Class Fair Hierarchical Policing (CFHP)

In This Section

This section provides information to configure CFHP QoS policies using the command line interface.

Topics in this section include:

Introduction

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.

With 9.0R1, multi-service 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 multi-service-site:

- Support for SAP only (no subscribers support)
- Assignment is for port only (not for card)
- Supported both in Ingress and Egress
- Policer Overrides are not supported under a multi-service-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 below:

```
*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
----------------------------------------------
```

Here the above mentioned sap-egress qos policy "3" will have policers parented to arbiters which are configured in the policer-control-policy "pcp" as in example above.

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 due to lower priority traffic. The child and parent policers operate in an atomic fashion, any conform effect on a child policer's bucket depth is canceled when the parent policer discards a packet. See Figure 34 for a description of policer bucket rate and packet flow interaction with bucket depth. See Figure 35 for a description of parent policer bucket and priority thresholds.

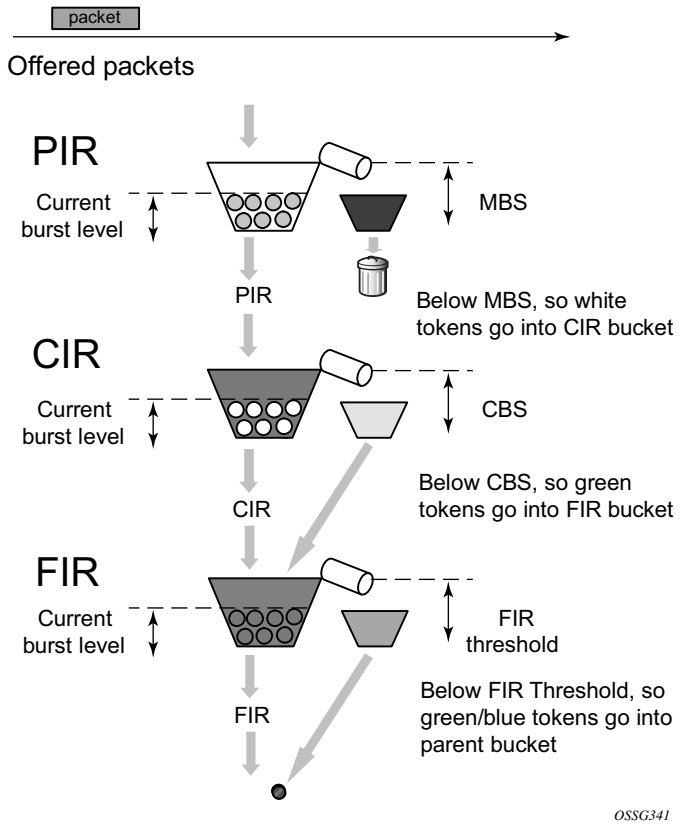

Figure 34: Policer Bucket Rate and Packet Flow Interaction with Bucket Depth

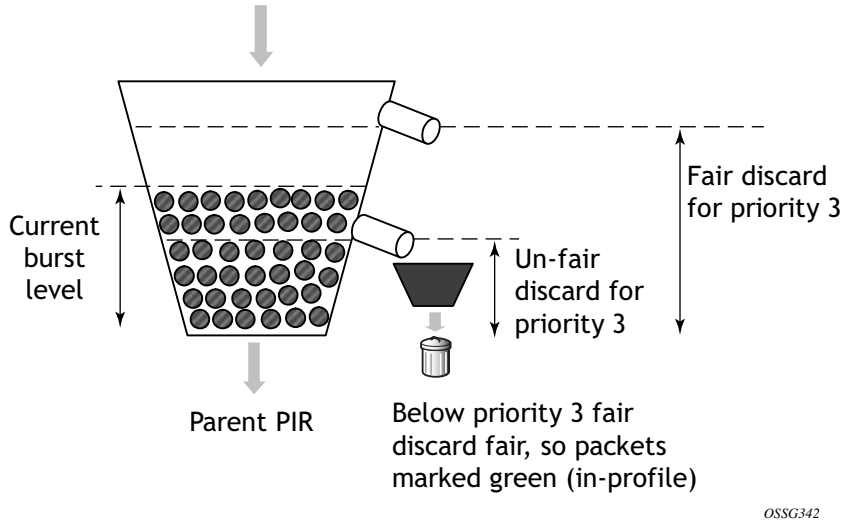

Figure 35: Parent Policer Bucket and Priority Thresholds

CFHP Ingress and Egress Use Cases

While ingress CFHP seems a natural fit based on how policers are typically used in today's networks, 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 simply to provide higher scale in the number of egress aggregate SLAs supported.

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.

Ingress CFHP Queuing

At ingress, CFHP output traffic is automatically mapped to a unicast or multipoint queue in order to reach the proper switch fabric destinations. In order to manage this automatic queuing function, newa shared queue policy has been created policer-output-queues. For modifying parameters in this shared-queue policy, refer to Shared-Queue QoS Policy Command Reference on page 527.

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. The multipoint queues within the policy are created on the IOM3-XP's 16/IMM multicast paths; 16 multicast paths are supported by default with 28 on 7950 XRS systems and 7750 12-e systems, with the latter having setting "tools perform the system set-fabric-speed fabric-speed-b." The multicast paths represent an available multicast switch fabric path - the number of each being controlled using the command:

CLI Syntax: `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 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 packets forwarding class to pick a hash table entry in order 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. This 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 will cause routed and VPLS (IGMP and PIM) snooped IP multicast groups to be managed by IMPM.

Any discards performed in the ingress shared queues will be 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.

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 will be 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.

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
----------------------------------------------
```

Note: To a local queue as in "queue 2" above, 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 when ever there is either a 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.

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.

Subscriber Destination String Queue Group Identification

When a subscriber is identified, a special destination string may optionally exist for the subscriber that is typically used to identify the subscriber's destination aggregation node.

On the subscriber's egress Ethernet port, the default policer-output-queues and other explicitly created queue groups may be configured to represent a destination node by defining the same destination string on the queue group. When the subscriber's destination string is defined, the system will search the subscriber's egress port for an egress queue group with the same string defined. If found, it will use that matched queue group instead of the default queue group. If a queue-group matching the string is not found, the subscriber identification event will not fail and the subscriber host will be mapped to default policer-output-queues.

The destination node-based queuing model is designed to provide the ability to shape the aggregate subscriber output to a destination aggregation node based on a queue group created for the specific purpose. On the queue group, a scheduling-policy is applied which defines the desired virtual scheduling behavior of the queues and aggregate maximum rate of the queue group. The

destination string matching function could be used to represent any arbitrary downstream bandwidth limit, not just an aggregation node. If the destination string is not present (null value), the default policer egress queue group ('policer-output-queues') on the subscriber's port will be used.

SAP Default Destination String

In order to simplify subscriber destination string provisioning, you can use a **def-inter-dest-id** command under the 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 top most 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.

CFHP Policer Control Policy

Provisioning CFHP entails creating policer control policies (policer-control-policy), 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.

Policer Control Policy Root Arbiter

Similar to a scheduler context within a scheduler-policy, the policer-control-policy contains objects called an arbiter 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 its 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.

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 thus becomes 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 thus the parent policer.

A child policer’s priority level on the root arbiter (directly or indirectly) defines which priority level discards thresholds will be associated with packets mapped to the child policer for use in the parent policer (assuming the packet is not discarded by its child policer).

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 will never be greater than a child policer's provisioned rate.

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 profile-mode in order to accept traffic from profile in or profile out forwarding classes or sub classes.

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 sub-classes 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.

Policer Enabled SAP QoS Policy Applicability

Policers are not created on a SAP or subscriber context until at least one forwarding class has been mapped to the policer. Simply creating a policer within a QoS policy does not cause policers to be created on the SAPs or subscriberss 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 or subscribers where the policy is applied. If the forwarding plane where the SAP or subscriber exists either doesn't support policers or has insufficient resources to create the policer for the object, the forwarding class mapping will fail.

Once a forwarding class is successfully mapped to a policer within the policy, attempting to apply the policy to a SAP or a subscriber where the policer cannot be created either due to lack of policer support or insufficient policer resources will 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 and HSMDA SAPs or subscribers.

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 context, the policer is treated as an orphan. The SAP or subscriber 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.

Profile Capped Policers

Profile capped mode has been introduced to enforce an overall in-profile burst limit to the CIR bucket for ingress undefined, ingress explicit in-profile, egress soft-in-profile and egress explicit in-profile packets. The default behavior when profile-capped mode is not enabled is to ignore the CIR output state when an explicit in-profile packet is handled by an ingress or egress policer. The explicit in-profile packets will consume CIR tokens up to 2xCBS 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 in-profile. Thus, the total amount of 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 from ingress as in-profile) are treated the same as explicit in-profile (unless explicitly reclassified as out-of-profile) and have an initial policer state of in-profile
- At both ingress and egress, any packet 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)

The idea is that a profile capped policer trusts the in-profile state determined at ingress classification or egress re-classification, the initial in-profile traffic is preferentially handled with the CIR bucket (2xCBS instead of 1xCBS used by undefined or soft-out-of-profile traffic) and the total amount of in-profile traffic output by the policer cannot exceed the CIR (including initial in-profile traffic).

One other aspect to consider with profile-capped mode is the effect on stat-mode behavior. As will be seen below, each stat-mode has a fixed number of counters in the NP and Q. The mapping of packets to a counter is also fixed by the offered packet state (profile in, profile out, undefined, soft-in-profile and soft-out-of-profile) in conjunction with the output state of the policer. Particularly of note is the egress policer stat-modes and the behavior of soft-in-profile (from ingress) and profile in (reclassified at egress) packets. 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 aspect that causes issues with ingress and egress stat-modes is the fact that initially green (profile in at ingress and egress as well as soft-in-profile at egress), packets can actually turn yellow in the policer output.

Table 48 demonstrates how the CIR rate and initial profile of each packet affects the output of normal (non-profile-capped) and profile-capped mode policers.

Table 48: Effect of Profile-Capped Mode on CIR Output

CIR Setting	Initial Profile State	Normal Mode	Capped Profile Mode	Notes
CIR=0	Ingress Undefined	Always Yellow	Always Yellow	CIR = 0 forces all packets to be yellow when profile-capped mode is enabled. In normal mode, all Profile In related packets are allowed to stay green.
	Ingress Profile In	Always Green	Always Yellow	
	Ingress Profile Out	Always Yellow	Always Yellow	
	Egress Soft-In-Profile	Always Green	Always Yellow	
	Egress Soft-Out-of-Profile	Always Yellow	Always Yellow	
	Egress Profile In	Always Green	Always Yellow	
	Egress Profile Out	Always Yellow	Always Yellow	
CIR=Max/PIR	Ingress Undefined	Always Green	Always Green	CIR never reaches non-conforming state.
	Ingress Profile In	Always Green	Always Green	
	Ingress Profile Out	Always Yellow	Always Yellow	
	Egress Soft-In-Profile	Always Green	Always Green	
	Egress Soft-Out-Of-Profile	Always Green	Always Green	
	Egress Profile In	Always Green	Always Green	
	Egress Profile Out	Always Yellow	Always Yellow	
0 < CIR < PIR	Ingress Undefined	Green below CBS Yellow at or above CBS	Green below CBS Yellow at or above CBS	
	Ingress Profile In	Always Green	Green below 2xCBS Yellow at or above 2xCBS	
	Ingress Profile Out	Always Yellow	Always Yellow	
	Egress Soft-In-Profile	Green below CBS Yellow at or above CBS	Green below 2xCBS Yellow at or above 2xCBS	
	Egress Soft-Out-Of-Profile	Green below CBS Yellow at or above CBS	Green below CBS Yellow at or above CBS	
	Egress Profile In	Always Green	Green below 2xCBS Yellow at or above 2xCBS	
	Egress Profile Out	Always Yellow	Always Yellow	

Policer Interaction with Initial 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 sub-class 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 sub-class associated with the packet is configured as profile in.
- out-of-profile—The forwarding class or sub-class 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 will 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) which 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 four different states relative to initial profile:

- soft-in-profile—The final result at ingress was in-profile and the profile of 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-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.

When an egress policer's CIR rate is set to 0 (or not defined), the policer will have no effect on the profile of packets offered to the policer. The soft-in-profile and hard-in-profile packets will remain in-profile while the soft-out-of-profile and hard-out-of-profile packets will remain out-of-profile.

Setting a non-zero rate for the egress policer’s CIR will modify this behavior, but only for Dot1P and DEI egress marking purposes. For egress IP header ToS field marking decisions, the policer’s CIR state will not change the profile used for the marking decision. Both soft-in-profile and hard-in-profile retain their inherent in-profile behavior and the soft-out-of-profile and hard-out-of-profile retain their inherent out-of-profile behavior.

For L2 marking decisions (Dot1P and DEI), the hard-in-profile and hard-out-of-profile packets ignore the egress policer’s CIR state. When the packet state is hard-in-profile, the in-profile Dot1P marking will be used and when DEI marking is enabled for the packets forwarding class it will be marked 0. When the packet state is hard-out-of-profile, the out-of-profile Dot1P marking will be used and when DEI marking is enabled for the packets forwarding class it will be marked 1.

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 will override the packets soft profile state. When the policer’s CIR is currently conforming, the output will be in-profile. When the CIR state is currently exceeding, the output will be out-of-profile. The Dot1P and DEI (when DE marking is configured) will reflect the CIR derived packet state.

Ingress ‘Undefined’ Initial Profile

Access ingress packets have one of three initial profile states prior to 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 sub-class within a forwarding class. The forwarding class or sub-class may be defined as explicit profile in or profile out (the default is no profile). When a packet’s forwarding class or sub-class 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 48 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.

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 once the combined explicit ‘in-profile’ rate and ‘undefined’ rate causes the bucket to reach ‘threshold-bc’.

The policer’s CIR bucket will indicate that the explicit ‘in-profile’ packets should be marked ‘out-of-profile’ once 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, once the packet is received at egress, an ingress ‘in-profile’ packet will be 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 desired for explicit in-profile packets, the packets should also be classified as priority high.

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 on page 718 is altered in one aspect. The CIR output state of yellow at the greater threshold is actually honored and the packet will be treated as out-of-profile. The packet will be sent to egress in the ‘soft-out-of-profile’ state in this case.

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 thus 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.

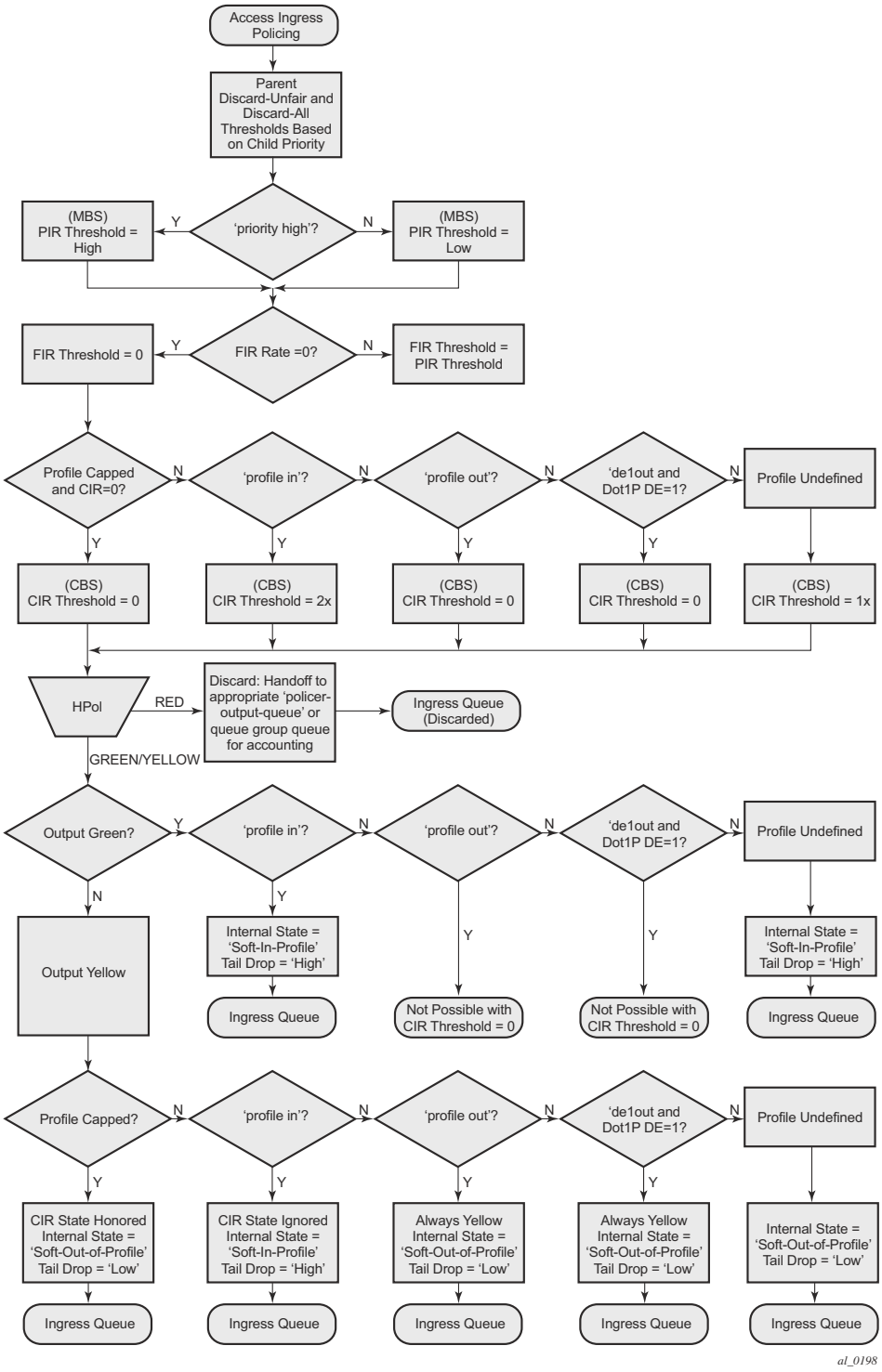

Figure 36: Ingress Policer Threshold Determination and Output Behavior

Egress Explicit Profile Reclassification

An egress profile reclassification overrides the ingress derived profile of a packet and may set it to either ‘hard-in-profile’ or ‘hard-out-of-profile’. A packet that has not been reclassified at egress retains its ‘soft-in-profile’ or ‘soft-out-of-profile’ status.

Egress 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.

Preserving Out of Profile State at Egress Policer

Traffic sent through an egress policer with a non zero CIR will be 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.

Egress Policer CIR Packet Handling without Profile Capped Mode

When an egress policer has been configured with a CIR (max or explicit rate other than ‘0’) and profile capped mode is not enabled, the policer’s CIR bucket state will override 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 been set to ‘0’ on the egress policer, the egress policer output state will be ‘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 similar to the ingress policer handling of initial ‘in-profile’ or ‘out-of-profile’ packets. When a packet has been reclassified as ‘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 will remain ‘in-profile’ even if the higher threshold is crossed. When a packet has been reclassified as ‘hard-out-of-profile’, it does not consume the egress policer’s CIR bucket depth and the policer output profile state remains ‘out-of-profile’.

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 on page 722 is modified in three aspects.

First, the soft-in-profile received from ingress is handled in a similar fashion as egress explicit **profile in** reclassification unless the packet has been reclassified to **profile out** at egress.

Second, explicit egress **profile in** and soft-in-profile that has not been reclassified to **profile out** at egress are allowed to be marked out-of-profile by an egress policer with CIR not set to 0.

Third, when the policer has a CIR = 0 rate (the default rate), all profile capped packets are treated as out-of-profile independent of the initial profile state.

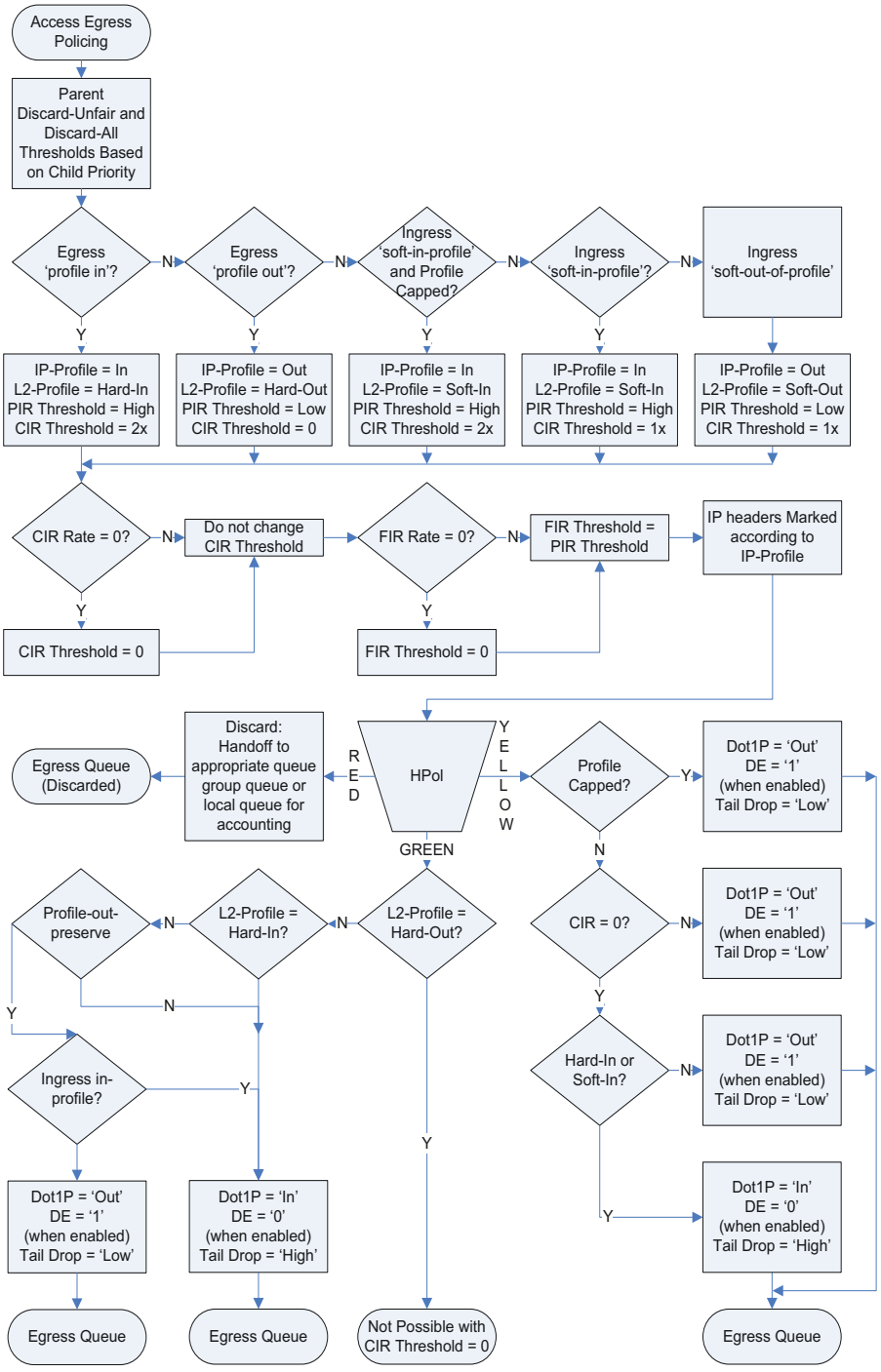

Figure 37: Egress Policer Threshold Determination and Output Behavior

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 will interact 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 de1out 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)

In order 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 will have a CIR defined, so the output green/yellow states will not exist. Also, not every policer will have 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 will be enforced, but the amount of in-profile and out-of-profile traffic output from the policer will not be 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. You can see the resources used and available by using the **tools dump system-resources** command. If insufficient resources exist, the change in the mode will fail without any change to the existing counters currently applied to the existing policers. If the QoS policy is being applied to a SAP or subscriber context and insufficient counter resources exist to implement the configured modes for the policers within the policy, the QoS policy will not be applied. For SAPs, this means the previous QoS policy will stay in effect. For subscribers, it could mean that the subscriber host event where the QoS policy is being applied will fail 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 will 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 will be 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 tail drop 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 sub-class 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 sub-class 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 sub-class provisioned as profile out

  –>Ignore the CIR on the policer and solely rely on the forwarding class or sub-class profile provisioning to the proper policer CIR output

Egress policers also have a default CIR set to 0, but in the egress case a value of 0 disables policer profiling altogether. Egress packets on a CIR disabled egress policer retain their offered profile state (soft-in-profile, soft-out-of-profile, hard-in-profile or hard-out-of-profile).

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 and when the CIR has a non-zero value. Also when overriding the policer's cir mode, make sure you 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

Egress Child Policer Stat-Mode

Egress policers have fewer stat-mode options due to the fact that they do not deal with offered packets with an undefined profile state. All packets received on the egress forwarding plane have been 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.

For egress, the possible types of offered packets include:

- Soft offered in-profile (from ingress)
- Soft offered out-of-profile (from ingress)
- Egress explicit in-profile (reclassified at egress)
- Egress explicit out-of-profile (reclassified at egress)

Similar to ingress, the possible output results are:

- Output green (in-profile)
- Output yellow (out-of-profile)
- Output red (discard)

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

Details of the output showing the stat-modes for ingress and egress child policers can be found in the Class Fair Hierarchical Policing for SAPs section of the SR OS Advanced Configuration Guide.

Profile Preferred Mode Root Policers

The profile-preferred option ensures that the root policer provides a preference to consume its PIR bucket tokens at a given priority level to packets which have their profile state been set to in-profile by the output of the child policer CIR bucket.

When this option is selected, all child policers parented to a root policer will have their FIR bucket track the state of the CIR bucket. In other words, a green packet will always be blue and a yellow packet will always be orange. When admitting packets from the child policers within a given priority level, orange packets will be allowed up to the "discard-unfair" threshold while blue packets will be allowed up to the "discard-all" threshold.

HPOL will no longer set the FIR bucket of the child policer based on fair share calculation. Instead, the ‘profile-preferred’ option forces the FIR bucket to track the CIR bucket’s decrement rate and the threshold chosen for the CIR bucket would also be used in the FIR bucket (instead of using the threshold associated with the PIR bucket).

The green/yellow output from the policer would be used for packet marking decisions. The blue/orange child policer input to the parent policer would chose the discard-orange or discard-all thresholds for the child policer’s priority level within the parent policer.

The net result is that explicit in-profile packets stay blue up to the high CBS threshold, undefined profile packets would stay blue up to the low CBS threshold (1x CBS) and explicit out-of-profile packets would always be orange due to a 0 CBS threshold. Orange packets would be discarded by the parent policer within the child policer’s priority level before the blue packets, preferring blue packets over orange once the discard-orange threshold is crossed.

The following is the CLI for the new option. The same option applies to overrides applied to the instances of a policer control policy under a SAP or subscriber 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
```

Note that the profile-preferred option provides us a way to configure a specific FIR (since 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 will always have PIR equal to

the min (child PIR, root PIR) and the FIR and CIR are fixed and equal. The child parenting weights are thus not used. This impacts the show commands, for example offered rate information will not be available. The output of some show commands (**show qos policer-hierarchy** ... **detail**) should be adjusted for profile-preferred configurations.

If an intermediate arbiter is present, then polling is offered at different rates since the child policer PIRs will be set based on this information so as to share the intermediate arbiter PIR in proportional to their parenting weight to the intermediate arbiter.

Interaction Between Profile Preferred and Profile Capped Mode

There is no requirement to restrict profile-preferred mode to only work when all children are profile-capped.