



CHAPTER 1

Quota Management Overview

Revised: February 2, 2011, OL-21082-04

Introduction

This chapter provides an overview of the Quota Manager (QM). The chapter contains the following sections:

- [Information About the Quota Management Solution, page 1-1](#)
- [Quota Manager Description, page 1-2](#)

Information About the Quota Management Solution

Quota Management is a type of policing of broadband user traffic that enforces policy actions based on integral and accumulative characteristics. Specifically, the QM controls the amount of consumed traffic per application and direction. The users consuming various amounts of the data traffic during specified period of time can help the provider to prevent abusive use of ISP resources shared by ISP customers and is an important step in ensuring improved user experience for the maximum number of subscribers.

The Service Control Application for Broadband (SCA-BB) solution provides powerful quota management capabilities enabling simple implementation of quota management for application traffic:

- Managed network resources—Upstream traffic volume, downstream traffic volume, number of sessions.
- External or internal quota replenishment schemes.
- Real-time notifications—Subscriber quota depletion, subscriber quota below threshold.
- Java API for Quota Management integration—Included within the SCE Subscriber Management API.
- Simultaneous different quota management schemes per subscriber group (policy package).
- Various actions upon quota depletion/breach—Bandwidth control, block, subscriber notification, real-time notification.
- Raw Data Record (RDR) based reporting.

SCA-BB supports three Quota Management operational and integration models that allow gradual investment and trade-off between integration/deployment complexity and functional offering:

- SCE Internal model—Time-based, auto-replenished quota.
- SM Quota Management model—Time-based, auto-replenished quota with preserved state.
- Flexible model—Integration with external QM.

Quota Manager Description

In versions of the Service Control Management Suite (SCMS) Subscriber Manager (SM) prior to Release 3.0, functionality existed to maintain subscriber quota levels across subscriber sessions. This functionality, removed in Release 3.0, was enhanced and reinstated.

The Quota Manager (QM) is now available as a component of the SM, which enables Service Control solution providers to manage, with a high degree of flexibility, subscriber quota.

- [QM Functionality, page 1-2](#)
- [Quota Manager Module, page 1-3](#)
- [Network Topology, page 1-3](#)
- [Quota Indications, page 1-3](#)
- [SCA BB Quota Buckets, page 1-4](#)
- [Quota Provisioning, page 1-5](#)
- [Sliding Window Model, page 1-6](#)
- [Multiple Thresholds of Subscriber Quota, page 1-6](#)

QM Functionality

The QM controls SCA-BB quota functionality and acts as an entry-level quota policy repository. The QM is an event driven solution leveraging the functionality of the Service Control Engine (SCE) Subscriber Application Programming Interface (API).

The quota manager provides the following functionality:

- Subscriber quota is preserved across subscriber sessions.
- Aggregation periods and amounts can be set on a per package basis.
- Quota allocation at the beginning of an aggregation period can be spread over time for different subscribers to avoid the buildup of traffic bursts at the start of the aggregation period.
- Subscriber quota is preserved across QM upgrade.

The QM supports:

- All SCE topologies (1+1 and MGSCP).
- High availability of the QM server (utilizing a Veritas Cluster Server [VCS]).
- Multiple quota thresholds allowing move subscribers to penalty packages upon certain quota threshold breach.

Subscribers can:

- Assign time-based quota for a period called an aggregation period while the consumption during this period is calculated in a sliding window model. For more information, see the [“Sliding Window Model” section on page 1-6](#).
- Assign 1-time quota that can only be replenished manually.
- Move between packages at any time, whether they are logged in or not.
- Purchase additional quota within an aggregation period.

Quota Manager Module

The QM module runs as a component on the SM. All of the logic to manage and maintain quotas runs on the SM server; therefore, you should configure the QM on the SM or load the configuration onto the SM. In a cluster setup, you must load the configuration onto each SM in the cluster. Subscriber quotas are stored in the SM database.

The QM uses the SCE Subscriber API to provision quota to subscribers upon request using the existing external quota functionality of the SCE.



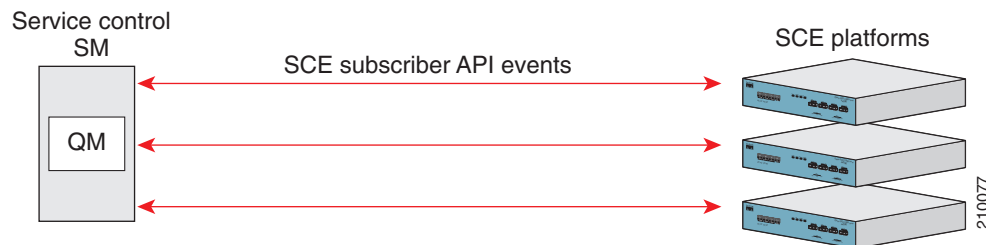
Note

Per package quota is configured on the SM and adds complexity to the integration or operation. Quota management also requires more management messages on the network.

Network Topology

The QM can serve more than one SCE. At any point in time, one subscriber is managed by one SCE. [Figure 1-1](#) shows the network topology of a system employing quota management.

Figure 1-1 Quota Manager Network Topology



Quota Indications

The QM is an event-driven component. All quota logic is performed as a response to quota indications that the SCE initiates. Therefore, the quota state is up-to-date in accordance with the last quota indication received.

Table 1-1 lists the quota indications and the quota responses to the indications.

Table 1-1 Quota Indications and Responses

Indication	Reason Generated	Quota Manager
Quota breach	Generated when a subscriber uses the entire available quota in the SCE.	Responds to this indication by providing more quota for the subscriber if the subscriber quota allocation is not depleted for the current aggregation period.
Remaining quota	Periodically generated to keep the QM updated with the quota remaining in the SCE.	In most cases, does not respond to this indication, but does write the quota value to the database to be stored until the subscriber next logs in. If the reported quota is below zero, responds by providing more quota for the subscriber if the subscriber quota allocation is not depleted for the current aggregation period.
Quota below threshold	Generated when the subscriber quota in the SCE drops below a predefined level.	Responds to this indication by providing more quota for the subscriber if the subscriber quota allocation is not depleted for the current aggregation period.
Quota state restore	Generated in response to a subscriber login event.	Responds to this indication by updating the subscriber quota in the SCE.



Note

Quota calculation uses the system date and time of the QM machine to calculate aggregation periods. During QM operation, if the system date and time is changed, we recommend you delete all quota information from the SM database to recalculate all aggregation periods according to the new date and time. This operation also causes a quota replenish for all subscribers. To remove quota information from the SM database, run the **p3subsdb --clear-all-states** command.

SCA BB Quota Buckets

The basic building block that the SCA BB uses to implement a specific quota is called a quota bucket. Each subscriber can be assigned with up to 16 quota buckets to maintain the utilization of subscriber traffic over a specific service.

The following network resources can be managed with a quota bucket:

- Traffic volume in units of Layer 3 kilobytes—Subscriber traffic consumption can be monitored separately per traffic direction such as from the subscriber (up) or to the subscriber (down).
- Number of sessions—The total number of network sessions classified to the services associated with the quota bucket.

The SCE provides real-time notifications and reporting of the quota breach, remaining quota periodic, quota state restore, and quota threshold events.

Depletion or breach of a quota bucket takes place once a monitored resource is consumed while the bucket is empty (consumed entirely). After the quota is depleted, the quota bucket causes all service rules associated with the quota bucket to execute breach state actions. Quota bucket replenishment (automatic or external) can bring the quota to a non-depleted state, and has the service rules associated with the quota bucket execute the normal (nonbreached) actions.

Upon quota depletion, the SCE platform can perform one or more actions, as defined in the service configuration settings. The following actions are:

- Send quota breach RDR.
- Signal a quota breach notification through Java API.
- Activate subscriber notification (HTTP redirect based notification).

Quota Provisioning

The QM uses quota provisioning to provide additional quota to the subscriber as a response to the following quota indications: Quota State Restore, Quota Below Threshold, and Quota Breach. Quota provisioning occurs only if quota is still available for the subscriber.

Quota provisioning is split into dosages to ensure that quota consumption is accounted for accordingly in the quota manager database. This ensures that in cases of failures, the amount of quota used, but not accounted for by a subscriber, is limited by the quota dosage size.

The QM performs the provisioning by adding quota to the SCE so that after the provisioning operation, the available quota in the SCE equals a configurable dosage size.

For example, suppose that the following configuration exists for bucket 1:

- Bucket size is 100 MB
- Dosage size is 10 MB
- Threshold size is 1 MB

The following quota provisioning takes place:

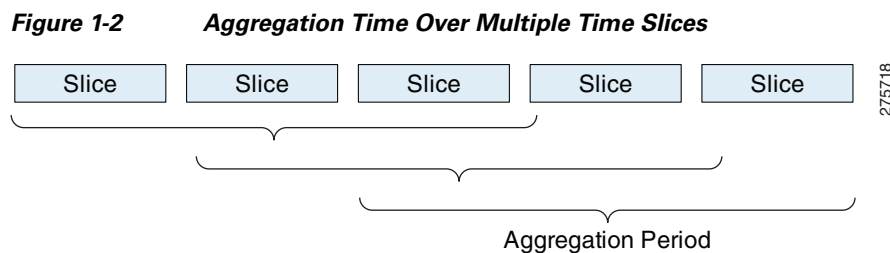
- When the subscriber logs in for the first time, the SCE initiates a Quota State Restore indication for this subscriber. This indication triggers the following actions:
 - 100 MB of quota is added to the QM database for the first aggregation period.
 - The QM provisions only 10 MB of quota to the SCE, as configured by the quota dosage value.
- After the subscriber consumes 9 MB of quota, the quota threshold (1 MB) is reached and the SCE initiates a Quota Below Threshold indication for this subscriber. This indication triggers the following actions:
 - The QM provisions a further 9 MB of quota to the SCE. After the provisioning operation, the remaining quota in the SCE equals the dosage size (10 MB).
 - At the same time, the QM updates its database to indicate that 9 MB of quota was consumed by the subscriber. After the provisioning operation, the remaining quota in the QM is 91 MB (for example, 100 MB minus 9 MB).

Sliding Window Model

Before Release 3.5.5, the QM measured the consumption of each subscriber for the fixed aggregation periods. At the end of the aggregation period, a new period starts with no memory about the consumption of the previous period (in other words, starting each aggregation period from scratch).

Starting in Release 3.5.5, the aggregation period is divided into multiple slices. Using the sliding window module, the SCE calculates the bandwidth consumption for each time slice or time period separately and saves the calculated bandwidth consumption in the QM. The SCE calculates the bandwidth consumption during the aggregation time over a configured number of time slices (N slices) or time periods. Calculating bandwidth consumption over the configured number of time periods ensures that the bandwidth consumption calculation is based on the average bandwidth consumption.

Figure 1-2 illustrates the aggregation time over multiple slices.



Multiple Thresholds of Subscriber Quota

The *Multiple Thresholds of Subscriber Quota* allows differentiating quota use between the groups of subscribers.

QM manages a bucket per subscriber and when a defined threshold is crossed, the QM changes a profile of the subscriber to the penalty profile according to the configuration. Subscribers that breach their quota are put in the penalty profile for a period of time. QM is also able to identify the consumption under a certain threshold. After the penalty time expires, QM verifies subscriber consumption and decides to which profile they should be moved (according to the configuration).



Note

Penalty profiles support single bucket per subscriber only.

Configuring the sliding window algorithm to n slices allows the sliding window algorithm to account for the penalty calculation for the $N - 1$ last slices and the current slice. For example, if $N = 3$ and the quota threshold is configured to x bytes over the aggregation time. The algorithm will take the consumption for the last two time slices and the consumption report received for the current time slice to calculate the average consumption for three time slices. If the subscriber consumed more than the average x bytes over the time slices, the subscriber is switched to the penalty package.

If a subscriber exceeds one of the thresholds, the subscriber is switched to the according package for a penalty time. If during the penalty time the subscriber again exceeds one of the thresholds, the subscriber is switched to the penalty package according to the configuration and the penalty timer is reset (starting to count the penalty time from zero).

Multiple Quota Thresholds—Example

In the following example, the requirement is to distinguish between the three groups of the subscribers:

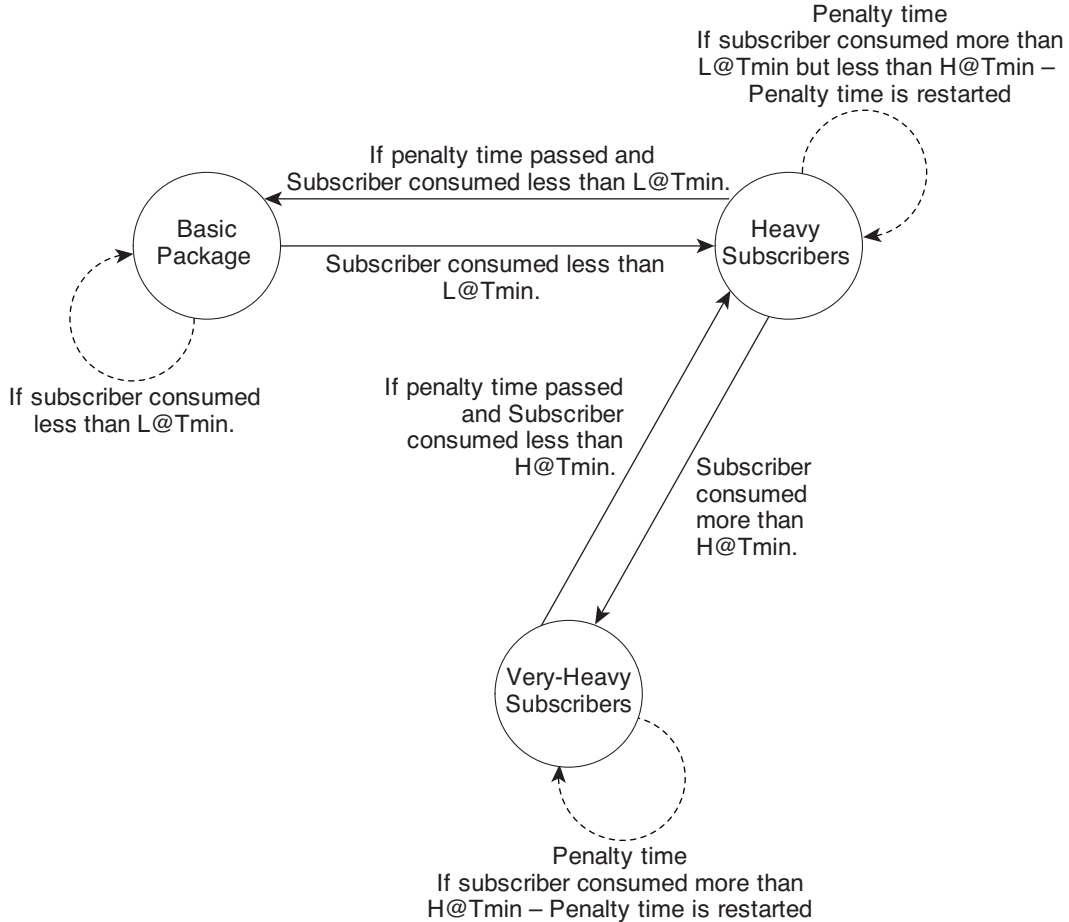
- Regular subscribers.
- Heavy subscribers—Subscribers that consumed more than 500 KB in the last 30 minutes.
- Very heavy subscribers—Subscribers that consumed more than 1000 KB in the last 30 minutes and

Regular subscribers are untouched, heavy subscribers are lightly controlled when the interface is close to congestion and very heavy subscribers are forcefully controlled when the interface is close to congestion.

The SCE sends RDRs to the QM. The QM detects heavy subscribers based on the amount of bandwidth (BW) these subscribers consumed in the last x minutes. Heavy subscribers are divided into two groups based on their consumption (crossed first threshold and crossed first and second thresholds) and each group is assigned a different package. The QM manages bucket per subscriber and when a threshold is crossed, the QM changes the subscriber package to either the heavy package or very-heavy package.

The Finite State Machine (FSM) as shown in [Figure 1-3](#) illustrates the new quota algorithm. Upon moving to a new package, the current status is checked, so a subscriber might immediately move two steps in a row.

Figure 1-3 Finite State Machine



Comments:

- (*) Aggregation period is restarted when subscriber is switched to a new package
- (*) Penalty time must be a product of the aggregation period ($Tmin$)

275933

Table 1-2 lists the parameters relevant to each threshold and quota package.

Table 1-2 Quota Threshold Parameters

	Quota profile/ package 1	Quota profile/ package 2	Quota profile/ package 3
Threshold on quota greater than/Move to	L/2	H/3	H/3
Threshold on quota less than/Move to	L/1	L/1	H/2
Assurance level	10	6	2

During the aggregation period, if the subscribers exceed the high threshold, the subscriber is moved to another package and the aggregation period start is reset to be the package change time. In addition, a penalty timer is reset.

On the first RDR after the aggregation period, the QM checks if the penalty time has passed and if the quota was below the low threshold. During the last aggregation period, if the quota was below the low threshold and the penalty time has passed, the package is changed.