
hp OpenView Service Quality Manager



SQM QoS Metrics Service Adapters User's Guide

Edition: 1.4

for the HP-UX Operating System

March 2007

© Copyright 2007 Hewlett-Packard Company, L.P.

Legal Notices

Warranty

The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

License requirement and U.S. Government legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright notices

© Copyright 2007 Hewlett-Packard Development Company, L.P.

Trademark notices

HP-UX Release 10.20 and later and HP-UX Release 11.00 and later (in both 32 and 64-bit configurations) on all HP 9000 computers are Open Group UNIX 95 branded products.

Java™ and all Java based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

Oracle® is a registered U.S. trademark of Oracle Corporation, Redwood City, California.

UNIX® is a registered trademark of The Open Group.

Windows®, and Windows NT® are U.S. registered trademarks of Microsoft Corporation.

Origin

Printed in France.

Contents

Chapter 1	7
Introduction to OpenView SQM QoSmetrics Service Adapter	7
1.1 Overview	7
1.1.1 Data Collection.....	7
1.1.2 Data Feeder Discovery	8
1.1.3 Service Instance Creation.....	8
Chapter 2	9
QoSmetrics Service Adapter User Guide	9
2.1 Software and Hardware Requirements.....	9
2.1.1 Software requirements.....	9
2.1.2 Hardware requirements	9
2.2 Installation	9
2.3 Loading Model in OV SQM	11
2.4 Running QoSmetrics SA with command line	11
2.5 Stopping the QoSmetrics SA	14
2.6 Setup for a OV SQM Platform.....	14
2.7 SA Startup by OV SQM Platform	14
2.8 Web Service Administration.....	14
2.9 Configuration.....	15
2.9.1 Log	15
2.9.2 Collected Data Configuration.....	17
2.10 Uninstall / Clean Up	17
Chapter 3	19
SQM / QoSmetrics Integration	19
3.1 Data Collection.....	19
3.1.1 Video Stream Data.....	19
3.1.2 Probe Statistic Data	20
3.1.3 Tables involved in data collection	20
3.2 IPTV QoSmetrics Model	20
3.2.1 Automatic Service Instantiation	21
3.2.2 Computed KQIs	22
3.3 Deployment Scenario.....	23
3.3.1 Simplest Case.....	23
3.3.2 Dataload.....	24

Preface

This document provides reference information to help you configure and use HP OpenView Service Quality Manager (OV SQM) QoSmetrics Service Adapter. This service adapter collects Video KQI collected in QoSmetrics NetAdvisor system.

Intended Audience

This document is intended for administrators and technical staff in charge of administrating an OpenView SQM platform.

Prerequisite Reading

This document assumes that you have read the OpenView SQM Overview and are familiar with OpenView SQM terminology and components.

Supported Software

The software versions referred to in this document are specified in chapter 2.1.1, “Software requirements”.

Typographical Conventions

Courier Font:

- Source code and examples of file contents.
- Commands that you enter on the screen.
- Path names
- Keyboard key names

Italic Text:

- Filenames, programs and parameters.
- The names of other documents referenced in this manual.

Bold Text:

- To introduce new terms and to emphasize important words.

Associated Documents

The OpenView SQM documentation set includes the following:

- *OpenView SQM Overview*
- *OpenView SQM Installation Guide*
- *OpenView SQM Administration Guide*
- *OpenView SQM IPMS Value Pack User's Guide*

Support

You can visit the HP OpenView support web site at:

<http://support.openview.hp.com/support.jsp>

This Web site provides contact information and details about the products, services, and support that HP OpenView offers.

HP OpenView online software support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support site to:

- Search for knowledge documents of interest
- Submit enhancement requests online
- Download software patches
- Submit and track progress on support cases
- Manage a support contract
- Look up HP support contacts
- Review information about available services
- Enter discussions with other software customers
- Research and register for software training

Chapter 1

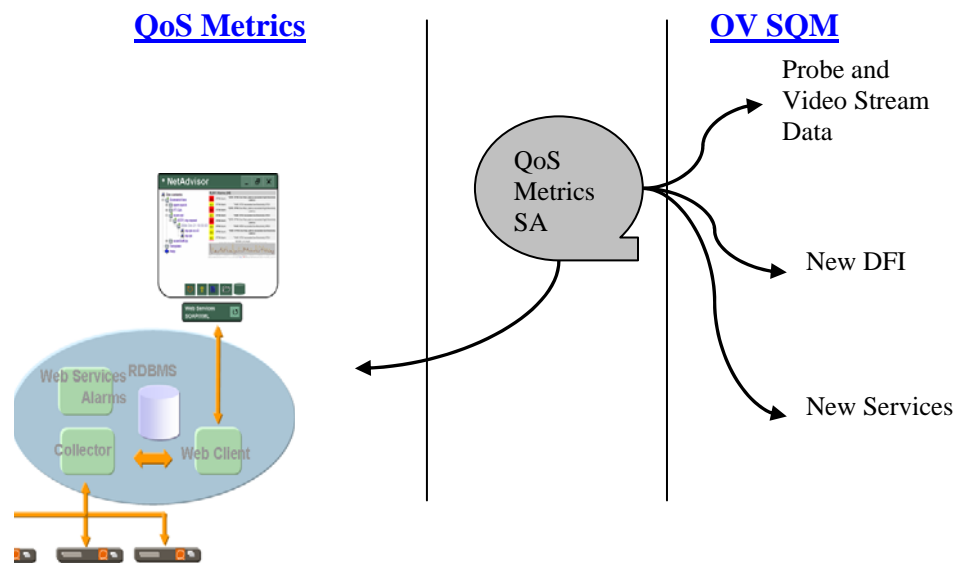
Introduction to OpenView SQM QoSmetrics Service Adapter

This chapter describes the main features of the OpenView (OV) Service Quality Manager (SQM) QoSmetrics Service Adapter.

1.1 Overview

The QoSmetrics Service Adapter collects Video KQI from a QoSmetrics NetAdvisor database and publishes them to an OV SQM platform.

In addition, the QoSmetrics Service Adapter is able to manage DFI discovery on the fly and to maintain a service instance hierarchy up-to-date.



QoSmetrics Service Adapter Overview

1.1.1 Data Collection

The QoSmetrics SA collects Video Stream measurement data from a QoSmetrics NetAdvisor database.

The collection process queries the NetAdvisor database to get the list of Active and Passive test campaigns and for each of them collects and publishes the associated results.

Data from Passive and Active tests are collected transparently, without having to know how tests and test campaigns are configured on the NetAdvisor system.

The QoS metrics Service adapter is able to collect from a starting point in the past, allowing to load NetAdvisor historical data in SQM.

This version of the QoS metrics Service adapter collects Video Stream KQI only. Data and Voice stream are not managed.

1.1.2 Data Feeder Discovery

A “Data Feeder Discovery” mode can be activated. When using this mode, the QoSmetrics Service adapter automatically creates unknown data feeder instances found during a collection step into OV SQM.

1.1.3 Service Instance Creation

A “Service Instance Creation” mode can be activated. When using this mode, the QoSmetrics Service adapter automatically create/update a set of service instances based on existing and discovered Data Feeder instances. This allows maintaining automatically a default instance model in SQM based on collected information, without any user interaction.

Chapter 2

QoSmetrics Service Adapter User Guide

This chapter describes how to install, run and configure the QoSmetrics Service adapter.

2.1 Software and Hardware Requirements

2.1.1 Software requirements

- HP-UX 11.11
- NetAdvisor Platform
- HP OpenView Service Quality Manager V1.4 (Kernel subset)
- Oracle JDBC driver related to Oracle platform in use.

2.1.2 Hardware requirements

For the complete hardware requirements, see the *HP OpenView Service Quality Manager Installation Guide*.

For the Service Adapter:

- 5 MB of available hard disk space (10 MB recommended)

2.2 Installation

The QoS Metrics Service Adapter can be installed on any director of an SQM platform.

To install the QoS Metrics Service Adapter, start the provided installer “SQMQOSMETRICSSA.bin and follow the instructions.

If Jython is not already installed on the platform, the installer will trigger the Jython installation. You have to follow installer instruction, and select:

- **Minimum (core)** as installation type
- `<TEMIP_SC_HOME>/jython2_2` as target directory.



After installation the following directories are created:

- TEMIP_SC_HOME/jython2_2: contains the Jython library and binary
- TEMIP_SC_HOME/ServiceAdapters/jython_sa: Contains framework for Jython Service Adapters.

Under the TEMIP_SC_HOME/ServiceAdapters/jython_sa directory the following directories are created:

- python: Additional Python library required for Jython
- sqm_lib: contains SQM libraries for Jython Service Adapters framework
- qosmetrics_sa: The Root Directory for the QoSmetrics Service Adapter.
- qosmetrics_sa/doc: contains the documentation.
- qosmetrics_sa/bin: contains the Service Adapter scripts and binaries.
- qosmetrics_sa/config: contains some configuration files for the QoSmetrics SA.
- qosmetrics_sa/model: contains the dedicated model (Rose Model and Generated XML files).

Once installed, you should start the QoS metrics SA with `-help` option to check that the installation is successful and also it will compile delivered files. You have to be root to perform the following commands.

```
#cd $TEMIP_SC_HOME /ServiceAdapters/jython_sa/qosmetrics_sa
# ./qossa.sh --help
...
*sys-package-mgr*: processing new jar,
'/opt/OV/SQM130/jre/lib/rt.jar'
*sys-package-mgr*: processing new jar,
'/opt/OV/SQM130/jre/lib/sunrsasign.jar'
*sys-package-mgr*: processing new jar,
'/opt/OV/SQM130/jre/lib/jsse.jar'
...
usage: QoSmetrics SA 0.1
```

2.3 Loading Model in OV SQM

If the IPMS Value Pack is loaded on your SQM platform, you do not have to go through these steps since the IPMS Value Pack includes the definition required to support the QoSMetrics probes.

If you do not have the IPMS Value Pack, a “Basic” configuration can be loaded on your SQM platform. This configuration can be used as a starting point for a more complex solution or stand-alone (with limited features).

The XML model files to load in SQM are located in `TEMIP_SC_HOME/ServiceAdapters/jython_sa/qosmetrics_sa/model/XML`.

The command to load this model in SQM is:

```
sqmadm>temip_sc load_model -f
$TEMIP_SC_HOME/ServiceAdapters/jython_sa
/qosmetrics_sa/model/XML
```

Note

If you do not intend to use automatic SI creation, you do not need to load the service definition that is provided with the service model. You can build your own Service Definition one using the provided Control Units. (See Chapter 3 for more details).

In any case, you still need to load the DFDs as provided by the solution.

2.4 Running QoSmetrics SA with command line

The QoSmetrics SA has to be launched by a command line with appropriate command line arguments.

```
# cd $TEMIP_SC_HOME/ServiceAdapters/jython_sa/qosmetrics/bin
# qossa.sh [options]
```

Command arguments:

- `-h, --help`:
Shows the help message and exit.
- `--verbose, --quiet`:
Specify verbose level at start up. Optional.

- --starttime=<Stat Time>:
Collection start time. Useful to load data in the past.
Accepted formats are epoch or “YYYY/MM/DDTHH:MI:SS”
Optional, defaulted to current time.
- --createsi:
When selected, the SA will trigger Service Instances creation/update after each collection. --discovery should be selected too.
- --reinit:si:
When selected, the SA will force re-creation of every Service Instances at startup.
- --discover:
Activate DFI discovery when selected.
- --qoshost <Host Name>
The name of the QoSmetrics NetAdvisor host.
Optional if QoS SA launched on NetAdvisor platform.
- --qosport <Port>
The port number to use to connect to the QoSmetrics NetAdvisor Database.
Note: The port used for connection and must be opened on target host.
Optional, defaulted to port 3306 (MySQL default).
- --srmhost <host name>
Specify the name of the host that hosts the OV SQM SRM database.
Note: The port 1521 is used for connection and must be opened on target host.
Optional, if QoSmetrics SA is launched on the host that hosts the SRM database.
- --period <Period in second>
The collection period of the service adapter in seconds.
Optional, defaulted to 1 minute.
- --afactor <Multiplication Factor>:
Acceleration factor when collecting data in the past. For each period, the SA will collect data within a time range equals to [period] * [acceleration factor].
When you collect data in the past, the afactor will tell you how long this collection will last. In the example below, the user wants to collect 1 year of past data
Example:
--starttime = “2005/11/01T00:00:00” (current time – 1 year)
period=60 (1 minutes)
afactor=365
→ It will then take $365/365 = 1$ days to collect all past data.
→ Each minutes, the QoS SA will collect 6 hours of data (1 minutes* 365).
The value of afactor must be selected correctly in order to have a dataload as fast as possible but it must not overload the SQM platform.
Optional, defaulted to 1.
- --resolution <value in seconds>:
To avoid flooding OV SQM with too many raw data, it’s possible to aggregate collected raw data. Instead of publishing every collected measure, the SA will only publish one aggregated message per resolution period. The aggregation

function is defined in the config/qosmetric_sa_config.py file.
Optional, defaulted to None (all raw data are published).

- --wSPORT=WSPORT :
Port for WebService Administration. Use this option if several QoSmetrics SA run on the same host.
Default: 8000.
- --logfile=LOG_FILE
Log file name. Use this option if several QoSmetrics SA run on the same host.
Optional, defaulted to [Installation directory]/qosmetrics_sa/qos_sa.log

Note

Please note that long option names are prefixed by two (2) minus characters (--).

Depending on where the QoSmetrics Service Adapter is launched, some arguments are optional.

For example, on an OV SQM platform the --network,--service,--daemon, --srmsid, --srmsid are useless and defaulted to OV SQM environment values.

To launch the QoSmetrics SA on a OV SQM platform, you just have to specify the name of the NetAdvisor host (--qoshost):

```
sqmadm> qossa.sh --qoshost=netadv1 --discover --reinit  
--createsi
```

To launch the QoS metrics SA on a W2K system with SQM UI Administration:

```
C:\>qossa.bat --service <port> --network "<network>"  
--daemon "tcp:<sqm_hostname>:<port>" --srmsid <sqm_hostname>  
-Q <netadvisor hostname>
```

The QoSmetrics Service Adapter command line has some others option that should not be used on a classical production system, but it's possible to use them in some specific cases:

- --srmsid=SRMSID:
This option has to be used only if the OV SQM SRM database SID has been changed.
Optional, defaulted to 'srm' (OV SQM default).
- --network=NETWORK:
Tibco Network specifier. Ex: " ;224.2.0.1".
Optional if OV SQM environment is set.
- --daemon=DAEMON:
Tibco Daemon. Example: tcp:2334 , tcp:hermine:234.
Optional if OV SQM environment is set.
- --service=SERVICE:
Starting Tibco Service for OV SQM (bus 52 - Raw data).
Ex: 2340. Optional if SQM environment is set.
- --nosrm:
SRM is not used to know existing DFI and their status. All collected measures are published.

- --keeplast:

Last collected values are kept in memory. This allows to get diagnostic using the getLastValues web service call.

If started successfully, you should see the following warning:

```
2006-11-30 10:32:13,825: WARNING: SA Collection Thread Started.  
2006-11-30 10:32:13,855: WARNING: QoS SA Administration Service  
listening on http://half:8000
```

It is possible to restart the QoSmetrics Service Adapters at any time to change some of the options. If you do this during an historical dataload, you have to use --starttime option with the last collection timestamp since restarting from last collection timestamp is not managed automatically in this version.

Note

Before restarting the QoSmetrics Service Adapter to change an option, consider to use the Web Service Administration tools. It is possible to change most settings while the Service Adapter is running.

2.5 Stopping the QoSmetrics SA

In the current version, the process has to be killed (Ctrl C or kill command).

2.6 Setup for a OV SQM Platform

Not available. The Service Adapter has to be stated using command line.

2.7 SA Startup by OV SQM Platform

Not available. The Service Adapter has to be stated using the command line. A custom script with suitable argument can be created and started with the nohup UNIX command.

2.8 Web Service Administration

The QoSmetrics SA can be administrated using some XML RPC call (Simple Web Service protocol).

In the current version the following actions are possible:

- setLogLevel(LogLevel Integer): Set log level for log file to specified value.
- setOutputLevel (LogLevel Integer): Set log level for console output to specified value.
- setPeriod(Period Int) : set the collection period to specified value (in seconds). The value is taken in account after next publication.
- setStartTime(stat_time int): Use carefully: set start time for next collection. This permits to recollect data without restating the SA.
- setAccelerationFactor(value Integer) : Set the new value for the acceleration factor. Effective only when collecting past data.
- setResolution(value Integer): Change the resolution for internal aggregation. If value = 0, then raw data are not aggregated by the QoS SA.

- getDFD: return the list of the managed DFD.
- getDFI(dfd string, dfd_version string): return the list of the managed DFI for selected DFD.
- getLastValues(): displays the list of last collected values. Effective only if –keeplast option is selected at QoS SA startup.

The utility client.py located [Installation directory]/qosmetrics/bin/ shows how to call the WebService Administration interface.

The steps required to make a call are:

```
#####
#IMPORT REQUIRED LIBRARY
#####
from xmlrpclib import ServerProxy
from xmlrpclib import Error
import logging
host=<specify hostname>
port=8000 #DEFAULT VALUE

#####
#CREATE THE SERVER PROXY
#####
server = ServerProxy("http://%s:%s " % (host,port)) # local
#####
# MAKE THE CALL
#####
try:
    server.setOutputLevel(logging.INFO)
except Error, v:
    print "ERROR", v
```

2.9 Configuration

2.9.1 Log

Log files can be configured in the [Jython SA directory]/qosmetrics_sa/config/ /qosmetric_sa_log.conf file.

It is possible to change default log level as well as rolling data (number of log files maintained and size of individual file).

You can use the –logfile option to change the destination of the logfile. Default value is in installation directory.

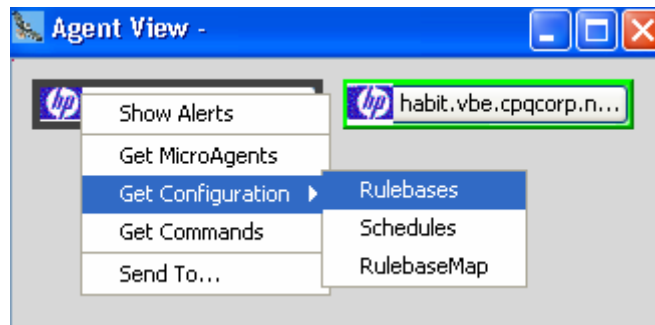
2.9.1.1 Monitor Logs in Hawk Display

If the QoSmetrics SA is installed on an OV SQM platform, it is possible to configure Tibco Hawk to display QoSmetrics SA log in Hawk Display console.

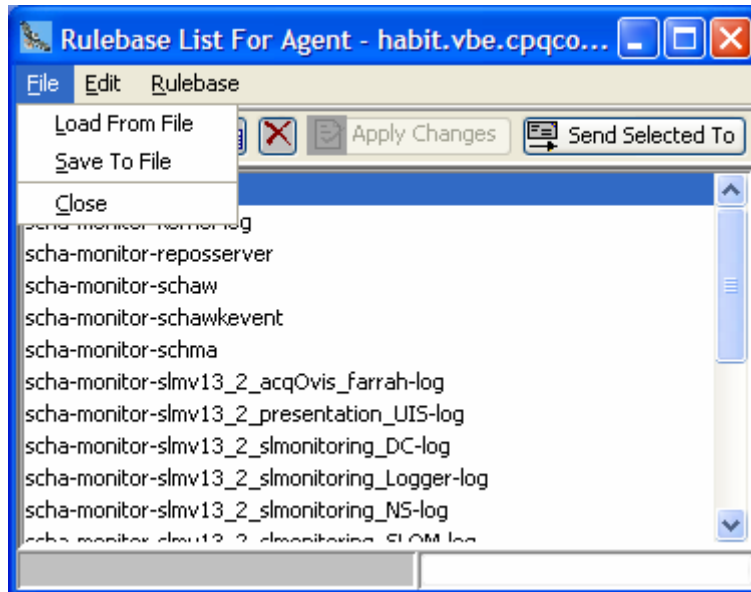
This step is optional and has no influence on the behavior of the SA.

For this, you have to load the QOSSA.hrb rules using the Tibco Hawk Display console.

In the Agent View, select the host where the QoSmetrics SA is supposed to run. Using the contextual menu, select GetConfiguration/Rulebases.



The “Rulebase list” windows opens, select “File / Load From File”:



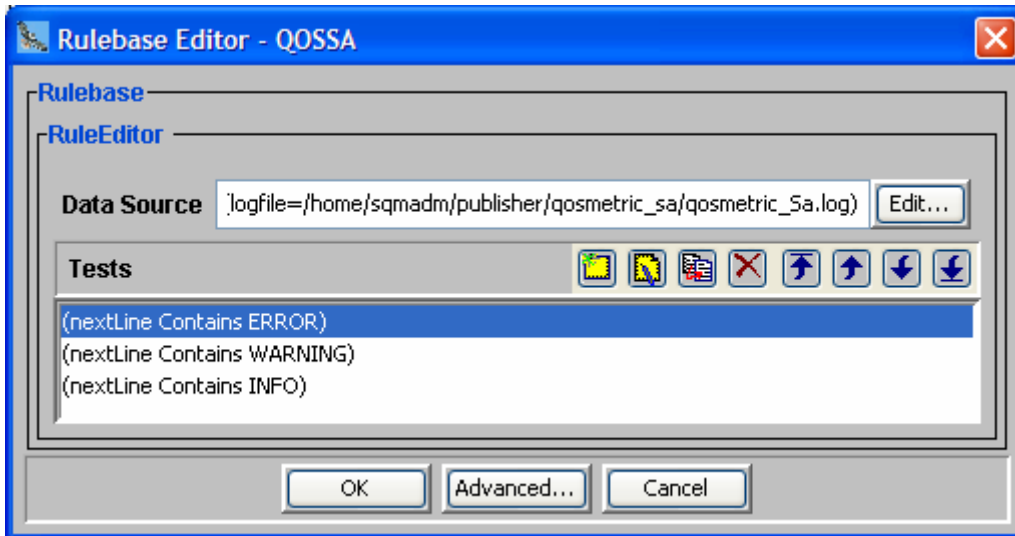
In the file browser, select the QOSSA.hrb file located in the [Installation directory]/qosmetrics_sa/config directory.

This adds a QOSSA rule in the list of the rulebase.

Double click on this rulebase to edit it. This opens the Rulebase editor window.



Double click on the “Logfile:onNewLine...” rule to edit it. This opens the following window:



You have to click edit to open the window below, and then change the logfile location in the Method Arguments:



Close all opened window by clicking “Ok”, then in the last “Rulebase list” window, click on “Apply Change”

2.9.2 Collected Data Configuration

It is possible to configure the collected data in the [Installation directory]/Jython_sa/qosmetrics_sa/config/qosmetric_sa_config.py file.

Be careful when updating this file since it can corrupt QoSmetrics SA behavior.

The file contains comments that explain what to do to add/remove collected parameters.

2.10 Uninstall / Clean Up

To uninstall, you just have to remove the installed directory.

To clean-up the OV SQM platform, you have to remove the Service Definition and the two Datafeeders Definitions.

```
# temp_sc delete sd -sd IPTVQoSMetric -force
# temp_sc_delete_dfd -dfd dpQProbeStat -dfdv v1_0 -force
# temp_sc_delete_dfd -dfd dpQVideoStream -dfdv v1_0 -force
```

Eventually you can use Tibco AMI to purge the SRM database.

SQM / QoSmetrics Integration

This chapter details the integration between QoSmetrics and SQM.

3.1 Data Collection

Two Data Feeders definitions are managed by the QoSmetrics SA.

- dpQ Video Stream: collect Passive/Active video data for a given stream (Channel) at a given measurement point (Probe).
- dpQ Probe Statistic: collect global statistics for a given probe.

The Video Stream data are collected in QoSmetrics results tables.

The “dpQ Video Stream” Data Feeder is able to collect every QoSmetrics Video data stored in the NetAdvisor database, but only a subset of these indicator is actually published in SQM. If needed, it is possible to update a configuration file and add missing parameters for the customer solution (see section 2.9.2).

Some parameters are converted into a more readable format by the QoSmetrics SA. (Integer micro-second → Float milliseconds).

Please refer to NetAdvisor Reference guide for details on parameter descriptions.

3.1.1 Video Stream Data

By default, the QoSmetrics Service Adapter publishes collected data directly into OV SQM.

If the `--resolution` option is used the raw data are aggregated by the QoSmetrics Service Adapter: for each video stream a single message is published per resolution period. The usage of the **resolution** can be useful to reduce the input rate into SQM.

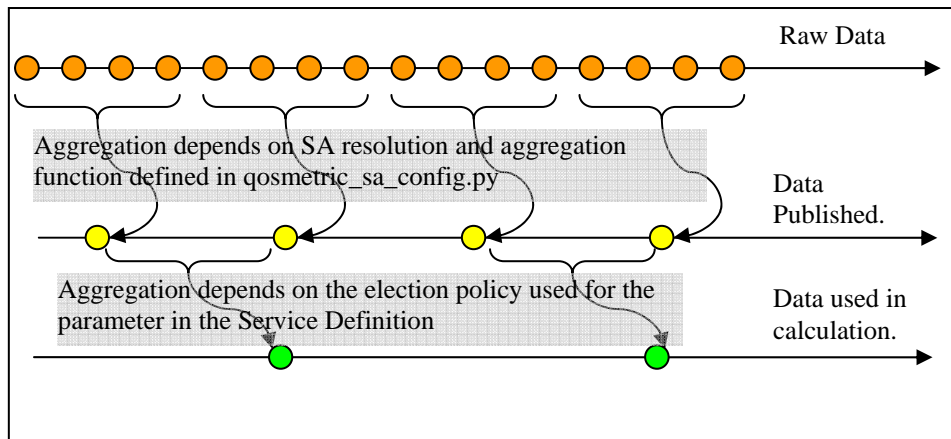
All Video Stream parameters are in auto-forward mode. This means that collected data are available in both SLA Monitoring UI and in SQM Reporting.

Useless data for reporting can be removed from the model by selecting the “Never Reported” flag in Service Designer.

Since data can be collected at a higher rate than the OV SQM calculation engine period, the calculation engine uses an “Election Policy” when computing secondary parameters. By default, the average is used in the QoSmetrics IPTV model, but for some indicators, parameters with other election policies have been defined in the model. For these parameters, the election policy is given in the parameter label.

Example: VFactor (min), VFactor (max).

The figure below illustrates the impact of the `--resolution` and the election policy in SQM.



3.1.2 Probe Statistic Data

The “dpQ Probe Statistic” Data Feeder collects statistical data for a given probe.

Every video stream that is analyzed by the probe is taken into account and synthetic results are computed.

The data feeder can also provide status and health information for a given probe.

The “dpQ Probe Statistic” publishes a set of data per datafeeder instance and per collection period.

3.1.3 Tables involved in data collection

This section describes how QoSmetrics tables are used for data collection.

Passive tests:

ResultTable1, ResultTable2, ResultTable3 tables are extracted from qosmetrix_v30.tblPassiveResultTables.

Then these tables are used to query test results.

Active Tests:

ResultTable1, ResultTable2, ResultTable3, ResultTable4 tables are extracted from qosmetrix_v30.tblUserResultTables.

Then these tables are used to query test results.

The ResultTableN are dynamic and named :
tbl_<ID1>_<N>_<ID2>_<RESULT_TYPE>

Where N in 1,2,3,4 and RESULT_TYPE depend on N and is in (VFACTOR|RTP|MP2PSI)

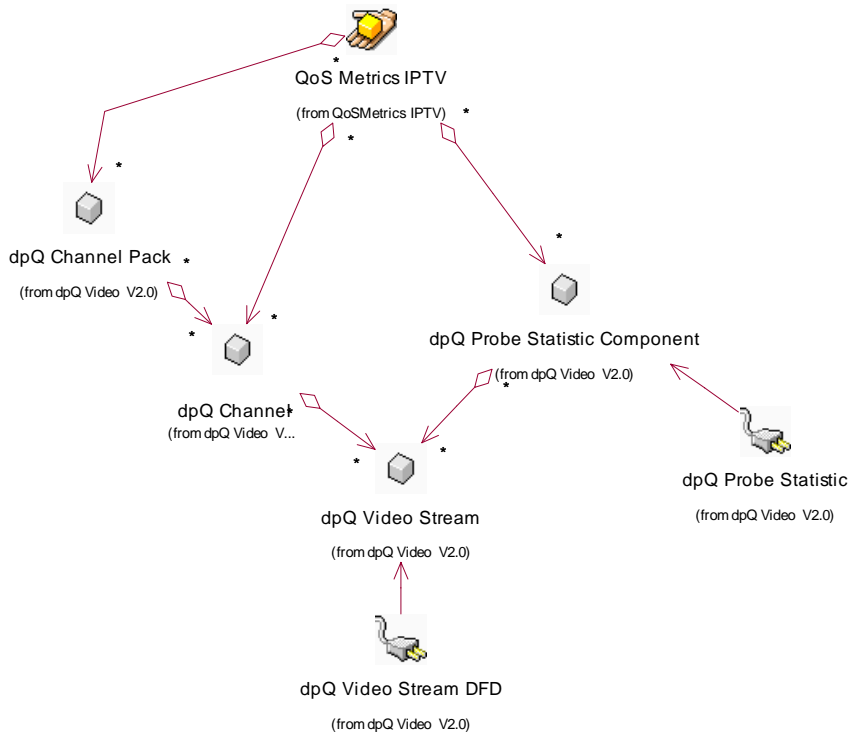
For more details on table structure consult QoSmetrics documentation.

3.2 IPTV QoSmetrics Model

The model is designed to cover two purposes:

- Provide the model definition for automatic service instantiation process implemented in QoSmetrics SA.
- Provide the basic components for collecting Video KPI in SQM.

If you want higher level indication, you can use the IPMS Value Packs or build you on model on top of these components.



Please refer to the “QoS Metrics IPTV.doc” document to get a full description of the QoSMetrics IPTV service model and KPIs available. This file is located in [Installation directory]/qosmetrics_sa/doc.

3.2.1 Automatic Service Instantiation

Each time new data feeder instances are discovered, the QoSMetrics SA can update the associated service instances.

The DFI MRP contains a “Cname” and “Probe” properties. The values of these properties are used to group DFI per channel and site. Thus, each DFI belongs to two service instances: one that groups all DFIs for the same probe (Probe XXX Services) and another that groups all DFIs for the same channel (Channel XXX Services).

Example:

Probe Milan 1 → Probe Milan 1 → Chanel1-Milan1
 → Chanel2-Milan1
 → Chanel2-Milan1

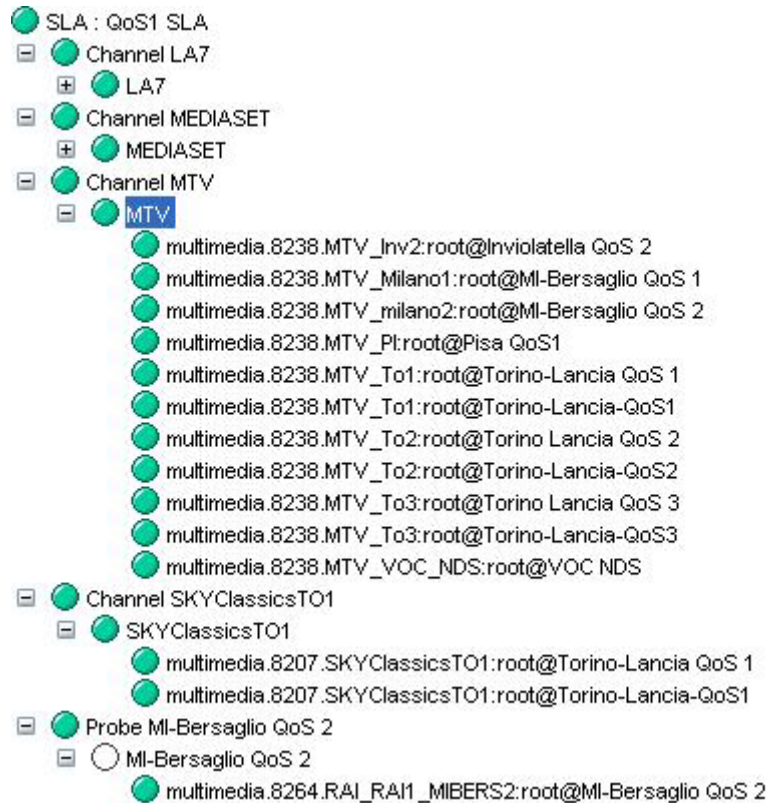
Channel 1 → Channel 1 → Chanel1-Milan1
 → Chanel1-Milan 2
 → Chanel1 – Roma 1

With this grouping it is possible to aggregate collected data per Region (probe) and per Channel.

Note

For each SI, a SCI with the same name is created. This is because SI cannot be shared with other SI, only SCI can be re-used in other SI.

Example of monitored SI:



Note

For Passive Test, the SI hierarchy is not maintained correctly yet.

3.2.2 Computed KQIs

This is a non-exhaustive list of the computed KQIs.

3.2.2.1 Video Stream Service Component Definition

This Service Component Definition (SCD) collects raw data from QoSmetrics NetAdvisor, but also provides Time Aggregated KQIs.

Two kinds of time aggregated parameters exist:

- The one that are aggregated by calculation engine thanks to the election policy: VFactor(min), VFactor(max)
- The one computed using SQM Time Aggregation feature: VFactor Avg (Running Hour)

3.2.2.2 Channel Service Component Definition

The Channel SCD aggregate several Video Stream components that collect data for the same channel, on different points of the network.

The main purpose of this SCD is to be able to pinpoint if the Video Stream has been degraded somewhere in the network.

For this, the SCD provide several KQI that compare data from several Video Streams and indicate if a problem occurs in the transition.

VFactor Lost: if > 0 , this means that video degradation has occurred for some location.

Please refer to the “QoS Metrics IPTV.doc” document to get a full description of the QoSmetrics IPTV service model and KPIs available.

3.2.2.3 Probe Service Component Definition

This SCD aggregates Video Stream data collected by the same probe.

It has two data sources: a “dpQ Probe Stat” data feeder and its child Video Stream SCI.

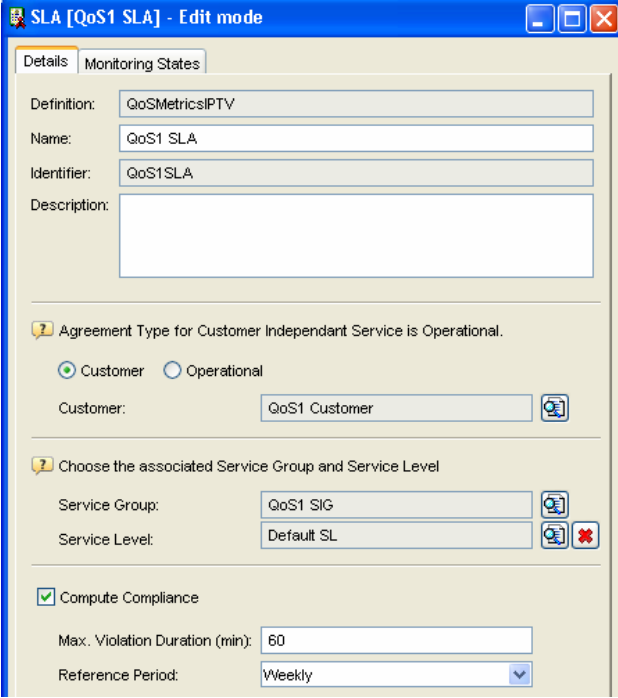
The main purpose of this SCD is to be able to control the overall quality of the IPTV service for a given location (probe).

Please refer to the “QoS Metrics IPTV.doc” document to get a full description of the QoSmetrics IPTV service model and KPIs available.

3.3 Deployment Scenario

3.3.1 Simplest Case

- Install the QoSmetrics SA on the OV SQM platform
- Load the delivered model.
- Load the default Service Level
- Create a new SLA that uses the Default Service Level and the Default SIG. The SLA must be customer dependent and be unlocked.



The screenshot shows a software window titled "SLA [QoS1 SLA] - Edit mode". It has two tabs: "Details" (selected) and "Monitoring States". The "Details" tab contains the following fields and options:

- Definition: QoSMetricsIPTV
- Name: QoS1 SLA
- Identifier: QoS1SLA
- Description: (empty text area)
- Agreement Type for Customer Independant Service is Operational:
 - Customer
 - Operational
- Customer: QoS1 Customer
- Choose the associated Service Group and Service Level:
 - Service Group: QoS1 SIG
 - Service Level: Default SL
- Compute Compliance
- Max. Violation Duration (min): 60
- Reference Period: Weekly

Start the QoS SA with `--createsi, --discover` options.

→ The NetAdvisor is fully monitored in your SQM platform.

Note

For the first collection step, you will have some warning messages indicating missing DFI. This is a normal behavior: when the discovery option is requested, the QoSmetrics SA automatically creates missing DFIs associated SIs.

3.3.2 Dataload

This section explains how to load historical data in the past, some specific tuning are needed.

- You need to specify the value for `--starttime` option. If you don't know it, you can look in `tblUserResultTables` (Date column). The `--starttime` option is convenient since you can specify a value in epoch format (the one stored in the `tblUserResultTables`) or in human readable format (“YYYY/MM/DDTHH:MI:SS”).
- Depending on the length of the history to load, the platform characteristic and the time you have to complete the dataload, you can tune the collection with the `--period, --afactor, --resolution` options.

The `--period` should be small (5 or 10) to have a high rate of collection steps.

The `--afactor` should be high (600) to speed-up the collection. A value of 700 allows loading 1 year of data in approximately 12 hours.

The `--resolution` depends on your requirements. If you tolerate raw aggregation for past data, you can use this parameter to lower input rate (300 => 1 measure published each 5 minutes of history). The resolution must be lower than `[afactor] * [period]` otherwise it's not effective and the value of `[afactor] * [period]` will be used instead.

Note

These parameters can be changed during the dataload if the initial ones are not suitable (using the Web Services).

