hp OpenView
Service Quality Manager

Service Adapter for

HP OpenView Service Desk

Installation, Configuration and User's Guide

Edition: 1.2

January 2005

Legal Notices

Warranty

Hewlett-Packard makes no warranty of any kind with regard to this manual, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

A copy of the specific warranty terms applicable to your Hewlett-Packard product can be obtained from your local Sales and Service Office.

Restricted Rights Legend

Use, duplication or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause in DFARS 252.227-7013.

Hewlett-Packard Company

United States of America

Rights for non-DOD U.S. Government Departments and Agencies are as set forth in FAR 52.227-19(c)(1,2).

Copyright Notices

Trademark Notices

HP-UX Release 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.

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

UNIX® is a registered trademark of The Open Group.

All other product names are the property of their respective trademark or service mark holders and are hereby acknowledged.

Contents

Preface

This document describes how to install, configure, and use the HP OpenView Service Quality Manager (SQM) Service Adapter for HP OpenView Service Desk. The Service Adapter is the application that provides SQM with the performance data used to measure the quality of service.

This document describes how to:

- Install the Service Adapter (and associated subsets)
- Set up the Service Adapter
- Start and stop the Service Adapter
- Customize the Service Adapter

Intended Audience

This document is intended for Service Quality Manager System administrators.

Required Knowledge

It is assumed that the reader is familiar with the functionality of Service Quality Manager and has previous experience of the following:

- System administration and operations
- HP OpenView Service Desk
- Service Level Management

It is assumed that the reader is familiar with the concepts described in the following books:

- HP OpenView Service Quality Manager Overview
- HP OpenView Service Quality Manager Service Adapter User's Guide

Software Versions

The software versions referred to in this document are specified in "Service Adapter Installation", on page 2-1.

Typographical Conventions

The following typographical conventions are used in this book:

Bold	New terms and to emphasize important words.
Italic	File names, programs, and parameters.
	The names of other documents referenced in this manual.
`Monospace`	Source code and examples of file contents.
	Commands that you enter on the screen.
	Pathnames.
	Keyboard key names.

Associated Documents

For a full list of Service Quality Manager user documentation, refer to the *HP OpenView Service Quality Manager Product Family Introduction*.

Support

Please visit our HP OpenView web site at:

http://openview.hp.com/

There you will find contact information as well as details about the products, services, and support HP OpenView has to offer.

The "HP OpenView support" area of the HP OpenView web site includes:

- Downloadable documentations
- Troubleshooting information
- Patches and updates
- Problem reporting
- Training information
- Support program information

Chapter 1

Introduction

The HP OpenView Service Quality Manager (SQM) uses a Service Adapter to collect data from HP OpenView Service Desk. It feeds Service Quality Manager with key quality indicators about Service Desk services. These indicators are computed from the following Service Desk related items:

- Service calls

- Incidents

This chapter provides a brief overview of the SQM components, and also introduces the Service Desk elements the Service Adapter uses to collect data.

For a detailed description of SQM, refer to the *HP OpenView Service Quality Manager Overview*.

For a detailed description of Service Adapters, refer to the *HP OpenView Service Quality Manager Service Adapter User's Guide*.

1.1 Service Quality Manager Presentation

SQM provides a complete service quality management solution. It consolidates quality indicators across all domains — telecom, IT networks, servers, and applications — providing end-to-end visibility on service quality. SQM links service quality degradations to potential effects on business, allowing network support personnel to address problems and prioritize actions proactively.

SQM monitors the service quality by aggregating information coming from all data sources, such as the network, the IT infrastructure, and the service provider's business processes. Using this information, service operators can pinpoint infrastructure problems and identify their potential affect on customers, services, and service level agreements (SLAs).

Figure1 shows the main Service Quality Manager components:

Figure 1: Service Quality Manager Main Components
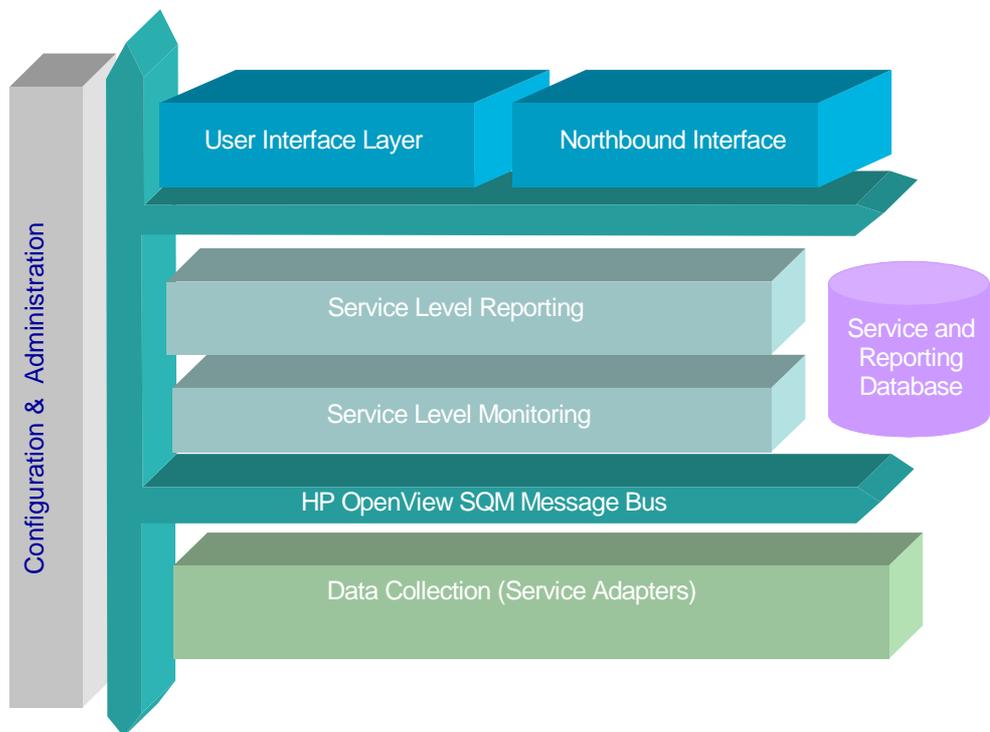
Service Adapters have two main interfaces to SQM:

- An interface to the Configuration and Administration layer.

- An interface to the Service Level Monitoring (SLM) layer.

The Configuration and Administration layer contains the configuration repository. This repository stores the information about how to connect to Service Desk, as well as the way to map the raw data from Service Desk to the SQM parameter format.

The Service and Reporting Database contains the SQM Service Repository, which in particular stores the parameters to be collected by the Service Adapter.

The Service Level Monitoring layer receives all measurement data from the Service Adapter. Then, it calculates the status of each service to determine whether a service has failed to meet a service level.

1.2 Service Adapter presentation

SQM uses a Service Adapter to connect to Service Desk and collect data. A Service Adapter acts as a gateway between Service Quality Manager and Service Desk, collecting data from Service Desk, calculating quality indicators, and feeding those indicators to SQM.

1.2.1 OVSD Service Adapter introduction

The HP OpenView Service Desk Service Adapter provides quality indicators derived from the data stored in Service Desk. With Service Desk, enterprises can streamline their critical service delivery and support processes into a single workflow. This Service Adapter extracts indicators about the responsiveness of the service delivering organization when receiving calls from their customers (service calls) or if the organization itself discovers issues (incidents) with a provided service. The Service Adapter enables a service operator to manage Service Level Agreements (SLAs) covering the:

- Number of open service calls, incidents, or both
- Average open time of service calls, incidents, or both
- Violations of deadlines

1.2.2 Service Desk overview

OpenView Service Desk is a product for Service Providers. Its functionality covers the following areas:

Figure 2: Overview of Service Desk functionality
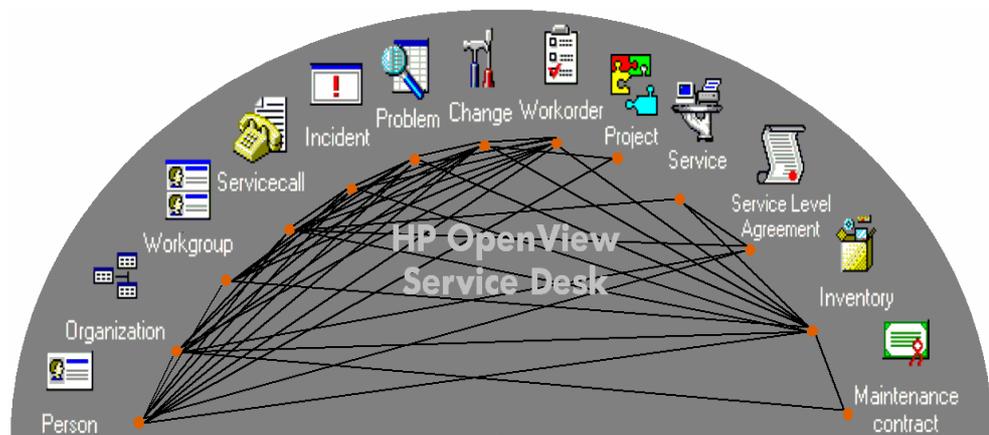
- help desk management
- knowledge management
- problem management
- incident management
- change management
- project management
- work management
- configuration/asset management

- service level management

1.2.2.1 Supported platforms

Service Desk Application Server:

- Microsoft Windows NT 4.0 Server Service Pack 6a

- Microsoft Windows 2000 Advanced server Service Pack 3

- Microsoft Windows 2003 Server

- HP-UX 11.0

- Sun Solaris 8

1.2.2.2 Architecture

The SD architecture consists of six layers: five horizontal ones and one vertical. These layers form an adaptation of the 'standard' 3-tier architecture style.

Figure 3: Overview of Service Desk layers and 3-tiers design
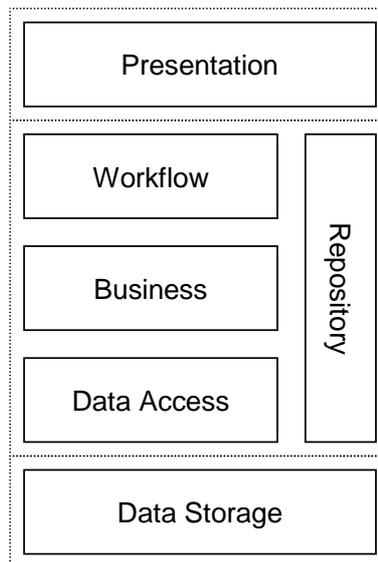
The Presentation layer offers the SD functionality to users. The Workflow and Business layers together contain the SD business logic. Data Storage takes care of persistent storage of SD information. The Repository manages 'meta'-information describing the object model and data mapping used by SD. The object model used by SD defines: Entities (classes), attributes of these Entities, and associations (e.g. inheritance and association) between them.

Presentation Layer

The Presentation layer offers the functionality to display a GUI for the user of SD. As the SA for SD does not interact with the SD GUI, I will not go into further details about the presentation layer.

Workflow Layer

The Workflow layer contains almost all business logic, in specialized subclasses of a kernel class. Some business rules (i.e. required fields and database rules) are enforced

by kernel classes, but the definition of these rules is stored in (the SD specific part of) the repository. The Workflow layer provides component interfaces to the data retrieved from the Business layer. The GUI never accesses these interfaces through their real class; instead it uses the root class. General-purpose methods are used in combination with generated enumerations and typecasting. The specialized methods (containing business rules, also know as 'hooks') are executed via a registering scheme.

The Workflow layer performs the mapping of user interface concepts to a Query object as used by the Data Access layer. Furthermore it indicates transaction boundaries.

The Workflow layer accesses the Business layer though a proprietary protocol named ITP (ITSM Transmission Protocol).

The Workflow is designed as a single-user component that is running on the same machine as its user. But it can be accessed (on the SD application server) from a different machine through ITP. Each logged-on user must have its own Workflow layer instance, both on the client and on the application server. Multiple Workflow layer instances can be created on one machine (application server) but these do have their own private state (most notably all entitlement information). This, and the fact that each instance needs a private thread, makes them quite 'heavy'.

The Workflow layer enforces an authorization scheme ('entitlement'). Requests to the Business layer always include extra information, so that only entities a user is allowed to see will be returned.

Business Layer

Together with the Data Access Layer the Business Layer makes up the application server. The function of the Business Layer is twofold:

- It contains the data-centric business rules. I.e. it prescribes how an operation is to be validated or completed.

- It provides services to the Workflow Layer. These services can best be characterized by their use of resources that are external to the Service Desk application: e.g. database access, transaction services, message queue services, mail services, etc. Consequently, integrations with other OpenView products and third-party products will also be handled by the Business Layer (note: this is a one-way integration. Service Desk can integrate with others from within the business layer, but when others want to integrate with Service Desk they can only do so by accessing the Workflow layer).

From a functional perspective one can say that the business layer provides the workflow layer with persistence, actions upon termination of an object's lifecycle, and in general, all actions that are performed by other services.

In connection with the Business Layer, a Workflow Layer is always running on the application server (that is, a Workflow Layer is running on both the client and the application server hardware). This Workflow Layer must be used to perform tasks specific to this layer from within the Business Layer, e.g. creating new entity instances.

Business rules are only maintained on the Business layer when they are data intensive. Rules that require few database interactions and small amounts of data are maintained in the Workflow layer. Business rules have very similar functionality available in both the Workflow and Business layer.

The Business layer does not maintain state information for the entities retrieved through it. It has a single instance for each entity type whose methods are the (stateless) services relating to that entity type.

Simple business rules can be enforced by the Business layer without Java coding. These rules are called 'database rules'; the rule definitions are stored in the Repository.

The Business layer is responsible for sending only required (as little as possible) information to the Workflow.

Data Access Layer

The Data Access layer provides a general-purpose mapping allowing database access through Query objects. Furthermore this layer provides facilities such as database connection pooling and 'query restriction' (i.e. data volume management).

Data Storage Layer

The data used by SD is stored in a third-party relational database management system. The Data Storage layer is not developed internally. The database systems supported by SD are Oracle and Microsoft SQL Server.

Repository

The Repository provides a single location for all meta-information for both the kernel and SD. This allows the product to be configured easily without changing the code. The most prominent example of such meta-information is the SD object model, which is used by all layers. The complete element list (and the layer that uses them):

- SD object model.

- Kernel object model, which is data centric.

- Database mapping (Data Access).

- Database rule definition (Business).

- External mapping model (Data exchange by Business).

- Entitlement information: roles (Workflow).

- Template definitions: default attribute values for new entities (Workflow).

- Dictionary: labels and messages (Presentation and database rules).

- Form definitions (Presentation).

- View definitions (Presentation).

- Action definitions (Presentation).

- General User Settings (Presentation).

The Repository stores its information in Data Storage; on start-up it reads back this information, using the Data Access layer. During runtime it stores this information in Java objects. A copy of these objects can be sent to clients. Clients can store these objects locally. When the repository information changes the objects get a new version number. Clients have to check the version numbers (when they connect to the application server) of their locally stored objects to see whether they can use these or have to request a new copy.

1.2.2.3   OVSD Integration as Data Source for SQM

As described above, SD manages a wide variety of different items. Most of these items are for configuration (e.g. Persons, Configuration Items, SLAs, Organizations, etc.) and therefore don't provide meaningful metrics. This section will describe the SD items that provide meaningful metrics and these metrics.

SD Item: Service Call

A service call records a request for support made by a user or customer. The user may only have a question about an application that is supported by the support organization. But when a computer does not work or a network performs badly, the user can also have a request to restore the performance of the computer or the network. A service call is created at the moment the request is received. After the service call is created, it is regularly updated with information to answer the question or to solve the problem of the user.

The status of the service call reflects where the service call is between being created and being solved. The number of status codes used between the moment the service call is created and the moment the service call is solved depends on the needs of your organization. The status code always reflects the place of the service call within the procedures of your organization. The status codes can be grouped. Open status codes are used from the moment the service call is created as long as the help desk or specialists are working on the service call to resolve it. Closed status codes are used from the moment the help desk or specialists believe the service call is resolved. Often the last available status code designates a service call where both user and service provider agree the service call is resolved.

You use impact and priority to specify the time given to resolve the service call. Impact reflects how much a service is impaired. If the service call records a question about how to use a utility, the utility is not impaired therefore the impact is low. If the service call is raised because the user cannot send or receive e-mail, it is likely that all users making use of the same e-mail server cannot send or receive e-mail. In this case the impact is high. Priority reflects the importance of the service for the user. The utility in the first example may be important for the user, this makes the priority of the service call high. In modern life e-mail is a very important service, however e-mail is not equally important for all departments or organizations. Some may not notice that the e-mail server was down for an hour; others may notice it within a few minutes.
Impact and priority are closely related. If a service is important for a user, then if the impact is low the priority may already be high. For example, a large number of sales orders are nowadays registered through e-mail. Even if the e-mail program of only one sales person does not function properly, the problem with the e-mail program should be solved with highest priority.

This table summarizes, for each service, the list of fields that are retrieved and calculated by the Service Adapter from service call items. The Open Duration, Over Deadline and Actual Duration parameters are aggregated from all the service calls related to a service and mapped to DFD parameters.

Table 1: Service call item fields used by the Service Adapter

Field	Comment
ID *	Unique ID generated by SD
Service	Functional ID, as name is not unique
Status *	Text mapped to Integer value
Impact	Text mapped to Integer value
Priority	Text mapped to Integer value
Open duration **	Equals to: (Actual finish - Registration time) if status is "closed" (current time – Registration time) if status is still "open"
Over deadline **	Equals to: (Actual finish – Deadline) if status is " closed" and value > 0, (Current time – Deadline) if status is still " open" and value > 0
Actual duration **	Equals to: (Actual finish – Actual start) if status is " closed", (Current time – Actual start) if status is still " open"

* by SD required fields

** The status "closed" or "open" is determined at the Service Adapter configuration step (see chapter 4.2.2.6).

SD Item: Incident

An incident holds information about the status of a service, a configuration item, or both. It informs Service Desk users about the reasons behind the status, the cause analysis for the incident and the actions taken to solve the situation. An incident in Service Desk is used in three ways:

- To provide information for customers explaining a situation

- To provide the basic information for specialists working on the described problem

- To notify a specialist colleague of a technical problem

At first sight service calls and incidents appear to be very similar. However, there are some basic differences.

- Service calls are created with information coming from customers.
  Incidents are created with information coming from specialists, network management tools or system management tools.

- Service calls support feedback to the customers.
  Incidents transfer information among specialists.

- Service calls are based on questions, remarks or requests concerning services.
  Incidents are based on information about the status of a service or configuration item.

This table summarizes, for each service, the list of fields that are retrieved and calculated by the Service Adapter for incident items. The Open Duration, Over Deadline and Actual Duration parameters are aggregated from all the incidents related to a service and mapped to DFD parameters.

Table 2: Incident item fields used by the Service Adapter

Field	Comment
ID *	Unique ID generated by SD
Service	Functional ID, as name is not unique
Status *	Text mapped to Integer value
Impact *	Text mapped to Integer value
Priority	Text mapped to Integer value
Open duration **	Equals to: (Actual finish - Registration time) if status is "closed", (Current time – Registration time) if is still "open"
Over deadline **	Equals to: (Actual finish – Deadline) if status is "closed" and value > 0, (Current time – Deadline) if status is still "open" and value > 0
Actual duration **	Equals to: (Actual finish – Actual start) if status is "closed", (Current time – Actual start) if status is still "open"

* by SD required fields

** The status "closed" or "open" is determined at the Service Adapter configuration step (see chapter 4.2.2.6).

1.2.3 How the Service Adapter works

The Service Adapter measures the quality of service related to the services defined in Service Desk application. For each service, the Service Adapter will collect and aggregate performance indicators collected on service calls or/and incidents.

To extract data from SD, the Service Adapter uses SD's Web-API. The SA will make direct use of SD's API and all configuration information known by SD.

The use of the Web-API allows the user to install the SA on a different machine than the real SD server itself. This helps the user to reduce the performance impact to its SD system.

To connect to SD the SA has to start a connection session to the SD server provided by the user. After the successful login, all the functionality of SD's Web-API is available to the SA. It provides the SA with classes for accessing all standard SD entities. This will make it much easier for the SA to request the needed data, as it does not have to deal with the database layout and configuration. The SA has just to request the needed entities and SD will take care of the translating them to the database mapping and actually do the real database query. SD will also take care of avoiding data inconsistencies if users are changing data while the SA makes a request for them.

The following picture illustrates three configurations supported by the SD Service Adapter to connect to the SD server:

- A first configuration where the Service Adapter is running on the same Windows system as the SD Server

- The second configuration shows a remote connection of the Service Adapter to the SD Server on HP-UX

- The third configuration shows a remote connection of Service Adapter running on sHP-UX to the SD Server on Windows.

OV SQM User Interfaces
Service Designer UI, SLA Administration &
Monitoring UI,

Windows XP

HPUX

OV SQM Core
Service Level Monitoring / Service
Level Reporting ...

Windows XP

Windows Acquisition director

OVSD SA

Windows Acquisition director

OVSD SA

Service Desk 4.5 Client

Service Desk 4.5 Client

Service Desk 4.5 Server

Windows XP

Service Desk 4.5 Server

SD db

database on
SQL server / Oracle

SD db
Oracle

HPUX

Acquisition director

OVSD SA

HPUX

Service Desk 4.5 Client

Service Desk 4.5 Server

SD db

Windows XP

database on
SQL server / Oracle

1.2.4   Data Feeder presentation

To collect data, the Service Adapter uses **data feeders**. A data feeder is a source of data for SQM. Each data feeder is defined by a **Data Feeder Definition** (DFD).

The data collected from Service Desk is defined by DFDs, their properties, and parameters, each of which is described below.

- DFDs
  DFDs are data structures that define what source data is collected from Service Desk - either service calls or incidents. They also define what types of service call or incident data are collected. DFD configuration information is stored in the SQM configuration repository.

- DFD parameters
  Parameters define what measurement data is sent by the Service Adapter to Service Quality Manager. They are stored in the SQM data repository.

- DFD properties
  Properties further define what source data is collected from Service Desk. For example, collecting data for a particular Service Desk service ID. They are stored in the SQM data repository.

- Measurement Reference Point
  The Measurement Reference Point (MRP) identifies the source of the data that is to be collected.

1.2.4.1   DFD presentation

Table 3: Service Desk DFDs

Name	Type	Description	Value
ItemType	String	Item to gather from Service Desk.	Either `serviceCall` or `incident`.
StateOpen	List	List of Service Desk status entries considered to mean the item is open.	See *Configuring Data Feeder Definition* on page 4-7.
StateClosed	List	List of Service Desk status entries considered to mean the item is closed.	See *Configuring Data Feeder Definition* on page 4-7.

1.2.4.2 DFD Parameters

Table 4: DFD Parameters

Parameter	Label	Description	Data type	Unit	Category	Cust. Dep.Flag
NbOpen	Number of open items	Number of currently open items.	Int		Other	False
NbClosed	Number of closed items	Number of closed items with a finished date since last collection.	Int		Other	False
NbOverdue	Number of overdue items	Number of items with a deadline in the past or before their close time.	Int		Other	False
OpenDuration	Average Open duration	Average open time of all reported items since their registration.	Int	Secs	Other	False
ActualDuration	Average Actual duration	Average open time of all reported items since their actual start time (see note below).	Int	Secs	Other	False
MaxOverDeadline	Max over deadline	The longest time past the deadline.	Int	Secs	Other	False
OverDeadline	Average over Deadline	Average time the items over the deadline are over the deadline.	Int	Secs	Other	False

Note

The actual start time needs to be set in Service Desk. Items without a set start time are ignored and no measurement data is collected.

1.2.4.3 DFD Properties

Table 5: DFD Properties

Property Name	Label	Description	Data type
ServiceID	ID of the Service Desk service	Internal ID of the Service Desk service as shown in the Service Desk user interface.	String
ServiceName	Name of the service	Name of the Service Desk service as it appears in the Service Desk interface.	String
Connector	Name of the Connector	The name of the connector to use to connect to the Service Desk server, as configured in "**Error! Reference source not found.**" on page 3-1.	String
SDServer	Name of the Service Desk server	The name of the Service Desk server. Not necessarily the host name. Can be an abstract name like "Backup Service Desk Server" or "Primary Service Desk Server"	String

1.2.4.4 Measurement Reference Point

A data feeder specifies the source data to be collected and the MRP specifies the source of that data.

The MRP contains all the information about a particular instance of a data feeder, the DFI, and thus provides the DFI with a unique name. In this way, the MRP distinguishes between two or more DFIs of the same DFD.

Syntax

For the Service Desk Service Adapter, the syntax of the MRP is:

```
OvSd  -[servicecall|incident]
      -[<SD_Server_Name>]
      -[<Service_ID>]
      -[<Service_Name>]
```

The MRP syntax has the following elements:

`OvSd`
   Is the prefix of each MRP name for this Service Adapter.

`servicecall` or `incident`
   Is the item type of the Service Desk service being monitored.

`<SD_Server_Name>`
   Is the name of the Service Desk server.

`<Service_ID>`
   Is the ID of the Service Desk service being monitored.

`<Service_Name>`
   Is the name of the Service Desk service being monitored.

Example

```
OvSd-incident-Main Server-107-Premium Exchange Service
```

1.3 Configuration overview

The Service Desk Service Adapter provides a configuration utility that connects to Service Desk, discovers as much configuration information as possible and creates all the configuration files needed to configure the Service Adapter for collecting the required measurement data. These configuration files are saved in XML format and are uploaded to the repositories.

1.3.1 Way to proceed

Figure 4 shows the steps you take to configure the Service Adapter:

Figure 4: Service Adapter configuration diagram
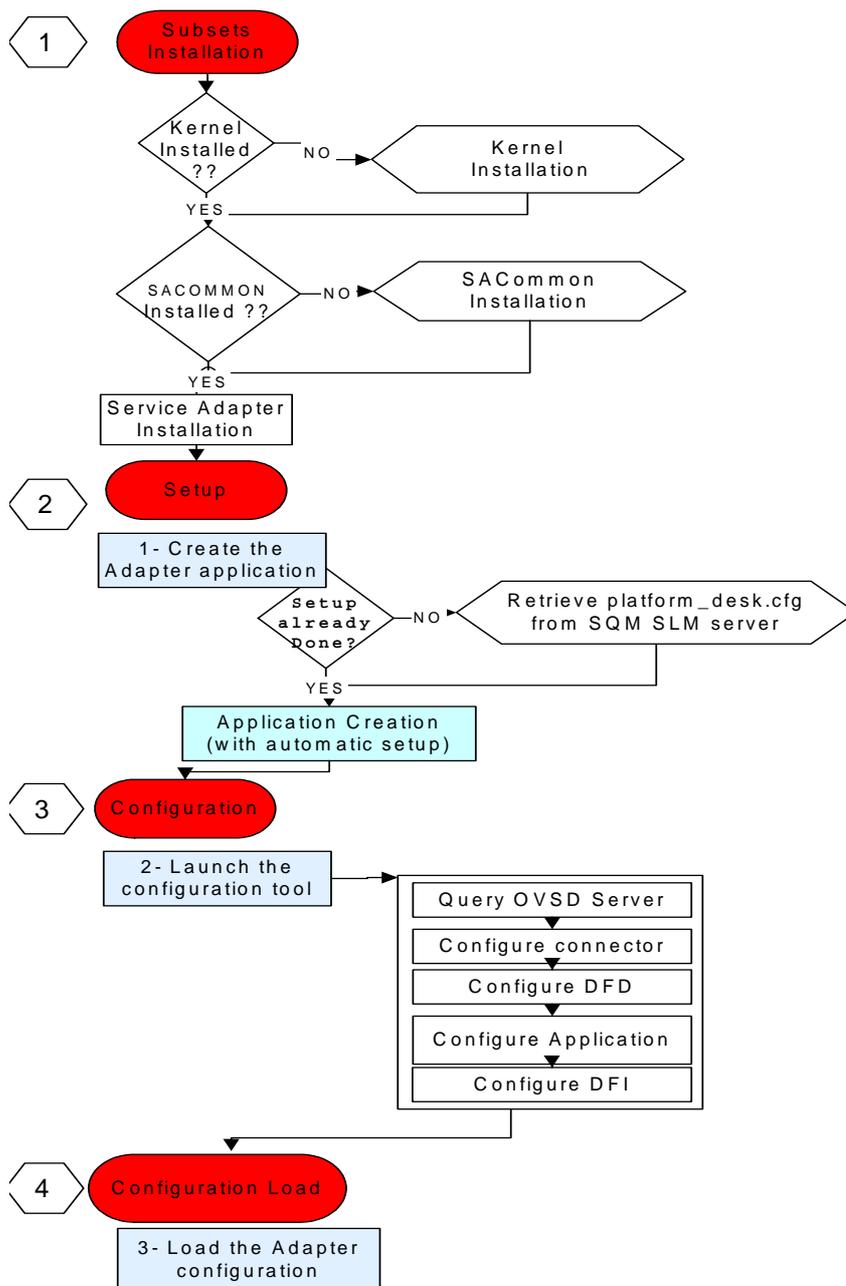
Chapter 2

Service Adapter Installation and Uninstallation on Windows

This chapter describes how to install the Service Adapter on Microsoft Windows. After you have completed the installation, follow the instructions in Chapter 4 to configure the adapter.

2.1 Software and Hardware requirements

2.1.1 Software requirements

- Windows XP
- HP OpenView Service Desk 4.5 up to Service Pack 9

2.1.2 Hardware requirements

For hardware requirements, refer to the HP OpenView Service Quality Manager Installation Guide.

2.2 Installing the software

If not already done, you first need to install the OV SQM V1.2 Windows Kernel, go to 2.2.1

If the Kernel has already been installed go directly to section 4.1

2.2.1 Installing the OV SQM Kernel

To install the Kernel, perform the following steps:

1. Insert the hp OpenView SQM Core CD in your CD-ROM drive, navigate in the `SQM-1.20.00-CORE\Windows\User_Interfaces` folder, and run the `SQMKERNEL-1.20.00.exe` installer.
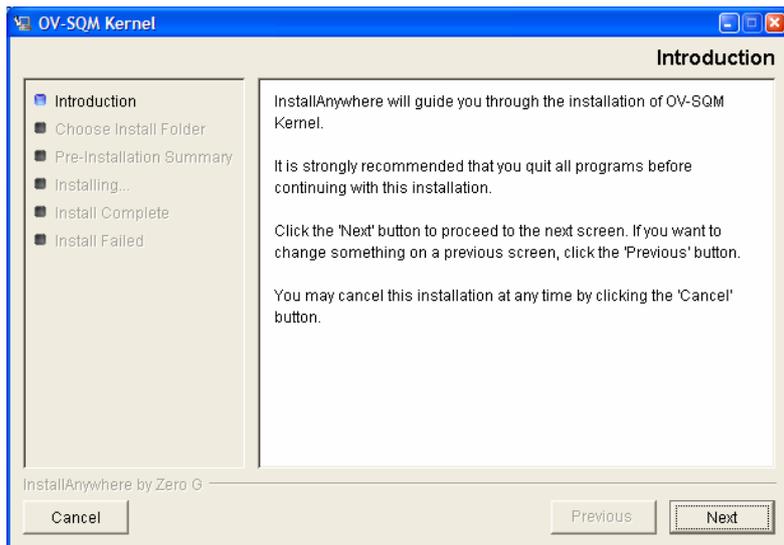
2. Click 'Next' to proceed.
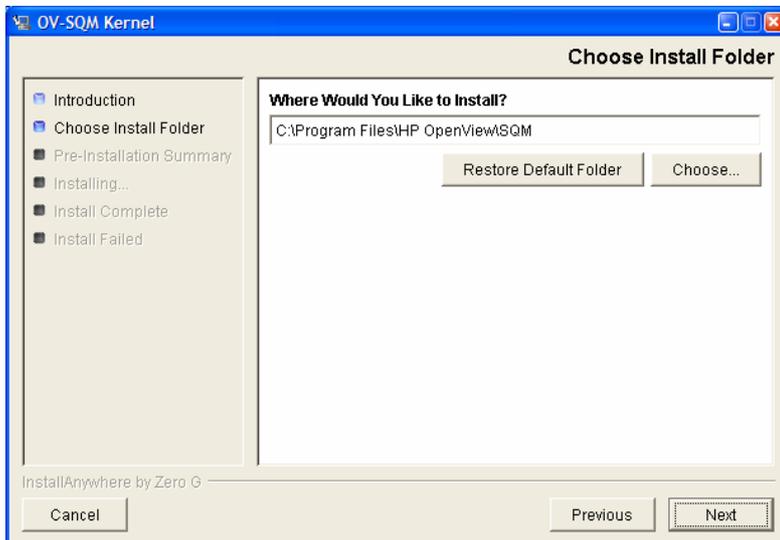
3. Select the destination directory of your OV SQM. If SQM has already been installed (for SLA Monitoring use for instance), you MUST install in the same directory. Click 'Next' to proceed.
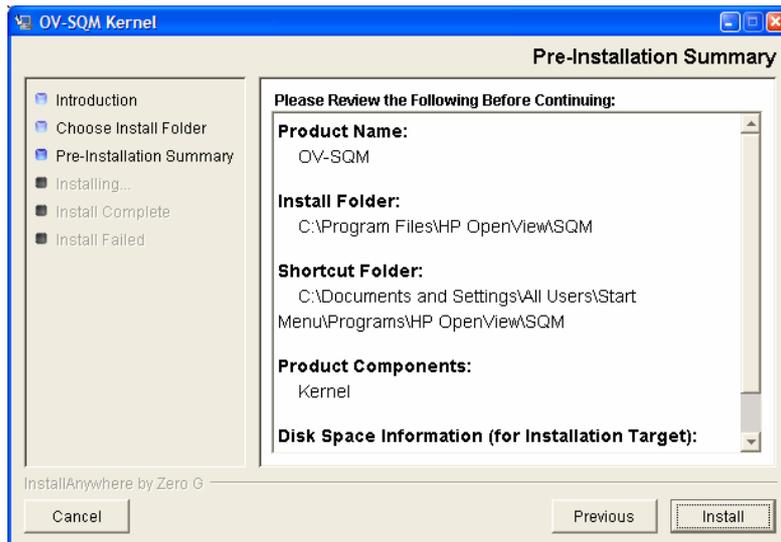
4. This window allows you to check the selected options. Click '**Install'** to perform else '**Previous'** to modify them.

5. Click '**Done'** to end installation process.

2.2.2   Installing the OVSD SA

To install the Service Adapter, perform the following steps:

First install the SA Common component (if already done go to step 10)

6. Insert the Service Adapter CD-ROM in your CD-ROM drive, navigate to the folder SQM-1.20.00-SAGTW\Windows and run the SQMSAGTWCOMMON-1.20.exe installer.

7. The software is installed and the **Install Complete** window is displayed.
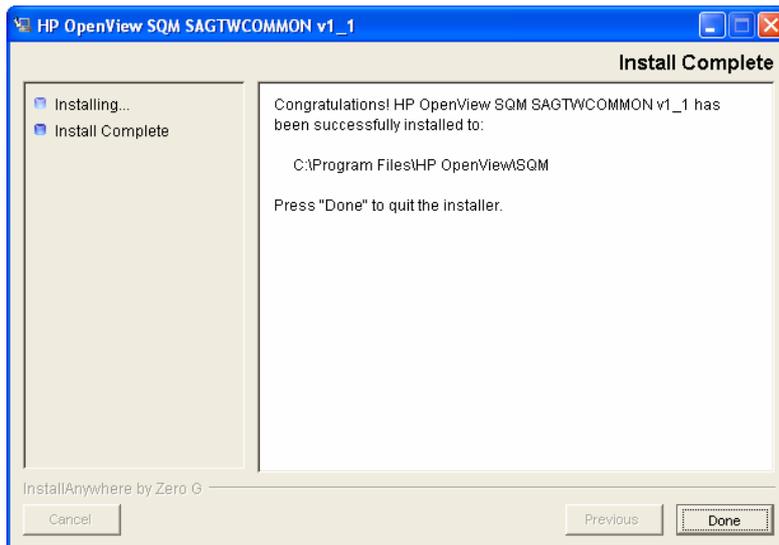
8. To end the installation process, click '**Done'**

9. Insert the Service Adapter CD-ROM in your CD-ROM drive and in the directory SQM-1.20.00-SAGTW\Windows run the SQMSAOVSD-1.20.exe installer.

10. The software is installed and the **Install Complete** window is displayed.
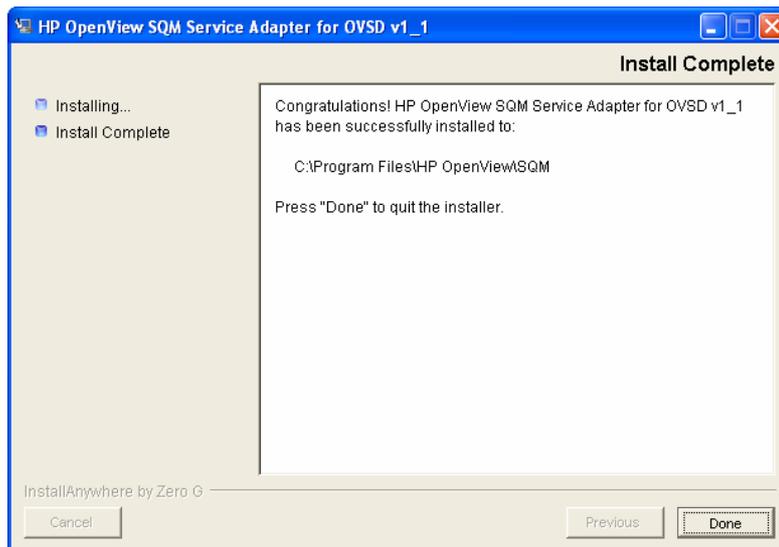
11. To end the installation process, click '**Done'**.

The list of installed directories and files is available in **Appendix A**.

2.2.3 Installing Service Desk Web API

The Service Adaptor communicates with the OV SD server thanks to the OV SD Web Api component.

The Web API version must be aligned with the OV SD Service pack installed on the OV SD server else the communication between both processes fails. To determine the version of installed on the OV SD Server, execute the following command on the server:

```
# /opt/OV/sd/server/bin/sd_version

Acronym:        SD
Version:        4.5.0588.0904 (SP9)
Company:        hp OpenView
```

```
Copyright:      Copyright (c) 2004 Hewlett-Packard Development
Company, L.P. All Rights Reserved.
Description:    service desk is a family member of the hp
OpenView suite.
ProductID:      B4321
ProductName:    hp OpenView service desk
```

By default the OV SD Service Adapter is delivered with the OV SD Web API of OV SD Service Pack 5. If your OV SD Server Service Pack is different it is mandatory to install the right OV SD Web Service API at the following location:

`%TEMIP_SC_HOME%\ServiceAdapters\OvSd\v1_2\lib\web-api.jar`

The OV SD SA kit is also delivered with the OV SD Web API Service Pack 9. To replace the Web API SP5 by the Web API SP9, execute the following commands:

```
> cd %TEMIP_SC_HOME%\ServiceAdapters\OvSd\v1_2\lib

> rename web-api.jar web-api-sp5.jar

> rename web-api-sp9.jar web-api.jar
```

Once the right Web Api is installed, if the OV SD Service Adapter is running, it is mandatory to re-start it.

2.2.4   Patch installation

SQM patches are delivered in the windows/patches delivery directory. If patch is delivered, refer to the associated README file to install it.

The installation on Windows completed. Please follow the instruction in Chapter 4 to configure the service adapter.

2.3   Uninstalling the software

To uninstall the Service Adapter software:

1. In Explorer, navigate to the following folder:

```
Select menu: All Programs -> HP OpenView -> SQM ->
ServiceAdapters -> Ovsd v1_2 -> Uninstall
```

2. The **Uninstall** window is displayed.

3. Click **Uninstall**. The software is uninstalled from your system.

4. To finish, click **Done** in the following window.

If no other Service Adapters (or Gateways) are installed on the system you can also uninstall the HP OpenView Service Adapter Common v1_2

5. In Explorer, navigate to the following folder:

```
Select menu: All Programs -> HP OpenView -> SQM ->
ServiceAdapters -> Common v1_2 -> Uninstall
```
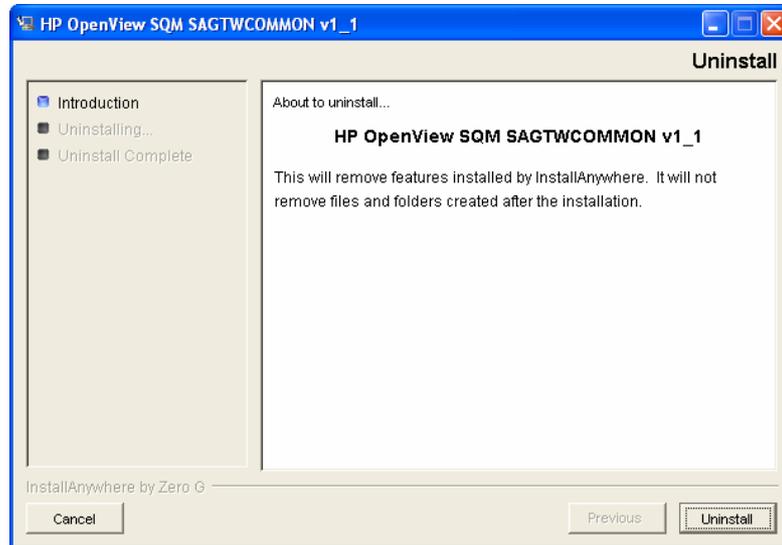

6. Click **Uninstall**. The software is uninstalled from your system.

Chapter 3

Service Adapter Installation and Uninstallation on UNIX

This chapter describes how to install the Service Adapter on HP-UX.

After you have completed the installation follow the instructions in to Chapter 5 to configure the service adapter.

3.1 Software and Hardware Requirements

3.1.1 Software Requirements

- HP-UX 11.11
- HP OpenView Service Desk 4.5 up to Service Pack 9

3.1.2 Hardware Requirements

For hardware requirements, refer to the *HP OpenView Service Quality Manager Installation Guide.*

- Minimum of 1.5 MB disk space.

3.2 Installing the Software

If not already done, you first need to install the OV SQM V1.2 Kernel, go to section 3.2.1.

Section 3.2.2 describes the required environment variables and user account for performing the installation and setup.

If the OV SQM V1.2 Kernel has been installed go directly to section 3.2.3

Section 3.2.4 explains how to install possible existing patch.

3.2.1 Installing the OV SQM Kernel

To install the Kernel perform the following steps:

1. If not already existing, create the SQM home directory. Under **root** account:

   #mkdir /opt/OV/SQM<xxx>

   Where <xxx> is the label you want to assign to the current release.

2. Mount the SQM Core CD-ROM on your system

3. Go to <mounted directory>/SQM-1.20.00

Run the '**sqm_install**' tool under the **root** account.
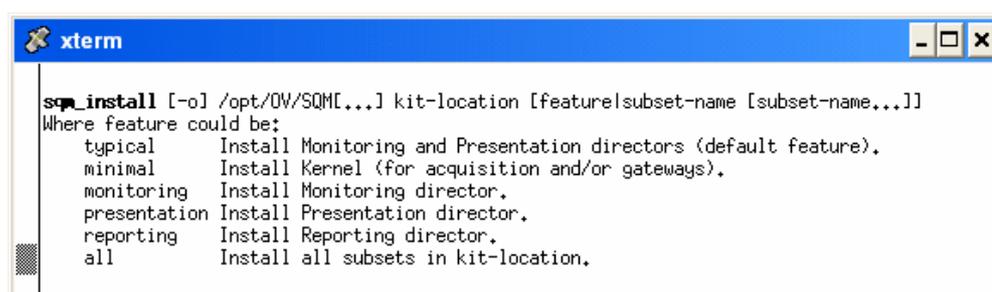
4. Select the 'minimal' feature
   Example: # **sqm_install /opt/OV/SQM<xxx> <mount directory>/SQM-1.20.00/HPUX/KIT minimal**

5. Press '**Enter'** to install the Kernel.

3.2.2 Required environment

1. **Sqmadm** user/**group**
   'sqmadm' group and user is necessary for the kernel/application setup and management. Please refer to *HP OpenView Service Quality Manager Installation Guide* for instructions on how to create the 'sqmadm' group and user.

2. Environment variables

   <u>If the kernel setup has already been done</u>, you have to source the **temip_sc_env.sh** file located under $TEMIP_SC_VAR_HOME directory.
   ```
   # .   $TEMIP_SC_VAR_HOME/temip_sc_env.sh
   ```

   <u>If the kernel setup is not yet done</u> (and before to install the SA Common and SA OvSd), you have to set your environment. Perform the following actions:
   - source environment variables:

   ```
   # . export TEMIP_SC_HOME=/opt/OV/SQM<xxx>
   # . $TEMIP_SC_HOME/jre/jre-setup.sh
   # . $TEMIP_SC_HOME/perl/perl-setup.sh
   ```

3.2.3 Installing the OVSD SA

To install the Service Adapter, perform the following steps:

3. Log on as **root** user.

4. First install the SA Common component if necessary (if already done go to step 7)

5. Insert the Service Adapter CD-ROM in your CD-ROM drive or go to the Service Desk Service Adapter installation directory and run the SQMSAGTWCOMMON-1.20.bin installer.
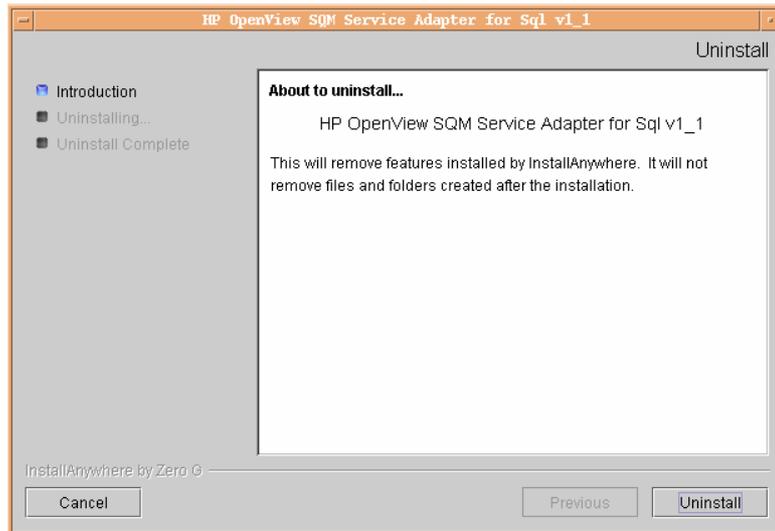
6. To end the installation process, click '**Done**'

7. Insert the Service Adapter CD-ROM in your CD-ROM drive or go to the Service Navigator Service Adapter installation directory and run the SQMOVSD`SA-1.20.bin` installer.
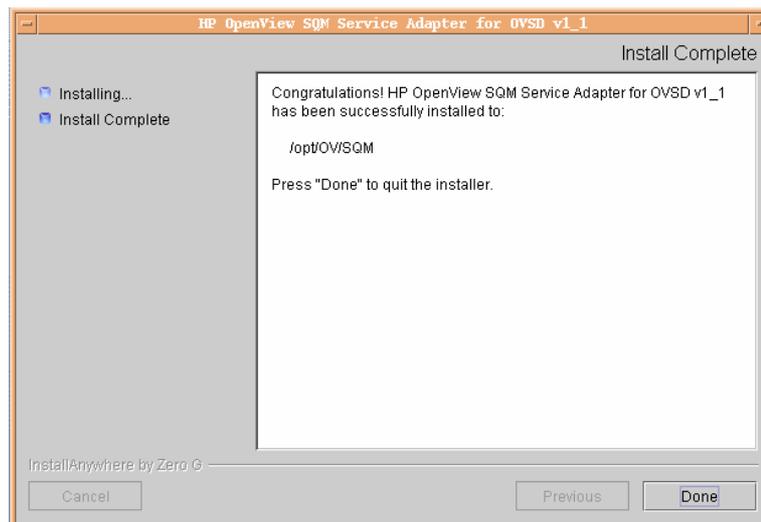
To end the installation process, click '**Done**'.

3.2.4 Installing Service Desk Web API

The Service Adaptor communicates with the OV SD server thanks to the OV SD Web Api component.

The Web API version must be aligned with the OV SD Service pack installed on the OV SD server else the communication between both processes fails. To determine the version of installed on the OV SD Server, execute the following command on the server:

```
# /opt/OV/sd/server/bin/sd_version

Acronym:        SD
Version:        4.5.0588.0904 (SP9)
Company:        hp OpenView
Copyright:      Copyright (c) 2004 Hewlett-Packard Development
Company, L.P. All Rights Reserved.
```

```
Description:    service desk is a family member of the hp
OpenView suite.
ProductID:      B4321
ProductName:    hp OpenView service desk
```

By default the OV SD Service Adapter is delivered with the OV SD Web API of OV
SD Service Pack 5. If your OV SD Server Service Pack is different it is mandatory to
install the right OV SD Web Service API at the following location:
$TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/lib/web-api.jar

The OV SD SA kit is also delivered with the OV SD Web API Service Pack 9. To
replace the Web API SP5 by the Web API SP9, execute the following commands
under root account:

```
# cd $TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/lib/

# mv web-api.jar web-api-sp5.jar

# mv web-api-sp9.jar web-api.jar
```

Once the right Web Api is installed, if the OV SD Service Adapter is running, it is
mandatory to re-start it.

3.2.5 Patch installation

To install a Patch, perform the following steps:

1. Log on as root

2. Go to <mounted directory>/SQM-1.20.00

3. Run the 'sqm_install' tool with '–p' option plus the 'minimal' feature.
    ```
    #sqm_install –p /opt/OV/SQM<xxx> <mounted directory>/SQM-
    1.20.00/HP-UX/Patches minimal
    ```
    This will install all Patches for the Kernel component.

4. Run the 'sqm_install' tool with '–p' option plus the 'SAGTWCOMMON' feature.
    ```
    #sqm_install –p /opt/OV/SQM<xxx> <mounted directory>/SQM-
    1.20.00/HP-UX/Patches SAGTWCOMMON
    ```
    This will install all Patches for the SAGTWCOMMON component.

5. Run the 'sqm_install' tool with '–p' option plus the 'SAOVSD' feature.
    ```
    #sqm_install –p /opt/OV/SQMV120 <mounted directory>/SQM-
    1.20.00/HP-UX/Patches SAOVSD
    ```
    This will install all Patches for the OVSD component.

The installation on Unix completed. Please follow the instruction in Chapter 5 to
configure the service adapter.

3.3 Uninstalling the Software

To uninstall the Service Adapter software:

1. Log on as **root** user

2. Perform the following commands on the SQM platform where the Service
    Adapter has been installed:

cd $TEMIP_SC_HOME

#./ServiceAdapters/OvSd/v1_2/UninstallerDataOVSD/Uninstall_OvSd
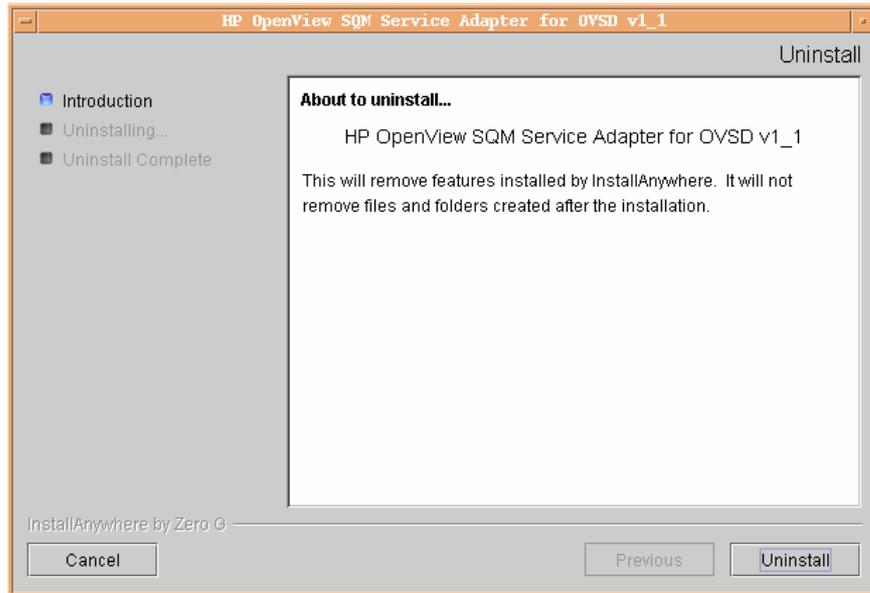
3. Click '**Uninstall'**. The software will be uninstalled from your system.

4. To finish, click '**Done'** in the following window.

5. If no other Service Adapters (or Gateways) are installed on the system you can also uninstall the HP OpenView Service Adapter Common v1_2

   # cd $TEMIP_SC_HOME

   #./ServiceAdapters/Common/v1_2/Uninstaller_SAGTWCOMMON/Uninstall_SAGTWCO MMON
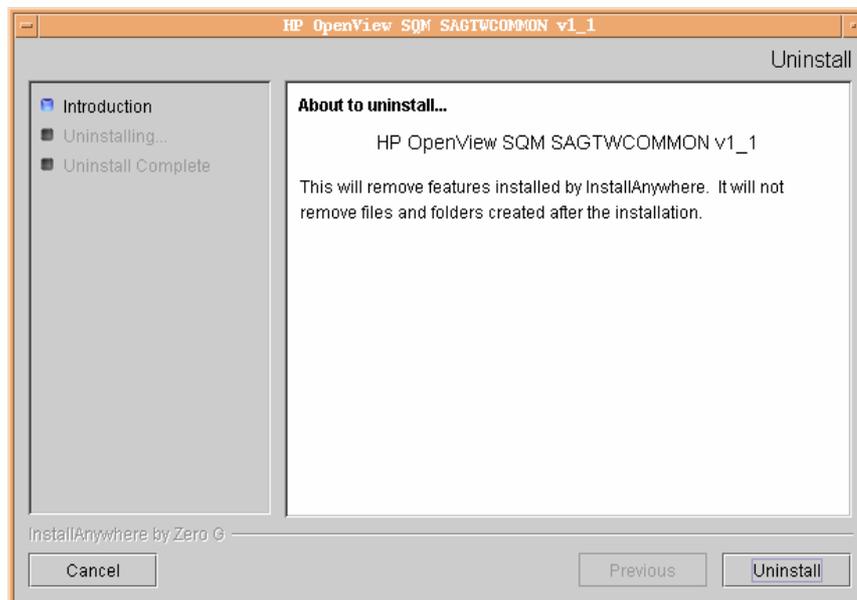
6. To finish, click '**Done'** in the following window.

Chapter 4

Service Adapter Setup on Windows

This chapter describes the way to perform the configuration of the OVSD service adapter.

The following subchapters will illustrate all the implementation steps by describing the configuration command and by giving basic examples.

Please note you will find at the end of this installation/configuration guide within the Appendix A a concrete scenario for the integrator.

It will introduce:

- Main OVSD architecture and their SQM implementation

- An example of naming policy management for acquisition director and application

Once installed, the following menu entries, available from the menu, allow performing the configuration of the OVSD:

All Programs → HP OpenView → SQM → ServiceAdapters → Ovsd v1_2
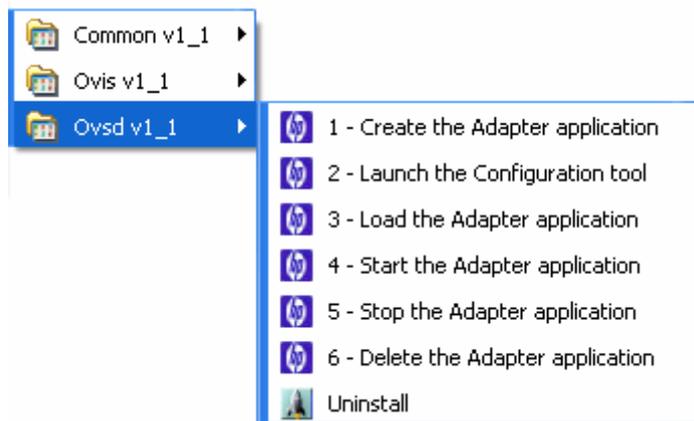
The Following table summarizes the steps to be done according to usual life cycles of the OVSD SA:

Create	(1) Create → (2) Configure → (3) Load → (4) Start
Update	(5) Stop → (2) Configure → (3) Load → (4) Start
Delete	(5) Stop → (6) Delete

All these steps are described in the following chapters.

4.1   Creating Application

Be aware that the create application will perform the kernel setup if not already done. If your Kernel setup has already been performed, refer to section 4.1.2 else perform first section 4.1.1

Caution

A Service Adapter application name has to be unique cross platform and director!

4.1.1   Kernel setup

The kernel setup is automatically called during the SA Application creation but the following step must be done in order to retrieve the platform description file:

Copy the platform description file:

- Copy the `$TEMIP_SC_VAR_HOME\setupconfig\platform_desc.cfg` file from the SQM SLM Primary Server (where all SQM core applications are running) to your local system into `%TEMIP_SC_HOME%\tmp` directory.

4.1.2   OVSD Application creation

To create the application, please launch the tool:

1. Select menu: All Programs → HP OpenView → SQM → ServiceAdapters → Ovsd v1_2 → Create the Adapter application.

   The OVSD application tool window is displayed.

2. You need to provide the following information:

- The platform name on which the application will be installed. (ex: **slmv12**)

- The director name on which the application will be installed. (ex: **acquisitionW2K**)

- The name of the OvSd Service Adapter Application (ex: **OvSd**)

3. When the creation is over, you are ready to close the window.

4. To check that the application has been correctly created you can perform the command: **temip_sc_show_platform –platform slmv12** either on the SQM SLM Primary Server (using the sqmadm user with all SQM environment variables set) or on the Windows acquisition platform, in a Windows command line with all SQM environment variables set (%TEMIP_SC_VAR_HOME%\temip_sc_env.bat).

Note: Please pay attention to the Service Adapter Application name (in this example: `'slmv12_acquisitionW2K_OvSd'`) which will be used during the whole life cycle of this application.

The following screenshot shows command line displaying the messages of a service adapter instance creation with windows kernel setup:

The following screenshot shows command line displaying the messages of a service adapter instance creation without windows kernel setup:

For more information, refer to the *HP OpenView Service Quality Manager Administration Guide*.

4.2 Configuring the Application

This chapter describes how to set up the HP OpenView Service Desk Service Adapter using the configuration tool.

To launch the OVSD configuration tool window:

Select menu: All Programs → HP OpenView → SQM → ServiceAdapters → Ovsd v1_2 → Launch the Configuration Tools.

The Command Prompt window is displayed with the configuration tool help:

```
Select C:\WINDOWS\System32\cmd.exe                                    _ □ ✕

usage: configure -discover -sainame <SA instance name>
                           -host <hostname>
                           -user <user>
                           -passwd <password>
                           -services <servicelist file>
                           -status <statuslist file>

        configure -connector -sainame <SA instance name>
                             -host <hostname>
                             -user <user>
                             -passwd <password>
                             [-name <connector name>]

        configure -dfd -sainame <SA instance name>
                       -name <DFD name>
                       -itemtype <"servicecall"¦"incident">
                       -status <statuslist file>

        configure -sai -sainame <SA instance name>
                       -connectors <connector1 [connector2 [...]]>
                       -datafeeders <datafeeder1 [datafeeder2 [...]]>

        configure -dfi -sainame <SA instance name>
                       -conname <connector>
                       [-servername <SD server name>]
                       -dfdname <DFD name>
                       -services <filename>
                       [-qty <dfi per message>]
```

4.2.1   Configuring the connector

To get data from the Service Desk server, the Service Adapter needs to log into
Service Desk. This configuration provides the Service Adapter connector with a valid
account name, password, and the host name of the Service Desk Server.

The connector contains the following settings:

- ApplPassword
  The password used to log into Service Desk application. The password is stored
  encrypted.

- ApplServerHost
  The host name of the Service Desk Server. Ex:sdserverhostname.domain.name

- ApplAccountName
  The account name used for logging in. This does need to be a user interface
  account, but it does need access rights to all necessary service calls and incidents.

4.2.1.1   Syntax

Select menu: All Programs → HP OpenView → SQM → ServiceAdapters → Ovsd
v1_2 → Launch the Configuration tool.
To configure a connector, use the following command at the **Command Prompt**:

```
configure       -connector
                -sainame <SA application name>
                -host <hostname>
                -user <user> -passwd <passwd>
                -name <connector_name>
                -h
```

This command has the following parameters:

`-connector`
    Starts the configuration tool in connector configuration mode.

`-sainame <SA application name>`
    The name of the Service Adapter Application to be configured (the one used
    during the create application (refer to 4.1)).

`-host <hostname>`

The host name of the Service Desk server.

`-user <user>`

The user account to log in to the Service Desk server.

`-passwd <passwd>`

The password for the Service Desk account.

`-name <connector_name>`

The name of the connector to be created. The name may contain only alphanumeric characters, plus the characters "-" and "_". If you do not specify a name, the host name is used, with underscores instead of periods.

`[-h]`

Provides online help.

4.2.1.2 Example

```
configure –connector –sainame slmv12_acquisitionW2K_OvSd -host
sdserverhostname.domain.name -user system -passwd servicedesk
-name hostname_service_desk
```

4.2.1.3 Output File

Service Desk changes or creates the file configure.dat in the %TEMIP_SC_VAR_HOME%\ServiceAdapters\OvSd\<SA application name>\config directory.

4.2.2 Discovering OVSD configuration

The next configuration step consists in retrieving the list of services and status information from the Service Desk server:

4.2.2.1 Syntax

Select menu: All Programs → HP OpenView → SQM → ServiceAdapters → Ovsd v1_2 → Launch the Configuration tool

To start the discovery, use the following command at the **Command Prompt**:

```
configure      -discover
               -sainame <SA application name>
               -host <hostname>
               -user <user>
               -passwd <passwd>
               -services <service list file>
               -status <status list file>
               [-h]
```

This command has the following parameters:

`-discover`

Starts the configuration tool in discovery mode.

`-sainame <SA application name>`

The name of the Service Adapter Application to be configured (the one used during the create application (refer to 4.1)).

`-host <hostname>`

The host name of the Service Desk server.

`-user <user>`

The user account to log in to the Service Desk server.

Important: This user must have at least the entry 'Service Level Manager` selected in his role. If not, the OVSD SA will not be able to retrieve values.

To check this, in the OV SD application:

- select Tools\System…\Security\Access\Role

- check if 'Service Level Manager' role has been selected

-passwd *<passwd>*
    The password for the Service Desk account.

-services *<service list file>*
    The name of the file to which the discovered Service IDs and names will be written. Any existing files are overwritten.

-status *<status list file>*
    The name of the file to which the discovered statuses of each help desk types (service call and incident) will be written. Any existing files are overwritten.

-h
    Provides online help. Optional.

4.2.2.2 Example

```
configure -discover –sainame slmv12_acquisitionW2K_OvSd -host
sdserverhostname.domain.name -user system -passwd servicedesk
-services discoveredservices.xml -status discoveredstatus.xml
```

4.2.2.3 Output Files

Service Desk creates the XML files specified by the -services and -status parameters in the %TEMIP_SC_VAR_HOME%\ServiceAdapters\OvSd\<SA application name>\config directory.

4.2.2.4 Editing the Discovery Files

After performing the discovery, you need to edit the configuration files created by the configuration tool to:

- Select what service data to collect

- Manage the different Service status values (see *Status List File* on page 4-6).

4.2.2.5 Services List File

The discovery creates the service list file. This file contains the list of all the services for which data could be collected. The OVSD services that are to be monitored can differ. This can be controlled with the `servicecalls` and `incidents` attributes of the OvSdService tag as shown in **Error! Reference source not found.**. By default service calls and incidents are monitored for each service (value "1" is used).

- to set collecting state on, set the appropriate attribute to 1

- to set collecting state off, set the attribute to 0.

Figure 5: Sample Service List File (discoveredservices.xml)

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE sc:OvSdCfgServiceList SYSTEM "DTD/tsc_OvSdConfigServiceList.dtd">
<sc:OvSdCfgService>
<sc:OvSdService id="281478229328186" name="E-mail" servicecalls="0"
incidents="1"/>
<sc:OvSdService id="281478229328184" name="Webhost" servicecalls="1"
incidents="1"/>
<sc:OvSdService id="281478229328182" name="Network" servicecalls="1"
incidents="1"/>
</sc:OvSdCfgServiceList>
```

4.2.2.6 Status List File

In addition to the list of services, the discovery also creates status file containing the possible service calls and incident category statuses.

Service Desk can manage an unlimited number of status entries for a service call or incident. The Service Adapter therefore needs to know which status entries should be treated as 'open' and which should be treated as 'closed'.

To do this, you must set the value of the status attribute of the corresponding XML tag to either open (the default) or closed. The Service Adapter ignores any status that is not in the list. The status attributes are shown in

Figure 6: Sample Status List File (discoveredstatus.xml)

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE sc:OvSdCfgStatusList SYSTEM "DTD/tsc_OvSdConfigStatusList.dtd">
<sc:OvSdCfgStatusList>
   <sc:OvSdServiceCallStatusList>
       <sc:OvSdServiceCallStatus name="Registered" ordering="3094610092" status="open"/>
       <sc:OvSdServiceCallStatus name="In Progress" ordering="3094610093" status="open"/>
       <sc:OvSdServiceCallStatus name="Waiting" ordering="3120234513" status="open"/>
       <sc:OvSdServiceCallStatus name="Completed" ordering="3094610094" status="closed"/>
       <sc:OvSdServiceCallStatus name="Informed" ordering="3094610095" status="closed"/>
       <sc:OvSdServiceCallStatus name="Closed" ordering="3094610096" status="closed"/>
   </sc:OvSdServiceCallStatusList>
   <sc:OvSdIncidentStatusList>
       <sc:OvSdIncidentStatus name="Registered" ordering="3094610250" status="open"/>
       <sc:OvSdIncidentStatus name="In Progress" ordering="3094610251" status="open"/>
       <sc:OvSdIncidentStatus name="Waiting" ordering="3120234522" status="open"/>
       <sc:OvSdIncidentStatus name="Completed" ordering="3094610252" status="closed"/>
       <sc:OvSdIncidentStatus name="Informed" ordering="3094610253" status="closed"/>
       <sc:OvSdIncidentStatus name="Closed" ordering="3094610254" status="closed"/>
   </sc:OvSdIncidentStatusList>
</sc:OvSdCfgStatusList>
```

4.2.3   Configuring Data Feeder Definitions

This configuration creates two DFDs that specify what source of data is collected from Service Desk - either service calls or incidents.

4.2.3.1   Syntax

Select menu: All Programs → HP OpenView → SQM → ServiceAdapters → Ovsd v1_2 → Launch the Configuration tool
Use the following command at the **Command Prompt** to add a DFD. Normally there are two DFDs, one for service calls and one for incidents.

```
configure      -dfd
               -sainame <SA application name>
               -status <status_list_file>
               -itemtype [servicecall|incident]
               -name <DFD_name>
               -h
```

This command has the following parameters:

-dfd
   Starts the configuration tool in DFD configuration mode.

-sainame <SA application name>
   The name of the Service Adapter Application to be configured (the one used during the create application (refer to 4.1)).

-status <status list file>
   The name of the file that contains a list of the different types of status and whether they are opened or closed.

-itemtype [servicecall|incident]
   The type of the DFD to be created.

-name <DFD_name>

The name of the DFD to be created. The name can contain only alphanumeric characters, plus the characters "-" and "_". The length of the name cannot exceed 12 characters and is prefixed automatically by the configuration tool with the Service Adapter name `OvSd`. The name must be unique within a Service Quality Manager environment.

Note that the DFD name needs to be defined according to the name used during the model definition step for this SA (for examples: scDfd, incidentDfd). The DFD name is unique for all the acquisition directors within the solution.

-h
Provides online help.

4.2.3.2 Examples

```
configure –dfd –sainame slmv12_acquisitionW2K_OvSd -status
discoveredstatus.xml -itemtype incident
-name incidentDfd

configure –dfd –sainame slmv12_acquisitionW2K_OvSd -status
discoveredstatus.xml –itemtype servicecall -name scDfd
```

4.2.3.3 Output File

Service Desk creates NewDFDReq_<DFD Name>_<DFD Version>.xml and DelDFDReq_<DFD name>_<DFD Version>.xml in the %TEMIP_SC_VAR_HOME%\ServiceAdapters\OvSd\<SA application name>\config directory.

4.2.4 Configuring the OVSD Application

You use the configuration tool to customize the connection and data feeder definitions information to be monitored by the Service Adapter Application.

4.2.4.1 Syntax

Select menu: All Programs → HP OpenView → SQM → ServiceAdapters → Ovsd v1_2 → Launch the Configuration tool
To configure the Service Adapter Application, use the following command:

```
configure     -sai
              -sainame <SA application name>
              -connectors <monitored_connectors_names>
              -datafeeders <monitored_datafeeders_names>
              -h
```

This command has the following parameters:

-sai
Starts the configuration tool in Service Adapter Application configuration mode.

-sainame <SA application name>
The name of the Service Adapter Application to be configured (the one used during the create application (refer to 4.1)).

-connectors <monitored_connectors_names>
A list of the connector names, separated by a space, to be monitored by the Service Adapter Instance. The connector name must correspond to a name already configured in step **Error! Reference source not found.**.

-datafeeders <monitored_datafeeders_names>
A list of the data feeder names, separated by a space, to be monitored by the Service Adapter Instance.

[-h]
Provides online help.

4.2.4.2 Example

```
configure -sai -sainame slmv12_acquisitionW2K_OvSd -connectors
xyz_service_desk -datafeeders incidentDfd svccallDfd
```

4.2.4.3 Output File

Service Desk creates the files OvSd_connectors_data.exp, OvSd_dfds_data.exp and
OvSd_<SA application name>.exp in the
%TEMIP_SC_HOME%\ServiceAdapters\OvSd\<SA application name>\config
directory.

4.2.5 Configuring Data Feeder Instances

You use the configuration tool to discover the list of DFIs that will be managed by
the Service Adapter Application.

4.2.5.1 Syntax

Select menu: All Programs → HP OpenView → SQM → ServiceAdapters → Ovsd
v1_2 → Launch the Configuration tool
To discover the DFIs, use the following command:

```
configure      -dfi
               -sainame <SA application name>
               -conname <connector_name>
               [-servername <SD server_name>]
               -dfdname <DFD_name>
               -services <service_list_file>
               [-qty <dfi per message>]
               [-h]
```

This command has the following parameters:

-dfi
   Starts the configure tool in DFI discovery mode.

-sainame *<SA application name>*
   The name of the Service Adapter Application to be configured (the one used
   during the create application (refer to 4.1)).

-conname *<connector_name>*
   The name of the connector.

[-servername *<SD_server_name>*]
   The name of the Service Desk server. If not provided, the Server Name of the
   connector is used by default.

-dfdname *<DFD_name>*
   The name of the DFD to which the DFIs will be assigned. The DFD name must
correspond to a previously configured DFD.

-services *<service_list_file>*
   The name of the file that contains all the services for which DFIs must be created

[- qty <dfi per message>]

   Specify the number of dfi declaration per generated message. Useful to control the size of
   DFI declaration messages (avoid timeout or log processing).
[-h]
   Provides online help.

4.2.5.2 Examples

```
configure -dfi -sainame slmv12_acquisitionW2K_OvSd -servername
sdserver -conname xyz_service_desk -dfdname incidentDfd -
```

```
services discoveredservices.xml –qty 25

configure –dfi -sainame slmv12_acquisitionW2K_OvSd –servername
sdserver -conname xyz_service_desk -dfdname svccallDfd -
services discoveredservices.xml –qty 25
```

4.2.5.3   Output File

Service Desk creates the files DeclareDFIReq_<DFD Name>_<DFI Id>.xml and a
file called for each DFI DelDFIReq_<DFD Name>_<DFI Id>.xml in the
%TEMIP_SC_VAR_HOME%\ServiceAdapters\OvSd\<SA application
name>\config directory.

4.2.6   Configuration Tool output

After you run the discovery, the following output files will be generated in the
%TEMIP_SC_VAR_HOME%\ServiceAdatapters\Ovsd\<SA application
name>\config directory:

- The OVSD Service Adapter instance configuration file
  The file *OvSd_<SA application Name> .exp* contains all application variables that
  are loaded into the repository at the application creation step described in the next
  section. The SA Application name has the following syntax:

  ```
  <SQM Platform Name>_<SQM Director Name>_OvSd
  ```

  where the SQM Platform and Director were chosen during SQM installation and
  configuration.

- The OVSD Service Adapter Connectors configuration file
  The file `OvSd_connectors_data.exp` contains connection parameters (Host,
  port, UserName …) to be used by the Service Adapter to access the OVSD system.
  This configuration file is loaded into the repository (see next section).

- The OVSD Service Adapter Datafeeder Configuration
  The file `OvSd_dfds_data.exp` contains the DFDs to be loaded into the
  repository.

- DFDs and DFIs
  Additional DFDs and instances XML files are generated to upload or delete the
  definitions and instances to or from the SRM.

  o *NewDFDReq_<DFDName>_<DFDVersion>.xml*
     DFD to be loaded into SRM

  o *DeclareDFIReq_<DFDName>_<DFIId>.xml*
     DFD to be loaded into the SRM

  o *DelDFIReq_<DFDName>_<DFIId>.xml*
     Delete DFI request to remove the DFI from the SRM.

  o *DelDFDReq_<DFDName>_<DFD Version>.xml*
     Delete DFD request to remove DFD from the SRM.

4.3   Loading configuration files

The final step is to load the configuration files to the SQM Service Repository
Manager and the Tibco central repository.

To be able to perform these steps:

- The SQM Kernel Subset must be installed, configured and started on the OVSD
  SA system

- The SQM platform has to be up and running on the SQM SLM Primary Server.

This step can be done once the configuration and the application creation of the OvSd SA has been done. The load of the application configuration will:

- Update the SQM repositories with the Data Feeder configurations and the OVSD connectors.

- Upload the DFD and DFI declarations

- Copy DFD definition to %TEMIP_SC_VAR_HOME%\Definitions\DFD. This option provides with the possibility to centralize all the Service Adapters Data Feeder definitions in a single directory for future use in the SQM Service Designer (Reverse engineering).

To launch the tool:

- Select menu: All Programs → HP OpenView → SQM → ServiceAdapters → Ovsd v1_2 → Load the Adapter application.

- You need to provide the following information:

  - The platform name on which the application has been created (ex: **slmv12**)

  - The director name on which the application has been created (ex: **acquisitionW2K**).

  - The name of the Service Adapter Application (ex: **OvSd**).

- Wait for the loading completion (this can take some time depending on the configuration to be loaded).

The configuration on Windows completed. Please follow the instruction in Chapter 7 to start the Service Adapter.

Chapter 5

Service Adapter Setup on Unix

Once the Service Adapter is installed, the **OvSd_setup.sh** shell script is created in:
$TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/utilities/. This tool must be executed from its directory. It provides the following options:

```
# cd $TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/utilities
# OvSd_setup.sh
Please select action to perform:
1) Create a new Adapter application
2) Load the Adapter configuration
3) Start the Adapter application
4) Stop the Adapter application
5) Delete the Adapter application
6) Quit configuration tool
```

The **configure.sh** script is created in:
$TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/utilities/. This tool is used to perform the configuration tasks of the OVSD Service Adapter.

The Following table summarizes the steps to be done according to the usual lifecycles of the OVSD SA:

Create	(1) Create → Configure → (2) Load → (3) Start
Update	(4) Stop → Configure → (2) Load → (3) Start
Delete	(4) Stop → (5) Delete

To create or update the Service Adapter, both OvSd_setup.sh and configure.sh tools are used.

All these steps are described in the following chapters.

5.1 Creating Application

Be aware that the create application will perform the kernel setup if not already done. If your Kernel setup has already been performed refer to section 5.1.2 else perform first section 5.1.1.

Caution

A Service Adapter application name has to be unique cross platform and director!

5.1.1 Kernel setup

The kernel setup is executed automatically during the SA Application creation but the following steps must be run in order to retrieve the platform description file:

1. Check that the environment is correct (refer to section 3.2.2)

2. Copy the platform description file
   '$TEMIP_SC_VAR_HOME/setupconfig/platform_desc.cfg'** from the SQM SLM Primary Server (where SQM is running) to the local system into **$TEMIP_SC_HOME/tmp directory.

5.1.2 OVSD application creation

1. Go into the setup tool directory:

```
cd $TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/utilities/
```

2. Use the following command to create the application (use **root** account):

```
OvSd_setup.sh
```

3. Select option: **'1) Create a new Adapter application'**

4. You need to provide the following information:

   > The platform name on which the application will be created (ex: **slmv12**).

   > The director name on which the application will be created (ex: **acquisition**).

   > The name of the Service Adapter Application (ex: **OvSd**).

5. Wait for the creation completion.

5.1.2.1 Example (case setup already done)

```
# cd $TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/utilities/
# OvSd_setup.sh
Please select action to perform:
1) Create a new Adapter application
2) Load the Adapter configuration
3) Start the Adapter application
4) Stop the Adapter application
5) Delete the Adapter application
6) Quit configuration tool
#? 1
Create the Adapter application
Please enter the platform: slmv12
Please enter the director: acquisition
Please enter the application: OvSd

The platform description file is a valid XML file!
TEMIP_SC_VAR_HOME=/var/opt/OV/SQM/slmv12
Application to setup (platform, director, application):
    slmv12, acquisition, OvSd
The platform description file is a valid XML file!
```

```
Start the setup of the specified addOn(s)

Traces are redirected in …

Directors were created successfully
Traces are redirected in …

Applications were created successfully

Operation Success.

Press [ENTER] to continue
```

> **Note:** Please pay attention to the Service Adapter Instance name (in this example:
> `'slmv12_acquisition_OvSd'`), which will be used during the whole life cycle of
> this application.

5.2 Configuring the Application

This chapter describes how to set up the HP OpenView Service Desk Service
Adapter. It explains how to use the configuration tool to query the Service Desk
Servers for configuration information. The chapter describes then how to edit the
configuration files to customize the Service Adapter.

Note

All the commands used in the following chapters must be launched from the *sqmadm*
user account (with SQM environment loaded).

5.2.1 Configuring the connector

The first step to configure the Service Adapter consists in retrieving connection
information.

To get data from Service Desk server, the Service Adapter needs to log into Service
Desk. This configuration step defines the Service Adapter connector.

The connector contains the following settings:

- ApplPassword
  The password used to log into Service Desk application. The password is stored
  encrypted.

- ApplServerHost
  The host name of the Service Desk Server. Ex:sdserverhostname.domain.name

- ApplAccountName
  The account name used for logging in. This does need to be a user interface
  account, but it does need access rights to all necessary service calls and incidents.

5.2.1.1 Syntax

Under sqmadm account:

 # $TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/configure.sh

```
        -connector
        -sainame <SA application name>
        -host <hostname>
        -user <user> -passwd <passwd>
```

```
                     -name <connector_name>
                     -h
```

This command has the following parameters:

`-connector`
   Starts the configuration tool in connector configuration mode.

`-sainame <SA application name>`
   The name of the Service Adapter Application to be configured (the one used during the create application (refer to 5.1.2)).

`-host <hostname>`
   The host name of the Service Desk server.

`-user <user>`
   The user account to log in to the Service Desk server.

`-passwd <passwd>`
   The password for the Service Desk account.

`-name <connector_name>`
   The name of the connector to be created. The name may contain only alphanumeric characters, plus the characters "-" and "_". If you do not specify a name, the host name is used, with underscores instead of periods.

`-h`
   Provides online help. Optional.

5.2.1.2   Example

```
configure.sh –connector –sainame slmv12_acquisition_OvSd -host
sdserverhostname.domain.name -user system -passwd servicedesk
-name hostname_service_desk
```

5.2.1.3   Output File

Service Desk changes or creates the file configure.dat in the $TEMIP_SC_VAR_HOME/ServiceAdapters/OvSd/<SA application name>/config directory.

5.2.2   Discovering Service Desk Configuration

The next step in configuring the Service Adapter consists in retrieving the list of services and status information from the Service Desk Server.

5.2.2.1   Syntax

Under sqmadm account:

$TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2 /configure.sh

```
                 -discover
                 -sainame <SA application name>
                 -host <hostname>
                 -user <user>
                 -passwd <passwd>
                 -services <service list file>
                 -status <status list file>
                 -h
```

This command has the following parameters:

`-discover`
   Starts the configuration tool in discovery mode.

`-sainame <SA application name>`

The name of the Service Adapter Application to be configured (the one used during the create application (refer to 5.1.2)).

-host *<hostname>*
The host name of the Service Desk server.

-user *<user>*
The user account to log in to the Service Desk server.
Important: This user must have at least the entry 'Service Level Manager` selected in his role. If not, the OVSD SA will not be able to retrieve values.
To check this, in the OV SD application:
- select Tools/System…/Security/Access/Role
- check if 'Service Level Manager' role has been selected

-passwd *<passwd>*
The password for the Service Desk account.

-services *<service list file>*
The name of the file to which the discovered Service IDs and names will be written. Any existing files are overwritten.

-status *<status list file>*
The name of the file to which the discovered statuses of each help desk types (service call and incident) will be written. Any existing files are overwritten.

-h
Provides online help. Optional.

5.2.2.2 Example

```
configure.sh -discover –sainame slmv12_acquisition_OvSd -host
sdserverhostname.domain.name -user system -passwd servicedesk
-services discoveredservices.xml -status discoveredstatus.xml
```

5.2.2.3 Output Files

Service Desk creates the XML files specified by the -services and -status parameters in the $TEMIP_SC_VAR_HOME/ServiceAdapters/OvSd/<SA application name>/config directory.

5.2.2.4 Editing the Discovery Files

After performing the discovery, you need to edit the configuration files created by the configuration tool to:

- Select what service data to collect

- Manage the different Service status values (see below).

5.2.2.5 Services List File

The discovery creates the service list file. This file contains the list of all the services for which data could be collected. The OVSD services that are to be monitored can differ. This can be controlled with the `servicecalls` and `incidents` attributes of the OvSdService tag as shown in **Error! Reference source not found.**. By default service calls and incidents are monitored for each service (value "1" is used).

- to set collecting state on, set the appropriate attribute to 1

- to set collecting state off, set the attribute to 0.

Figure 7: Sample Service List File (discoveredservices.xml)

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE sc:OvSdCfgServiceList SYSTEM "DTD/tsc_OvSdConfigServiceList.dtd">
```

```
<sc:OvSdCfgServiceList xmlns:sc="http://www.compaq.com/TeMIP/ServiceCenter">
     <sc:OvSdCfgService id="281478248923386" incidents="1"
         name="OVO Services and Processes" servicecalls="1"/>
     <sc:OvSdCfgService id="281478249382214" incidents="1" name="SAP"
servicecalls="1"/>
     <sc:OvSdCfgService id="281478229328186" incidents="1" name="E-MAIL"
servicecalls="1"/>
     <sc:OvSdCfgService id="281478229328184" incidents="1" name="Desktop"
servicecalls="1"/>
     <sc:OvSdCfgService id="281478229328188" incidents="1"
         name="Database Management" servicecalls="1"/>
     <sc:OvSdCfgService id="281478229328190" incidents="1" name="Webhost"
servicecalls="1"/>
     <sc:OvSdCfgService id="281478229328192" incidents="1" name="Server"
servicecalls="1"/>
     <sc:OvSdCfgService id="281478229328194" incidents="1" name="Network"
servicecalls="1"/>
     <sc:OvSdCfgService id="281478229328196" incidents="1" name="DBA"
servicecalls="1"/>
     <sc:OvSdCfgService id="281478229328198" incidents="1"
         name="HP Hardware Service" servicecalls="1"/>
</sc:OvSdCfgServiceList>
```

5.2.2.6   Status List File

In addition to the list of services, the discovery also creates status file containing the possible service calls and incident category statuses.

Service Desk can manage an unlimited number of status entries for a service call or incident. The Service Adapter therefore needs to know which status entries should be treated as 'open' and which should be treated as 'closed'.

To do this, you must set the value of the status attribute of the corresponding XML tag to either open (the default) or closed. The Service Adapter ignores any status that is not in the list. The status attributes are shown in

Figure 8: Sample Status List File (discoveredstatus.xml)

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE sc:OvSdCfgStatusList SYSTEM "DTD/tsc_OvSdConfigStatusList.dtd">
<sc:OvSdCfgStatusList xmlns:sc="http://www.compaq.com/TeMIP/ServiceCenter">
  <sc:OvSdServiceCallStatusList>
    <sc:OvSdServiceCallStatus name="Registered"
      ordering="3094610092" status="open"/>
    <sc:OvSdServiceCallStatus name="InProgress"
      ordering="3094610093" status="open"/>
    <sc:OvSdServiceCallStatus name="Completed" ordering="3094610094" status="open"/>
    <sc:OvSdServiceCallStatus name="Informed" ordering="3094610095" status="open"/>
    <sc:OvSdServiceCallStatus name="Closed" ordering="3094610096" status="open"/>
    <sc:OvSdServiceCallStatus name="Waiting" ordering="3120234513" status="open"/>
  </sc:OvSdServiceCallStatusList>
  <sc:OvSdIncidentStatusList>
    <sc:OvSdIncidentStatus name="Registered" ordering="3094610250" status="open"/>
    <sc:OvSdIncidentStatus name="InProgress" ordering="3094610251" status="open"/>
    <sc:OvSdIncidentStatus name="Completed" ordering="3094610252" status="open"/>
    <sc:OvSdIncidentStatus name="Informed" ordering="3094610253" status="open"/>
    <sc:OvSdIncidentStatus name="Closed" ordering="3094610254" status="open"/>
    <sc:OvSdIncidentStatus name="Waiting" ordering="3120234522" status="open"/>
  </sc:OvSdIncidentStatusList>
</sc:OvSdCfgStatusList
```

5.2.3   Configuring Data Feeder Definitions

This configuration creates two DFDs that specify what source of data is collected from Service Desk - either service calls or incidents.

5.2.3.1 Syntax

Use the following command (from **sqmadm** account) to add a DFD. Normally there are two DFDs, one for service calls and one for incidents.

$TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2 /configure.sh

```
-dfd
-sainame <SA application name>
-status <status_list_file>
-itemtype [servicecall|incident]
-name <DFD_name>
-h
```

This command has the following parameters:

-dfd
>   Starts the configuration tool in DFD configuration mode.

-sainame *<SA application name>*
>   The name of the Service Adapter Application to be configured (the one used during the create application (refer to 5.1.2)).

-status *<status list file>*
>   The name of the file that contains a list of the different types of status and whether they are opened or closed.

-itemtype [servicecall|incident]
>   The type of the DFD to be created.

-name *<DFD_name>*
>   The name of the DFD to be created. The name can contain only alphanumeric characters, plus the characters "-" and "_". The length of the name cannot exceed 12 characters and is prefixed automatically by the configuration tool with the Service Adapter name OvSd. The name must be unique within a Service Quality Manager environment.
>   Note that the DFD name needs to be defined according to the name used during the model definition step for this SA (for examples: DFDSrvCall, DFDincident). The DFD name is unique for all the acquisition directors within the solution.

-h
>   Provides online help. Optional.

<div align="center">Note</div>

The configuration tool will add 'OvSd' as prefix to the *<DFD name>* given as parameter. When using it through other tools, you must not add this prefix to the *<DFD name>*.

5.2.3.2 Example

```
configure.sh –dfd –sainame slmv12_acquisition_OvSd -status
discoveredstatus.xml -itemtype incident -name DFDIncident

configure.sh –dfd –sainame slmv12_acquisition_OvSd -status
discoveredstatus.xml –itemtype servicecall -name DFDSrvcCall
```

5.2.3.3 Output File

The following files are created:

- NewDFDReq_OvSd<DFD Name>.<DFD Version>.xml
  Used to upload the DFD to the SQM Service Repository

- DelDFDReq_<DFD name>.<DFD Version>.xml
  Used to delete the DFD from the SQM Service Repository.

The generated files are located in:
`$TEMIP_SC_VAR_HOME/ServiceAdapters/OvSd/<SA application name>/config/`

5.2.4   Configuring the OVSD Application

You use the configuration tool to customize the connection and data feeder definitions information to be monitored by the Service Adapter Application.

5.2.4.1   Syntax

To configure the Service Adapter Application, use the following command (from **sqmadm** account):

```
configure.sh -sai
            -sainame <SA application name>
            -connectors <monitored_connectors_names>
            -datafeeders <monitored_datafeeders_names>
            -h
```

The command has the following parameters:

`-sai`

Starts the configuration tool in Service Adapter Application configuration mode.

`-sainame <SA application_name>`
The name of the Service Adapter Application to be configured (the one used during the create application (refer to 5.1.2)).

`-connectors <connector1 [connector2 […]]>`

A list of the connector names, separated by a space, to be monitored by the Service Adapter Application. A connector mentioned in this list has to be previously declared in the phase 'configuring the connector'.

`-datafeeders <datafeeder1 [datafeeder2 […]]>`

A list of the data feeder names, separated by a space, to be monitored by the Service Adapter Application. A data feeder mentioned in this list has to be previously declared in the phase 'configuring Data Feeder Definitions'.

`-h`
Provides online help. Optional.

5.2.4.2   Example

```
configure.sh –sai -sainame slmv12_acquisition_OvSd  -connectors
hostname_service_desk -datafeeders DFDIncident DFDSrvcCall
```

5.2.4.3   Output files

Service Desk creates the following files:

- `OvSd_connectors_data.exp`
  List of connectors to be uploaded to the SQM Repository.

- `OvSd_dfds_data.exp`
  List of DataFeederDefConfig to be uploaded to the SQM Repository.

- `OvSd_slmv12_acquisition_OvSd.exp`
  File used to create the Service Desk application.

The generated files are located in:
`$TEMIP_SC_VAR_HOME/ServiceAdapters/OvSd/<SA application name>/config/`

5.2.5 Configuring Data Feeder Instances

You use the configuration tool to discover the list of DFIs that will be managed by the Service Adapter Application.

5.2.5.1 Syntax

To discover the DFIs, use the following command (from **sqmadm** account):

```
configure.sh -dfi
            -sainame <SA application name>
            -conname <connector_name>
            [-servername <SD server_name>]
            -dfdname <DFD_name>
            -services <service_list_file>
            [-qty <dfi per message>]
            [-h]
```

```
This command has the following parameters:
```

-dfi
   Starts the configure tool in DFI discovery mode.

-sainame <SA application_name>
   The name of the Service Adapter Application to be configured (the one used during the create application (refer to 5.1)).

-conname <connector_name>
   The name of the connector. Should be an already defined connector. When specifying the connector, replace all "." characters by "_".

[-servername <SD_server_name>]
   The name of the Service Desk server. If not provided, the Server Name of the connector is used by default.

-dfdname <DFD_name>
   The name of the DFD to which the DFIs will be assigned. Should be an already defined DFD name.

-services <service_list_file>
   The name of the file that contains all the service IDs for which DFIs must be created. This file is created during the discovery.

[-qty <dfi per message>]
   Specify the number of dfi declaration per generated message. Useful to control the size of DFI declaration messages (avoid timeout or log processing).
[-h]
   Provides online help.

5.2.5.2 Example

```
configure.sh –dfi -sainame slmv12_acquisition_OvSd –servername
sdserver -conname xyz_service_desk -dfdname DFDincident -
services discoveredservices.xml –qty 25

configure.sh –dfi -sainame slmv12_acquisition_OvSd –servername
sdserver -conname xyz_service_desk -dfdname DFDSvcCall -
services discoveredservices.xml –qty 25
```

5.2.5.3 Output Files

The following files are created:

- DeclareDFIReq_OvSd<DFD Name><DFI Id>.xml for all DFIs
  Used to upload the DFIs to the SQM Service Repository.

- DelDFIReq_<*DFI Id*>.xml for each DFI defined.
  Used to remove the DFIs from the SQM Service Repository.

The generated files are located in:
```
$TEMIP_SC_VAR_HOME/ServiceAdapters/OvSd/<SA application
name>/config/
```

5.3 Loading configuration files

The final step is to load the configuration files to the SQM Service Repository Manager and the Tibco central repository.

The setup tool is located in:
```
$TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/utilities/
```

To be able to perform these steps:

- The SQM Kernel Subset must be installed, configured and started on the OVSD SA system

- The SQM platform has to be up and running on the SQM SLM Primary Server.

This step can be done once the configuration and the application creation of the OvSd SA has been done. The load of the application configuration will:

- Update the SQM repositories with the Data Feeder configurations and the OVSD connectors.

- Upload the DFD and DFI declarations

- Copy DFD definition to $TEMIP_SC_VAR_HOME/Definitions/DFD. This option provides with the possibility to centralize all the Service Adapters Data Feeder definitions in a single directory for future use in the SQM Service Designer (Reverse engineering).

To run the tool (from **sqmadm** account):

1. OvSd_setup.sh

2. Select option: **'*2) Load the Adapter configuration*'**

3. You need to provide the following information (i.e. the SA Application name: the one used during the create application (refer to 5.1)):

   - The platform name on which the application has been created (ex: **slmv12**)

   - The director name on which the application has been created (ex: **acquisition**)

   - The name of the Service Adapter Application (ex: **OvSd**).

4. Wait for the loading completion (this can take some time depending on the configuration to be loaded)

The configuration on Unix is completed. Please follow the instruction in Chapter 7 to start the service adapter.

Chapter 6

Service Adapter Operation for Windows

6.1 SA Start/Stop

6.1.1 Starting the Service Adapter Application

Before starting the Service Adapter, ensure that the SQM Kernel is running on both systems (Windows and HP-UX SQM SLM Primary Server). Also ensure that the full SQM monitoring feature is running on the HP-UX SLM Primary Server (refer to the *SQM Administration Guide* for these configuration steps).

Launch the tool:

1. Select menu: All Programs → HP OpenView → SQM → ServiceAdapters → OvSd v1_2→ Start the Adapter Application.

   The OVSD application tool window is displayed.

2. You need to provide the following information:

   - The platform name on which the application has been created (ex: **slmv12**)

   - The director name on which the application will be installed. (ex: **acquisitionW2K**)

   - The name of the Service Adapter Application. (ex: **OvSd**)

3. When the operation is over, you are ready to close the window.

4. To check if the application is running, you can list the processes in the Windows Task Manager and check that there is a newly **sqm_invoque.exe** process created.

For more information, refer to the *HP OpenView Service Quality Manager Administration Guide*.

6.1.2 Stopping the Service Adapter Application

To stop the Service Adapter:

Launch the tool:

1. Select menu: All Programs→ HP OpenView → SQM → ServiceAdapters → OvSd v1_2 → Stop the Adapter Application.

   The Ovsd application tool window is displayed.

2. You need to provide the following information:

- The platform name on which the application has been created (ex: **slmv12**)

- The director name on which the application has been created (ex: **acquisitionW2K**).

- The name of the Service Adapter Application (ex: **OvSd**).

3. When the operation is over, you are ready to close the window.

6.1.3 Starting and Stopping the Service Adapter Collection

The collection of OVSD data starts only when the Service Adapter Application is instructed to do so by the SQM Service Repository Manager. This is controlled from the SQM SLA Admin User Interface. For more information, see the *HP OpenView SQM SLA Admin User Interface User's Guide*

6.2 Maintenance

6.2.1 Create a new Application

Creating a new Service Adapter Application implies following steps:

- Create the Service Adapter application. (Refer to chapter 4.1)

- Configure the Service Adapter Application. (Refer to chapter 4.2)

- Load the Service Adapter configuration. (Refer to chapter 4.3)

- Start the Service Adapter Application (Refer to chapter 6.1.1)

6.2.2 Update an existing Application

Updating an existing Service Adapter Application implies following steps:

- Stop the Service Adapter Application to be updated. (Refer to chapter 6.1.2)

- Configure the Service Adapter Application (Refer to chapter 4.2)

- Load the Service Adapter configuration. (Refer to chapter 4.3)

- Start the Service Adapter Application (Refer to chapter 6.1.1)

6.2.3 Delete an existing Application

To run tool:

1. Stop the Service Adapter (Refer to chapter 6.1.2)

2. Delete all Data Feeder Instances associated to this Service Adapter Instance.

3. Select menu: All Programs → HP OpenView → SQM → ServiceAdapters → Ovsd v1_2 → Delete the Adapter application.

4. You need to provide the following information (i.e.: the SA application name (the one used during the create application (refer to4.1.2))):

- The platform name on which the application has been created (ex: **slmv12**)

- The director name on which the application has been created (ex: **acquisitionW2K**).

- The name of the Service Adapter Application (ex: **OvSd**).

5. When the operation is over, you can close the window.

6.2.4 Scheduling Configuration

The Service Adapter can be installed in an OVSD environment where new OVSD services are changed daily. In this case, the Service Adapter configuration can be scheduled every day.

Following steps must be done in order to update the OVSD SA configuration. These steps can be automated in a script using the corresponding actions:

STEP	ACTION(s) TO BE DONE …
Stop	"C:\Program Files\HP OpenView\SQM\ServiceAdapters\OvSd\v1_2\utilities\setup.cmd" -appli stop
Configure	cd %TEMIP_SC_HOME%\ServiceAdapters\OvSd\v1_2\utilities then call the 'configure' tool with appropriate arguments
Load	"C:\Program Files\HP OpenView\SQM\ServiceAdapters\OvSd\v1_2\utilities\setup.cmd" -appli load
Start	"C:\Program Files\HP OpenView\SQM\ServiceAdapters\OvSd\v1_2\utilities\setup.cmd" -appli start

Warning

If you do this, you cannot restrict the data that is collected nor change the default parameter labels. **All data** is collected.

Chapter 7

Service Adapter Operation for UNIX

7.1 SA Start/Stop

7.1.1 Starting the Service Adapter Application

When the Service Adapter application starts it connects automatically to the Service Desk Server (with the connection information provided at the setup). To start the Service Adapter you must be logged in as `sqmadm` and all SQM environment variables must be set. Ensure that the SQM Kernel is running. For more information, refer to the *HP OpenView SQM Administration Guide*.

The setup tool is located in:
`$TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/utilities/`

Use the following command (from **sqmadm** account) to start the application:

1. OvSd_setup.sh

2. Select option: **'3**) *Start the Adapter application***'**

3. You need to provide the following information (i.e. the SA application name: the one used during the create application (refer to 5.1)):

▪ The platform name on which the application has been created (ex: **slmv12**)

▪ The director name on which the application has been created (ex: **acquisition)**

▪ The name of the Service Adapter Application (ex: **OvSd**).

Wait for the application start completion.

Example

```
# OvSd_setup.sh
Please select action to perform:
1) Create a new Adapter application
2) Load the Adapter configuration
3) Start the Adapter application
4) Stop the Adapter application
5) Delete the Adapter application
6) Quit configuration tool
#? 3
Start the Adapter application
Please enter the platform [slmv12] : <ENTER>
Please enter the director [acquisition] : <ENTER>
Please enter the application [OvSd] : <ENTER>
launch start of application OvSd on director acquisition,
platform slmv12
Processing
/tibco/private/adapter/ServiceCenter/PlatformDescription/slmv12
/platform ...
```

```
Application OvSd is starting. Check Alerts in Hawk Display to
get the startup status.
Operation Success.
```

4. To check if the application is running, you can list the processes running on HP-
   UX with the command `temip_sc_show` and check that there is a newly
   sqm_SAOvSd process created.

7.1.2   Stopping the Service Adapter Application

To stop the Service Adapter application, you have to be connected as sqmadm user
and have all SQM environment variables set (please refer to *SQM Administration
Guide* for more information).

The setup tool is located in:
`$TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/utilities/`

Use the following command (from **sqmadm** account) to stop the application:

to stop the application:

1. OvSd_setup.sh

2. Select option: '4) Stop the Adapter application'

3. You need to provide the following information (i.e. the SA Application name: the
   one used during the create application (refer to 5.1)):

▪ The platform name on which the application has been created (ex: **slmv12**)

▪ The director name on which the application has been created (ex: **acquisition)**

▪ The name of the Service Adapter Application (ex: **OvSd**).

Wait for the application stop completion.

<u>Example</u>
```
# OvSd_setup.sh
Please select action to perform :
1) Create a new Adapter application
2) Load the Adapter configuration
3) Start the Adapter application
4) Stop the Adapter application
5) Delete the Adapter application
6) Quit configuration tool
#? 4
Stop the Adapter application
Please enter the platform [slmv12] : <ENTER>
Please enter the director [acquisition] : <ENTER>
Please enter the application [OvSd] : <ENTER>
Processing
/tibco/private/adapter/ServiceCenter/PlatformDescription/slmv12
/platform ...
Application OvSd is not running
Operation Success.
```

7.1.3   Starting and Stopping the Service Adapter Collection

The collection of OVSD data starts only when the Service Adapter Application is
instructed to do so by the SQM Service Repository Manager. This is controlled from
the SQM SLA Admin User Interface. For more information, see the *HP OpenView
SQM SLA Admin User Interface User's Guide*

7.2 Maintenance

7.2.1 Create a new Application

Creating a new SA Application implies following steps:

- Create the SA application. (Refer to 5.1.2 OVSD application creation)

- Configure the SA Application. (Refer to 5.2)

- Load the Application configuration. (Refer to 5.3 Loading configuration files)

- Start the SA Application. (Refer to 7.1.1 Starting the Service Adapter)

7.2.2 Update an existing Application

Updating an existing SA Application implies following steps:

- Stop the SA Application. (Refer to 7.1.2 Stopping the Service Adapter)

- Configure the SA Application. (Refer to 5.2)

- Load the Application configuration. (Refer to 5.3 Loading configuration files)

- Start the SA Application. (Refer to 7.1.1 Starting the Service Adapter)

7.2.3 Delete an existing Application

To delete the Service Adapter Application, you have to be connected as sqmadm user and have all SQM environment variables set (please refer to *SQM Administration Guide* for more information).

The setup tool is located in:
`$TEMIP_SC_HOME/ServiceAdapters/`**`OvSd`**`/v1_2/utilities/`

Use the following command (from **sqmadm** account) to delete the application:
1.   OvSd_setup.sh

2.   Select option: '5) Delete the Adapter application'

3.   You need to provide the following information (i.e. the SA application name: the one used during the create application (refer to 5.1)):

- The platform name on which the application has been created (ex: **slmv12**).

- The director name on which the application has been created (ex: **acquisition)**

- The name of the Service Adapter Application (ex: **OvSd**).

Wait until the delete is over.

Example

```
# OvSd_setup.sh
Please select action to perform :
1) Create a new Adapter application
2) Load the Adapter configuration
3) Start the Adapter application
4) Stop the Adapter application
5) Delete the Adapter application
6) Quit configuration tool
#? 5
Stop the Adapter application
Please enter the platform [slmv12] : <ENTER>
Please enter the director [acquisition] : <ENTER>
Please enter the application [OvSd] : <ENTER>
```

```
Processing
/tibco/private/adapter/ServiceCenter/PlatformDescription/slmv12
/platform ...
Application OvSd :
        applicationType  = Monitored
        host             = herbi
Application OvSd is NOT RUNNING


WARNING : Do you want to delete the application y/n ? [n] y
```

7.2.4 Scheduling Configuration

The Service Adapter can be installed in an OVSD environment where new OVSD services are changed daily. In this case, the Service Adapter configuration can be scheduled every day.

Following steps must be done in order to update the OVSD SA configuration. These steps can be automated in a script using the corresponding actions:

STEP	ACTION(s) TO BE DONE …
Stop	${TEMIP_SC_HOME}/ServiceAdapters/OvSd/v1_2/utilities/setup.pl -sa OvSd -appli stop
Configure	cd $TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/utilities
	then call the 'configure.sh' tool with appropriate arguments
Load	${TEMIP_SC_HOME}/ServiceAdapters/OvSd/v1_2/utilities/setup.pl -sa OvSd -appli load
Start	${TEMIP_SC_HOME}/ServiceAdapters/OvSd/v1_2/utilities/setup.pl –sa OvSd -appli start

Warning

If you do this, you cannot restrict the data that is collected nor change the default parameter labels. **All data** is collected.

Appendix A

Scenario

The scenario will show the right methods to implement OV SQM. Among others, it will present a typical architecture and the OVSD SA configuration steps.

The scenario will also underline important points like naming policy, discovery and management steps.

Introduction

Using a fictitious company, XYZ Corporation, the scenario describes how to set up the Service Adapter as described within the chapter 4.1

Like most large companies, XYZ Corporation has a Help Desk service to provide customer support and internal troubleshooting. To further improve the Help Desk, XYZ Corporation has purchased and installed HP OpenView Service Desk.

The following schema displays mainly the XYZ infrastructure with the location of the OV SQM SLM Primary Server and the servers, which host the service desk servers and databases.

Figure 9: XYZ Corporation Network
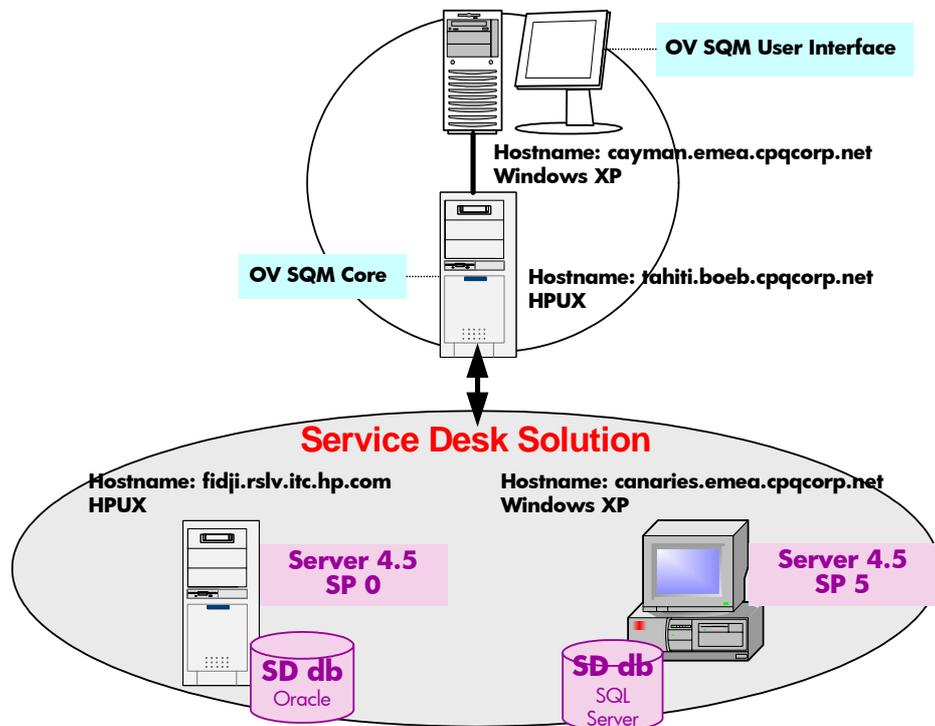
XYZ has defined a distributed architecture with:

- A first OVSD server hosted on HPUX (fidji). This server focuses on the service calls opened by XYZ Customers. Its version is 4.5 without service pack.

- A second one is hosted on windows XP (canaries) and is dedicated to the incidents generated for the OSS domain. Its version is the last one 4.5 service pack 5.

The XYZ Company wants to gather performance information from these two OV Service Desk solutions. The OV SD Service Adapter will collect key performance indicators from selected OVSD services.

Implementation

The first step of the OVSD Service Adapter implementation is to define the director architecture.

In this scenario, the acquisition director is installed on the windows server.

The OVSD Service Adapter is located within the windows acquisition director.

Two Service Adapter Applications are defined, one is dedicated to retrieve performance information from the UNIX SD Server whereas the second one collects locally OV Service Desk data.

On the following picture, the detailed technical information is mentioned; please take attention to the useful naming policy regarding the director and the Service Adapter Application names

- The platform name

- The windows acquisition director name

- The OVSD service adapter application name

- The mandatory configuration information from the OVSD server (user/passwd, OVSD host)

Figure 10: XYZ Corporation Network
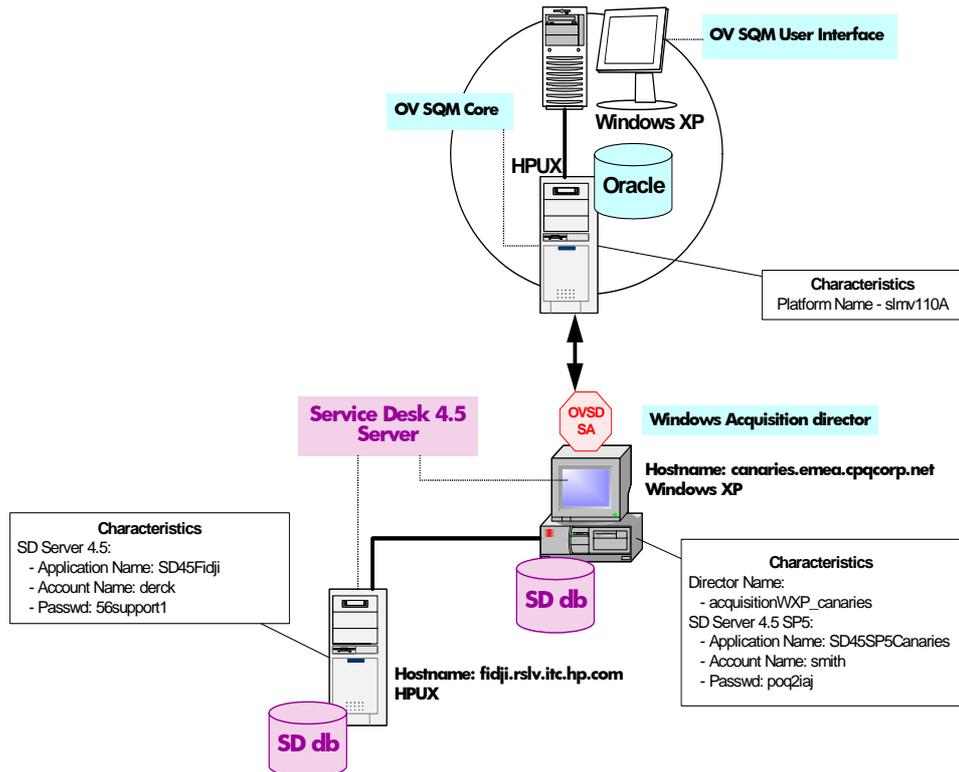
Setup

The setup of this solution is composed of several steps:

- **Creation of the Service Adapter Application**

This operation is performed by the OVSD SA "1- Create the Adapter Application" tool and launched on the host canaries.emea.cpqcorp.net.

Platform Name:

> **slmv120A**

Director name:

> The director naming policy is acquisition<operating system>_<Director Host>

> **acquisitionWXP_canaries**

Application name:

> The SA Application naming policy is SD<version>SP<version><SD Host>

> **SD45SP5Canaries**

> **SD45Fidji**

- **Configuration of the Service Adapter Application**

This operation is performed by the OVSD SA "2- Launch the Configuration tool". Please note that each Application configuration process should be immediately followed by the "3-Load the Adapter Configuration"

This sequence is only advisable to increase your understanding of the global Application implementation and facilitate Application configuration sequence.

The discovery files naming policy is <hostname>Services.xml &
<hostname>Status.xml

For SD45SP5Canaries:

```
configure -discover -sainame
slmv120A_acquisitionWXP_canaries_SD45SP5Canaries –host
canaries.emea.cpqcorp.net –user smith –passwd qop2iaj –services
canariesServices.xml –status canariesStatus.xml
```

canariesServices.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE sc:OvSdCfgServiceList SYSTEM "DTD//tsc_OvSdConfigServiceList.dtd">
<sc:OvSdCfgService>
      <sc:OvSdService id="281478229328183" name="System XYZ" servicecalls="0" incidents="1"/>
      <sc:OvSdService id="281478229328184" name="Network XYZ" servicecalls="0" incidents="1"/>
      <sc:OvSdService id="281478229328185" name="Application XYZ" servicecalls="0"
incidents="1"/>
</sc:OvSdCfgServiceList>
```

canariesStatus.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE sc:OvSdCfgStatusList SYSTEM "DTD/tsc_OvSdConfigStatusList.dtd">
<sc:OvSdCfgStatusList>
    <sc:OvSdIncidentStatusList>
        <sc:OvSdIncidentStatus name="Registered" ordering="3094610250" status="open"/>
        <sc:OvSdIncidentStatus name="In Progress" ordering="3094610251" status="open"/>
        <sc:OvSdIncidentStatus name="unknow" ordering="3120234522" status="closed"/>
        <sc:OvSdIncidentStatus name="Resolved" ordering="3094610252" status="closed"/>
        <sc:OvSdIncidentStatus name="Clear" ordering="3094610253" status="closed"/>
        <sc:OvSdIncidentStatus name="Closed" ordering="3094610254" status="closed"/>
    </sc:OvSdIncidentStatusList>
</sc:OvSdCfgStatusList>
```

```
configure -connector -sainame
slmv120A_acquisitionWXP_canaries_SD45SP5Canaries -host
canaries.emea.cpqcorp.net –user smith -passwd qop2iaj –name
canariesConnector

configure -dfd -sainame
slmv120A_acquisitionWXP_canaries_SD45SP5Canaries -status
canariesStatus.xml -itemtype incident -name incidentDfd

configure -dfd -sainame
slmv120A_acquisitionWXP_canaries_SD45SP5Canaries -status
canariesStatus.xml -itemtype servicecall -name scDfd

configure -sai -sainame
slmv120A_acquisitionWXP_canaries_SD45SP5Canaries
-connectors canariesConnector -datafeeders incidentDfd scDfd

configure -dfi -sainame
slmv120A_acquisitionWXP_canaries_SD45SP5Canaries
-servername canaries.emea.cpqcorp.net -conname canariesConnector
-dfdname incidentDfd –services canariesServices.xml
configure -dfi -sainame
slmv120A_acquisitionWXP_canaries_SD45SP5Canaries
-servername canaries.emea.cpqcorp.net -conname canariesConnector
–dfdname scDfd
-services canariesServices.xml
```

For SD45Fidji:

```
configure -discover -sainame
slmv120A_acquisitionWXP_canaries_SD45Fidji –host
```

```
fidji.rsvl.itc.hp.com –user derck –passwd 56support1 –services
fidjiServices.xml
–status fidjiStatus.xml
```

fidjiServices.xml

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE sc:OvSdCfgServiceList SYSTEM "DTD//tsc_OvSdConfigServiceList.dtd">
<sc:OvSdCfgService>
      <sc:OvSdService id="281478229328187" name="E-mail" servicecalls="1" incidents="0"/>
      <sc:OvSdService id="281478229328182" name="Webhost" servicecalls="1" incidents="0"/>
      <sc:OvSdService id="281478229328188" name="Network" servicecalls="1" incidents="0"/>
</sc:OvSdCfgServiceList>
```

fidjiStatus.xml

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE sc:OvSdCfgStatusList SYSTEM "DTD/tsc_OvSdConfigStatusList.dtd">
<sc:OvSdCfgStatusList>
    <sc:OvSdServiceCallStatusList>
        <sc:OvSdServiceCallStatus name="Registered" ordering="3094610092" status="open"/>
        <sc:OvSdServiceCallStatus name="In Progress" ordering="3094610093" status="open"/>
        <sc:OvSdServiceCallStatus name="Waiting" ordering="3120234513" status="open"/>
        <sc:OvSdServiceCallStatus name="Completed" ordering="3094610094" status="closed"/>
        <sc:OvSdServiceCallStatus name="Informed" ordering="3094610095" status="closed"/>
        <sc:OvSdServiceCallStatus name="Closed" ordering="3094610096" status="closed"/>
    </sc:OvSdServiceCallStatusList>
</sc:OvSdCfgStatusList>
```

```
configure -connector -sainame
slmv120A_acquisitionWXP_canaries_SD45Fidji -host
fidji.rsvl.itc.hp.com –user derck –passwd 56support1 –name
fidjiConnector

configure -dfd -sainame
slmv120A_acquisitionWXP_canaries_SD45Fidji -status
fidjiStatus.xml -itemtype incident -name incidentDfd
configure -dfd -sainame
slmv120A_acquisitionWXP_canaries_SD45Fidji -status
fidjiStatus.xml -itemtype servicecall -name scDfd

configure -sai -sainame
slmv120A_acquisitionWXP_canaries_SD45Fidji -connectors
fidjiconnector -datafeeders incidentDfd scDfd

configure -dfi -sainame
slmv120A_acquisitionWXP_canaries_SD45Fidji –servername
fidji.rsvl.itc.hp.com -conname fidjiConnector -dfdname
incidentDfd –services fidjiServices.xml
configure -dfi -sainame slmv120A_acquisitionWXP_canaries_SD45SP5
–servername fidji.rsvl.itc.hp.com –conname fidjiConnector –
dfdname scDfd –services fidjiServices.xml
```

Appendix B

Windows Installation Directory Structure

The following directories and files are installed:

```
TEMIP_SC_HOME\DTD\tsc_OvSdConfigServiceList.dtd
TEMIP_SC_HOME\DTD\tsc_OvSdConfigStatusList.dtd

TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\config
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\config\SCPlatform_SCDirector_SCApplication.p
roperties
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\lib
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\lib\web-api.jar
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\lib\README.TXT
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\jar
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\jar\TeSCSAOvSd.jar
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\properties
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\properties\TeSCOvSd.properties
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\properties\TeSCOvSd_Messages.properties
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\properties\TeSCOvSd_Version.properties
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\repository
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\repository\OvSd_template.exp
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\repository\OvSd_setup.cfg
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\bin
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\bin\OvSd_v1_2_launch.bat
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\utilities
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\utilities\configure.cmd
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\utilities\launchTools.cmd
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\utilities\launchTools.ico
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\utilities\SAOvSd_launcher.bat
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\utilities\setup.cmd
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\utilities\setup.ico
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\utilities\setup.pl
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\utilities\Welcome.txt
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\UninstallerDataOVSD
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\UninstallerDataOVSD\.com.zerog.registry.xml
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\UninstallerDataOVSD\Uninstall Ovsd.exe
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\UninstallerDataOVSD\Uninstall Ovsd.lax
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\UninstallerDataOVSD\uninstall.jar
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\UninstallerDataOVSD\resource
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\UninstallerDataOVSD\resource\iawin32.dll
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\UninstallerDataOVSD\resource\remove.exe
TEMIP_SC_HOME\ServiceAdapters\OvSd\v1_2\UninstallerDataOVSD\resource\ZGWin32LaunchHe
lper.exe
```

Unix Installation Directory Structure

The following directories and files are installed:

```
TEMIP_SC_HOME/DTD/tsc_OvSdConfigServiceList.dtd
TEMIP_SC_HOME/DTD/tsc_OvSdConfigStatusList.dtd
TEMIP_SC_HOME/adapter/bin/OvSd_v1_2_launch.sh
TEMIP_SC_HOME/etc/addOn/OvSd_v1_2_unix.tmpl_cfg
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/UninstallerDataOVSD
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/UninstallerDataOVSD/Uninstall_OvSd
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/UninstallerDataOVSD/Uninstall_OvSd.lax
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/UninstallerDataOVSD/uninstaller.jar
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/bin
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/bin/OvSd_v1_2_launch.sh
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/config
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/config/SCPlatform_SCDirector_SCApplication.p
roperties
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/jar
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/jar/TeSCSAOvSd.jar
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/lib
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/lib/README.TXT
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/lib/web-api.jar
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/properties
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/properties/TeSCOvSd.properties
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/properties/TeSCOvSd_Messages.properties
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/properties/TeSCOvSd_Version.properties
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/repository
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/repository/OvSd_setup.cfg
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/repository/OvSd_template.exp
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/utilities
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/utilities/OvSd_setup.sh
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/utilities/Welcome.txt
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/utilities/configure.sh
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/utilities/launchTools.sh
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/utilities/setup.sh
TEMIP_SC_HOME/ServiceAdapters/OvSd/v1_2/utilities/setup.pl
```

Appendix C

Troubleshooting

For troubleshooting information, refer to the *HP OpenView SQM Administration Guide*.

Appendix D

Acronyms

The following table lists the acronyms commonly used in this document:

Term	Description
API	Application programming interface
DFD	Data feeder definition
DFI	Data feeder instance
MRP	Measurement reference point
SAI	Service Adapter Application name (or Service Adapter Instance name)
SLA	Service level agreement
SLM	Service level management
SLO	Service level objective
SRM	Service Repository Manager
XML	eXtensible Mark-up Language