingress undefined
- ingress explicit in-profile
- egress soft-in-profile
- egress explicit inplus
- egress explicit in-profile

The default behavior when profile-capped mode is not enabled is to ignore the CIR output state when an explicit inplus-profile (egress only) and in-profile packet is handled by an ingress or egress policer. The explicit in-profile packets consume CIR tokens up to two times the CBS at which point the bucket stops incrementing and the CIR output for that type of packet enters the non-conforming state. However, the non-conforming state is ignored by the forwarding plane and the packet continues to be handled as inplus-profile or in-profile. Therefore, the total amount of inplus-profile or in-profile traffic can be greater than the configured CIR.

The profile-capped mode makes two changes:

- At egress, soft-in-profile packets (packets received at ingress as in-profile) are treated the same as explicit in-profile packets (unless explicitly reclassified as out-of-profile) and have an initial policer state of in-profile.
- At both ingress and egress, any packets output from the policer with a non-conforming CIR state are treated as out-of-profile (out-of-profile state is ignored for initial in-profile packets when profile-capped mode is not enabled).

A profile-capped policer trusts the in-profile state determined at ingress classification or egress reclassification, so the initial in-profile traffic is preferentially handled with the CIR bucket (two times the

CBS instead of CBS used by undefined or soft-out-of-profile traffic) and the total amount of inplus-profile or in-profile traffic output by the policer cannot exceed the CIR (including initial in-profile traffic).

Profile-capped mode has an effect on stat-mode behavior. Each stat mode has a fixed number of counters in the forwarding plane. The mapping of packets to a counter is also fixed by the offered packet state (profile inplus, profile in, profile out, undefined, soft-in-profile and soft-out-of-profile) in conjunction with the output state of the policer. In the non-capped mode, soft-in-profile is considered undefined while in capped mode it is considered to be equivalent to profile in. Another potential issue with ingress and egress stat-modes is the fact that green packets (that is, those that are profile in at ingress and egress, or soft-in-profile at egress) can turn yellow in the policer output.

Table 17: Effect of profile-capped mode on CIR output describes how the CIR rate and initial profile of each packet affects the output of normal (non-profile-capped) and profile-capped mode policers.

Table 17: Effect of profile-capped mode on CIR output

CIR setting	Initial profile state	Normal mode	Profile-capped mode	Notes
CIR=0	Ingress undefined	Always out-of-profile	Always out-of-profile	When CIR = 0, the CIR has no effect on the packet profile except for ingress-undefined packets. If **profile-capped** is used, this forces all packets to be out-of-profile except for those explicitly reprofiled to exceed-profile.
	Ingress profile in	Always in-profile	Always out-of-profile	
	Ingress profile out	Always out-of-profile	Always out-of-profile	
	Egress soft-in-profile	Always in-profile	Always out-of-profile	
	Egress soft-out-of-profile	Always out-of-profile	Always out-of-profile	
	Egress profile inplus	Always inplus-profile	Always out-of-profile	
	Egress profile in	Always in-profile	Always out-of-profile	
	Egress profile out	Always out-of-profile	Always out-of-profile	
	Egress profile exceed	Always exceed-profile	Always exceed-profile	
CIR=Max/PIR	Ingress undefined	Always in-profile	Always in-profile	CIR never reaches non-conforming state.
	Ingress profile in	Always in-profile	Always in-profile	
	Ingress profile out	Always out-of-profile	Always out-of-profile	

CIR setting	Initial profile state	Normal mode	Profile-capped mode	Notes
	Egress soft-in-profile	Always in-profile	Always in-profile	
	Egress soft-out-of-profile	Always in-profile	Always in-profile	
	Egress profile inplus	Always inplus-profile	Always inplus-profile	
	Egress profile in	Always in-profile	Always in-profile	
	Egress profile out	Always out-of-profile	Always out-of-profile	
	Egress profile exceed	Always exceed-profile	Always exceed-profile	
0 < CIR < PIR	Ingress undefined	In-profile below CBS Out-of-profile at or above CBS	In-profile below CBS Out-of-profile at or above CBS	
	Ingress profile in	Always in-profile	In-profile below two times CBS Out-of-profile at or above two times CBS	
	Ingress profile out	Always out-of-profile	Always out-of-profile	
	Egress soft-in-profile	In-profile below CBS Out-of-profile at or above CBS	In-profile below two times CBS Out-of-profile at or above two times CBS	
	Egress soft-out-of-profile	In-profile below CBS Out-of-profile at or above CBS	In-profile below CBS Out-of-profile at or above CBS	
	Egress profile inplus	Always inplus-profile	Inplus-profile below two times CBS Out-of-profile at or above two times CBS	

CIR setting	Initial profile state	Normal mode	Profile-capped mode	Notes
	Egress profile in	Always in-profile	In-profile below two times CBS Out-of-profile at or above two times CBS	
	Egress Profile Out	Always Out-of-profile	Always Out-of-profile	
	Egress Profile Exceed	Always Exceed-profile	Always Exceed-profile	

9.10 Policer interaction with profile, discard eligibility, and ingress priority

Packets that are offered to an ingress policer may have three different states relative to initial profile:

undefined — either the forwarding class or subclass associated with the packet is not explicitly configured as profile in; profile out or de-1-out-profile is enabled and the dot1p DE bit is set to zero

in-profile — the forwarding class or subclass associated with the packet is configured as profile in

out-of-profile — the forwarding class or subclass associated with the packet is configured as profile out or de-1-out-profile is enabled, and the dot1p DE bit is set to 1

Ingress policed packets are not subject to ingress queue CIR profiling within the ingress policer output queues. While the unicast and multipoint shared queues used by the system for ingress queuing of policed packets may have a CIR rate defined, this CIR rate is only used for rate-based dynamic priority scheduling purposes. The state of the CIR bucket while forwarding a packet from a policer-output-queues shared queue does not alter the packets ingress in-profile or out-of-profile state derived from the ingress policer.

Priority high and low are used in the child policer's PIR leaky bucket to choose one of two discard thresholds (threshold-be-low and threshold-be-high) that are derived from the child policer's MBS and high-priority-only parameters. The high threshold is directly generated by the MBS value. The low threshold is generated by reducing the MBS value by the high-priority-only percentage. A packet's priority is determined while the packet is evaluated against the ingress classification rules in the sap-ingress QoS policy.

Packets that are offered to an egress policer may have six different states relative to their initial profile:

soft-in-profile — the final result at ingress was in-profile and the packet's profile has not been reclassified at egress

soft-out-of-profile — the final result at ingress was out-of-profile and the packet's profile has not been reclassified at egress

hard-inplus-profile — the profile of the packet has been reclassified at egress as profile inplus

hard-in-profile — the profile of the packet has been reclassified at egress as profile in

hard-out-of-profile — the profile of the packet has been reclassified at egress as profile out

hard-exceed-profile — the profile of the packet has been reclassified at egress as profile exceed

When an egress policer's CIR rate is set to 0 (or not defined), the policer has no effect on the profile of packets offered to the policer. An exception to this is when **enable-exceed-pir** is configured under the policer. In this case, the exceed-profile state of a packet takes precedence over the hard-out/in/inplus reclassification, specifically, traffic that is reprofiled to exceed within a SAP egress policer has an exceed-profile state regardless of whether it was reclassified at egress to hard-out, hard-in, or hard-inplus.

Setting a non-zero rate for the egress policer's CIR modifies this behavior for DSCP, IP precedence, dot1p, and DEI egress marking purposes. Hard-inplus-profile and hard-in-profile retain their inherent inplus-profile or in-profile behavior and the hard-out-of-profile and hard-exceed-profile retain their inherent out-of-profile or exceed-profile behavior.

When the egress packet state is soft-in-profile and soft-out-of-profile and the policer's CIR is configured as non-zero, the current CIR state of the policer's CIR bucket overrides the packet's soft profile state. When the policer's CIR is currently conforming, the output is in-profile. When the CIR state is currently exceeding, the output is out-of-profile.

Hard-exceed-profile packets are discarded by default by an egress policer. If **enable-exceed-pir** is configured, the hard-exceed-profile packets are forwarded and, when the PIR state is exceeding, all packets are forwarded with an exceed-profile state.

For egress marking decisions, the hard-inplus-profile, hard-in-profile, and hard-out-of-profile packets ignore the egress policer's CIR state. When the packet state is hard-inplus-profile or hard-in-profile, the in-profile dot1p marking is used, and when DEI marking is enabled for the packet's forwarding class, the exceed-profile traffic is marked 0. When the packet state is hard-out-of-profile or hard-exceed-profile, the out-of-profile dot1p marking is used, unless explicit dot1p exceed-profile marking is configured, in which case the exceed-profile traffic is marked with the configured value, and when DEI marking is enabled for the packets forwarding class, the exceed-profile traffic is marked 1.

The dot1p, outerDot1p, and DEI (when DE marking is configured) reflect the CIR- and PIR-derived packet state. If the **enable-dscp-prec-remarking** command is enabled, the DSCP and IP precedence reflect the CIR- and PIR-derived packet state.

9.10.1 Ingress 'undefined' initial profile

Access ingress packets have one of three initial profile states before processing by the policer:

- undefined
- profile in
- profile out

The SAP ingress QoS policy classification rules map each packet to either a forwarding class or a subclass within a forwarding class. The forwarding class or subclass may be defined as explicit profile in or profile out (the default is no profile). When a packet's forwarding class or subclass is explicitly defined as profile in or profile out, the packet's priority is ignored, and it is not handled by the ingress policer as profile 'undefined'.

See Table 17: Effect of profile-capped mode on CIR output to track the ingress behavior of initial profile and the effect of the CIR bucket on that initial state.

At egress, an ingress policer output of 'in-profile' is treated as 'soft-in-profile' and an ingress policer output of 'out-of-profile' is treated as 'soft-out-of-profile'. Each may be changed by egress profile reclassification or by an egress policer with a CIR rate defined.

9.10.2 Ingress explicitly 'in-profile' state packet handling without profile-capped mode

Packets that are explicitly 'in-profile' remain 'in-profile' in the ingress forwarding plane and are not affected by the ingress policer CIR bucket state when profile-capped mode is not enabled. They do not bypass the policer's CIR leaky bucket but are extended with a greater threshold than the CBS derived 'threshold-bc'. This allows the 'undefined' packets to backfill the remaining conforming CIR bandwidth after accounting for the explicit 'in-profile' packets. This does not prevent the sum of the explicit 'in-profile' from exceeding the configured CIR rate, but it does cause the 'undefined' packets that are marked 'in-profile' to diminish to zero when the combined explicit 'in-profile' rate and 'undefined' rate causes the bucket to reach 'threshold-bc'.

The policer's CIR bucket indicates that the explicit 'in-profile' packets should be marked 'out-of-profile' when the bucket reaches the greater threshold, but this indication is ignored by the ingress forwarding plane. All explicit 'in-profile' packets remain in-profile within the ingress forwarding plane. However, when the packet is received at egress, an ingress 'in-profile' packet is treated as 'soft-in-profile' and the profile may be changed either by explicit profile reclassification or by an egress policer with a CIR rate defined.

Explicit in-profile packets do not automatically use the high-priority threshold ('threshold-be-high') within the child policer's PIR bucket. If preferential burst tolerance is needed for explicit in-profile packets, the packets should also be classified as priority high.

9.10.3 Ingress explicitly 'in-profile' state packet handling with profile-capped mode

When profile-capped mode is enabled, the packet handling behavior defined in Ingress 'undefined' initial profile is altered in one aspect. The CIR output state of yellow at the greater threshold is actually honored and the packet is treated as out-of-profile. The packet is sent to egress in the 'soft-out-of-profile' state in this case.

9.10.4 Ingress explicit 'out-of-profile' state packet handling

Packets that are explicitly 'out-of-profile' remain 'out-of-profile in the ingress forwarding plane. Unlike initially 'in-profile' packets, they do not consume the policer's CIR bucket depth (accomplished by setting the 'threshold-bc' to 0) and therefore do not have an impact on the amount of 'undefined' marked as 'in-profile' by the policer.

While explicit 'out-of-profile' packets remain out-of-profile within the ingress forwarding plane, the egress forwarding plane treats ingress out-of-profile packets as 'soft-out-of-profile' and the profile may be changed either by explicit profile reclassification or by an egress policer with a CIR rate defined.

9.10.5 Egress explicit profile reclassification

An egress profile reclassification overrides the ingress-derived profile of a packet and may set it to hard-inplus-profile, hard-in-profile, hard-out-of-profile, or hard-exceed-profile. A packet that has not been reclassified at egress retains its soft-in-profile or soft-out-of-profile status.

Egress inplus-profile and in-profile (including soft-in-profile and hard-in-profile) packets use the child policer's high threshold-be value within the child policer's PIR bucket while soft-out-of-profile and hard-out-of-profile packets use the child policer's low threshold-be value. Egress hard-exceed-profile packets are

not subject to any threshold control in the child's PIR bucket if **enable-exceed-pir** is configured; otherwise, they are discarded.

9.10.6 Preserving out of profile state at egress policer

Traffic sent through an egress policer with a non-zero CIR is reprofiled by default based on the CIR threshold of the egress policer. To accommodate designs where traffic is set to be out-of-profile at ingress, and the out-of-profile state is required to be maintained by an egress policer, the parameter **profile-out-preserve** can be configured under the egress policer. Explicit egress reclassification to the profile takes precedence over the profile-out-preserve operation.

9.10.7 Egress policer CIR packet handling without profile-capped mode

When an egress policer has been configured with a CIR (maximum or explicit rate other than 0) and profile-capped mode is not enabled, the policer's CIR bucket state overrides the ingress soft-in-profile or soft-out-of-profile state much like the ingress policer handles initial profile undefined packets. If the CIR has not been defined or set to 0 on the egress policer, the egress policer output state is in-profile for soft-in-profile packets and out-of-profile for soft-out-of-profile packets.

If a packet's profile has been reclassified at egress, the new profile classification is handled in the same way as the ingress policer handling of initial in-profile or out-of-profile packets. When a packet has been reclassified as hard-inplus-profile or hard-in-profile, it is applied to the egress policer's CIR bucket using a threshold-bc higher than the threshold-bc derived from the policer's CBS parameter, but the policer output profile state remains inplus-profile or in-profile even if the higher threshold is crossed. When a packet has been reclassified as hard-out-of-profile or hard-exceed-profile, it does not consume the egress policer's CIR bucket depth and the policer output profile state remains out-of-profile or exceed-profile.

9.10.8 Egress policer CIR packet handling with profile-capped mode

When profile-capped mode is enabled, the egress packet handling described in Egress policer CIR packet handling without profile-capped mode is modified in three ways.

First, the soft-in-profile packets received from ingress are handled in the same way as egress explicit profile in reclassification, unless the packet has been reclassified to profile out or profile exceed at egress.

Second, explicit egress inplus-profile, in-profile, and soft-in-profile packets that have not been reclassified to out-of-profile or exceed-profile at egress are allowed to be marked out-of-profile by an egress policer with a CIR not set to 0.

Third, when a policer has a CIR set to 0 (the default rate), all profile-capped packets are treated as out-of-profile independent of the initial profile state, except for exceed-profile packets that remain as exceed-profile.

9.10.9 Forwarding traffic exceeding PIR in egress policers

An egress policer can be configured to forward traffic that enters the policer with an exceed-profile state or exceeds its operational PIR instead of dropping it. The traffic exceeding the PIR is assigned an exceed-profile state. This is supported for any configured (not dynamic) policer in a SAP egress QoS policy, which

can be used for both SAPs and subscribers, and in an egress queue group template that can be used on egress network ports.

When **enable-exceed-pir** is configured under the policer, the exceed-profile state of a packet takes precedence over the hard-out/in/inplus reclassification, specifically, traffic that is reprofiled to exceed within a SAP egress policer has an exceed-profile state regardless of whether it was reclassified at egress to hard-out, hard-in, or hard-inplus.

The **stat-mode offered-total-cir-exceed** command provides forward and drop counters for exceed-profile traffic, as follows:

```
configure
    qos
        queue-group-templates
            egress
                queue-group <queue-group-name> create
                    policer <policer-id> create
                        enable-exceed-pir
                        stat-mode offered-total-cir-exceed
                    exit
                exit
            exit
        exit
        sap-egress <policy-id> create
            policer <policer-id> create
                enable-exceed-pir
                stat-mode offered-total-cir-exceed
            exit
        exit
    exit
exit
```

The dot1p, outer dot1p, DSCP, and precedence can be remarked for the exceed-profile traffic.

9.10.10 Post egress policer packet forwarding class and profile state remapping

Packets processed by a SAP or subscriber egress child policer can have their forwarding class and profile state remapped to a different forwarding class and profile state after exiting the policer. Remapping is achieved by using a post-policer mapping policy containing mapping statements that determine the remapping of packets based on their forwarding class and profile state.

The post-policer mapping policy is configured as follows:

```
configure
    qos
        post-policer-mapping <mapping-policy-name> [create]
            description <description-string>
            fc <fc-name> profile <profile> [create]
                maps-to fc <fc-name> profile <profile>
```

where:

- <mapping-policy-name> is the name of mapping policy, up to 32 characters.
- <description> is the description text string.
- <fc-name> can be af, be, ef, h1, h2, l1, l2, or nc.
- <profile> can be in, out, exceed, or inplus.

Up to 32 mapping statements are supported within a policy, covering the maximum combinations of eight forwarding classes and four profile states. A maximum of seven post-policer mapping policies can be configured per system.

After being configured, the post-policer mapping policy must be applied within a SAP egress QoS policy with, at most, one mapping policy per SAP egress QoS policy:

```
configure
    qos
        sap-egress <policy-id> [create]
            post-policer-mapping <mapping-policy-name>
```

Packets entering a child policer are assigned a forwarding class and profile state. The configuration of the policer can change the profile state of the exiting packet, but not the forwarding class. If a post-policer mapping policy is applied within the SAP egress QoS policy, a packet exiting the policer with a specific forwarding class and profile state can be remapped to a different forwarding class and profile state. For example, consider the following post-policer mapping policy:

```
configure
    qos
        post-policer-mapping ppm1 create
            fc ef profile exceed create
                maps-to fc be profile out
            exit
        exit
```

Packets exiting an egress child policer with forwarding class "ef" and profile state "exceed" have their forwarding class remapped to "be" and their profile state to "out".

The mapping applies to all policers within the SAP egress QoS policy, including regular child policers and policers configured in an IP/IPv6 criterion action statement, except for dynamic policers. The remapping does not affect the policer statistics or the parent policer processing (root arbiter) as it occurs after each of these.

The new forwarding class is used to select the egress queue on which the post-policer traffic is placed. The new profile is used to determine the congestion control handling in that queue and its pool, specifically the drop tail or slope that is applied.

Egress packet remarking is based on the marking configured for the forwarding class and profile of the traffic after being policed but before it is remapped.

Remapping a subset of packets from one forwarding class to another could result in out-of-order packets being received at the destination if there is congestion or different latency characteristics on the paths of the different forwarding classes.

9.10.11 Ingress child policer stat-mode

A policer has multiple types of input traffic and multiple possible output states for each input traffic type. These variations differ between ingress and egress.

For ingress policing, each offered packet has a priority and a profile state. The priority is used by the policer to choose either the high- or low-priority PIR threshold-be. Every offered packet is either priority high or priority low. The offered profile state defines how a packet interacts with the policers CIR bucket state. The combinations of priority and initial profile are as follows:

- offered priority low, undefined profile

- offered priority low, explicit profile in
- offered priority low, explicit profile out
- offered priority high, undefined profile
- offered priority high, explicit profile in
- offered priority high, explicit profile out

Note: When de-1-out-profile is enabled, DEI = 0 is considered as undefined profile and DEI = 1 is considered the same as profile out.

The possible output results for the ingress policer are:

- output green (in-profile)
- output yellow (out-of-profile)
- output red (discard)

To conserve counter resources, the system supports a policer stat-mode command that is used to identify what counters are actually needed for the policer. Not every policer has a CIR defined, so the output green/yellow states do not exist. Also, not every policer has both high- and low-priority or explicit in-profile or out-of-profile offered traffic types. Essentially, the **stat-mode** command allows the counter resources to be allocated based on the accounting needs of the individual policers.

Setting the **stat-mode** does not modify the packet handling behavior of the policer. For example, if the configured stat-mode does not support in-profile and out-of-profile output accounting, the policer is not blocked from having a configured CIR rate. The CIR rate is enforced, but the amount of in-profile and out-of-profile traffic output from the policer is not counted separately (or maybe not at all based on the configured stat-mode).

A policer is created with minimal counters sufficient to provide total offered and total discarded (the total forwarded is computed as the sum of the offered and discarded counters). The **stat-mode** is defined within the **sap-ingress** or **sap-egress** QoS policy in the policer context. When defining the **stat-mode**, the counter resources needed to implement the mode must be available on all forwarding planes where the policer has been created using the QoS policy unless the policer instance has a stat-mode override defined. Use the **tools dump resource-usage card fp** command to see the resources used and available. If insufficient resources exist, the change in the mode fails without any change to the existing counters currently applied to the existing policers. If the QoS policy is being applied to a SAP or multiservice site context and insufficient counter resources exist to implement the configured modes for the policers within the policy, the QoS policy is not applied. For SAPs, this means the previous QoS policy stays in effect. For subscribers, it could mean that the subscriber host event where the QoS policy is being applied fails and the subscriber host may be blocked or removed.

A stat-mode with at least minimal stats is required before the policer can be assigned to a parent arbiter using the parent command.

Successfully changing the stat-mode for a policer causes the counters associated with the policer to reset to zero. Any collected stats on the object the policer is created on also reset to zero.

The system uses the forwarding plane counters to generate accounting statistics and for calculating the operational PIR and FIR rates for a set of children when they are managed by a policer-control-policy. Only the offered counters are used in hierarchical policing rate management. When multiple offered stats are maintained for a child policer, they are summed to derive the total offered rate for each child policer.

All ingress policers have a default CIR value of 0, meaning that by default, all packets except packets classified as profile in is output by the policer as out-of-profile. This may have a negative impact on egress

marking decisions (if in-profile and out-of-profile have different marking values) and on queue congestion handling (WRED or queue drop tail decisions when out-of-profile is less preferred). The following options exist to address this potential issue:

- If all packets handled by the policer must be output as in-profile by the policer, either the packet's forwarding class or subclass can be defined as profile in or the CIR on the policer can be defined as max
- If some packets must be output as in-profile while others output as out-of-profile, three options exist:
  - The CIR may be left at '0' while mapping the packets that must be output as in-profile to a forwarding class or subclass provisioned as profile in.
  - The CIR may be set to max while mapping the packets that must be output as out-of-profile to a forwarding class or subclass provisioned as profile out.
  - Ignore the CIR on the policer and solely rely on the forwarding class or subclass profile provisioning to the correct policer CIR output.

Make sure to use the correct stat mode if the policer's CIR is explicitly not set or is set to 0. The **no-cir** version of the stat-mode must be used when the CIR has a non-zero value. Also, when overriding the policer's CIR mode, make sure to override the stat-mode instance (CIR override can be performed using SNMP access).

Ingress supported stat-modes are:

- no-stats
- minimal - default
- offered-profile-no-cir
- offered-priority-no-cir
- offered-limited-profile-cir
- offered-profile-cir
- offered-priority-cir
- offered-total-cir
- offered-profile-capped-cir
- offered-limited-capped-cir

9.10.12 Egress child policer stat-mode

All packets received on the egress forwarding plane are profiled as either in-profile or out-of-profile. The egress forwarding plane treats the ingress-derived profile as a soft state that may be either overridden by an egress profile reclassification or by a CIR rate enforced by an egress policer.

Egress policers have a default CIR set to 0, but in the egress case a value of 0 disables policer profiling. Egress packets on a CIR-disabled egress policer retain their offered profile state (soft-in-profile, soft-out-of-profile, hard-inplus, hard-in-profile, hard-out-of-profile, or hard-exceed-profile) unless the **enable-exceed-pir** command is configured, in which case the exceed-profile state of a packet takes precedence over the hard-out/in/inplus reclassification.

For egress, the possible types of offered packets include:

- soft offered in-profile (from ingress)
- soft offered out-of-profile (from ingress)

- egress explicit inplus-profile (reclassified at egress)
- egress explicit in-profile (reclassified at egress)
- egress explicit out-of-profile (reclassified at egress)
- egress explicit exceed-profile (reclassified at egress)

The possible output results are:

- output inplus-profile
- output in-profile
- output out-of-profile
- output discard or exceed-profile

The stat-mode command follows the same counter resource rules as ingress.

Egress supported stat-modes are:

- no-stats
- minimal - default
- offered-profile-no-cir
- offered-profile-cir
- offered-total-cir
- offered-limited-capped-cir
- offered-profile-capped-cir
- offered-total-cir-exceed
- offered-four-profile-no-cir
- offered-total-cir-four-profile

Details of the output showing the stat-modes for ingress and egress child policers are in the Class Fair Hierarchical Policing for SAPs section of the *SR OS Advanced Configuration Guide*.

9.11 Profile-preferred mode root policers

Configuring the **profile-preferred** option gives preference for inplus-profile packets or in-profile packets to consume the root policer PIR bucket tokens at a given priority level. This preference applies to packets whose profile is either configured explicitly or set by the output of the child policer CIR bucket.

When this option is selected, all child policers parented to a root policer have their FIR bucket track the state of the CIR bucket. In other words, an inplus-profile or in-profile packet is always blue and an out-of-profile packet is always orange. When admitting packets from the child policers within a specific priority level, orange packets are allowed up to the discard-unfair threshold while blue packets are allowed up to the discard-all threshold. If a child policer is configured to forward traffic exceeding its PIR, the exceed-profile traffic does not contribute to the parent policer bucket depth with respect to any of its thresholds.

The **profile-preferred** option forces the FIR bucket to track the CIR bucket's decrement rate and the threshold chosen for the CIR bucket is also be used in the FIR bucket (instead of using the threshold associated with the PIR bucket).

The inplus/in/out/exceed-profile output from the policer is used for packet marking decisions. The blue/orange child policer input to the parent policer chooses the discard-orange or discard-all thresholds for the child policer's priority level within the parent policer.

Explicit inplus-profile and in-profile packets stay blue up to the high CBS threshold, undefined profile packets stay blue up to the low CBS threshold (1x CBS) and explicit out-of-profile packets are always orange because of a 0 CBS threshold. Orange packets are discarded by the parent policer within the child policer's priority level before the blue packets, preferring blue packets over orange when the discard-orange threshold is crossed.

Use the following CLI syntax to configure the **profile-preferred** option. This option also applies to overrides applied to instances of a policer control policy under a SAP or multiservice site context.

```
config qos
      policer-control-policy policy-name [create]
      no policer-control-policy policy-name
            description "description-string"
            no description
            root
                  max-rate {kilobits-per-second | max}
                  no max-rate
                  [no] profile-preferred
                  priority-mbs-thresholds
                        min-thresh-separation size [bytes | kilobytes]
                        no min-thresh-separation
                        priority level
                              mbs-contribution size [bytes | kilobytes] [fixed]
                              no mbs-contribution
```

The profile-preferred option provides us a way to configure a specific FIR (because it uses the CIR as FIR). In the direct-parented case (no intermediate arbiters present at all) the child policers do not need to have their offered rate polled as each policer always has PIR equal to the min (child PIR, root PIR) and the FIR and CIR are fixed and equal. The child parenting weights are therefore not used. This impacts the show commands, for example, offered rate information is not available. The output of some show commands (for example, **show qos policer-hierarchy card detail**) should be adjusted for profile-preferred configurations.

If an intermediate arbiter is present, then polling is offered at different rates because the child policer PIRs are set based on this information so as to share the intermediate arbiter PIR in proportional to their parenting weight to the intermediate arbiter.

9.12 Child policer hierarchical QoS parenting

Policers can be parented into the QoS hierarchy used for queue and scheduler bandwidth control, referred to as hierarchical QoS (HQoS). This allows the bandwidth of policers, queues, and schedulers to be managed in the same HQoS hierarchy.

HQoS builds a scheduling hierarchy for a queue by parenting it into a scheduler or port scheduler. The schedulers can be parented into other schedulers to create multiple tiers or into a port scheduler that can exist on a Vport or port.

Parenting a policer into HQoS is supported at egress for subscribers and SAPs, with multiservice sites (MSS) supported for SAPs. A post-policer local queue is not supported with HQoS managed policers, nor are queues that are mapped by the **use-fc-mapped-queue** parameter in a criteria action statement.

To enable the parenting of an egress policer into HQoS, the following command is configured in the SAP egress QoS policy:

```
configure
  qos
    sap-egress policy-id
      policers-hqos-manageable
    exit
```

When the **policers-hqos-manageable** command is configured, all policers in the SAP egress QoS policy, except for dynamic policers, can be managed by HQoS. To be managed by HQoS, the policer must be configured with either a **scheduler-parent** or **port-parent** command or be orphaned to an egress port scheduler applied on a Vport or port.

The **no policers-hqos-manageable** command results in policers not being managed by HQoS.

If the **policers-hqos-manageable** command is used, the **parent-location sla**, **policers enable-exceed-pir**, or **policers stat-mode no-stats** commands may not be used within an SAP egress QoS policy.

Egress policers can be parented into a scheduler in a scheduler policy using the **scheduler-parent** command:

```
configure
  qos
    sap-egress policy-id
      policer policer-id
        scheduler-parent scheduler-name [weight weight]
        [level level] [cir-weight cir-weight]
        [cir-level cir-level]
      exit
    exit
```

When a scheduler is specified, no checks are performed as to whether the scheduler exists. If the *scheduler-name* does not exist, the policer is placed in an orphaned operational state. The policer accepts packets but is not bandwidth-limited by a virtual scheduler or the scheduler hierarchy applied to the SAP or subscriber. Consequently, an orphan policer operates in the same way as a non-HQoS-managed policer. On a SAP, the orphan state is indicated in the **show service sap-using** command output with the SapEgressPolicerMismatch flag. This flag is automatically cleared when the *scheduler-name* becomes available on the egress SAP.

The **level** and **cir-level** keywords define the level in the HQoS hierarchy to which the policer parents for the above-CIR and within-CIR bandwidth distribution passes, respectively. If the **cir-level** is set to 0, the policer does not get any bandwidth allocated in the within-CIR pass.

The **weight** and **cir-weight** keywords define the relative weight of this policer in comparison with other child policers, queues, or schedulers at the same level when competing for bandwidth on the parent *scheduler-name* at the above-CIR and within-CIR bandwidth distribution passes, respectively. If the **weight** or **cir-weight** is set to 0, the policer receives bandwidth only after other children with a non-zero weight at this level are serviced.

If **limit-unused-bandwidth** is configured in the HQoS hierarchy to which the policer is parented, only the offered rate increasing in the last sampled period is used to determine that the policer has accumulated work.

Egress policers can also be parented into a port scheduler using the **port-parent** command:

```
configure
  qos
    sap-egress policy-id
```

```
      policer policer-id
        port-parent [weight weight] [level level]
        [cirweight cir-weight] [cir-level cir-level]
```

If the exit port used by the policed traffic is configured with a port scheduler but the policer has neither a scheduler parent nor a port parent, or if it is orphaned (its scheduler parent does not exist), then the policer is fostered by the port scheduler.

When parenting to a port scheduler, the subscriber profile or SAP can use an egress aggregate rate limit to control its traffic rate.

The policer **scheduler-parent** and **port-parent** commands are mutually exclusive; configuring one overrides the other. These commands and with the policer **parent** command (that parents the policer to an arbiter) are also mutually exclusive.

The system does not prevent the configuration of a policer control policy for a SAP, multiservice site, or subscriber using HQoS managed policers. The arbiters for these policer control policies are not used but are allocated, so must be accounted for when considering scaling.

The configuration of **profile-out-preserve** and **profile-capped** is permitted for HQoS policers with these configurations affecting the within-CIR and above-CIR statistics for the HQoS managed policer.

The purpose of parenting a policer to HQoS is to measure the policer traffic in the HQoS hierarchy so that the configured HQoS bandwidth allocation can be enforced. When traffic passes through the policer, it exits through an access egress queue group queue. If the queue group queue was also parented to the same HQoS hierarchy, the policed traffic would be measured twice: one time through the HQoS managed policer, then a second time through a post-policer access egress queue group queue. To prevent the traffic from being measured the second time, the queue group queues must be configured so that they are not managed by HQoS, as follows:

```
configure
  qos
    queue-group-templates
      egress
        queue-group queue-group-name
          no queues-hqos-manageable
```

The default for the **queues-hqos-manageable** command is to allow the queues to be managed by HQoS.

When **no queues-hqos-manageable** is configured, all queues in the egress queue group instances using this template are not managed by HQoS. This command and the configuration of policers and queue **packet-byte-offset** within the egress queue group template are mutually exclusive. The configuration of **no queues-hqos-manageable** is permitted in the default egress **policer-output-queue** queue group template, which avoids the need to create additional queue groups when policers managed by HQoS are used.

When a queue group template with **no queues-hqos-manageable** is configured under a port's Ethernet access egress context, the configuration of an aggregate rate or scheduler policy is not permitted under that context, nor are parent overrides for any of the queues in the queue group. If a port scheduler is configured on the port, the queue group queues are not parented to the port scheduler.

The configuration of an **encap-offset** within the egress of a subscriber profile does not apply to policer traffic that exits through egress queue group queues.

A queue group configured with **no queues-hqos-manageable** should only be used for post-policer traffic from policers in a SAP egress QoS policy configured with **policers-hqos-manageable**. In this case, the traffic is only measured once by HQoS.

Avoid the following scenarios as they may cause HQoS results to be inaccurate:

- Passing traffic through policers in a SAP egress QoS policy configured with **policers-hqos-manageable**, exiting through a queue group queue with its queue group template configured with **queues-hqos-manageable** causes the traffic to be measured twice by HQoS.
- Passing traffic through policers in a SAP egress QoS policy configured with **no policers-hqos-manageable**, exiting through a queue group queue with its queue group template configured with **no queues-hqos-manageable** causes the traffic not to be measured by HQoS.
- Passing traffic that is redirected in a SAP egress QoS policy to the queue group queue that has **no queues-hqos-manageable** configured in its queue group template causes the traffic not to be measured by HQoS.
- Parenting policers to a scheduler policy for SAPs or subscribers active on a distributed mode LAG with member ports on multiple FPs on the same line card causes a policer to be instantiated for each FP on the line card, potentially resulting in a higher share of traffic than intended.

For each of the preceding cases, the following applies:

- for SAPs, a mismatch flag, *SapEgressHQosMgmtMismatch*, is displayed in the **show service id** *service-id* **sap** *sap-id* **detail** command output.
- for subscribers, the **show service active-subscribers detail** command output indicates *Egr hqos mgmt status* : *mismatch* under the SLA profile instance. The output displays *Egr hqos mgmt status* : *enabled* when policer HQoS parenting is correctly configured.

10 Frequently used QoS terms

10.1 Overview

This section provides definitions for frequently used QoS terminology.

The following terms are used in router Hierarchical QoS to describe the operation and maintenance of a virtual scheduler hierarchy and are presented for reference purposes.

10.2 Above-CIR distribution

Above-CIR distribution is the second phase of bandwidth allocation between a parent scheduler and its child queues and child schedulers. The bandwidth that is available to the parent scheduler after the within-CIR distribution is distributed among the child members using each child's level (to define strict priority for the above-CIR distribution), weight (the ratio at a specific level with several children), and the child's rate value. A rate value equal to the child's CIR value results in a child not receiving any bandwidth during the above-CIR distribution phase.

10.3 Available bandwidth

Available bandwidth is the bandwidth usable by a parent scheduler to distribute to its child queues and schedulers. The available bandwidth is limited by the parent's schedulers association with its parent scheduler. If the parent scheduler has a parent of its own and the parent schedulers defined rate value, then available bandwidth is distributed to the child queues and schedulers using a within-CIR distribution phase and an above-CIR distribution phase. Distribution in each phase is based on a combination of the strict priority of each child and the relative weight of the child at that priority level. Separate priority and weight controls are supported per child for each phase.

10.4 CBS

The Committed Burst Size (CBS) specifies the relative amount of reserved buffers for a specific ingress network XMA or MDA forwarding class queue or egress network port forwarding class queue. The value is entered as a percentage.

10.5 CIR

The Committed Information Rate (CIR) defines the amount of bandwidth committed to the scheduler or queue.

- For schedulers, the CIR value can be explicitly defined or derived from summing the child member CIR values.
- On a queue, the CIR value is explicitly defined.

  The CIR rate for ingress queues controls the in-profile and out-of-profile policing and ultimately egress in-profile and out-of-profile marking. Queue CIR rates also define the hardware fairness threshold at which the queue is no longer prioritized over other queues.

A child's (queue or scheduler) CIR is used with the CIR level parameter to determine the child's committed bandwidth from the parent scheduler. When multiple children are at the same strict CIR level, the CIR weight further determines the bandwidth distribution at that level.

10.6 CIR level

The CIR level parameter defines the strict level at which bandwidth is allocated to the child queue or scheduler during the within-CIR distribution phase of bandwidth allocation. All committed bandwidth (determined by the CIR defined for the child) is allocated before any child receives non-committed bandwidth. Bandwidth is allocated to children at the higher CIR levels before children at a lower level. A child CIR value of zero or an undefined CIR level results in bandwidth allocation to the child only after all other children receive their provisioned CIR bandwidth. When multiple children share a CIR level, the CIR weight parameter further defines bandwidth allocation according to the child's weight ratio.

10.7 CIR weight

The CIR weight parameter defines the weight within the CIR level given to a child queue or scheduler. When multiple children share the same CIR level on a parent scheduler, the ratio of bandwidth given to an individual child is dependent on the ratio of the weights of the active children. A child is considered active when a portion of the offered load is within the child's defined CIR rate. The ratio is calculated by first adding the CIR weights of all active children, then dividing each child's CIR weight by the sum. If a child's CIR level parameter is not defined, that child is not included in the within-CIR distribution and the CIR weight parameter is ignored. A CIR weight of zero forces the child to receive bandwidth only after all other children at that level have received their within-CIR bandwidth. When several children share a CIR weight of zero, all are treated equally.

10.8 Child

Child is a logical state of a queue or scheduler that has been configured with a valid parent scheduler association. The child/parent association is used to build the hierarchy among the queues and schedulers.

10.9 Level

The level parameter defines the strict priority level for a child queue or scheduler with regards to bandwidth allocation during the above-CIR distribution phase on the child's parent scheduler. This allocation of bandwidth is done after the within-CIR distribution is finished. All child queues and schedulers receive

the remaining bandwidth according to the strict priority level in which they are defined with higher levels receiving bandwidth first and lower levels receiving bandwidth if available.

10.10 MBS

The Maximum Burst Size (**MBS**) command specifies the relative amount of the buffer pool space for the maximum buffers for a specific ingress network XMA or MDA forwarding class queue or egress network port forwarding class queue. The value is entered as a percentage.

10.11 Offered load

Offered load is evaluated per child in the scheduler hierarchy. The offered load is the amount of bandwidth a child queue or scheduler can use to accommodate the data passing through the child. It is separated into two portions: within-CIR and above-CIR. Within-CIR offered load is the portion of bandwidth required to meet the child's CIR value. It can be less than the CIR value but never greater. If the forwarding requirement for the child is greater than the CIR value, the remaining is considered to be the above-CIR offered load. The sum of the within-CIR and above-CIR offered load cannot be greater than the maximum rate defined for the child.

10.12 Orphan

When a child queue is configured with a parent scheduler specified but the parent scheduler does not exist on the object the queue is created on, the state is considered orphaned.

An orphaned state is not the same condition as when a queue is not defined with a parent association. Orphan states are cleared when the parent scheduler becomes available on the object. This can occur when a scheduler policy containing the parent scheduler name is applied to the object that the queue exists on or when the scheduler name is added to the scheduler policy already applied to the object that the queue exists on.

10.13 Parent

A scheduler becomes a parent when a queue or scheduler defines it as its parent. A queue or scheduler can be a child of only one scheduler. When defining a parent association on a child scheduler, the parent scheduler must already exist in the same scheduler policy and on a scheduler tier higher (numerically lower) than the child scheduler. Parent associations for queues are only checked when, when an instance of the queue is created on a SAP.

10.14 Queue

A queue is where packets that are forwarded are buffered before scheduling. Packets are not actually forwarded through the schedulers; they are forwarded from the queues directly to ingress or egress interfaces. The association between the queue and the virtual schedulers is intended to accomplish

bandwidth allocation to the queue. Because the offered load is derived from queue utilization, bandwidth allocation is dependent on the queue distribution among the scheduler hierarchy. Queues can be tied to only one scheduler within the hierarchy.

10.15 Rate

The rate defines the maximum bandwidth that is made available to the scheduler or queue. The rate is defined in kilobits per second (kb/s).

- On a scheduler, the rate setting is used to limit the total bandwidth allocated to the scheduler's child members.
- For queues, the rate setting is used to define the Peak Information Rate (PIR) at which the queue can operate.

10.16 Root scheduler

A scheduler that has no parent scheduler association (is not a child of another scheduler) is considered to be a root scheduler. With no parent scheduler, bandwidth used by a root scheduler is dependent on offered load of child members, the maximum rate defined for the scheduler, and total overall available bandwidth. Any scheduler can be a root scheduler. Because parent associations are not allowed in Tier 1, all schedulers in Tier 1 are considered be a root scheduler.

10.17 Scheduler policy

A scheduler policy represents a particular grouping of virtual schedulers that are defined in specific scheduler tiers. The tiers and internal parent associations between the schedulers establish the hierarchy among the virtual schedulers. A scheduler policy can be applied to either a multiservice site or to a Service Access Point (SAP). When the policy is applied to a site or SAP, the schedulers in the policy are instantiated on the object and are available for use by child queues directly or indirectly associated with the object.

10.18 Tier

A tier is an organizational configuration used within a scheduler policy to define the place of schedulers created in the policy. Three tiers are supported: Tier 1, Tier 2, and Tier 3. Schedulers defined in Tier 2 can have parental associations with schedulers defined in Tier 1. Schedulers defined in Tier 3 can have parental associations with schedulers defined at Tiers 1 or 2. Queues can have parental associations with schedulers at any tier level.

10.19 Virtual scheduler

A virtual scheduler, defined by a name (text string), is a logical configuration used as a parent to a group of child members that are dependent upon a common parent for bandwidth allocation. The virtual scheduler can also be a child member to another parent virtual scheduler and receive bandwidth from that parent to distribute to its child members.

10.20 Weight

The weight parameter defines the weight within the above-CIR level given to a child queue or scheduler. When several children share the same level on a parent scheduler, the ratio of bandwidth give to an individual child is dependent on the ratio of the weights of the active children. A child is considered active when a portion of the offered load is above the CIR value (also bounded by the child's maximum bandwidth defined by the child's rate parameter). The portion of bandwidth given to each child is based on the child's weight compared to the sum of the weights of all active children at that level. A weight of zero forces the child to receive bandwidth only after all other children at that level have received their above-CIR bandwidth. When several children share a weight of zero, all are treated equally.

10.21 Within-CIR distribution

Within the CIR distribution process is the initial phase of bandwidth allocation between a parent scheduler and its child queues and child schedulers. The bandwidth that is available to the parent scheduler is distributed first among the child members using each child's CIR level (to define a strict priority for the CIR distribution), CIR weight (the ratio at a specific CIR level with several children), and the child's CIR value. A CIR value of zero or an undefined CIR level causes a child to not receive any bandwidth during the CIR distribution phase. If the parent scheduler has any bandwidth remaining after the within-CIR distribution phase, it is distributed using the above-CIR distribution phase.

11 Standards and protocol support

Note:
The information provided in this chapter is subject to change without notice and may not apply to all platforms.

Nokia assumes no responsibility for inaccuracies.

11.1 Bidirectional Forwarding Detection (BFD)

RFC 5880, *Bidirectional Forwarding Detection (BFD)*

RFC 5881, *Bidirectional Forwarding Detection (BFD) IPv4 and IPv6 (Single Hop)*

RFC 5882, *Generic Application of Bidirectional Forwarding Detection (BFD)*

11.2 Border Gateway Protocol (BGP)

draft-hares-idr-update-attrib-low-bits-fix-01, *Update Attribute Flag Low Bits Clarification*

draft-ietf-idr-add-paths-guidelines-08, *Best Practices for Advertisement of Multiple Paths in IBGP*

draft-ietf-idr-best-external-03, *Advertisement of the best external route in BGP*

draft-ietf-idr-bgp-gr-notification-01, *Notification Message support for BGP Graceful Restart*

draft-ietf-idr-bgp-optimal-route-reflection-10, *BGP Optimal Route Reflection (BGP-ORR)*

draft-ietf-idr-error-handling-03, *Revised Error Handling for BGP UPDATE Messages*

draft-ietf-idr-link-bandwidth-03, *BGP Link Bandwidth Extended Community*

RFC 1772, *Application of the Border Gateway Protocol in the Internet*

RFC 1997, *BGP Communities Attribute*

RFC 2385, *Protection of BGP Sessions via the TCP MD5 Signature Option*

RFC 2439, *BGP Route Flap Damping*

RFC 2545, *Use of BGP-4 Multiprotocol Extensions for IPv6 Inter-Domain Routing*

RFC 2858, *Multiprotocol Extensions for BGP-4*

RFC 2918, *Route Refresh Capability for BGP-4*

RFC 4271, *A Border Gateway Protocol 4 (BGP-4)*

RFC 4360, *BGP Extended Communities Attribute*

RFC 4364, *BGP/MPLS IP Virtual Private Networks (VPNs)*

RFC 4456, *BGP Route Reflection: An Alternative to Full Mesh Internal BGP (IBGP)*

RFC 4486, *Subcodes for BGP Cease Notification Message*

RFC 4659, *BGP-MPLS IP Virtual Private Network (VPN) Extension for IPv6 VPN*

RFC 4684, *Constrained Route Distribution for Border Gateway Protocol/MultiProtocol Label Switching (BGP/MPLS) Internet Protocol (IP) Virtual Private Networks (VPNs)*

RFC 4724, *Graceful Restart Mechanism for BGP* – helper mode

RFC 4760, *Multiprotocol Extensions for BGP-4*

RFC 4798, *Connecting IPv6 Islands over IPv4 MPLS Using IPv6 Provider Edge Routers (6PE)*

RFC 5004, *Avoid BGP Best Path Transitions from One External to Another*

RFC 5065, *Autonomous System Confederations for BGP*

RFC 5291, *Outbound Route Filtering Capability for BGP-4*

RFC 5396, *Textual Representation of Autonomous System (AS) Numbers* – asplain

RFC 5492, *Capabilities Advertisement with BGP-4*

RFC 5668, *4-Octet AS Specific BGP Extended Community*

RFC 6286, *Autonomous-System-Wide Unique BGP Identifier for BGP-4*

RFC 6368, *Internal BGP as the Provider/Customer Edge Protocol for BGP/MPLS IP Virtual Private Networks (VPNs)*

RFC 6793, *BGP Support for Four-Octet Autonomous System (AS) Number Space*

RFC 6810, *The Resource Public Key Infrastructure (RPKI) to Router Protocol*

RFC 6811, *Prefix Origin Validation*

RFC 6996, *Autonomous System (AS) Reservation for Private Use*

RFC 7311, *The Accumulated IGP Metric Attribute for BGP*

RFC 7606, *Revised Error Handling for BGP UPDATE Messages*

RFC 7607, *Codification of AS 0 Processing*

RFC 7752, *North-Bound Distribution of Link-State and Traffic Engineering (TE) Information Using BGP*

RFC 7911, *Advertisement of Multiple Paths in BGP*

RFC 7999, *BLACKHOLE Community*

RFC 8092, *BGP Large Communities Attribute*

RFC 8097, *BGP Prefix Origin Validation State Extended Community*

RFC 8212, *Default External BGP (EBGP) Route Propagation Behavior without Policies*

RFC 8277, *Using BGP to Bind MPLS Labels to Address Prefixes*

RFC 9294, *Application-Specific Link Attributes Advertisement Using the Border Gateway Protocol - Link State (BGP LS)*

RFC 9494, *Long-Lived Graceful Restart for BGP*

11.3 Bridging and management

IEEE 802.1AB, *Station and Media Access Control Connectivity Discovery*

IEEE 802.1ad, *Provider Bridges*

IEEE 802.1AX, *Link Aggregation*

IEEE 802.1D, *MAC Bridges*

IEEE 802.1p, *Traffic Class Expediting*

IEEE 802.1Q, *Virtual LANs*

IEEE 802.1s, *Multiple Spanning Trees*

IEEE 802.1w, *Rapid Reconfiguration of Spanning Tree*

11.4 Certificate management

RFC 4210, *Internet X.509 Public Key Infrastructure Certificate Management Protocol (CMP)*

RFC 4211, *Internet X.509 Public Key Infrastructure Certificate Request Message Format (CRMF)*

RFC 5280, *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*

RFC 6712, *Internet X.509 Public Key Infrastructure -- HTTP Transfer for the Certificate Management Protocol (CMP)*

RFC 7030, *Enrollment over Secure Transport*

RFC 7468, *Textual Encodings of PKIX, PKCS, and CMS Structures*

11.5 Ethernet

IEEE 802.3x, *Ethernet Flow Control*

11.6 Ethernet VPN (EVPN)

draft-ietf-bess-evpn-ipvpn-interworking-15, *EVPN Interworking with IPVPN*

draft-ietf-bess-evpn-l3mh-proto-00, *EVPN Multi-Homing support for L3 services*

draft-rbickhart-evpn-ip-mac-proxy-adv-04, *Proxy MAC-IP Advertisement in EVPN*

RFC 7432, *BGP MPLS-Based Ethernet VPN*

RFC 8214, *Virtual Private Wire Service Support in Ethernet VPN*

RFC 8365, *A Network Virtualization Overlay Solution Using Ethernet VPN (EVPN)*

RFC 8560, *Seamless Integration of Ethernet VPN (EVPN) with Virtual Private LAN Service (VPLS) and Their Provider Backbone Bridge (PBB) Equivalents*

RFC 9047, *Propagation of ARP/ND Flags in an Ethernet Virtual Private Network (EVPN)*

RFC 9135, *Integrated Routing and Bridging in Ethernet VPN (EVPN)*

RFC 9136, *IP Prefix Advertisement in Ethernet VPN (EVPN)*

RFC 9161, *Operational Aspects of Proxy ARP/ND in Ethernet Virtual Private Networks*

RFC 9251, *Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) Proxies for Ethernet VPN (EVPN)*

11.7 gRPC Remote Procedure Calls (gRPC)

cert.proto version 0.1.0, *gNOI Certificate Management Service*

file.proto version 0.1.0, *gNOI File Service*

gnmi.proto version 0.8.0, *gNMI Service Specification*

gnmi_ext.proto, *gNMI Commit Confirmed Extension*

gnmi_ext.proto, *gNMI Config Subscription Extension*

gnmi_ext.proto, *gNMI Depth Extension*

system.proto version 1.0.0, *gNOI System Service*

tunnel.proto version 0.2, *gRPC Tunnel Service*

PROTOCOL-HTTP2, *gRPC over HTTP2*

11.8 Intermediate System to Intermediate System (IS-IS)

draft-ietf-isis-mi-02, *IS-IS Multi-Instance*

draft-ietf-lsr-igp-ureach-prefix-announce-01, *IGP Unreachable Prefix Announcement* – without U-Flag and UP-Flag

draft-kaplan-isis-ext-eth-02, *Extended Ethernet Frame Size Support*

ISO/IEC 10589:2002 Second Edition, *Intermediate system to Intermediate system intra-domain routeing information exchange protocol for use in conjunction with the protocol for providing the connectionless-mode Network Service (ISO 8473)*

RFC 1195, *Use of OSI IS-IS for Routing in TCP/IP and Dual Environments*

RFC 2973, *IS-IS Mesh Groups*

RFC 3359, *Reserved Type, Length and Value (TLV) Codepoints in Intermediate System to Intermediate System*

RFC 3719, *Recommendations for Interoperable Networks using Intermediate System to Intermediate System (IS-IS)*

RFC 3787, *Recommendations for Interoperable IP Networks using Intermediate System to Intermediate System (IS-IS)*

RFC 5120, *M-ISIS: Multi Topology (MT) Routing in IS-IS*

RFC 5130, *A Policy Control Mechanism in IS-IS Using Administrative Tags*

RFC 5301, *Dynamic Hostname Exchange Mechanism for IS-IS*

RFC 5302, *Domain-wide Prefix Distribution with Two-Level IS-IS*

RFC 5303, *Three-Way Handshake for IS-IS Point-to-Point Adjacencies*

RFC 5304, *IS-IS Cryptographic Authentication*

RFC 5305, *IS-IS Extensions for Traffic Engineering TE*

RFC 5306, *Restart Signaling for IS-IS* – helper mode

RFC 5308, *Routing IPv6 with IS-IS*

RFC 5309, *Point-to-Point Operation over LAN in Link State Routing Protocols*

RFC 5310, *IS-IS Generic Cryptographic Authentication*

RFC 6213, *IS-IS BFD-Enabled TLV*

RFC 6232, *Purge Originator Identification TLV for IS-IS*

RFC 6233, *IS-IS Registry Extension for Purges*

RFC 7775, *IS-IS Route Preference for Extended IP and IPv6 Reachability*

RFC 7794, *IS-IS Prefix Attributes for Extended IPv4 and IPv6 Reachability* – sections 2.1 and 2.3

RFC 7981, *IS-IS Extensions for Advertising Router Information*

RFC 7987, *IS-IS Minimum Remaining Lifetime*

RFC 8202, *IS-IS Multi-Instance* – single topology

RFC 8570, *IS-IS Traffic Engineering (TE) Metric Extensions* – Min/Max Unidirectional Link Delay metric for flex-algo, RSVP, SR-TE

RFC 8919, *IS-IS Application-Specific Link Attributes*

RFC 9885, *Multi-Part TLVs in IS-IS*

11.9 Internet Protocol (IP) general

RFC 768, *User Datagram Protocol*

RFC 793, *Transmission Control Protocol*

RFC 854, *Telnet Protocol Specifications*

RFC 1350, *The TFTP Protocol (revision 2)*

RFC 2784, *Generic Routing Encapsulation (GRE)*

RFC 3164, *The BSD syslog Protocol*

RFC 4250, *The Secure Shell (SSH) Protocol Assigned Numbers*

RFC 4251, *The Secure Shell (SSH) Protocol Architecture*

RFC 4252, *The Secure Shell (SSH) Authentication Protocol* – publickey, password

RFC 4253, *The Secure Shell (SSH) Transport Layer Protocol*

RFC 4254, *The Secure Shell (SSH) Connection Protocol*

RFC 4511, *Lightweight Directory Access Protocol (LDAP): The Protocol*

RFC 4513, *Lightweight Directory Access Protocol (LDAP): Authentication Methods and Security Mechanisms* – TLS

RFC 4632, *Classless Inter-domain Routing (CIDR): The Internet Address Assignment and Aggregation Plan*

RFC 5082, *The Generalized TTL Security Mechanism (GTSM)*

RFC 5246, *The Transport Layer Security (TLS) Protocol Version 1.2* – TLS client, RSA public key

RFC 5289, *TLS Elliptic Curve Cipher Suites with SHA-256/384 and AES Galois Counter Mode (GCM)*

RFC 5425, *Transport Layer Security (TLS) Transport Mapping for Syslog* – RFC 3164 with TLS

RFC 5656, *Elliptic Curve Algorithm Integration in the Secure Shell Transport Layer* – ECDSA

RFC 5925, *The TCP Authentication Option*

RFC 5926, *Cryptographic Algorithms for the TCP Authentication Option (TCP-AO)*

RFC 6398, *IP Router Alert Considerations and Usage* – MLD

RFC 6528, *Defending against Sequence Number Attacks*

RFC 8446, *The Transport Layer Security (TLS) Protocol Version 1.3*

RFC 8907, *The Terminal Access Controller Access-Control System Plus (TACACS+) Protocol*

11.10 Internet Protocol (IP) multicast

RFC 1112, *Host Extensions for IP Multicasting*

RFC 2236, *Internet Group Management Protocol, Version 2*

RFC 2365, *Administratively Scoped IP Multicast*

RFC 2375, *IPv6 Multicast Address Assignments*

RFC 2710, *Multicast Listener Discovery (MLD) for IPv6*

RFC 3376, *Internet Group Management Protocol, Version 3*

RFC 3446, *Anycast Rendevous Point (RP) mechanism using Protocol Independent Multicast (PIM) and Multicast Source Discovery Protocol (MSDP)*

RFC 3590, *Source Address Selection for the Multicast Listener Discovery (MLD) Protocol*

RFC 3618, *Multicast Source Discovery Protocol (MSDP)*

RFC 3810, *Multicast Listener Discovery Version 2 (MLDv2) for IPv6*

RFC 4541, *Considerations for Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) Snooping Switches*

RFC 4604, *Using Internet Group Management Protocol Version 3 (IGMPv3) and Multicast Listener Discovery Protocol Version 2 (MLDv2) for Source-Specific Multicast*

RFC 4610, *Anycast-RP Using Protocol Independent Multicast (PIM)*

RFC 4611, *Multicast Source Discovery Protocol (MSDP) Deployment Scenarios*

RFC 5059, *Bootstrap Router (BSR) Mechanism for Protocol Independent Multicast (PIM)*

RFC 5186, *Internet Group Management Protocol Version 3 (IGMPv3) / Multicast Listener Discovery Version 2 (MLDv2) and Multicast Routing Protocol Interaction*

RFC 5384, *The Protocol Independent Multicast (PIM) Join Attribute Format*

RFC 7761, *Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)*

RFC 8487, *Mtrace Version 2: Traceroute Facility for IP Multicast*

11.11 Internet Protocol (IP) version 4

RFC 791, *Internet Protocol*

RFC 792, *Internet Control Message Protocol*

RFC 826, *An Ethernet Address Resolution Protocol*

RFC 1034, *Domain Names - Concepts and Facilities*

RFC 1035, *Domain Names - Implementation and Specification*

RFC 1191, *Path MTU Discovery* – router specification

RFC 1519, *Classless Inter-Domain Routing (CIDR): an Address Assignment and Aggregation Strategy*

RFC 1812, *Requirements for IPv4 Routers*

RFC 1918, *Address Allocation for Private Internets*

RFC 2131, *Dynamic Host Configuration Protocol*; Relay only

RFC 2132, *DHCP Options and BOOTP Vendor Extensions* – DHCP

RFC 2401, *Security Architecture for Internet Protocol*

RFC 3021, *Using 31-Bit Prefixes on IPv4 Point-to-Point Links*

RFC 3046, *DHCP Relay Agent Information Option (Option 82)*

RFC 3768, *Virtual Router Redundancy Protocol (VRRP)*

RFC 4884, *Extended ICMP to Support Multi-Part Messages* – ICMPv4 and ICMPv6 Time Exceeded

11.12 Internet Protocol (IP) version 6

RFC 2464, *Transmission of IPv6 Packets over Ethernet Networks*

RFC 2529, *Transmission of IPv6 over IPv4 Domains without Explicit Tunnels*

RFC 3122, *Extensions to IPv6 Neighbor Discovery for Inverse Discovery Specification*

RFC 3315, *Dynamic Host Configuration Protocol for IPv6 (DHCPv6)*

RFC 3587, *IPv6 Global Unicast Address Format*

RFC 3596, *DNS Extensions to Support IP version 6*

RFC 3633, *IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6*

RFC 3736, *Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6*

RFC 3971, *SEcure Neighbor Discovery (SEND)*

RFC 4007, *IPv6 Scoped Address Architecture*

RFC 4191, *Default Router Preferences and More-Specific Routes* – Default Router Preference

RFC 4193, *Unique Local IPv6 Unicast Addresses*

RFC 4291, *Internet Protocol Version 6 (IPv6) Addressing Architecture*

RFC 4443, *Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification*

RFC 4861, *Neighbor Discovery for IP version 6 (IPv6)*

RFC 5095, *Deprecation of Type 0 Routing Headers in IPv6*

RFC 5722, *Handling of Overlapping IPv6 Fragments*

RFC 5798, *Virtual Router Redundancy Protocol (VRRP) Version 3 for IPv4 and IPv6* – IPv6

RFC 5952, *A Recommendation for IPv6 Address Text Representation*

RFC 6164, *Using 127-Bit IPv6 Prefixes on Inter-Router Links*

RFC 8021, *Generation of IPv6 Atomic Fragments Considered Harmful*

RFC 8200, *Internet Protocol, Version 6 (IPv6) Specification*

11.13 Internet Protocol Security (IPsec)

draft-ietf-ipsec-isakmp-mode-cfg-05, *The ISAKMP Configuration Method*

draft-ietf-ipsec-isakmp-xauth-06, *Extended Authentication within ISAKMP/Oakley (XAUTH)*

RFC 2401, *Security Architecture for the Internet Protocol*

RFC 2403, *The Use of HMAC-MD5-96 within ESP and AH*

RFC 2404, *The Use of HMAC-SHA-1-96 within ESP and AH*

RFC 2405, *The ESP DES-CBC Cipher Algorithm With Explicit IV*

RFC 2406, *IP Encapsulating Security Payload (ESP)*

RFC 2407, *IPsec Domain of Interpretation for ISAKMP (IPsec DoI)*

RFC 2408, *Internet Security Association and Key Management Protocol (ISAKMP)*

RFC 2409, *The Internet Key Exchange (IKE)*

RFC 2410, *The NULL Encryption Algorithm and Its Use With IPsec*

RFC 2560, *X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP*

RFC 3526, *More Modular Exponential (MODP) Diffie-Hellman group for Internet Key Exchange (IKE)*

RFC 3566, *The AES-XCBC-MAC-96 Algorithm and Its Use With IPsec*

RFC 3602, *The AES-CBC Cipher Algorithm and Its Use with IPsec*

RFC 3706, *A Traffic-Based Method of Detecting Dead Internet Key Exchange (IKE) Peers*

RFC 3947, *Negotiation of NAT-Traversal in the IKE*

RFC 3948, *UDP Encapsulation of IPsec ESP Packets*

RFC 4106, *The Use of Galois/Counter Mode (GCM) in IPsec ESP*

RFC 4109, *Algorithms for Internet Key Exchange version 1 (IKEv1)*

RFC 4301, *Security Architecture for the Internet Protocol*

RFC 4303, *IP Encapsulating Security Payload*

RFC 4307, *Cryptographic Algorithms for Use in the Internet Key Exchange Version 2 (IKEv2)*

RFC 4308, *Cryptographic Suites for IPsec*

RFC 4434, *The AES-XCBC-PRF-128 Algorithm for the Internet Key Exchange Protocol (IKE)*

RFC 4543, *The Use of Galois Message Authentication Code (GMAC) in IPsec ESP and AH*

RFC 4754, *IKE and IKEv2 Authentication Using the Elliptic Curve Digital Signature Algorithm (ECDSA)*

RFC 4835, *Cryptographic Algorithm Implementation Requirements for Encapsulating Security Payload (ESP) and Authentication Header (AH)*

RFC 4868, *Using HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 with IPsec*

RFC 4945, *The Internet IP Security PKI Profile of IKEv1/ISAKMP, IKEv2 and PKIX*

RFC 5019, *The Lightweight Online Certificate Status Protocol (OCSP) Profile for High-Volume Environments*

RFC 5282, *Using Authenticated Encryption Algorithms with the Encrypted Payload of the IKEv2 Protocol*

RFC 5903, *ECP Groups for IKE and IKEv2*

RFC 5996, *Internet Key Exchange Protocol Version 2 (IKEv2)*

RFC 5998, *An Extension for EAP-Only Authentication in IKEv2*

RFC 6379, *Suite B Cryptographic Suites for IPsec*

RFC 6380, *Suite B Profile for Internet Protocol Security (IPsec)*

RFC 6960, *X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP*

RFC 7296, *Internet Key Exchange Protocol Version 2 (IKEv2)*

RFC 7321, *Cryptographic Algorithm Implementation Requirements and Usage Guidance for Encapsulating Security Payload (ESP) and Authentication Header (AH)*

RFC 7383, *Internet Key Exchange Protocol Version 2 (IKEv2) Message Fragmentation*

RFC 7427, *Signature Authentication in the Internet Key Exchange Version 2 (IKEv2)*

RFC 8784, *Mixing Preshared Keys in the Internet Key Exchange Protocol Version 2 (IKEv2) for Post-quantum Security*

11.14 Label Distribution Protocol (LDP)

draft-pdutta-mpls-ldp-adj-capability-00, *LDP Adjacency Capabilities*

draft-pdutta-mpls-ldp-v2-00, *LDP Version 2*

RFC 3037, *LDP Applicability*

RFC 3478, *Graceful Restart Mechanism for Label Distribution Protocol* – helper mode

RFC 5036, *LDP Specification*

RFC 5283, *LDP Extension for Inter-Area Label Switched Paths (LSPs)*

RFC 5443, *LDP IGP Synchronization*

RFC 5561, *LDP Capabilities*

RFC 5919, *Signaling LDP Label Advertisement Completion*

11.15 Multiprotocol Label Switching (MPLS)

RFC 3031, *Multiprotocol Label Switching Architecture*

RFC 3032, *MPLS Label Stack Encoding*

RFC 3270, *Multi-Protocol Label Switching (MPLS) Support of Differentiated Services* – E-LSP

RFC 3443, *Time To Live (TTL) Processing in Multi-Protocol Label Switching (MPLS) Networks*

RFC 5332, *MPLS Multicast Encapsulations*

RFC 5884, *Bidirectional Forwarding Detection (BFD) for MPLS Label Switched Paths (LSPs)*

RFC 6424, *Mechanism for Performing Label Switched Path Ping (LSP Ping) over MPLS Tunnels*

RFC 7308, *Extended Administrative Groups in MPLS Traffic Engineering (MPLS-TE)*

RFC 7746, *Label Switched Path (LSP) Self-Ping*

11.16 Network Address Translation (NAT)

RFC 4787, *Network Address Translation (NAT) Behavioral Requirements for Unicast UDP*

RFC 5382, *NAT Behavioral Requirements for TCP*

RFC 5508, *NAT Behavioral Requirements for ICMP*

11.17 Network Configuration Protocol (NETCONF)

RFC 5277, *NETCONF Event Notifications*

RFC 6020, *YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)*

RFC 6022, *YANG Module for NETCONF Monitoring*

RFC 6241, *Network Configuration Protocol (NETCONF)*

RFC 6242, *Using the NETCONF Protocol over Secure Shell (SSH)*

RFC 6243, *With-defaults Capability for NETCONF*

RFC 8071, *NETCONF Call Home and RESTCONF Call Home* – NETCONF

RFC 8342, *Network Management Datastore Architecture (NMDA)* – Startup, Candidate, Running and Intended datastores

RFC 8525, *YANG Library*

RFC 8526, *NETCONF Extensions to Support the Network Management Datastore Architecture* – <get-data> operation

11.18 Media sanitization

NIST Special Publication 800-88 Revision 1, *Guidelines for Media Sanitization* – CF, MMC, SSD, SD, USB

11.19 Open Shortest Path First (OSPF)

RFC 1765, *OSPF Database Overflow*

RFC 2328, *OSPF Version 2*

RFC 3101, *The OSPF Not-So-Stubby Area (NSSA) Option*

RFC 3509, *Alternative Implementations of OSPF Area Border Routers*

RFC 3623, *Graceful OSPF Restart Graceful OSPF Restart* – helper mode

RFC 3630, *Traffic Engineering (TE) Extensions to OSPF Version 2*

RFC 4811, *OSPF Out-of-Band Link State Database (LSDB) Resynchronization* – OSPFv2

RFC 4812, *OSPF Restart Signaling* – OSPFv2

RFC 4576, *Using a Link State Advertisement (LSA) Options Bit to Prevent Looping in BGP/MPLS IP Virtual Private Networks (VPNs)*

RFC 4577, *OSPF as the Provider/Customer Edge Protocol for BGP/MPLS IP Virtual Private Networks (VPNs)*

RFC 5185, *OSPF Multi-Area Adjacency*

RFC 5243, *OSPF Database Exchange Summary List Optimization*

RFC 5250, *The OSPF Opaque LSA Option*

RFC 5309, *Point-to-Point Operation over LAN in Link State Routing Protocols*

RFC 5642, *Dynamic Hostname Exchange Mechanism for OSPF*

RFC 6549, *OSPFv2 Multi-Instance Extensions*

RFC 6987, *OSPF Stub Router Advertisement*

RFC 7471, *OSPF Traffic Engineering (TE) Metric Extensions* – Min/Max Unidirectional Link Delay metric for flex-algo, RSVP, SR-TE

RFC 7684, *OSPFv2 Prefix/Link Attribute Advertisement*

RFC 7770, *Extensions to OSPF for Advertising Optional Router Capabilities*

RFC 8920, *OSPF Application-Specific Link Attributes*

11.20 Path Computation Element Protocol (PCEP)

draft-alvarez-pce-path-profiles-04, *PCE Path Profiles*

draft-ietf-pce-binding-label-sid-15, *Carrying Binding Label/Segment Identifier (SID) in PCE-based Networks.* – MPLS binding SIDs

RFC 5440, *Path Computation Element (PCE) Communication Protocol (PCEP)*

RFC 8231, *Path Computation Element Communication Protocol (PCEP) Extensions for Stateful PCE*

RFC 8253, *PCEPS: Usage of TLS to Provide a Secure Transport for the Path Computation Element Communication Protocol (PCEP)*

RFC 8281, *PCEP Extensions for PCE-initiated LSP Setup in a Stateful PCE Model*

RFC 8408, *Conveying Path Setup Type in PCE Communication Protocol (PCEP) Messages*

RFC 8664, *Path Computation Element Communication Protocol (PCEP) Extensions for Segment Routing*

11.21 Pseudowire (PW)

draft-ietf-l2vpn-vpws-iw-oam-04, *OAM Procedures for VPWS Interworking*

MFA Forum 12.0.0, *Multiservice Interworking - Ethernet over MPLS*

MFA Forum 13.0.0, *Fault Management for Multiservice Interworking v1.0*

MFA Forum 16.0.0, *Multiservice Interworking - IP over MPLS*

RFC 3916, *Requirements for Pseudo-Wire Emulation Edge-to-Edge (PWE3)*

RFC 3985, *Pseudo Wire Emulation Edge-to-Edge (PWE3)*

RFC 4385, *Pseudo Wire Emulation Edge-to-Edge (PWE3) Control Word for Use over an MPLS PSN*

RFC 4446, *IANA Allocations for Pseudowire Edge to Edge Emulation (PWE3)*

RFC 4447, *Pseudowire Setup and Maintenance Using the Label Distribution Protocol (LDP)*

RFC 4448, *Encapsulation Methods for Transport of Ethernet over MPLS Networks*

RFC 5085, *Pseudowire Virtual Circuit Connectivity Verification (VCCV): A Control Channel for Pseudowires*

RFC 5659, *An Architecture for Multi-Segment Pseudowire Emulation Edge-to-Edge*

RFC 5885, *Bidirectional Forwarding Detection (BFD) for the Pseudowire Virtual Circuit Connectivity Verification (VCCV)*

RFC 6073, *Segmented Pseudowire*

RFC 6310, *Pseudowire (PW) Operations, Administration, and Maintenance (OAM) Message Mapping*

RFC 6391, *Flow-Aware Transport of Pseudowires over an MPLS Packet Switched Network*

RFC 6575, *Address Resolution Protocol (ARP) Mediation for IP Interworking of Layer 2 VPNs*

RFC 6718, *Pseudowire Redundancy*

RFC 6829, *Label Switched Path (LSP) Ping for Pseudowire Forwarding Equivalence Classes (FECs) Advertised over IPv6*

RFC 6870, *Pseudowire Preferential Forwarding Status bit*

RFC 7023, *MPLS and Ethernet Operations, Administration, and Maintenance (OAM) Interworking*

RFC 7267, *Dynamic Placement of Multi-Segment Pseudowires*

RFC 7392, *Explicit Path Routing for Dynamic Multi-Segment Pseudowires* – ER-TLV and ER-HOP IPv4 Prefix

RFC 8395, *Extensions to BGP-Signaled Pseudowires to Support Flow-Aware Transport Labels*

11.22 Quality of Service (QoS)

RFC 2430, *A Provider Architecture for Differentiated Services and Traffic Engineering (PASTE)*

RFC 2474, *Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers*

RFC 2597, *Assured Forwarding PHB Group*

RFC 3140, *Per Hop Behavior Identification Codes*

RFC 3246, *An Expedited Forwarding PHB (Per-Hop Behavior)*

11.23 Remote Authentication Dial In User Service (RADIUS)

draft-oscca-cfrg-sm3-02, *The SM3 Cryptographic Hash Function*

RFC 2865, *Remote Authentication Dial In User Service (RADIUS)*

RFC 2866, *RADIUS Accounting*

RFC 3162, *RADIUS and IPv6*

RFC 6613, *RADIUS over TCP* – with TLS

RFC 6614, *Transport Layer Security (TLS) Encryption for RADIUS*

RFC 6929, *Remote Authentication Dial-In User Service (RADIUS) Protocol Extensions*

11.24 Resource Reservation Protocol - Traffic Engineering (RSVP-TE)

RFC 2702, *Requirements for Traffic Engineering over MPLS*

RFC 2747, *RSVP Cryptographic Authentication*

RFC 2961, *RSVP Refresh Overhead Reduction Extensions*

RFC 3097, *RSVP Cryptographic Authentication -- Updated Message Type Value*

RFC 3209, *RSVP-TE: Extensions to RSVP for LSP Tunnels*

RFC 4124, *Protocol Extensions for Support of Diffserv-aware MPLS Traffic Engineering*

RFC 4125, *Maximum Allocation Bandwidth Constraints Model for Diffserv-aware MPLS Traffic Engineering*

RFC 4127, *Russian Dolls Bandwidth Constraints Model for Diffserv-aware MPLS Traffic Engineering*

RFC 4561, *Definition of a Record Route Object (RRO) Node-Id Sub-Object*

RFC 5817, *Graceful Shutdown in MPLS and Generalized MPLS Traffic Engineering Networks*

11.25 Routing Information Protocol (RIP)

RFC 1058, *Routing Information Protocol*

RFC 2080, *RIPng for IPv6*

RFC 2082, *RIP-2 MD5 Authentication*

RFC 2453, *RIP Version 2*

11.26 Segment Routing (SR)

RFC 8287, *Label Switched Path (LSP) Ping/Traceroute for Segment Routing (SR) IGP-Prefix and IGP-Adjacency Segment Identifiers (SIDs) with MPLS Data Planes*

RFC 8426, *Recommendations for RSVP-TE and Segment Routing (SR) Label Switched Path (LSP) Coexistence*

RFC 8476, *Signaling Maximum SID Depth (MSD) Using OSPF* – node MSD

RFC 8491, *Signaling Maximum SID Depth (MSD) Using IS-IS* – node MSD

RFC 8660, *Segment Routing with the MPLS Data Plane*

RFC 8661, *Segment Routing MPLS Interworking with LDP*

RFC 8665, *OSPF Extensions for Segment Routing*

RFC 8667, *IS-IS Extensions for Segment Routing*

RFC 8669, *Segment Routing Prefix Segment Identifier Extensions for BGP*

RFC 9256, *Segment Routing Policy Architecture*

RFC 9350, *IGP Flexible Algorithm*

11.27 Simple Network Management Protocol (SNMP)

draft-blumenthal-aes-usm-04, *The AES Cipher Algorithm in the SNMP's User-based Security Model* – CFB128-AES-192 and CFB128-AES-256

draft-ietf-isis-wg-mib-06, *Management Information Base for Intermediate System to Intermediate System (IS-IS)*

draft-ietf-mpls-ldp-mib-07, *Definitions of Managed Objects for the Multiprotocol Label Switching, Label Distribution Protocol (LDP)*

draft-ietf-mpls-lsr-mib-06, *Multiprotocol Label Switching (MPLS) Label Switching Router (LSR) Management Information Base Using SMIv2*

draft-ietf-mpls-te-mib-04, *Multiprotocol Label Switching (MPLS) Traffic Engineering Management Information Base*

draft-ietf-ospf-mib-update-08, *OSPF Version 2 Management Information Base*

draft-ietf-vrrp-unified-mib-06, *Definitions of Managed Objects for the VRRP over IPv4 and IPv6* – IPv6

ESO-CONSORTIUM-MIB revision 200406230000Z, *esoConsortiumMIB*

IANA-ADDRESS-FAMILY-NUMBERS-MIB revision 200203140000Z, *ianaAddressFamilyNumbers*

IANAifType-MIB revision 200505270000Z, *ianaifType*

IANA-RTPROTO-MIB revision 200009260000Z, *ianaRtProtoMIB*

IEEE8021-CFM-MIB revision 200706100000Z, *ieee8021CfmMib*

IEEE8021-PAE-MIB revision 200101160000Z, *ieee8021paeMIB*

IEEE8023-LAG-MIB revision 200006270000Z, *lagMIB*

LLDP-MIB revision 200505060000Z, *lldpMIB*

RFC 1157, *A Simple Network Management Protocol (SNMP)*

RFC 1212, *Concise MIB Definitions*

RFC 1215, *A Convention for Defining Traps for use with the SNMP*

RFC 1724, *RIP Version 2 MIB Extension*

RFC 1901, *Introduction to Community-based SNMPv2*

RFC 2021, *Remote Network Monitoring Management Information Base Version 2 using SMIv2*

RFC 2206, *RSVP Management Information Base using SMIv2*

RFC 2578, *Structure of Management Information Version 2 (SMIv2)*

RFC 2579, *Textual Conventions for SMIv2*

RFC 2580, *Conformance Statements for SMIv2*

RFC 2787, *Definitions of Managed Objects for the Virtual Router Redundancy Protocol*

RFC 2819, *Remote Network Monitoring Management Information Base*

RFC 2856, *Textual Conventions for Additional High Capacity Data Types*

RFC 2863, *The Interfaces Group MIB*

RFC 2864, *The Inverted Stack Table Extension to the Interfaces Group MIB*

RFC 2933, *Internet Group Management Protocol MIB*

RFC 3014, *Notification Log MIB*

RFC 3273, *Remote Network Monitoring Management Information Base for High Capacity Networks*

RFC 3410, *Introduction and Applicability Statements for Internet Standard Management Framework*

RFC 3430, *Simple Network Management Protocol (SNMP) over Transmission Control Protocol (TCP) Transport Mapping*

RFC 3411, *An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks*

RFC 3412, *Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)*

RFC 3413, *Simple Network Management Protocol (SNMP) Applications*

RFC 3414, *User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)*

RFC 3415, *View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*

RFC 3416, *Version 2 of the Protocol Operations for the Simple Network Management Protocol (SNMP)*

RFC 3417, *Transport Mappings for the Simple Network Management Protocol (SNMP)* – SNMP over UDP over IPv4

RFC 3418, *Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*

RFC 3419, *Textual Conventions for Transport Addresses*

RFC 3434, *Remote Monitoring MIB Extensions for High Capacity Alarms*

RFC 3584, *Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework*

RFC 3593, *Textual Conventions for MIB Modules Using Performance History Based on 15 Minute Intervals*

RFC 3635, *Definitions of Managed Objects for the Ethernet-like Interface Types*

RFC 3826, *The Advanced Encryption Standard (AES) Cipher Algorithm in the SNMP User-based Security Model*

RFC 3877, *Alarm Management Information Base (MIB)*

RFC 4001, *Textual Conventions for Internet Network Addresses*

RFC 4022, *Management Information Base for the Transmission Control Protocol (TCP)*

RFC 4113, *Management Information Base for the User Datagram Protocol (UDP)*

RFC 4273, *Definitions of Managed Objects for BGP-4*

RFC 4292, *IP Forwarding Table MIB*

RFC 4293, *Management Information Base for the Internet Protocol (IP)*

RFC 4878, *Definitions and Managed Objects for Operations, Administration, and Maintenance (OAM) Functions on Ethernet-Like Interfaces*

RFC 7420, *Path Computation Element Communication Protocol (PCEP) Management Information Base (MIB) Module*

RFC 7630, *HMAC-SHA-2 Authentication Protocols in the User-based Security Model (USM) for SNMPv3*

11.28 Timing

RFC 3339, *Date and Time on the Internet: Timestamps*

RFC 5905, *Network Time Protocol Version 4: Protocol and Algorithms Specification*

RFC 8573, *Message Authentication Code for the Network Time Protocol*

11.29 Two-Way Active Measurement Protocol (TWAMP)

RFC 5357, *A Two-Way Active Measurement Protocol (TWAMP)* – server, unauthenticated mode

RFC 5938, *Individual Session Control Feature for the Two-Way Active Measurement Protocol (TWAMP)*

RFC 6038, *Two-Way Active Measurement Protocol (TWAMP) Reflect Octets and Symmetrical Size Features*

RFC 8545, *Well-Known Port Assignments for the One-Way Active Measurement Protocol (OWAMP) and the Two-Way Active Measurement Protocol (TWAMP)* – TWAMP

RFC 8762, *Simple Two-Way Active Measurement Protocol* – unauthenticated

RFC 8972, *Simple Two-Way Active Measurement Protocol Optional Extensions* – unauthenticated

RFC 9503, *Simple Two-Way Active Measurement Protocol (STAMP) Extensions for Segment Routing Networks* – excluding Sections 3, 4.1.2 and 4.1.3

11.30 Virtual Private LAN Service (VPLS)

RFC 4761, *Virtual Private LAN Service (VPLS) Using BGP for Auto-Discovery and Signaling*

RFC 4762, *Virtual Private LAN Service (VPLS) Using Label Distribution Protocol (LDP) Signaling*

RFC 5501, *Requirements for Multicast Support in Virtual Private LAN Services*

RFC 6074, *Provisioning, Auto-Discovery, and Signaling in Layer 2 Virtual Private Networks (L2VPNs)*

RFC 7117, *Multicast in Virtual Private LAN Service (VPLS)*

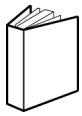
11.31 Yet Another Next Generation (YANG)

RFC 6991, *Common YANG Data Types*

RFC 7950, *The YANG 1.1 Data Modeling Language*

RFC 7951, *JSON Encoding of Data Modeled with YANG*

Customer document and product support



Customer documentation

Customer documentation welcome page

Technical support

Product support portal

Documentation feedback

Customer documentation feedback