
HP OpenView Service Quality Manager



Datamart Installation, Configuration and Administration Guide

Edition: 1.4

for the HP-UX Operating System

April 2007

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

Legal Notices

Warranty

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.

The information contained herein is subject to change without notice.

Restricted Rights 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 2004-2006, 2007 Hewlett-Packard Development Company, L.P.

Trademark Notices

Adobe®, Acrobat®, and PostScript® are trademarks of Adobe Systems Incorporated.

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.

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

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

UNIX® is a registered trademark of The Open Group.

Contents

Preface	5
Part I.....	7
Installing the OpenView Service Quality Manager DataMart.....	7
Part II.....	9
Configuration and Administration.....	9
Chapter 1	11
DataMart Architecture Overview.....	11
1.1 General Architectural Overview	11
Chapter 2	13
DataMart General Configuration.....	13
2.1 Configuration Input Data List	13
Chapter 3	17
Creating and Configuring the DataMart Databases	17
3.1 General	17
3.1.1 Database Naming Plan.....	17
3.2 Security	17
Chapter 4	19
Configuring the DataMart Migration Component.....	19
4.1 Initial Configuration	19
4.1.1 General Configuration Fields.....	19
4.1.2 Database Related Configuration Fields.....	22
4.1.3 Advanced Configuration Fields.....	24
4.2 Reloading the DataMart Configuration - "hot reload".....	26
4.3 Stopping the DataMart Migration Component	27
Chapter 5	29
Deleting the DataMart Databases	29
5.1 Logging On.....	29
5.2 Deleting the Databases.....	29
Chapter 6	30

Troubleshooting	30
6.1 Dump.....	30
6.2 Log and Trace Files	31
6.2.1 Log File	31
6.2.2 Trace Files	31
Glossary	33
Index	37

Preface

The OpenView Service Quality Manager DataMart is part of the Hewlett-Packard solution for Service Quality Management. This manual describes the installation and the configuration of the DataMart in the context of the complete OpenView Service Quality Manager solution.

Intended Audience

This document is intended for personnel who install and configure the OpenView Service Quality Manager DataMart. Prior knowledge of the OpenView Service Quality Manager platform and core components is a prerequisite to fully appreciate its contents.

Supported Software

The supported software referred to in this document is as follows:

Product Version	Operating System
OpenView Service Quality Manager 1.4	HP-UX 11.11

The term UNIX is used as a generic reference to the operating system, unless otherwise specified.

Typographical Conventions

Courier Font:

- Source code and examples of file contents.
- Commands that you enter on the screen.
- Pathnames
- 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 following documents contain useful reference information:

- *HP OpenView Service Quality Manager Administration Guide.*
- *HP OpenView Service Quality Manager Reference Guide for Oracle Use.*
- *Oracle Database Concepts 10g Release 2.*
- *HP OpenView Service Quality Manager DataMart User's Guide.*
- *HP OpenView Service Quality Manager Installation 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

Installing the OpenView Service Quality Manager DataMart

The OpenView Service Quality Manager DataMart (a.k.a. DataMart) is installed and set-up with SQM. For more information please refer to the hp OV SQM installation guide.

This guide provides a detailed description of all DataMart configuration capabilities.

Part II

Configuration and Administration

This part provides the configuration instructions for the OpenView Service Quality Manager DataMart.

DataMart Architecture Overview

This chapter describes the architecture of the DataMart. All the configuration related features are detailed in the following chapters, which show what the SQM administrator will have to configure.

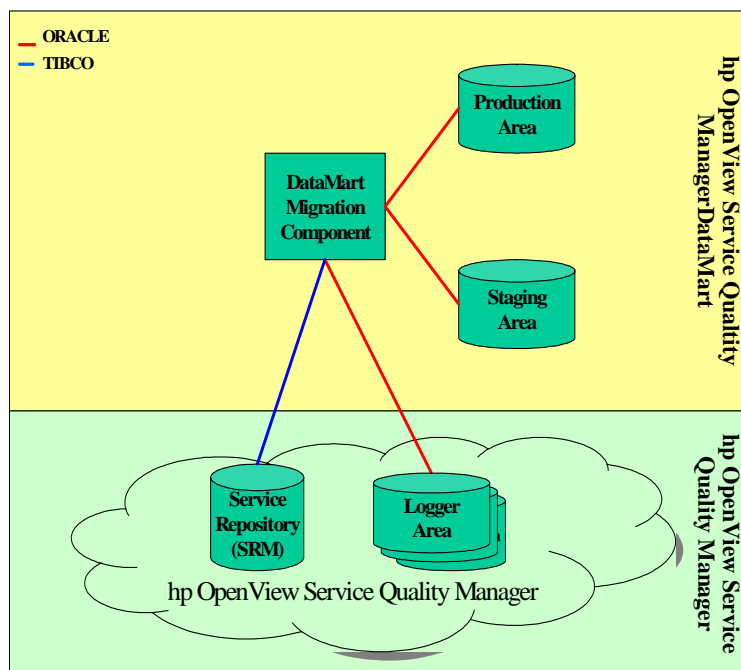
1.1 General Architectural Overview

The OpenView Service Quality Manager DataMart provides the data warehousing functionality basis for the SQM product.

The DataMart is composed of a TIBCO component called the DataMart Migration Component.

The DataMart architecture is described hereafter:

Figure 1 General Architectural Overview



The DataMart is composed of:

- The *DataMart Migration Component*: this is a TIBCO adapter, written in Java, which performs all of the processing, including reading and processing the logger records and constructing the production area.
- The Oracle databases: the *production* area and the *staging* area are all Oracle databases created and filled by the DataMart functions; more precisely, by the DataMart Migration Component.

For more information on the DataMart, refer to the *OpenView Service Quality Manager DataMart User's Guide*.

Chapter 2

DataMart General Configuration

2.1 Configuration Input Data List

This chapter provides a list of all the configuration data required for the DataMart database and for the DataMart Migration Component.

The following table summarizes the data configuration used for the database creation. These data are defined in the configuration file *temip_database_product_configuration.cfg* located in:

`$TEMIP_SC_HOME/oracle/product/conf`

where *product* stands for:

- *dmprod* for the Production database.
- *dmstag* for the Staging database.

Table 1 DataMart Database Configuration

Production Database	Default Value
Production SID	dmprod
Production User Name	production
DataMart Granularity(min)	5
Number of years used to generate the time dimension	2
Staging Database	Default Value
Staging SID	dmstag
Staging User Name	staging

The next table summarizes all the DataMart Migration Component configuration fields stored in the Central Configuration Repository. Refer to the *OpenView Service Quality Manager Administration Guide* on how to use the Central Configuration Repository. A description of each of the following fields is provided in Section 4.1.

Table 2 DataMart Migration Component Data Configuration

General Data Configuration Fields	Default Values
DataMart Granularity (min)	5
Data Collection Interval (min)	%%SCDataAcquisitionPeriod(min)%% ¹
Log Reading Period (min)	%%SCDataStagingPeriod(min)%%
Summarization Period (min)	%%SCDataSummarizationPeriod%%
Rows Commit Number	200
Production Area Management Mode	Delete
SLA Status Storage Duration (quarters)	3
Raw Data Storage Duration (days)	31
Hourly Summarized Data Storage Duration (months)	1
Daily Summarized Data Storage Duration (months)	1
Weekly Summarized Data Storage Duration (weeks)	53
Monthly Summarized Data Storage Duration (months)	12
Quarterly Summarized Data Storage Duration (quarters)	20
Yearly Summarized Data Storage Duration (years)	10
Enumeration Maximum Number of Values	20
List of Archive Database References	No default value
TNS Directory	%%SCTeMIPNameSpaceDirectory%%
TNS Namespace	%%SCTeMIPNameSpace%%
Violation State Level	%%SCStateViolationLevel%%
Degradation State Level	%%SCStateDegradationLevel%%
XML Validation	%%SCXmlMsgValidation%%
SRM Request Reply Timeout(ms)	%%SCInternalRequestRepliesTimeout(ms)%%
SRM Request Reply Retry Nb	%%SCInternalRequestRepliesRetryNb%%
Log Record Processing Retry Number	1

¹ The variables starting and ending with “%%” refer to TIBCO global variables. Refer to the *OpenView Service Quality Manager Configuration and Administration Guide* for their values.

Max Number Of Log Records To Process	100000
Do Not Process Performance Measures In Degraded Mode	true
Init Model At Startup	false
Max Number Of SRM Records	1000
Scheduled Late Performance Calculation	true
Store No Value Parameters	true

Database Configuration Fields	
DriverType	Thin
DBProductionHost	No default value
DBProductionSID	dmprod
ProductionListenerPort	1521
ProductionUserName	production
ProductionPassword	production
ProductionNetServiceName	(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=__DBProductionHost__)(PORT=__ProductionListenerPort__)))CONNECT_DATA=(SID=__DBProductionSID__)))
DBStagingHost	No default value
DBStagingSID	dmstag
StagingListenerPort	1521
StagingUserName	staging
StagingPassword	staging
StagingNetServiceName	(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=__DBStagingHost__)(PORT=__StagingListenerPort__)))CONNECT_DATA=(SID=__DBStagingSID__)))

Chapter 3

Creating and Configuring the DataMart Databases

3.1 General

The database creation is carried out with the *temip_create_database* script. Refer to the *OpenView Service Quality Manager Reference Guide for Oracle Use* for more details on this tool and for any Oracle configuration questions.

3.1.1 Database Naming Plan

The table below summarizes the pre-defined naming plan for the DataMart databases.

Table 3 DataMart Database SID

HP OpenView Service Quality Manager DataMart Databases	Pre-Defined SID
Production Database	dmprod
Staging Database	dmstag

It is recommended not to change the default naming schema in order to deploy a homogeneous platform.

3.2 Security

3.2.1.1 User Authentication

Each DataMart database has its own user to connect to. The following table lists the usernames created within each specific database.

Table 4 DataMart Database User Names

DataMart Databases	Dedicated User Names
Production Database	production
Staging Database	staging

3.2.1.2 User Rights

None of the above listed users is granted "DBA" right.

Each user has a limited set of roles and privileges. The following table summarizes the granted privileges per user (production, staging).

Table 5 Staging User Granted Privileges

Staging User Privileges	Description
Select_catalog_role	This role is provided for accessing data dictionary views and packages
Alter session	Issue ALTER SESSION statements
Create table	Create tables in grantee's schema
Create sequence	Create sequences in grantee's schema
Create database link	Create private database links in grantee's schema
Unlimited tablespace	Use an unlimited amount of any tablespace. This privilege overrides any specific quotas assigned.

Table 6 Production User Granted Privileges

Production User Privileges	Description
Select_catalog_role	This role is provided for accessing data dictionary views and packages
Alter session	Issue ALTER SESSION statements
Create table	Create tables in grantee's schema
Create sequence	Create sequences in grantee's schema
Create database link	Create private database links in grantee's schema
Unlimited tablespace	Use an unlimited amount of any tablespace. This privilege overrides any specific quotas assigned.

Configuring the DataMart Migration Component

The DataMart Migration Component is configured with a set of fields that are permanently stored within the Tibco repository.

4.1 Initial Configuration

The DataMart Migration Component Tibco Adapter configuration can be performed at two stages:

- Before starting the DataMart Migration Component for the first time.
- While the DataMart Migration Component is already running.

The difference between these two stages concerns the data that can be updated depending on when this is done. Indeed, not all the fields can be updated once the DataMart Migration Component has been initialized (databases created).

The DataMart Migration Component parameters are stored in the central SQM repository. They are accessible in Read/Write mode from the Tibco Adapter Administrator tool, the Tibco Designer. The parameters that are specific to the DataMart Migration Component are described in the following sections.

4.1.1 General Configuration Fields

4.1.1.1 DataMart Granularity

The DataMart Granularity (or Time Grain) is the granularity of the Time dimension.

Important Note

The value of this configuration field cannot be modified. It cannot especially be changed by the "ReloadConfig" AMI method.

The associated TIBCO configuration field is named "*DataMart Granularity(min)*" and its default value is 5.

4.1.1.2 Data Collection Interval

This parameter is obsolete since SQM 1.3. It has been removed from the Tibco repository.

4.1.1.3 Log Reading Period

The Log Reading Period field is the scheduling period of the staging process. It must be greater than or equal to the DataMart Granularity, and must be a multiple of it. For example, every Log Reading session, N performance facts for the same service or component are added to the corresponding fact table.

The associated TIBCO configuration field is named "*Log Reading Period(min)*" and its default value is the `%%SCDataStagingPeriod(min)%%` global variable.

When the staging process takes more than the Log Reading Period, the next Log Reading session is started as soon as the current one finishes.

4.1.1.4 Summarization Period

The Summarization Period field is the scheduling period of the summarization process. It must be greater than or equal to the Log Reading Period, and must be a multiple of it (that is, every N Log Reading session, a summarization is performed).

The associated TIBCO configuration field is named "*Summarization Period(min)*" and its default value is the `%%SCDataSummarizationPeriod(min)%%` global variable.

When the summarization process takes more than the Summarization Period, the next Summarization session is started as soon as the current one finishes.

4.1.1.5 Production Area Management Mode

The Production Area Management Mode field is the action performed during each archiving session, once a day. It concerns only the facts and not the dimensions, and only for the data matching the *Production Area Management Storage Duration*.

The list of possible values is:

- **Delete:** All data are deleted from the *Production Area*.
- **NoAction:** No action is performed.

The associated TIBCO configuration field is named "*Production Area Management Mode*" and its default value is "*Delete*". For example, if the *SLA Status Storage Duration* is equal to 3, then the corresponding facts for this SLA will be kept for three quarters.

4.1.1.6 Raw Data Storage Duration

Only the data with a *TimeId* (which corresponds to the Creation Time Stamp of the fact) column (raw data fact tables) older than this value will be deleted (mode *Delete*) from the *Production Area*.

If the *Production Area Management Mode* attribute is set to *NoAction* this attribute is not taken into account.

The associated TIBCO configuration field is named "*Raw Data Storage Duration(days)*" and its default value is *31*.

4.1.1.7 Hourly Summarized Data Storage Duration

Only the data with a *TimeId* (which corresponds to the Creation Time Stamp of the fact) column (hourly summarized fact tables) older than this value will be deleted (mode *Delete*) from the *Production Area*.

If the *Production Area Management Mode* attribute is set to *NoAction*, this attribute is not taken into account. The associated TIBCO configuration field is named "*Hourly Summarized Data Storage Duration(months)*" and its default value is *1*.

4.1.1.8 Daily Summarized Data Storage Duration

Only the data with a *TimeId* (which corresponds to the Creation Time Stamp of the fact) column (daily summarized fact tables) older than this value will be deleted (mode *Delete*) from the *Production Area*.

If the *Production Area Management Mode* attribute is set to *NoAction* this attribute is not taken into account.

The associated TIBCO configuration field is named "*Daily Summarized Data Storage Duration(months)*" and its default value is *1*.

4.1.1.9 Weekly Summarized Data Storage Duration

Only the data with a *TimeId* (which corresponds to the Creation Time Stamp of the fact) column (weekly summarized fact tables) older than this value will be deleted (mode *Delete*) from the *Production Area*.

If the *Production Area Management Mode* attribute is set to *NoAction* this attribute is not taken into account.

The associated TIBCO configuration field is named "*Weekly Summarized Data Storage Duration(weeks)*" and its default value is *53*.

4.1.1.10 Monthly Summarized Data Storage Duration

Only the data with a *TimeId* (which corresponds to the Creation Time Stamp of the fact) column (monthly summarized fact tables) older than this value will be deleted (mode *Delete*) from the *Production Area*.

If the *Production Area Management Mode* attribute is set to *NoAction* this attribute is not taken into account.

The associated TIBCO configuration field is named "*Monthly Summarized Data Storage Duration(months)*" and its default value is *12*.

4.1.1.11 Quarterly Summarized Data Storage Duration

Only the data with a *TimeId* (which corresponds to the Creation Time Stamp of the fact) column (quarterly summarized fact tables) older than this value will be deleted (mode *Delete*) from the *Production Area*.

If the *Production Area Management Mode* attribute is set to *NoAction* this attribute is not taken into account.

The associated TIBCO configuration field is named "*Quarterly Summarized Data Storage Duration(quarters)*" and its default value is *20*.

4.1.1.12 Yearly Summarized Data Storage Duration

Only the data with a *TimeId* (which corresponds to the Creation Time Stamp of the fact) column (yearly summarized fact tables) older than this value will be deleted (mode *Delete*) from the *Production Area*.

If the *Production Area Management Mode* attribute is set to *NoAction* this attribute is not taken into account.

The associated TIBCO configuration field is named "*Yearly Summarized Data Storage Duration(years)*" and its default value is *10*.

4.1.1.13 List of Logger Database References

This field contains the list of TIBCO references pointing to the Logger Configuration Area of each Logger Component. This represents the set of Archive Areas that will all be read by the DataMart Migration Component.

The associated TIBCO configuration field is named "*List of Archive Database References*" and there is no default value.

4.1.1.14 TNS Directory

The TNS Directory is retrieved from the SQM Repository.

The associated TIBCO configuration field is named "*TNS Directory*" and its default value is the `%%SCTeMIPNameSpaceDirectory%%` global variable.

4.1.1.15 TNS Namespace

The TNS Namespace is retrieved from the SQM Repository.

The associated TIBCO configuration field is named "*TNS Namespace*" and its default value is the `%%SCTeMIPNameSpace%%` global variable.

4.1.1.16 Violation State Level

The Violation State Level field is retrieved from the SQM Repository. Note that the modification of this value has an impact on the Health Indicators calculation.

The associated TIBCO configuration field is named "*Violation State Level*" and its default value is the `%%SCStateViolationLevel%%` global variable.

4.1.1.17 Degradation State Level

The Degradation State Level field is retrieved from SQM Repository. Note that the modification of this value has an impact on the Health Indicators calculation.

The associated TIBCO configuration field is named "*Degradation State Level*" and its default value is the `%%SCDegradationStateLevel%%` global variable.

4.1.2 Database Related Configuration Fields

4.1.2.1 Driver Type

This field indicates the JDBC driver that should be used for the DataMart Migration Component.

The associated TIBCO configuration field is named "*Driver Type*" and the possible values are:

- *Thin*: The JDBC thin driver is a 100% pure Java driver that uses Java sockets to connect directly to a database server. Because it is written entirely in Java, this driver is platform-independent. It does not require any additional Oracle software on the client side.
- *Oci*: The OCI driver is a driver that makes calls to the OCI (Oracle Call Interface) that is written in C, to interact with an Oracle database server, thus using native and Java methods. This driver requires an Oracle client installation, and therefore is Oracle platform-specific. The OCI driver usually executes faster than the thin driver.

The default value is the *thin* driver.

4.1.2.2 Production Host

This field represents the host name hosting the Production Area.

The associated TIBCO configuration field is named *"DBProductionHost"* and there is no default value.

4.1.2.3 Production SID

This field represents the Oracle SID for the Production Area.

The associated TIBCO configuration field is named *"DBProductionSID"* and its default value is *"dmprod"*.

4.1.2.4 Production Listener Port

This field represents the Oracle listener port for the Production Area access.

The associated TIBCO configuration field is named *"ProductionListenerPort"* and its default value is *1521*.

4.1.2.5 Production User Name

This field represents the Oracle username for the Production Area access.

The associated TIBCO configuration field is named *"ProductionUserName"* and its default value is *"production"*.

4.1.2.6 Production Password

This field represents the Oracle password for the Production Area access.

The associated TIBCO configuration field is named *"ProductionPassword"* and its default value is *"production"*.

4.1.2.7 Production Net Service Name

This parameter defines the Oracle Net Service that targets the Production Oracle Database.

The associated TIBCO configuration field is named *"ProductionNetServiceName"* and the default value is

```
"(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)
(HOST=__DBProductionHost__) (PORT=__ProductionListenerPort__))
(CONNECT_DATA=( SID=__DBProductionSID__)))"
```

Note that this parameter is not used when the driver type is set to *thin* driver.

4.1.2.8 Staging Host

This field represents the host name hosting the Staging Area.

The associated TIBCO configuration field is named *"DBStagingHost"* and there is no default value.

4.1.2.9 Staging SID

This field represents the Oracle SID for the Staging Area.

The associated TIBCO configuration field is named *"DBStagingSID"* and its default value is *"dmstag"*.

4.1.2.10 Staging Listener Port

This field represents the Oracle listener port for the Staging Area access.

The associated TIBCO configuration field is named "*StagingListenerPort*" and its default value is *1521*.

4.1.2.11 Staging User Name

This field represents the Oracle username for the Staging Area access.

The associated TIBCO configuration field is named "*StagingUserName*" and its default value is "*staging*".

4.1.2.12 Staging Password

This field represents the Oracle password for the Staging Area access.

The associated TIBCO configuration field is named "*StagingPassword*" and its default value is "*staging*".

4.1.2.13 Staging Net Service Name

This parameter defines the Oracle Net Service that targets the Staging Oracle database.

The associated TIBCO configuration field is named "*StagingNetServiceName*" and the default value is

```
"(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)
(HOST=__DBStagingHost__) (PORT=__StagingListenerPort)))
(CONNECT_DATA=( SID=__DBStagingSID__)))".
```

Note that this parameter is not used when the driver type is set to *thin* driver.

4.1.3 Advanced Configuration Fields

4.1.3.1 Rows Commit Number

The Rows Commit Number field is the number of rows between two SQL Commits. This design variable permits the optimization of the rollback segments size, because the volume of data to store in the rollback segment is never greater than the "Rows Commit Number". This mechanism is used while copying data into the *Production Area*.

The associated TIBCO configuration field is named "*Rows Commit Number*" and its default value is *1000*.

4.1.3.2 Enumeration Maximum Number of Values

The Enumeration Maximum Number of Values is the number of distinct values to store per enumeration datatype.

The associated TIBCO configuration field is named "*Enumeration Maximum Number of Values*" and its default value is *20*.

This means that only the 20 first (in the order they appear in the definition) enumeration values will be stored and summarized.

4.1.3.3 XML Validation

This field can be used for debugging purposes. It is used to check the validity of all the XML messages present in the DataMart Migration Component. This field should not be modified except if it is explicitly requested by the SQM support.

The associated TIBCO configuration field is named "*XML Validation*" and its default value is *false*.

4.1.3.4 SRM Request Reply Timeout(ms)

This field is used to tune the time allowed to the SRM component before considering its reply is late, hence implying a timeout.

The associated TIBCO configuration field is named "*SCInternalRequestRepliesTimeout(s)*" and its default value is *10*.

4.1.3.5 Max Number of SRM Records

This field is used to tune the size of replies sent by the SRM. If more than the specified number of records are to be returned then the replies are managed by a handle more mechanism and sent in several messages.

4.1.3.6 Log Record Processing Retry Number

This field is used by the DataMart Migration Component while processing the records read from the Logger Component export table.

Two cases can arise when an error occurs at this stage:

- The error is considered as *fatal*: the record is flagged as processed and will be removed at the end of the staging session. Then the DataMart Migration Component continues its normal process.
- The error is considered as *non fatal* (temporary or transient problem): the record is not flagged as process, but the number of attempts to process this record is incremented. Then the DataMart Migration Component stops its staging session. When the column value is greater than the Log Record Processing Retry Number, it is considered that the error can not be recovered. Therefore the error is considered as *fatal*: the export table line is discarded, and the DataMart Migration Component continues with the remaining lines in the export table.

4.1.3.7 Max Number Of Log Records To Process

This parameter indicated the maximum number of messages to be processed by the DataMart in a staging session, before it enters the degraded mode default is 100000. When the number of records exceeds this value, the parameters with a reporting property set to "Disable When Late" are ignored.

4.1.3.8 Do Not Process Performance Measures In Degraded Mode

This parameter can be valued to true (the default) or false. When set to true, the datamart does not process the performance messages when the number of records to process exceeds the safety level - calculated as a percentage (150% by default) of the maximum number of log records to process. When set to false, the datamart never exclude messages.

4.1.3.9 Init Model At Startup

This parameter can be valued to true or false (the default). When set to true, the datamart queries the Service Repository Manager during its starting phase in order to build or update its internal representation of the Service Models. If it makes the startup phase longer, it avoids on-the-fly discovery that can impact runtime performances.

4.1.3.10 Max OS Lag Time (days)

This parameter is obsolete since SQM 1.3. It has been removed from the Tibco repository.

4.1.3.11 Pad Up To Staging Start Time

This parameter is obsolete since SQM 1.3. It has been removed from the Tibco repository.

4.1.3.12 Max Number Of SRM Records

This parameter specifies the maximal number of service instance that should be retrieved by the SRM in a unique reply; the default is set to 1000. If the number to retrieve is greater than the fixed limit multiple replies will be sent in a handle more mode.

4.1.3.13 Scheduled Late Performance Calculation

This parameter valued to True or False specifies if the late calculation sessions created by the `sqm_dm_load_perf_correction` tool, are to be launched automatically by the DataMart at the same time as the Summarization or manually using the dedicated AMI.

4.1.3.14 Store No Value Parameters

This parameter is valued to true (the default) or false. Before SQM 1.4 the datamart used to discard measures with NoValue flag set to true. In SQM 1.4, the datamart process these measures and computes a new indicator giving the time while a parameter has no value. This parameter indicates if the datamart stores the No Value measures or discard them as in SQM 1.3.

4.2 Reloading the DataMart Configuration - "hot reload"

The SQM user can update the DataMart configuration and request the DataMart Migration Component to reload its updated configuration. The DataMart Migration Component then applies the new configuration to the next staging and summarization sessions. All the previously described fields except the DataMart Granularity can be updated by using the TIBCO Designer application.

The Reload Configuration directive is used to force the reload of the DataMart Migration Component configuration. Refer to the *OpenView Service Quality Manager Configuration and Administration Guide* for more information on how to use the Designer and the Hawk Display applications.

This method can be called through the HawkDisplay application. Once the DataMartHawkMicroAgent is selected, you can select the reloadConfig method and invoke it as shown in the following figure:

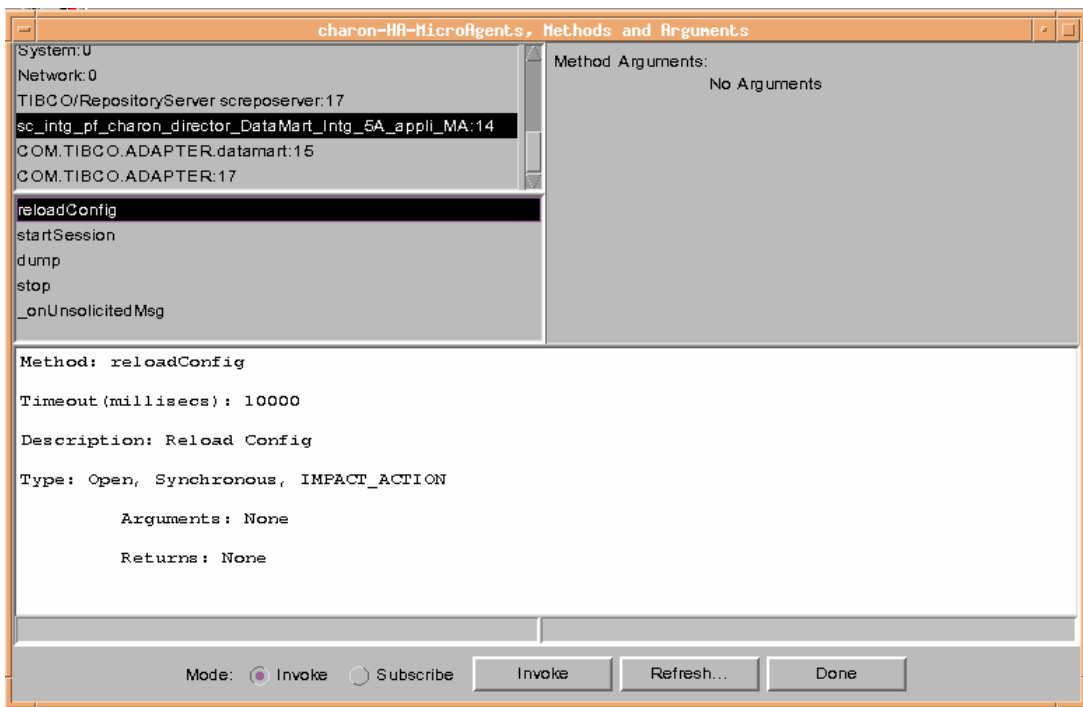


Figure 2 Invoking the reloadConfig AMI method

This method does not require any arguments. Once invoked, the DataMart Migration Component will reload its configuration for the next staging/summarization sessions.

4.3 Stopping the DataMart Migration Component

The DataMart Migration Component can be stopped at any time by using the *temip_sc_application_stop* script. Refer to the *OpenView Service Quality Manager Administration Guide* for more information on how to use this script. Once invoked, the DataMart Migration Component will stop once the currently running sessions (if any) are terminated.

When the DataMart Migration Component is restarted, it reads the new data made available in the Logger database(s). The read information is then transformed into raw facts that are finally stored into the Production Area. As for the cold start, the DataMart Migration Component does not recover any of the available model information for the objects that may have been created before being restarted. Indeed, the DataMart Migration Component can be plugged into an already functioning system. It is then filled and built at every staging session.

Note

When you turn the 'Init Model At Startup' to true the DataMart will first query for model updates before starting to process collected measures.

Deleting the DataMart Databases

The database deletion is done with the *temip_delete_database* script. Refer to the *OpenView Service Quality Manager Reference Guide for Oracle Use* for more details on this tool and for any Oracle configuration questions.

5.1 Logging On

Before starting the DataMart database deletion, log-on to the administrator account on the system where the DataMart databases have been created.

5.2 Deleting the Databases

Usually you should invoke *temip_delete_database* without any parameters, in order to enter an interactive mode that displays a list of detected (existing) databases from which you choose the one you want to delete. You can also invoke this script with the required Database SID as parameter.

Table 5-1 DataMart Database SID

DataMart Databases	Pre-Defined SID
Production Database	dmprod
Staging Database	dmstag

Note

You must manually edit the *oratab* and *listener.ora* files² to remove static information related to the destroyed database.

² Those files are located in the \$ORACLE_ROOT/network/admin directory

Chapter 6

Troubleshooting

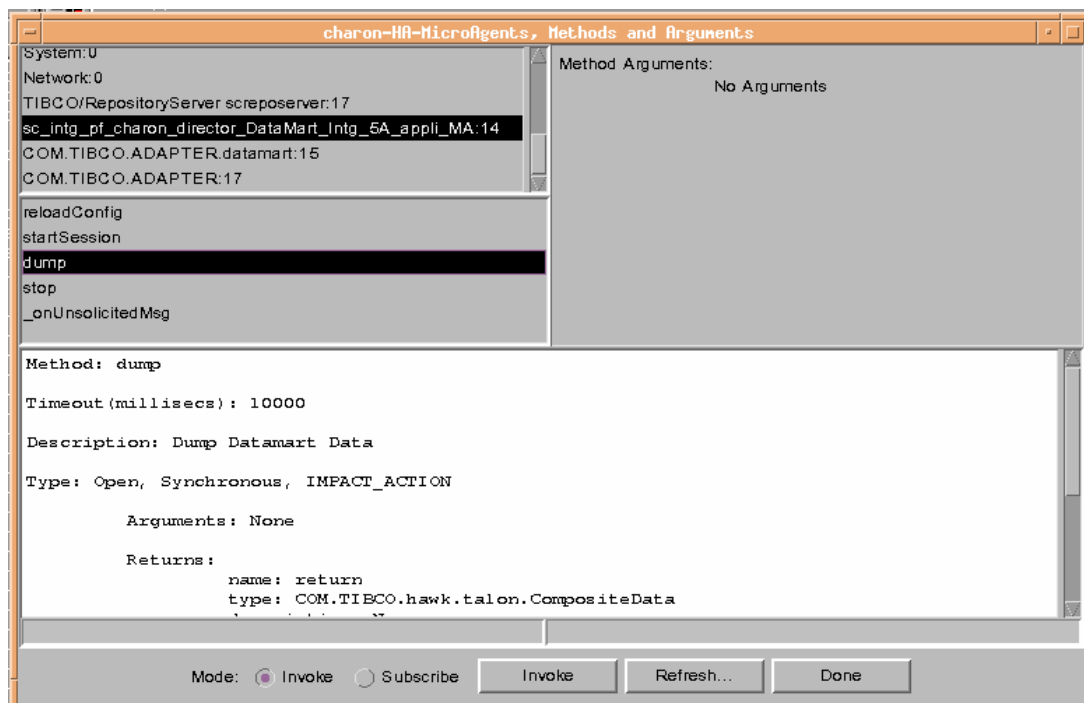
6.1 Dump

The DataMart Migration Component implements a dump AMI method allowing the display of its internal data:

- The DataMart Migration Component configuration.
- The currently processed message (if any).
- The currently active session (staging/summarization or archiving session).

This method can be called through the HawkDisplay application. Once the DataMartHawkMicroAgent is selected, you can select the dump method and invoke it as shown in the following figure:

Figure 3 Invoking the Dump AMI Method



6.2 Log and Trace Files

6.2.1 Log File

In order to help troubleshooting of the DataMart Migration Component, a new log line is appended to the file in the `$TEMIP_SC_VAR_HOME/log` directory when an unexpected error occurs. The log file name is *SCPlatform_SCDirector_SCAApplication.log*. You can check the content of the log file to discover and understand the problem.

To activate the logging feature, modify the *SCPlatform_SCDirector_SCAApplication..properties* file located in the `$TEMIP_SC_VAR_HOME/DataMart/config` directory. Set the `MTLog.level` to "ALL" to have full logs.

```
MTLog.level=ALL
```

The possible values are:

- ALL: full logs.
- MINOR: only CRITICAL, MAJOR and MINOR logs are dumped.
- MAJOR: only the CRITICAL and MAJOR logs are dumped.
- CRITICAL: only the CRITICAL logs are dumped.
- OFF: disable the logging mechanism.

Once you have modified this field, you have to stop and restart the DataMart Migration Component. The Reload Configuration AMI method does not take into account those modifications. You have to use the *temip_sc_stop_application* and *temip_sc_start_application* scripts. Refer to the *OpenView Service Quality Manager Administration Guide* for more information on how to use these scripts.

6.2.2 Trace Files

6.2.2.1 Java Trace File

The DataMart Migration Component can generate trace files. These traces are located in the `$TEMIP_SC_VAR_HOME/trace` folder. The trace file name is *SCPlatform_SCDirector_SCAApplication.log*

To activate the trace logging, modify the *SCPlatform_SCDirector_SCAApplication..properties* file located in the `$TEMIP_SC_VAR_HOME/DataMart/config` directory. Set the `TraceLog.level` to "All" to have full traces.

```
TraceLog.level=All
```

The possible values are:

- ALL: full traces.
- FINEST: only FINE, FINER and FINEST traces are dumped.
- FINER: only the FINE and FINER traces are dumped.
- FINE: only the FINE traces are dumped.
- OFF: disable the tracing mechanism.

Once you have modified this field, you have to stop and restart the DataMart Migration Component. The Reload Configuration AMI method does not take into account those modifications. You have to use the *temip_sc_stop_application* and

temip_sc_start_application scripts. Refer to the *OpenView Service Quality Manager Administration Guide* for more information on how to use these scripts.

Changing the Java Trace level without stopping the DataMart Migration Component

If you want to change the Java Trace level without stopping the DataMart Migration Component, you can use the `SetTraceLogLevel` AMI method. Refer to the *OpenView Service Quality Manager Administration Guide* for more information

6.2.2.2 SQL Trace Files

SQL traces can be also dumped in the above introduced trace file. Trace level can be set at runtime in each database as appropriate

Production and staging database traces can be enabled separately using a script in `$STEMIP_SC_HOME/support`. The script is *set_dm_db_trace.sh*

Production database,

```
SH> set_dm_db_trace.sh -prod  
(ALL|FINE|FINER|FINEST|OFF)
```

Staging database,

```
SH> set_dm_db_trace.sh -stag  
(ALL|FINE|FINER|FINEST|OFF)
```

Meanwhile they can also be set in one script,

```
SH> set_dm_db_trace.sh -prod  
(ALL|FINE|FINER|FINEST|OFF) -stag  
(ALL|FINE|FINER|FINEST|OFF)
```

To get the current trace level, you can execute *set_dm_db_trace.sh* directly without any parameter

```
SH> set_dm_db_trace.sh
```


Glossary

Terminology commonly used in the OpenView Service Quality Manager.

Auto instantiate (SLA Administration)

This action automatically creates an Instance of the Object selected. When the instance is created, the initial values of its instance variables are assigned.

BI

See Business Intelligence.

Business Intelligence (BI)

A broad category of applications and technologies for gathering, storing, analyzing, and providing access to data that help users to make better business decisions.

Collected binding

Describes how *collected parameters* are filled from *measurement parameters*; either directly assigned or through a more complex expression.

Collected parameters

Known as KPI in the TMF, They represent the parameters collected from the Service Adapters (*measurement parameters*) and mapped into SQM *service component* parameters.

Computed binding

Describes how *computed parameters* are filled from *collected parameters*: either directly assigned or through a more complex expression.

Computed parameters

Known as KQI in the TMF, they represent the parameters calculated from *collected parameters*.

CNM

See Customer Network Management.

Customer

Companies or organizations that make use of the *services* offered by a *service provider*, based on a contractual relationship.

Customer Network Management

Customer Network Management is enabled by means of tools that provide business customers with access to management information originating from the service provider.

Data collection interval

The interval of time over which performance parameters are retrieved from the monitored service resources. This interval does **not** have to be the same as the *measurement interval* because *service adapters* or service resources may buffer statistics.

Data feeder

OpenView Service Quality Manager's source of data. A data feeder models service resources by defining one or more service parameters.

Data feeder definition

The static definition of a data feeder that models service resources by defining one or more service parameters.

Degraded service

The presence of anomalies or defects that cause degradation of the *quality of service*, but do not result in the total failure of the *service*.

Instantiate (SLA Administration)

Instantiate differs from Auto Instantiate in that items are instantiated individually.

Measurement interval

The interval of time over which each service parameter is measured. For example, a parameter may be the number of discarded packets, measured over a 15-minute measurement interval.

Measurement parameters

They represent the parameters directly collected by the Service Adapters. These parameters are defined in the *Data Feeders*.

Measurement Reference Point (MRP) naming scheme

This is the formal description of how the measurement point name is built, that is, by concatenating the values of Data Feeder properties and fixed strings.

Mobile Virtual Network Operator

A mobile operator that does not own its own spectrum and usually does not have its own network infrastructure. Instead, MVNOs have business arrangements with traditional mobile operators to buy **minutes of use** for sale to their own customers.

MRP

See Measurement Reference Point.

MVNO

See mobile virtual network operator.

Parameter

A value or set of values that are periodically updated and that help determine the quality of service.

Parameter objective

A set of objectives for the parameters belonging to a service.

Property

Special static parameters that are given a value only when an instance of an OpenView Service Quality Manager **Object** is created. For example, a Service Component can have a property called "location".

QoS

See quality of service.

Quality of Service (QoS)

The ITU-T has defined quality of service as "the collective effect of service performances that determine the degree of satisfaction of a user of the service".

Service

A Service is a set of independent functions (Service Components) that consist of hardware and software elements and an underlying communications medium. A Service can include anything from a single leased-line service, to a complex application, such as vision conferencing.

Service availability

A measurement made in the context of a *service level agreement* that is expressed as a percentage. This percentage indicates the time during which the *service* is operational at the respective *service access points*.

ServiceCenter Repository

The ServiceCenter Repository is the storage center for all Service Quality Manager data. It receives data from the various Service Quality Manager interfaces and each interface can request information from the Repository.

Service component

An independent function that is part of a *service*, such as a hardware or software element, or the underlying communications medium.

Service component instance

The instance of a Service Component Definition that is active in the network, such as an instance of the IPAccess Service Component definition called "pop".

Service Level (SL)

Defines Service Parameters and operational data enforced by the Service Level Agreement (for example, Max Jitter < 10 ms).

Service Level Agreement (SLA)

There are two types of Service Level Agreement, the **Customer** Agreement: a contract between a *service provider* and a *customer*, that specifies in measurable terms what the service provider supplies to its customers, and the Operational Service Level Agreement, that specifies in measurable terms the operational levels of the Service. A *service level agreement* is composed of individual objectives.

Service Level Objective (SLO)

The set of objectives for the parameters belonging to a Service or Service Component.

Service parameter

See *parameter*.

Service provider

A company or organization that provides *services* as a business. Service providers may operate networks or may integrate the services of other providers.

Service Instance (SI)

The instantiated service definition that is active in the network, such as an instance of the video service definition called "Paris".

Service Instance Group (SIG)

A **group** of *service instances* against which the *service availability* must be reported. Each *service instance* belongs to one or more Service Instance Groups

and each SIG contains at least one Service Instance. The relationship between the SIG and the Service Instances is defined in their *service level agreement*.

Service quality parameters

They represent *computed* and *collected* parameters

SI

See Service Instance.

SIG

See Service Instance Group.

SL

See Service Level.

SLA

See Service Level Agreement.

SLO

See service level objective.

Subscriber

The entity responsible for the payment of charges incurred by one or more users.

User

An entity designated by a customer to use the services of a telecommunication network, such as a person using a UMTS mobile station as a portable telephone.

Index

C	
Central Configuration Repository	14
D	
Daily Summarized Data Storage Duration	21
Data Collection Interval	19
Database	
creating the	17
deleting the	29
naming plan	17
user authentication	17
user privileges	18
user rights	18
DataMart	
logical architecture	11
DataMart Granularity	19
DataMart Migration Component	
configuring the	19
definition of	12
dump	30
reloading the configuration	26
stopping the	27
Degradation State Level	22
Driver Type	22
E	
Enumeration Maximum Number of Values	24
H	
Hourly Summarized Data Storage Duration	20
L	
List of Logger Database References	22
Log File	31
Log Reading Period	20
M	
Monthly Summarized Data Storage Duration	21
P	
Production Area Management Mode	20
Production Host	23
Production Listener Port	23
Production Net Service Name	23
Production Password	23
Production SID	23
Production User Name	23
Q	
Quarterly Summarized Data Storage Duration	21
R	
Raw Data Storage Duration	20
Rows Commit Number	24
S	
Staging Host	23
Staging Listener Port	24
Staging Net Service Name	24
Staging Password	24
Staging SID	23
Staging User Name	24
Summarization Period	20
T	
TNS Directory	22
TNS Namespace	22
Trace File	
Java	31
SQL	32
V	
Violation State Level	22
W	
Weekly Summarized Data Storage Duration	21
X	
XML Validation	25
Y	
Yearly Summarized Data Storage Duration	21

